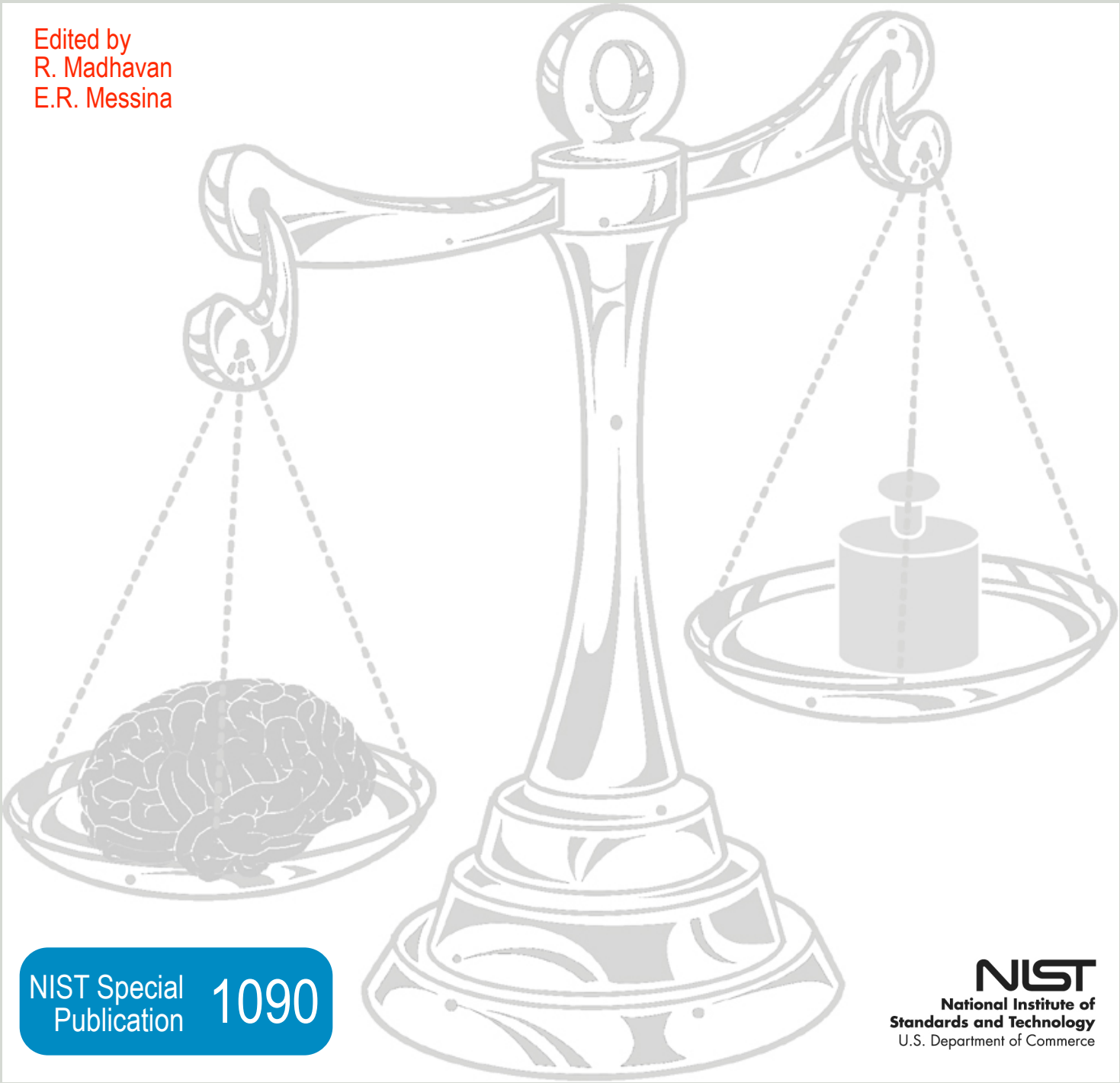
PERFORMANCE METRICS
FOR
INTELLIGENT SYSTEMS (PERMIS) WORKSHOP

National Institute of Standards and Technology, Gaithersburg, Maryland USA
August 19 - 21, 2008

Edited by
R. Madhavan
E.R. Messina



NIST Special Publication 1090

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

AUTHENTICATED
U.S. GOVERNMENT
INFORMATION
GPO

FOREWORD

The 2008 Performance Metrics for Intelligent Systems (PerMIS'08) Workshop will be the eighth in the series that started in 2000, targeted at defining measures and methodologies of evaluating performance of intelligent systems. The workshop has proved to be an excellent forum for discussions and partnerships, dissemination of ideas, and future collaborations in an informal setting. Attendees usually include researchers, graduate students, practitioners from industry, academia, and government agencies.

PerMIS'08 aims at *identifying and quantifying contributions of functional intelligence towards achieving success*. Our working definition of functional intelligence is "the ability to act appropriately in an uncertain environment, where appropriate action is that which increases the probability of success", and success is "the achievement of behavioral goals" (J. Albus, 1991). In addition to the main theme, as in previous years, the workshop will focus on applications of performance measures to practical problems in commercial, industrial, homeland security, and military applications. Topic areas include, but are not limited to:

Defining and measuring aspects of a system:

- The level of autonomy
- Human-robot interaction
- Collaboration & coordination
- Taxonomies
- Biologically inspired models

Evaluating components within intelligent system

- Sensing and perception
- Knowledge representation, world models, ontologies
- Planning and control
- Learning and adaption
- Reasoning

Infrastructural support for performance evaluation

- Testbeds and competitions for intercomparisons
- Instrumentation and other measurement tools
- Simulation and modeling support

Technology readiness measures for intelligent systems

Applied performance measures in various domains, e.g.,

- Intelligent transportation systems
- Emergency response robots (search and rescue, bomb disposal)
- Homeland security systems
- De-mining robots
- Defense robotics
- Hazardous environments (e.g., nuclear remediation)
- Industrial and manufacturing systems
- Space/Aerial robotics
- Medical Robotics & assistive devices

PerMIS'08 will feature five plenary addresses and seven special sessions. The plenary speakers are world-class experts in their own field and we are confident that the attendees will be able to benefit from their presentations. This year, there is a special session for every (parallel) general session. Over the course of three days, there will be twelve sessions related to performance of intelligent systems covering an array of topics from medical systems to manufacturing, mobile robotics to virtual automation, human-system interaction to biologically inspired models, and much more.

Special thanks are due to the Program Committee for publicizing the workshop, the special session organizers for proposing interesting topics and bringing together researchers related to their sessions, and the reviewers who provided feedback to the authors, and helped us to assemble an excellent program. We much appreciate the authors submitting their papers to this workshop and for sharing their thoughts and experiences related to their research with the workshop attendees.

PerMIS'08, is sponsored by NIST with technical co-sponsorship of the IEEE Washington Section Robotics and Automation Society Chapter and in-cooperation with the Association for Computing Machinery (ACM) Special Interest Group on Artificial Intelligence (SIGART). As in previous years, the proceedings of PerMIS will be indexed by INSPEC, Compendex, ACM's Digital Library, and are released as a NIST Special Publication. Springer Publishers are back again this year to raffle off some of the books that will be displayed at their booth during the course of the workshop. Selected papers from this workshop will be considered for inclusion in an edited book volume by Springer. We gratefully acknowledge the support of our sponsors.

We sincerely hope that you enjoy the presentations and the social programs!

Raj Madhavan
Program Chair

Elena Messina
General Chair

SPONSORS

NIST
National Institute of Standards and Technology

IEEE

acm

PROGRAM COMMITTEE

General Chair:

Elena Messina (Intelligent Systems Division, NIST, USA)

Program Chair:

Raj Madhavan (Oak Ridge National Laboratory/NIST, USA)

S. Balakirsky (NIST USA)
R. Bostelman (NIST USA)
F. Bonsignorio (Heron Robots Italy)
G. Berg-Cross (EM & I USA)
J. Bornstein (ARL USA)
P. Courtney (PerkinElmer UK)
J. Evans (USA)
D. Gage (XPM Tech. USA)
J. Gunderson (GammaTwo USA)
L. Gunderson (GammaTwo USA)
S. K. Gupta (UMD USA)
A. Jacoff (NIST USA)
S. Julier (Univ. College London UK)
M. Lewis (UPitt USA)
R. Lakaemper (Temple Univ USA)
T. Kalmar-Nagy (Texas A & M USA)
A. del Pobil (Univ. Jaume-I Spain)
S. Ramasamy UALR USA)
L. Reeker (NIST USA)
C. Schlenoff (NIST USA)
M. Shneier (NIST USA)
E. Tunstel (JHU-APL USA)

Table of Contents

Foreword vii

Sponsors viii

Program Committee ix

Plenary Addresses x

Workshop Program xiii

Author Index xix

Acknowledgements xxi

Technical Sessions

TUE-AM1 Performance Evaluation

Evolution of the SCORE Framework to Enhance Field-Based Performance Evaluations of Emerging Technologies
[Brian Weiss, Craig Schlenoff] 1

Reliability Estimation and Confidence Regions from Subsystem and Full System Tests via Maximum Likelihood
[James Spall] 9

Fuzzy-Logic-Based Approach for Identifying Objects of Interest in the PRIDE Framework
[Zeid Kootbally, Craig Schlenoff, Raj Madhavan, Sebti Foufou] 17

Identifying Objects in Range Data Based on Similarity Transformation Invariant Shape Signatures
[Xiaolan Li, Afzal Godi, Asim Wagan] 25

Stepfield Pallets: Repeatable Terrain for Evaluating Robot Mobility
[Adam Jacoff, Anthony Downs, Ann Virts, Elena Messina] 29

Potential Scaling Effects for Asynchronous Video in Multirobot Search
[Prasanna Velagapudi, Paul Scerri, Katia Sycara, Huadong Wang, Michael Lewis]. . . 35

TUE-AM2 Special Session I: Cognitive Systems of EU Cognition Programme

*Cognitive Systems of EU Cognition Programme**
[Patrick Courtney]

The Rat's Life Benchmark: Competing Cognitive Robots
[Olivier Michel, Fabien Rohrer] 43

The iCub Humanoid Robot: An Open Platform for Research in Embodied Cognition
[Giorgio Metta, Giulio Sandini, David Vernon, Lorenzo Natale, Francesco Nori] 50

An Open-Source Simulator for Cognitive Robotics Research: The Prototype of the iCub Humanoid Robot Simulator
[Vadim Tikhanoff, Angelo Cangelosi, Paul Fitzpatrick, Giorgio Metta, Lorenzo Natale, Francesco Nori] 57

Symbiotic Robot Organisms: REPLICATOR and SYMBRION Projects
[Serge Kernbach, Eugen Meister, Florian Schlachter, Kristof Jebens, Marc Szymanski, Jens Liedke, Davide Laneri, Lutz Winkler, Thomas Schmickl, Ronald Thenius, Paolo Corradi, Leonardo Ricotti] 62

Virtual Agent Modeling in the RASCALLI Platform
[Christian Eis, Marcin Skowron, Brigitte Krenn] 70

TUE-PM1 Human-System Interaction

Evaluation Criteria for Human-Automation Performance Metrics
[Birsen Donmez, Patricia Pina, Mary Cummings] 77

Assessing Measures of Coordination Demand Based on Interaction Durations
[Michael Lewis, Jijun Wang] 83

The Gestural Joystick and the Efficacy of the Path Tortuosity Metric for Human/Robot Interaction
[Richard Voyles, Jaewook Bae, Roy Godzdanker] 91

Modeling of Thoughtful Behavior with Dynamic Expert System
[Vadim Stefanuk] 98

TUE-PM2 Special Session II: Architectures for Unmanned Systems

*UAV Architectures**
[George Vachtsevanos]

*Architectures for Unmanned Systems**
[James Albus]

*Levels-of-Autonomy of the ASTM F41 Unmanned Maritime Vehicles Standard**
[Mark Rothgeb]

Ontological Perspectives for Autonomy Performance
[Hui-Min Huang, Elena Messina, Tsai Hong, Craig Schlenoff] 101

WED-AM1 Metrics & Measures

Robotic Systems Technical and Operational Metrics Correlation
[Jason Schenk, Robert Wade] 108

Survey of Domain-Specific Performance Measures in Assistive Robotic Technology
[Katherine Tsui, Holly Yanco, David Feil-Seifer, Maja Mataric] 116

Refining the Cognitive Decathlon
[Robert Simpson, Charles Twardy] 124

Using Metrics to Optimize a High Performance Intelligent Image Processing Code
[Scott Spetka, Susan Emen, George Ramseyer, Richard Linderman] 132

Measurement Techniques for Multiagent Systems
[Robert Lass, Evan Sultanik, William Regli] 134

*RoboCupRescue Robot League: 2008 Overview**
[Adam Jacoff, Andreas Birk, Johannes Pellenz, Ehsan Mihankhah, Raymond Sheh, Satoshi Tadokoro]

WED-AM2 Special Session III: Performance Metrics for Perception in Intelligent Manufacturing

Performance of Super-Resolution Enhancement for Flash LADAR Data
[Shuowen Hu, Susan Young, Tsai Hong] 143

Performance Evaluation of Laser Trackers
[Bala Muralikrishnan, Daniel Sawyer, Christopher Blackburn, Steven Phillips, Bruce Borchardt, Tyler Estler] 149

Preliminary Analysis of Conveyor Dynamic Motion for Automation Applications
[Jane Shi] 156

3D Part Identification Based on Local Shape Descriptors
[Xiaolan Li, Afzal Godil, Asim Wagan] 162

Calibration of a System of a Gray-Value Camera and an MDSI Range camera
[Tobias Hanning, Aless Lasaruk] 167

Dynamic 6DOF Metrology for Evaluating a Visual Servoing System
[Tommy Chang, Tsai Hong, Mike Shneier, German Holguin, Johnny Park, Roger Eastman]. 173

WED-PM1 Autonomous Systems

Integrating Reification and Ontologies for Mobile Autonomous Robots
[James Gunderson, Louise Gunderson] 181

Quantification of Line Tracking Solutions for Automotive Applications
[Jane Shi, Rick Rourke, Dave Groll, Peter Tavora] 189

Mobile Robotic Surveying Performance for Planetary Surface Site Characterization
[Edward Tunstel] 200

Evaluating Situation Awareness of Autonomous Systems
[Jan Gehrke] 206

WED-PM2 Special Session IV: Results from a Virtual Manufacturing Automation Competition

NIST/IEEE Virtual Manufacturing and Automation Competition: From Earliest Beginnings to Future Directions
[Stephen Balakirsky, Raj Madhavan, Chris Scrapper] 214

Analysis of a Novel Docking Technique for Autonomous Robots
[George Henson, Michael Maynard, Xinlian Liu, George Dimitoglou] 220

Partitioning Algorithm for Path Determination of Automated Robotic Part Delivery System in Manufacturing Environments
[Payam Matin, Ali Eydgahi, Ranjith Chowdary] 224

Algorithms and Performance Analysis for Path Navigation of Ackerman-Steered Autonomous Robots
[George Henson, Michael Maynard, George Dimitoglou, Xinlian Liu] 230

THU-AM1 Model-based Performance Assessment

Wireless Communications in Tunnels for Urban Search and Rescue Robots
[Kate Remley, George Hough, Galen Koepke, Dennis Camell, Robert Johnk, Chriss Grosvenor] 236

A Performance Assessment of Calibrated Camera Networks for Construction Site Monitoring
[Itai Katz, Nicholas Scott, Kam Saidi] 244

A Queuing-Theoretic Framework for Modeling and Analysis of Mobility in WSNs
[Harsh Bhatia, Rathinasamy Lenin, Aarti Munjal, Srini Ramaswamy, Sanjay Srivastava] 248

Towards Information Networks to Support Composable Manufacturing
[Mahesh Mani, Albert Jones, Junho Shin, Ram Sriram] 254

3D Reconstruction of Rough Terrain for USARSim using a Height-map Method
[Gael Roberts, Stephen Balakirsky, Sebti Foufou] 259

THU-AM2 Special Session V: Quantitative Assessment of Robot-generated Maps

*Characterizing Robot-Generated Maps: The Importance of Representations and Objective Metrics**
[Chris Scrapper, Raj Madhavan, Stephen Balakirsky]

Using Virtual Scans to Improve Alignment Performance in Robot Mapping
[Rolf Lakeamper, Nagesh Adluru] 265

The Role of Bayesian Bounds in Comparing SLAM Algorithms Performance
[Andrea Censi] 271

Map Quality Assessment
[Asim Wagan, Afzal Godil, Xiaolan Li] 278

Discussion: Roadmap for Map Evaluation Frameworks

THU-PM1 Special Session VI: Biologically Inspired Models of Intelligent Systems

Introduction to Biological Inspiration for Intelligent Systems
[Gary Berg-Cross] 283

*Overview of Biologically Inspired Cognitive Architectures (BICA)**
[Alexei Samsonovich]

*Recent modeling and Rapid Prototyping Experience Aimed at Building Architectures of Cognitive Agents**
[Giorgio Ascoli]

Applying Developmental-Inspired Principles to the Field of Developmental Robotics
[Gary Berg-Cross] 288

Discussion of Biologically Inspired Models
[Panel]

THU-PM2 Special Session VII: Medical Robotics

Overcoming Barriers to Wider Adoption of Mobile Telerobotic Surgery: Engineering, Clinical and Business Challenges
[Gerald Moses, Charles Doarn, Blake Hannaford, Jacob Rosen] 293

Calibration of a Computer Assisted Orthopaedic Hip Surgery Phantom
[Daniel Sawyer, Nick Dagalakis, Craig Shakarji, Yong Kim] *297*

HLPR Chair – A Novel Patient Transfer Device
[Roger Bostelman, James Albus, Joshua Johnson] *302*

Robotic Navigation in Crowded Environments: Key Challenges for Autonomous Navigation Systems
[James Ballantyne, Salman Valibeik, Ara Darzi, Guang-Zhong Yang] *306*

*Note: * Presentation Only*

FOREWORD

The 2008 Performance Metrics for Intelligent Systems (PerMIS'08) Workshop is the eighth in the series that started in 2000, targeted at defining measures and methodologies of evaluating performance of intelligent systems. The workshop has proved to be an excellent forum for discussions and partnerships, dissemination of ideas, and future collaborations in an informal setting. Attendees usually include researchers, graduate students, practitioners from industry, academia, and government agencies.

PerMIS'08 aims at *identifying and quantifying contributions of functional intelligence towards achieving success*. Our working definition of functional intelligence is "the ability to act appropriately in an uncertain environment, where appropriate action is that which increases the probability of success", and success is "the achievement of behavioral goals" (J. Albus, 1991). In addition to the main theme, as in previous years, the workshop focuses on applications of performance measures to practical problems in commercial, industrial, homeland security, and military applications. Topic areas include, but are not limited to:

Defining and measuring aspects of a system:

- The level of autonomy
- Human-robot interaction
- Collaboration & coordination
- Taxonomies
- Biologically inspired models

Evaluating components within intelligent system

- Sensing and perception
- Knowledge representation, world models, ontologies
- Planning and control
- Learning and adaption
- Reasoning

Infrastructural support for performance evaluation

- Testbeds and competitions for intercomparisons
- Instrumentation and other measurement tools
- Simulation and modeling support

Technology readiness measures for intelligent systems

Applied performance measures in various domains, e.g.,

- Intelligent transportation systems
- Emergency response robots (search and rescue, bomb disposal)
- Homeland security systems
- De-mining robots
- Defense robotics
- Hazardous environments (e.g., nuclear remediation)
- Industrial and manufacturing systems
- Space/Aerial robotics
- Medical Robotics & assistive devices

PerMIS'08 features five plenary addresses and seven special sessions. The plenary speakers are world-class experts in their own field and we are confident that the attendees will be able to benefit from their presentations. This year, there is a special session for every (parallel) general session. Over the course of three days, there will be twelve sessions related to performance of intelligent systems covering an array of topics from medical systems to manufacturing, mobile robotics to virtual automation, human-system interaction to biologically inspired models, and much more.

Special thanks are due to the Program Committee for publicizing the workshop, the special session organizers for proposing interesting topics and bringing together researchers related to their sessions, and the reviewers who provided feedback to the authors, and helped us to assemble an excellent program. We much appreciate the authors submitting their papers to this workshop and for sharing their thoughts and experiences related to their research with the workshop attendees.

PerMIS'08, is sponsored by NIST with technical co-sponsorship of the IEEE Washington Section Robotics and Automation Society Chapter and in-cooperation with the Association for Computing Machinery (ACM) Special Interest Group on Artificial Intelligence (SIGART). As in previous years, the proceedings of PerMIS will be indexed by INSPEC, Compendex, ACM's Digital Library, and are released as a NIST Special Publication. Springer Publishers was back again this year to raffle off some of the books that were displayed at their booth during the course of the workshop. Selected papers from this workshop are being included in an edited book volume by Springer. We gratefully acknowledge the support of our sponsors.

We sincerely hope that you enjoy the presentations and the social programs!

Raj Madhavan
Program Chair

Elena Messina
General Chair

SPONSORS

NIST
National Institute of Standards and Technology

IEEE

acm

PROGRAM COMMITTEE

General Chair:

Elena Messina (Intelligent Systems Division, NIST, USA)

Program Chair:

Raj Madhavan (Oak Ridge National Laboratory/NIST, USA)

S. Balakirsky (NIST USA)
R. Bostelman (NIST USA)
F. Bonsignorio (Heron Robots Italy)
G. Berg-Cross (EM & I USA)
J. Bornstein (ARL USA)
P. Courtney (PerkinElmer UK)
J. Evans (USA)
D. Gage (XPM Tech. USA)
J. Gunderson (GammaTwo USA)
L. Gunderson (GammaTwo USA)
S. K. Gupta (UMD USA)
A. Jacoff (NIST USA)
S. Julier (Univ. College London UK)
M. Lewis (UPitt USA)
R. Lakaemper (Temple Univ USA)
T. Kalmar-Nagy (Texas A & M USA)
A. del Pobil (Univ. Jaume-I Spain)
S. Ramasamy UALR USA)
L. Reeker (NIST USA)
C. Schlenoff (NIST USA)
M. Shneier (NIST USA)
E. Tunstel (JHU-APL USA)

PLENARY SPEAKER

Mr. Alan Schultz

Navy Center for Applied Research in Artificial Intelligence (NCARAI), USA

Cognitively Enhanced Intelligent Systems

Tues. 8:30 am

ABSTRACT

We hypothesize that adding computational cognitive reasoning components to intelligent systems will result in three benefits:
Most if not all intelligent systems must interact with humans, who are the ultimate users of these systems. Giving the system cognitive models can enhance the human-system interface by allowing more common ground in the form of cognitively plausible representations and qualitative reasoning. By using cognitive models, reasoning mechanisms and representations, we believe that we can yield a more effective and efficient interface that accommodates the user.

Since the resulting system in interacting with the human, giving it behaviors that are more natural to the human can also result in more natural interactions between the human and the intelligent system. For example, mobile robots that must work collaboratively with humans can actually result in less effective interactions if its behaviors are alien or non-intuitive to the human. By incorporating cognitive models, we can develop systems whose behavior is more expected and natural.

One key interest is in measuring the performance of intelligent systems. We propose that an intelligent system that is cognitively enhanced can be more directly compared to human level performance. Further, if cognitive models of human performance have been developed in creating the intelligent system, we can directly compare the intelligent systems behavior and performance in the task to the human subject behavior and performance.

In this talk, I will present several instantiations of developing cognitively enhanced intelligent systems.

BIOGRAPHY

Alan C. Schultz is the Director of the Navy Center for Applied Research in Artificial Intelligence at the Naval Research Laboratory in Washington, DC. His research is in the areas of human-robot interaction, cognitive robotics, evolutionary robotics, learning in robotic systems, and adaptive systems. He is the recipient of an Alan Berman Research Publication Award, and has published over 90 articles on HRI, machine learning and robotics. Alan is currently the co-chair of the AAAI Symposia Series, and chaired the 1999 and 2000 AAAI Mobil Robot Competition and Exhibitions.

PLENARY SPEAKER

Prof. Allison Okamura

The Johns Hopkins University, USA

Haptics in Medical Robotics: Surgery, Simulation, and Rehabilitation

Tues. 2:00 pm

ABSTRACT

Haptics is the science and technology of experiencing and creating touch sensations. This talk will examine the role of haptics in three types of medical systems: surgical robotics, surgical simulators, and rehabilitation robotics. Robot-assisted surgery can improve the outcomes of medical procedures by enhancing accuracy and minimally invasive access, thereby reducing patient trauma and recovery time. However, the current lack of force and tactile information is hypothesized to compromise system performance. With approaches ranging from psychophysical studies to control systems engineering, we are designing teleoperated robots capable of providing haptic feedback in challenging surgical environments. Haptic information is also needed for accurate surgical simulation. Surgical simulators present a safe and potentially effective method for surgical training, and can also be used in robot-assisted surgery for pre- and intra-operative planning. I will describe experiments to determine the mechanics of interaction between surgical instruments and tissues, as well as techniques for accurate patient-specific modeling. Finally, rehabilitation through robotically enabled orthotics and prosthetics inherently requires understanding and appropriate generation of haptic interactions. Our recent work in this area includes motor control augmentation with an exoskeleton robot, and studies of the role of haptic proprioception in prosthetic limb use.

BIOGRAPHY

Allison M. Okamura received the BS degree from the University of California at Berkeley in 1994, and the MS and PhD degrees from Stanford University in 1996 and 2000, respectively, all in mechanical engineering. She is currently an associate professor of mechanical engineering and the Decker Faculty Scholar at Johns Hopkins University. She is associate director of the Laboratory for Computational Sensing and Robotics and a thrust leader of the NSF Engineering Research Center for Computer-Integrated Surgical Systems and Technology. Her awards include the 2005 IEEE Robotics Automation Society Early Academic Career Award, the 2004 US NSF CAREER Award, the 2004 JHU George E. Owen Teaching Award, and the 2003 JHU Diversity Recognition Award. Her research interests are haptics, teleoperation, medical robotics, virtual environments and simulators, prosthetics, rehabilitation engineering, and engineering education.

PLENARY SPEAKER

Prof. Erwin Prassler

Applied Science Institute, Germany

Incremental Integration, Evaluation, and Harmonization of Components of a Reference Platform for Service Robotics

Wed. 8:30 am

ABSTRACT

In industrial robotics, system integration is a rather common business. Robot manufacturers typically team up with a number of so-called system integrators, which design robot cells, assembly lines and entire manufacturing plants out of "standardized" components, such as manipulators, sensors, tools, and conveyor systems.

In service robotics the situation is in no way comparable. Service robots are typically considered as mass products, which are designed like dish washers or play stations. System integration is simply part of the regular product design.

It would be rather irrelevant to discuss this issue any further, if the design of a service robot for some specific application was a task like the design of a dish washer. As a matter of fact, the two tasks have not much if anything at all in common.

The design of a service robot is more the result of the ingenuity of an engineer rather of established procedures or methodologies or even technologies. Typically every new service robot is design from scratch. Not too seldom, the service task itself and the operational constraints are not too well understood, neither is the business model under which the automation of a service could become an economic success. A plenitude of components such as sensors, actuators, operating systems, algorithms are available but no common recipe for integrating and compiling them into a competitive product.

Service robotics today is in a situation very much comparable to the situation of the car industry in 1885, when Carl Benz built the first car. The industry is virtually not existing. Potential players and investors are skeptical because not only a realistic market but also a realistic technology assessment gives them a rather fuzzy picture.

This situation has motivated the German Ministry for Education and Research to invest into a so-called technology platform for service robotics. Other funding agencies such as the European Commission are implementing similar initiatives.

In my presentation I will talk about the German Service Robotics Initiative, which as a major activity pushes the development of such a technology platform. The platform is considered as a vehicle for understanding and managing the requirements for system integration in service robotics. I will talk about a first approach of this Initiative to incrementally integrate, evaluate and harmonize available off the shelf components and their interfaces to simplify and accelerate the development of new service robots. I will also talk about the lessons learned in this Initiative and how they are currently being picked up in other initiatives to promote the development of harmonized and/or standardized building blocks for service robots.

BIOGRAPHY

Erwin Prassler received a master's degree in Computer Science from the Technical University of Munich in 1985 and a Ph.D. in Computer Science from the University of Ulm in March 1996. For his doctoral dissertation he received the AKI dissertation award in September 1997. Between 1986 and 1989, Dr. Prassler held positions as a member of the scientific staff at the Technical University of Munich and as a guest researcher in the Computer Science Department at the University of Toronto. In fall 1989, he joined the Research Institute for Applied Knowledge Processing (FAW) in Ulm, where he headed a research group working in the field of mobile robots and service robotics between 1994 and 2003. In 1999, Dr. Prassler entered a joint affiliation with Gesellschaft fur Produktionssysteme (GPS) in Stuttgart, where directed the department for Project Management and Technology Transfer. In this function, Dr. Prassler coordinated the MORPHA project (Interaction and Communication between Humans and Intelligent Robot Assistents, www.morpha.de) one of six national research projects in the field of Human Machine Interaction funded by the German Ministry for Education and Research. In March 2004, Dr. Prassler was appointed as an Associate Professor at the Bonn-Aachen International Center for Information Technology. Together with Prof. Rolf Dieter Schraft, director of Fraunhofer IPA in Stuttgart, he is currently co-ordinating the German Service Robotice Initiatve DESIRE (www.service-robotik-initiative.de), a joint national research project involving 7 academic and 6 industrial partners.

PLENARY SPEAKER

Prof. Alonzo Kelly

Carnegie Mellon University, USA

Various Tradeoffs and Metrics of Performance for Field Robots

Wed. 2:00 pm

ABSTRACT

A mature systems engineering discipline is exemplified by aerospace engineering where purpose-built vehicles are designed while regularly consulting system level performance models to help guide the design optimization process. Robotics has not yet identified such rich and universal performance models but useful performance models do arise naturally in the performance of the work. This talk will discuss a large number of field robotic systems in an attempt to identify some system level constraints, tradeoffs and metrics which seem to be valuable in formulating the quest for an optimal system. Examples include the hard constraints of safe real-time replanning, the optimal update rate of a visual servo, the related tradeoff between systematic and random error accumulation in mapping, and the relative completeness of planning search spaces and its affect on winning robot races.

BIOGRAPHY

Dr. Alonzo Kelly is an associate professor at the Robotics Institute of Carnegie Mellon University. He has also worked as a member of the technical staff at MD Robotics, Canada and at NASA's Jet Propulsion Laboratory. His research typically concerns wheeled mobile robots operating in both structured and unstructured environments. His work spans many sub-specialties of mobile robots including control, position estimation, mapping, motion planning, simulation, and human interfaces. It also spans many application areas including outdoor unmanned ground vehicles, agricultural and mining vehicles, planetary rovers, and indoor automated guided vehicles.

PLENARY SPEAKER

Prof. Sunil Kumar Agrawal

University of Delaware, USA

Robotic Exoskeletons for Gait Assistance and Training of the Motor Impaired

Thurs. 8:30 am

ABSTRACT

Robotics is emerging as a promising tool for training of human functional movement. The current research in this area is focused primarily on upper extremity movements. This talk describes novel designs of three lower extremity exoskeletons, intended for gait assistance and training of motor-impaired patients. The design of each of these exoskeletons is novel and different. Force and position sensors on the exoskeleton provide feedback to the user during training. The exoskeletons have undergone tests on healthy and chronic stroke survivors to assess their potential for treadmill training. These results will be presented. GBO is a Gravity Balancing un-motorized Orthosis which can alter the gravity acting at the hip and knee joints during swing. ALEX is an Actively driven Leg Exoskeleton which can modulate the foot trajectory using motors at the joints. SUE is a bilateral Swing-assist Un-motorized Exoskeleton to propel the leg during gait. This research was supported by NIH through a BRP program.

BIOGRAPHY

Prof. Agrawal received a Ph.D. degree in Mechanical Engineering from Stanford University in 1990. He is currently the Director of Mechanical Systems Laboratory. He has published close to 200 journal and conference papers and 2 books in the areas of controlled mechanical systems, dynamic optimization, and robotics. Dr. Agrawal is a Fellow of the ASME and his other honors include a Presidential Faculty Fellowship from the White House in 1994, a Bessel Prize from Germany in 2003, and a Humboldt US Senior Scientist Award in 2007. He has served on editorial boards of numerous journals published by ASME and IEEE.

August 19

PROGRAM

PERMIS

TUESDAY

08:15	**Welcome & Overview**
08:30	**Plenary Presentation:** **Alan Schultz** ***Cognitively Enhanced Intelligent Systems***
09:30	**Coffee Break**
10:00	**TUE-AM1 Performance Evaluation** ***Chairs: James Spall & Craig Schlenoff*** • Evolution of the SCORE Framework to Enhance Field-Based Performance Evaluations of Emerging Technologies [Brian Weiss, Craig Schlenoff] • Reliability Estimation and Confidence Regions from Subsystem and Full System Tests via Maximum Likelihood [James Spall] • Fuzzy-Logic-Based Approach for Identifying Objects of Interest in the PRIDE Framework [Zeid Kootbally, Craig Schlenoff, Raj Madhavan, Sebti Foufou] • Identifying Objects in Range Data Based on Similarity Transformation Invariant Shape Signatures [Xiaolan Li, Afzal Godil, Asim Wagan] • Stepfield Pallets: Repeatable Terrain for Evaluating Robot Mobility [Adam Jacoff, Anthony Downs, Ann Virts, Elena Messina] • Potential Scaling Effects for Asynchronous Video in Multirobot Search [Prasanna Velagapudi, Paul Scerri, Katia Sycara, Huadong Wang, Michael Lewis]
12:30	**Lunch**
14:00	**Plenary Presentation:** **Allison Okamura** ***Haptics in Medical Robotics: Surgery, Simulation, and Rehabilitation***
15:00	**Coffee Break**
15:30	**TUE-PM1 Human-System Interaction** ***Chairs: Michael Lewis & Birsen Donmez*** • Evaluation Criteria for Human-Automation Performance Metrics [Birsen Donmez, Patricia Pina, Mary Cummings] • Assessing Measures of Coordination Demand Based on Interaction Durations [Michael Lewis, Jijun Wang] • The Gestural Joystick and the Efficacy of the Path Tortuosity Metric for Human/Robot Interaction [Richard Voyles, Jaewook Bae, Roy Godzdanker] • Modeling of Thoughtful Behavior with Dynamic Expert System [Vadim Stefanuk]
19:00	**Reception**

PERMIS PROGRAM

August 19 TUESDAY

08:15	**Welcome & Overview**
08:30	**Plenary Presentation:** **Alan Schultz** ***Cognitively Enhanced Intelligent Systems***
09:30	**Coffee Break**
10:00	**TUE-AM2 Special Session I: Cognitive Systems of EU Cognition Programme** ***Organizer: Patrick Courtney*** • Cognitive Systems of EU Cognition Programme* [Patrick Courtney] • The Rat's Life Benchmark: Competing Cognitive Robots [Olivier Michel, Fabien Rohrer] • The iCub Humanoid Robot: An Open Platform for Research in Embodied Cognition [Giorgio Metta, Giulio Sandini, David Vernon, Lorenzo Natale, Francesco Nori] • An Open-Source Simulator for Cognitive Robotics Research: The Prototype of the iCub Humanoid Robot Simulator [Vadim Tikhanoff, Angelo Cangelosi, Paul Fitzpatrick, Giorgio Metta, Lorenzo Natale, Francesco Nori] • Symbiotic Robot Organisms: REPLICATOR and SYMBRION Projects [Serge Kernbach, Eugen Meister, Florian Schlachter, Kristof Jebens, Marc Szymanski, Jens Liedke, Davide Laneri, Lutz Winkler, Thomas Schmickl, Ronald Thenius, Paolo Corradi, Leonardo Ricotti] • Virtual Agent Modeling in the RASCALLI Platform [Christian Eis, Marcin Skowron, Brigitte Krenn]
12:30	**Lunch**
14:00	**Plenary Presentation:** **Allison Okamura** ***Haptics in Medical Robotics: Surgery, Simulation, and Rehabilitation***
15:00	**Coffee Break**
15:30	**TUE-PM2 Special Session II: Architectures for Unmanned Systems** ***Organizers: Roger Bostelman & James Albus*** • UAV Architectures* [George Vachtsevanos] • Architectures for Unmanned Systems* [James Albus] • Levels-of-Autonomy of the ASTM F41 Unmanned Maritime Vehicles Standard* [Mark Rothgeb] • Ontological Perspectives for Autonomy Performance [Hui-Min Huang, Elena Messina, Tsai Hong, Craig Schlenoff]
19:00	**Reception**

*Presentation Only

August

PROGRAM

PERMIS

20

WEDNESDAY

08:15	Overview
08:30	**Plenary Presentation:** **Erwin Prassler** ***Incremental Integration, Evaluation, and Harmonization of Components of a Reference Platform for Service Robotics***
09:30	Coffee Break
10:00	**WED-AM1 Metrics & Measures** ***Chairs: Scott Spetka & Robert Wade*** • Robotic Systems Technical and Operational Metrics Correlation [Jason Schenk, Robert Wade] • Survey of Domain-Specific Performance Measures in Assistive Robotic Technology [Katherine Tsui, Holly Yanco, David Feil-Seifer, Maja Mataric] • Refining the Cognitive Decathlon [Robert Simpson, Charles Twardy] • Using Metrics to Optimize a High Performance Intelligent Image Processing Code [Scott Spetka, Susan Emeny, George Ramseyer, Richard Linderman] • Measurement Techniques for Multiagent Systems [Robert Lass, Evan Sultanik, William Regli] • RoboCupRescue Robot League: 2008 Overview* [Adam Jacoff, Andreas Birk, Johannes Pellenz, Ehsan Mihankhah, Raymond Sheh, Satoshi Tadokoro]
12:30	Lunch
14:00	**Plenary Presentation:** **Alonzo Kelly** ***Various Tradeoffs and Metrics of Performance for Field Robots***
15:00	Coffee Break
15:30	**WED-PM1 Autonomous Systems** ***Chairs: James Gunderson & Edward Tunstel*** • Integrating Reification and Ontologies for Mobile Autonomous Robots [James Gunderson, Louise Gunderson] • Quantification of Line Tracking Solutions for Automotive Applications [Jane Shi, Rick Rourke, Dave Groll, Peter Tavora] • Mobile Robotic Surveying Performance for Planetary Surface Site Characterization [Edward Tunstel] • Evaluating Situation Awareness of Autonomous Systems [Jan Gehrke]
18:30	Banquet

*Presentation Only

08:15	**Overview**
08:30	**Plenary Presentation:** **Erwin Prassler** ***Incremental Integration, Evaluation, and Harmonization of Components of a Reference Platform for Service Robotics***
09:30	**Coffee Break**
10:00	**WED-AM2 Special Session III: Performance Metrics for Perception in Intelligent Manufacturing** ***Organizers: Tsai Hong & Roger Eastman*** • Performance of Super-Resolution Enhancement for Flash LADAR Data [Shuowen Hu, Susan Young, Tsai Hong] • Performance Evaluation of Laser Trackers [Bala Muralikrishnan, Daniel Sawyer, Christopher Blackburn, Steven Phillips, Bruce Borchardt, Tyler Estler] • Preliminary Analysis of Conveyor Dynamic Motion for Automation Applications [Jane Shi] • 3D Part Identification Based on Local Shape Descriptors [Xiaolan Li, Afzal Godil, Asim Wagan] • Calibration of a System of a Gray-Value Camera and an MDSI Range camera [Tobias Hanning, Aless Lasaruk] • Dynamic 6DOF Metrology for Evaluating a Visual Servoing System [Tommy Chang, Tsai Hong, Mike Shneier, German Holguin, Johnny Park, Roger Eastman]
12:30	**Lunch**
14:00	**Plenary Presentation:** **Alonzo Kelly** ***Various Tradeoffs and Metrics of Performance for Field Robots***
15:00	**Coffee Break**
15:30	**WED-PM2 Special Session IV: Results from a Virtual Manufacturing Automation Competition** ***Organizers: Stephen Balakirsky, Raj Madhavan & Chris Scrapper*** • NIST/IEEE Virtual Manufacturing and Automation Competition: From Earliest Beginnings to Future Directions [Stephen Balakirsky, Raj Madhavan, Chris Scrapper] • Analysis of a Novel Docking Technique for Autonomous Robots [George Henson, Michael Maynard, Xinlian Liu, George Dimitoglou] • Partitioning Algorithm for Path Determination of Automated Robotic Part Delivery System in Manufacturing Environments [Payam Matin, Ali Eydgahi, Ranjith Chowdary] • Algorithms and Performance Analysis for Path Navigation of Ackerman-Steered Autonomous Robots [George Henson, Michael Maynard, George Dimitoglou, Xinlian Liu]
18:30	**Banquet**

August

PROGRAM

PERMIS

THURSDAY 21

08:15	Overview
08:30	**Plenary Presentation:** **Sunil Kumar Agrawal** ***Robotic Exoskeletons for Gait Assistance and Training of the Motor Impaired***
09:30	Coffee Break
10:00	**THU-AM1 Model-based Performance Assessment** ***Chairs: Kate Remley & Kam Saidi*** • Wireless Communications in Tunnels for Urban Search and Rescue Robots [Kate Remley, George Hough, Galen Koepke, Dennis Camell, Robert Johnk, Chriss Grosvenor] • A Performance Assessment of Calibrated Camera Networks for Construction Site Monitoring [Itai Katz, Nicholas Scott, Kam Saidi] • A Queuing-Theoretic Framework for Modeling and Analysis of Mobility in WSNs [Harsh Bhatia, Rathinasamy Lenin, Aarti Munjal, Srini Ramaswamy, Sanjay Srivastava] • Towards Information Networks to Support Composable Manufacturing [Mahesh Mani, Albert Jones, Junho Shin, Ram Sriram] • 3D Reconstruction of Rough Terrain for USARSim using a Height-map Method [Gael Roberts, Stephen Balakirsky, Sebti Foufou]
12:30	Lunch
14:00	**THU-PM1 Special Session VI: Biologically Inspired Models of Intelligent Systems** ***Organizer: Gary Berg-Cross*** • Introduction to Biological Inspiration for Intelligent Systems [Gary Berg-Cross] • Overview of Biologically Inspired Cognitive Architectures (BICA)* [Alexei Samsonovich] • Recent modeling and Rapid Prototyping Experience Aimed at Building Architectures of Cognitive Agents* [Giorgio Ascoli] • Applying Developmental-Inspired Principles to the Field of Developmental Robotics [Gary Berg-Cross] • Discussion of Biologically Inspired Models [Panel]
16:00	Coffee Break
16:30	Adjourn

*Presentation Only

PERMIS PROGRAM

August 7 THURSDAY

08:15	Overview
08:30	**Plenary Presentation:** **Sunil Kumar Agrawal** ***Robotic Exoskeletons for Gait Assistance and Training of the Motor Impaired***
09:30	Coffee Break
10:00	**THU-AM2 Special Session V: Quantitative Assessment of Robot-generated Maps** ***Organizers: Chris Scrapper, Raj Madhavan & Stephen Balakirsky*** • Characterizing Robot-Generated Maps: The Importance of Representations and Objective Metrics* [Chris Scrapper, Raj Madhavan, Stephen Balakirsky] • Using Virtual Scans to Improve Alignment Performance in Robot Mapping [Rolf Lakeamper, Nagesh Adluru] • The Role of Bayesian Bounds in Comparing SLAM Algorithms Performance [Andrea Censi] • Map Quality Assessment [Asim Wagan, Afzal Godil, Xiaolan Li] • Discussion: Roadmap for Map Evaluation Frameworks
12:30	Lunch
14:00	**THU-PM2 Special Session VII: Medical Robotics** ***Organizer: Ram Sriram*** • Overcoming Barriers to Wider Adoption of Mobile Telerobotic Surgery: Engineering, Clinical and Business Challenges [Gerald Moses, Charles Doarn, Blake Hannaford, Jacob Rosen] • Calibration of a Computer Assisted Orthopaedic Hip Surgery Phantom [Daniel Sawyer, Nick Dagalakis, Craig Shakarji, Yong Kim] • HLPR Chair – A Novel Patient Transfer Device [Roger Bostelman, James Albus, Joshua Johnson] • Robotic Navigation in Crowded Environments: Key Challenges for Autonomous Navigation Systems [James Ballantyne, Salman Valibeik, Ara Darzi, Guang-Zhong Yang]
16:00	Coffee Break
16:30	Adjourn

*Presentation Only

AUTHOR INDEX

Albus, J. THU-PM2
Adluru, N. THU-AM2
Ascoli, G. THU-PM1
Bae, J. TUE-PM1
Balakirsky, S. WED-PM2
.....THU-AM1
.....THU-AM2
Ballantyne, J. THU-PM2
Berg-Cross, G. THU-PM1
.....THU-PM1
Bhatia, H. THU-AM1
Birk, A. WED-AM1
Blackburn, C. WED-AM2
Borchardt, B. WED-AM2
Bostelman, R. THU-PM2
Camell, D. THU-AM1
Cangelosi, A. TUE-AM2
Censi, A. THU-AM2
Chang, T. WED-AM2
Chowdary, R. WED-PM2
Corradi, P. TUE-AM2
Courtney, P. TUE-AM2
Cummings, M. TUE-PM1
Dagalakis, N. THU-PM2
Darzi, A. THU-PM2
Dimitoglou, G. WED-PM2
.....WED-PM2
Doarn, C. THU-PM2
Donmez. B. TUE-PM1
Downs, A. TUE-AM1
Eastman, R. WED-AM2
Eis, C. TUE-AM2
Emeny, S. WED-AM1
Estler, T. WED-AM2
Eydgahi, A. WED-PM2
Feil-Seifer, D. WED-AM1
Fitzpatrick, P. TUE-AM2
Foufou, S. TUE-AM1
.....THU-AM1
Gehrke, J. WED-PM1
Godil, A. TUE-AM1
.....WED-AM2
.....THU-AM2
Godzdanker, R. TUE-PM1
Groll, D. WED-PM1
Grosvenor, C. THU-AM1
Gunderson, J. WED-PM1
Gunderson, L. WED-PM1
Hannaford, B. THU-PM2
Hanning, T. WED-AM2
Henson, G. WED-PM2
.....WED-PM2
Holguin, G. WED-AM2
Hong, T. TUE-PM2
.....WED-AM2
.....WED-AM2
Hough, G. THU-AM1
Hu, S. WED-AM2
Huang, H-M. TUE-PM2
Jacoff, A. TUE-AM1
.....WED-AM1
Jebens, K. TUE-AM2
Johnk, R. THU-AM1
Johnson, J. THU-PM2
Jones, A. THU-AM1
Katz, I. THU-AM1
Kernbach, S. TUE-AM2
Kim, Y. THU-PM2
Koepke, G. THU-AM1
Kootbally, Z. TUE-AM1
Krenn, B. TUE-AM2
Lakeamper, R. THU-AM2
Laneri, D. TUE-AM2
Lasaruk, A. WED-AM2
Lass, R. WED-AM1
Lenin, R. THU-AM1
Lewis, M. TUE-AM1
.....TUE-PM1
Li, X. TUE-AM1
.....WED-AM2
.....THU-AM2
Liu, X. WED-PM2
.....WED-PM2
Liedke, J. TUE-AM2
Linderman, R. WED-AM1
Madhavan, R. TUE-AM1
.....WED-PM2
.....THU-AM2
Mani, M. THU-AM1
Mataric, M. WED-AM1
Matin, P. WED-PM2
Maynard, M. WED-PM2
.....WED-PM2
Meister, E. TUE-AM2
Messina, M. TUE-AM1
.....TUE-PM2
Metta, G. TUE-AM2
.....TUE-AM2
Michel, O. TUE-AM2
Mihankhah, E. WED-AM1
Moses, G. THU-PM2
Munjal, A. THU-AM1
Muralikrishnan, B. ..WED-AM2
Natale, L. TUE-AM2
.....TUE-AM2
Nori, F. TUE-AM2
.....TUE-AM2
Park, J. WED-AM2
Pellenz, J. WED-AM1
Phillips, S. WED-AM2
Pina, P. TUE-PM1
Ramaswamy, S. THU-AM1
Ramseyer, G. WED-AM1
Regli, W. WED-AM1
Remley, K. THU-AM1
Ricotti, L. TUE-AM2
Roberts, G. THU-AM1
Rohrer, F. TUE-AM2
Rosen, J. THU-PM2
Rothgeb, M. TUE-PM2
Rourke, R. WED-PM1
Saidi, K. THU-AM1
Samsonovich, A. THU-PM1
Sandini, G. TUE-AM2
Sawyer, D. WED-AM2
.....THU-PM2
Scerri, P. TUE-AM1
Schenk, J. WED-AM1
Schlachter, F. TUE-AM2
Schlenoff, C. TUE-AM1
.....TUE-AM1
.....TUE-PM2
Schmickl, T. TUE-AM2
Scott, N. THU-AM1
Scrapper, C. WED-PM2
.....THU-AM2
Shakarji, C. THU-PM2
Sheh, R. WED-AM1
Shi, J. WED-AM2
.....WED-PM1
Shin, J. THU-AM1

Shneier, M. WED-AM2
Simpson, R. WED-AM1
Skowron, M. TUE-AM2
Spall, J. TUE-AM1
Spetka, S. WED-AM1
Sriram, R. THU-AM1
Srivastava, S. THU-AM1
Stefanuk, V. TUE-PM1
Sultanik, E. WED-AM1
Sycara, K. TUE-AM1
Szymanski, M. TUE-AM2
Tadokoro, S. WED-AM1
Tavora, P. WED-PM1
Thenius, R. TUE-AM2
Tikhanoff, V. TUE-AM2
Tsui, K. WED-AM1
Tunstel, E. WED-PM1
Twardy, C. WED-AM1
Vachtsevanos, G. TUE-PM2
Valibeik, S. THU-PM2
Velagapudi, P. TUE-AM1
Vernon, D. TUE-AM2
Virts, A. TUE-AM1
Voyles, R. TUE-PM1
Wade, R. WED-AM1
Wagan, A. TUE-AM1
.....WED-AM2
.....THU-AM2
Wang, H. TUE-AM1
Wang, J. TUE-PM1
Weiss. B. TUE-AM1
Winkler, L. TUE-AM2
Yanco, H. WED-AM1
Yang, G-Z. THU-PM2
Young, S. WED-AM2

ACKNOWLEDGMENTS

These people provided essential support to make this event happen. Their ideas and efforts are very much appreciated.

Website and Proceedings

Debbie Russell

Local Arrangements

Jeanenne Salvermoser

Conference and Registration

Mary Lou Norris

Kathy Kilmer

Angela Ellis

Teresa Vicente

Intelligent Systems Division
Manufacturing Engineering Laboratory
National Institute of Standards and Technology
100 Bureau Drive, MS-8230
Gaithersburg, MD 20899
http://www.isd.mel.nist.gov/

Evolution of the SCORE Framework to Enhance Field-Based Performance Evaluations of Emerging Technologies

Brian A. Weiss and Craig Schlenoff
National Institute of Standards and Technology
100 Bureau Drive, MS 8230
Gaithersburg, Maryland 20899 USA
+1.301.975.4373, +1.301.975.3456
brian.weiss@nist.gov, craig.schlenoff@nist.gov

ABSTRACT
NIST has developed the System, Component, and Operationally-Relevant Evaluations (SCORE) framework as a formal guide for designing evaluations of emerging technologies. SCORE captures both technical performance and end-user utility assessments of systems and their components within controlled and realistic environments. Its purpose is to present an extensive (but not necessarily exhaustive) picture of how a system would behave in a realistic operating environment. The framework has been applied to numerous evaluation efforts over the past three years producing valuable quantitative and qualitative metrics. This paper will present the building blocks of the SCORE methodology including the system goals and design criteria that drive the evaluation design process. An evolution of the SCORE framework in capturing utility assessments at the capability level of a system will also be presented. Examples will be shown of SCORE's successful application to the evaluation of the soldier-worn sensor systems and two-way, free-form spoken language translation technologies.

Categories and Subject Descriptors
C.4 [**Performance of Systems**]: *measurement techniques, modeling techniques, performance attributes.*

General Terms
Measurement, Documentation, Performance, Experimentation, Verification.

Keywords
SCORE, DARPA, ASSIST, TRANSTAC, performance evaluation, elemental tests, vignette tests, task tests, speech translation, soldier-worn sensor.

1. INTRODUCTION
As intelligent systems emerge and take shape, it is important to understand their capabilities and limitations. Evaluations are a means to assess both quantitative technical performance and qualitative end-user utility. System, Component and Operationally Relevant Evaluations (SCORE) is a unified set of criteria and software tools for defining a performance evaluation approach for intelligent systems. It provides a comprehensive evaluation blueprint that assesses the technical performance of a system and its components through isolating and changing variables as well as capturing end-user utility of the system in realistic use-case environments. SCORE is unique in that:

- It is applicable to a wide range of technologies, from manufacturing to defense systems
- Elements of SCORE can be decoupled and customized based upon evaluation goals
- It has the ability to evaluate a technology at various stages of development, from conceptual to full maturation
- It combines the results of targeted evaluations to produce an extensive picture of a systems' capabilities and utility

Section 2 introduces the SCORE framework and its initial evaluation design structure. Section 3 presents SCORE's first applications in evaluating technologies developed under the Defense Advanced Research Projects Agency's (DARPA) Advanced Soldier Sensor Information System and Technology (ASSIST), Phase I and II program along with DARPA's Spoken Language Communication and Translation System for Tactical Use (TRANSTAC) Phase II program. Section 4 discusses the evolution of the framework necessitated by the advancing goals of the ASSIST and TRANSTAC programs. Section 5 describes some future efforts (outside of the above military-based programs) that are expected to use the SCORE framework. Section 6 concludes the paper.

2. BACKGROUND
2.1 SCORE Development
Intelligent systems tend to be complex and non-deterministic, involving numerous components that are jointly working together to accomplish an overall goal. Existing approaches to measuring such systems often focus on evaluating the system as a whole or individually evaluating some of the components under very controlled, but limited, conditions. These approaches do not comprehensively and quantitatively assess the impact of variables such as environmental variables (e.g, weather) and system variables (e.g., processing power, memory size) on the system's overall performance. The SCORE framework, with its comprehensive evaluation criteria and software tools, is developed to enhance the ability to quantitatively and qualitatively evaluate intelligent systems at the component level -- and the system level -- in both controlled and operationally-relevant environments.

SCORE leverages the multi-level Steves/Scholtz evaluation framework that defines metrics and measures in the context of

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
ACM ISBN 978-1-60558-293-1/08/08.

system goals and evaluation objectives, and combines these assessments for an overall evaluation of a system [1]. SCORE takes the framework a step further by identifying specific system goals and areas of interest. It is built around the premise that, in order to get a comprehensive picture of how a system performs in its actual use-case environment, technical performance should be evaluated at the component and system levels [2]. Additionally, system level utility assessments should be performed to gain an understanding of the value the system provides to the end-users. SCORE defines three evaluation goal types:

- *Component Level Testing – Technical Performance* – This evaluation type involves decomposing a system into components to isolate those subsystems that are critical to system operation. Ideally, all of the components together, should include all facets of the system and yield a complete evaluation.
- *System Level Testing – Technical Performance* – This evaluation type is intended to assess the system as a whole, but in an ideal environment where test variables can be isolated and controlled. The benefit is that tests can be performed using a combination of test variables and parameters, where relationships can be determined between system behavior and these variables and parameters based upon the technical performance analysis.
- *System Level Testing – Utility Assessments* – This evaluation class assesses a system's utility, where utility is defined as the value the application provides to the end-user. In addition, usability is assessed which includes effectiveness, learnability, flexibility, and user attitude towards the system. The advantage of this evaluation mode is that system's utility and value can still be addressed even when the system design and user-interface are not yet finalized (i.e. the working version in place is not perfected).

For each of these three goal types, the following evaluation elements are pertinent:

- Identification of the system or component to be assessed
- Definition of the goal/objective(s)/metrics/ measures
  - Goal – For a particular assessment, the goal is influenced by whether the intent of the evaluation is to inform or validate the system design. The state of system maturity also weighs heavily on the goal specification
  - Objectives – Evaluation objectives are used to separate evaluation concerns. These evaluation concerns also include identifying how different variables impact system performance and determining which should be fixed and which should be modified during testing.
  - Metrics/measures – Depending upon the type of evaluation, either technical performance metrics or utility metrics would be employed.
- Specification of the testing environment(s) – Selecting a testing environment is influenced by a range of aspects including system maturity, intended use-case environments, physical issues, site suitability, etc.
- Identification of participants – The system users, whether they are the technology developers and/or end-users needs to be determined. Actors that will be indirectly interacting with the system through role-playing within the environment also need to be identified.
- Specification of participant training – Technology users must be properly instructed (and have time to practice) on how to appropriately interact/engage the systems. Likewise, the environmental actors require guidance as to how they should perform throughout the test(s).
- Specification of data collection methods – As measures and metrics are specified, data capture methods must be formulated.
- Specification of the use-case scenarios – The evaluation architect must devise the use scenario(s) under which the system (or component) will be tested.

Considering each of these evaluation elements, SCORE takes a tiered approach to measuring the performance of intelligent systems. At the lowest level, SCORE uses component level tests to isolate specific components and then systematically modifies variables that could affect the performance of that component to determine those variables' impact. Typically, this is performed for each relevant component within the system. At the next level, the overall system is tested in a highly structured environment to understand the performance of individual variables on the system. Lastly, the technology is immersed in a richer scenario that evokes typical situations and surroundings in which the end-user is asked to perform an overall mission or procedure in a highly-relevant environment which stresses the overall system's capabilities. Formal surveys and semi-structured interviews are used to assess the usefulness of the technology to the end-user.

3. INITIAL APPLICATIONS

SCORE was initially applied to intelligent systems developed under the DARPA ASSIST and TRANSTAC programs. The SCORE-based evaluations also provided the researchers and end-users with the information needed to determine if and when the technology will be ready for actual use. The SCORE framework identified various key components of the system and evaluated them both independently and as a whole, thus helping to determine the impact of the individual components on the performance of the overall system. This detailed analysis allowed the evaluation team, and the sponsor, to more accurately target the aspects of the systems that were shown to provide the greatest benefit to the overall advancement of the technology. Prior to adopting SCORE, DARPA did not have this level of necessary detail about system and component performance.

3.1 ASSIST – Phase I and II

The DARPA ASSIST program is an advanced technology research and development program whose objective is to exploit soldier-worn sensors to augment a Soldier's mission recall and reporting capability to enhance situational knowledge within Military Operations in Urban Terrain (MOUT) environments [3]. This program is split into two tasks with the NIST Independent Evaluation Team (IET) focused on evaluating task 2 technology. This task stresses passive collection and automated activity/object recognition capabilities in the form of algorithms, software, and tools that will undergo system integration in future efforts.

The process of applying the SCORE framework to the ASSIST evaluations begins with identifying the specific technologies. The technologies were developed by three different research teams. It should be noted that there is no single, fully-integrated ASSIST system, so each team focused their attention on some unique and/or overlapping technologies. The Phase I and Phase II capabilities are broken out as follows:

- Image/Video Data Analysis Capabilities
  - Object Detection/ Image Classification (Phase I)
  - Arabic Text Translation (Phase I)
  - Face Recognition and Matching (Phase II)
- Audio Data Analysis Capabilities
  - Sound Recognition/Speech Recognition (Phase I)
  - Shot Localization/Weapon Classification (Phase I)
- Soldier Activity Data Analysis Capabilities
  - Soldier State Identification/Localization (Phase I and II)

Further explanation of these technologies can be found in [3] and [4]. The next crucial step is to determine the evaluation goals/ objectives and metrics/measures. As outlined by DARPA, at a high level, they are:

1. The accuracy of object/event/activity identification and labeling.
2. The system's ability to improve its classification performance through learning.
3. The utility of the system in enhancing operational effectiveness.

Guided by the SCORE framework, component and system level technical performance tests are developed to handle metrics 1 and 2, while system level utility assessments are designed to address the third metric. The quantitative performance tests are accomplished through elemental tests, while the qualitative tests are done through vignette tests.

3.1.1 Elemental Tests

This test type was used to measure technical performance at both the component and system levels [4]. Specifically, this test type afforded the designer the ability to place tight controls on the testing environment including modifying specific test variables in order to measure their impact on a technology's performance. The elemental tests that were developed across the ASSIST Phase I and II evaluations include:

- Arabic text translation – This test was designed to evaluate the Arabic text translation ability at both the component and system levels. Component level elemental tests include specific measurements of the technology's ability to 1) Identify Arabic text in an image, 2) Extract Arabic text from an image, and 3) Translate Arabic text to English text. The system level elemental test measured the technology's *start-to-finish* ability from capturing an image of Arabic text and to successfully translating the text into English.
- Face recognition and matching – Likewise, this elemental test evaluated the face recognition technology at the component and system levels. The component level test occurred in the form of an offline evaluation where test images of faces were directly fed into a computer running the matching algorithm and compared against a preloaded watchlist of images. Accuracy measures were calculated based upon the system's output as compared to the ground truth. The system level test evaluated the full hardware/software technology package in a controlled environment by measuring the time and accuracy for the system to capture a person's image and match them against a watchlist.
- Object detection/image classification – This elemental test evaluated these technologies at the system level. The test began with end-users capturing feature/object-laden images of the environment with the evaluation team analyzing their output of the number of objects detected/images classified.
- Shot localization/weapon classification - A system level elemental test was designed to evaluate the accuracy of this technology's ability to detect gunshots, calculate a shot's trajectory, localize a shot's origin, identify the caliber of bullet fired and classify the weapon that fired the shot (see Figure 1 for an example output).
- Soldier state/localization – A system level elemental test was created to assess the ASSIST system's ability to characterize a Soldier's actions within indoor and outdoor environments.
- Sound/speech recognition – A system level elemental test was devised to evaluate the technology's ability to detect specific sounds within the environment.

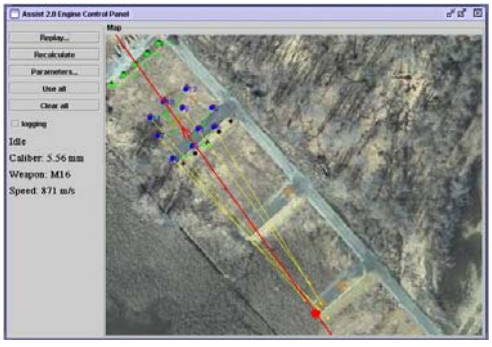


Figure 1: Shot localization/weapon classification output

Once the technologies and their respective elemental tests were ascertained, the next step was to define the specific metrics and measures. This step also included identifying the influential variables that impact performance, specifically highlighting which variables should be fixed along with those that should be altered during the test(s). More information on this step with respect to the ASSIST evaluations can be found in [2] [4].

It was now time to identify a suitable testing environment for each of these elemental tests. It was determined that the system-level elemental tests would be conducted at a MOUT site at the Aberdeen Proving Grounds in Maryland. The only exception to this was the shot localization/weapon classification test. Since live gunfire was necessary to accurately assess this technology and safety restrictions were in place at the MOUT site, this technology was evaluated at a live fire range adjacent to the MOUT site. Locating a test environment for the component level elemental tests was less taxing since these could be run practically anywhere since they were run on common personal computers (PCs).

Choosing participants was the next step, specifically those that will use the technology (whether it be the members of the end-user population or the technology developers) and those that will indirectly interact with the systems (including those playing roles within the environments). Per DARPA's instructions, Phase I evaluations had the technology developers use/wear their ASSIST systems and shadow the movements of partner Soldiers. This restriction was reduced as both researchers and end-users (Soldiers) used/wore the systems throughout the Phase II evaluations.

Training of these personnel played a critical role in the evaluations. For Phase I that called for the developers to use their own systems, the training consisted of familiarizing these

personnel with the scope of the elemental tests, not the technology (since they were the ones that created the systems). However, when the Soldiers stepped in to use the technology during later evaluations, they had to be trained not only on the scope of the elemental tests, but also on how to use the technology. Likewise, the actors in the environment (e.g. the people whose faces were captured to support the face recognition technology, the shooters who fired the weapons to test the shot localization/weapon classification technology, etc.) all had to be trained on their roles.

Additionally, it was necessary to determine how data was to be collected from the ASSIST technologies (and the environment, where necessary). Successfully undertaking this task required that the technology outputs are known (which, according to the SCORE framework, are highlighted as the technologies for evaluation are identified) along with realizing what critical data could be captured from the environment. Data collection can be as simple as measuring the amount of time it takes for the face recognition/matching algorithm to return a match. It can also require more complex actions such as an IET member noting specific actions of the system-wearer into a voice-recorder and then comparing those actions with corresponding times (from their audible notes) to that of a technology system-output log file. For each of the described elemental tests, data collection methods were determined based upon the available output data and the metrics necessary for each evaluation.

Going hand-in-hand with determining the data collection methodology and the required personnel were the scenarios in which the components/systems were tested. These use-case scenarios were developed based upon the expected concept(s) of operation (CONOPS) while keeping in mind the technology's current state of maturity. Specifically, CONOPS is a "formal document that employs users' terminology and a specific, prescribed format to describe the rationale, uses, operating concept, capabilities and benefits of a system" [5]. The challenge in this step is that CONOPS do not often exist for emerging technologies. To surmount this obstacle, the IET developed use-case scenarios with end-user and technology developer input. These test scenarios are presented in great detail in [2] [4].

Going through these SCORE-prescribed steps in order to assess technical performance at the component and system levels produced comprehensive evaluations for the above mentioned ASSIST technologies. SCORE was also applied to develop system level utility assessments in the form of vignette tests.

3.1.2 Vignette Tests

This test type was used to perform System Level Testing – Utility Assessment of the ASSIST technologies [6]. In this case, utility is defined as the value that a technology or piece of equipment provides to an end-user. Utility assessments were uniquely designed given the technology's state of maturity. Typically, a system's utility can still be evaluated prior to its full development where the intent of the assessment is to inform on the system design. Assessments done at the end of a technology's development cycle are intended to validate the value of the system. The former evaluation type is known as formative while the latter is defined as summative.

Since the ASSIST technologies were young in development, these formative vignette tests took the form of several operationally-relevant, mini-mission scenarios where end-users employed the technology in use-case situations to accomplish their mission objectives. Informing the developers about the capabilities of the ASSIST technology became the goal in the design and execution of the SCORE-driven utility evaluations. It should be noted that all of the ASSIST Phase I and II technologies were evaluated under vignette tests with the exception of the shot localization/weapon classification due to safety considerations.

Measures were identified in the form of end-user surveys and semi-structured interviews. The end-users (Soldiers in the case of the ASSIST evaluations) were presented with a suite of survey questions that they answered with respect to their recent experiences with the technology. Furthermore, the Soldiers were interviewed (without the technology developers being present) to gain further insight into what features/capabilities they liked, what they didn't like, and what improvements should be made. The responses were rolled up into technology utility assessments.

The Aberdeen MOUT site presented a small-scale, middle-Eastern-like village where Soldiers frequently train. This test environment provided over a dozen single-story and two-story buildings that challenged the ASSIST technology-laden end-users.

The participants selected to use the technology and to interact with the end-users in the environment were chosen in an identical manner to that of the individuals selected for the elemental tests. Phase I started with the researchers wearing their own technologies and shadowing the Soldiers during the elemental and vignette tests while Phase II put the technology directly on the Soldiers. In both phases, extras/environmental actors were employed to bring about more realism in the vignette test environment. Training for these participants is similar for what was done in support of the elemental tests. When the Soldiers were wearing the technology, they were provided specific training by the research teams so they would be competent in the systems' basic operations.

Some of the data collection methods are already presented in the form of survey instruments and semi-structured interviews. Additionally, several evaluation team members were strategically placed within the environment to observe the Soldiers, the researchers (when they wearing the technology during Phase I), and the extras acting within the environment.

In parallel, the specific vignette mission scenarios were created. After considering the various SCORE-prescribed factors and interviewing subject matter experts, the following mission-scenarios were used throughout the various evaluations in Phase I and II included:

- Presence patrol with deliberate search
- Presence patrol leading to a cordon and search
- Presence patrol and improvised explosive device site reconnaissance
- Assessment of local village with respect to an upcoming election
- Presence patrol leading to checkpoint operations

Prior to the execution of each mission, the Soldiers were briefed on their specific objectives and told to react accordingly to the environment based upon their tactical training. The Soldiers were also reminded of the available ASSIST technologies at their disposal and instructed to use them as they see fit to accomplish their mission objectives.

Using the SCORE framework, elemental and vignette tests were designed and executed to provide the program sponsor with the

requested data in addition to informing the researchers on the state of their technologies. The next subsection will show how SCORE has been applied to evaluate another technology.

3.2 TRANSTAC – Phase II

TRANSTAC is another DARPA advanced technology and research program whose goal is to demonstrate capabilities to rapidly develop and field free-form, two-way speech-to-speech translation systems enabling English and foreign language speakers to communicate with one another in real-world tactical situations where an interpreter is unavailable [7]. Several prototype systems have been developed under this program for numerous military applications including force protection and medical screening. The technology has been demonstrated on PDA (personal digital assistant) and laptop platforms. DARPA asked NIST to assess these systems starting in Phase II of this program (another team evaluated these systems during Phase I). Five different research teams presented systems for evaluations during this phase.

The NIST IET applied the SCORE framework to this program because this approach would scale well as the systems continued to mature and DARPA wanted both technical performance and utility assessments of the technology. Specifically, the following test types were conducted during the Phase II evaluations:

1. System usability testing – providing overall scores to the capabilities of the whole system.
2. Software component testing – evaluating components of a system to see how well they perform in isolation.

The IET implemented a two-part test methodology to produce these metrics. Metric 1 was evaluated through the use of structured scenarios within live evaluations, while metric 2 was evaluated through the use of pre-recorded utterances within offline evaluations.

3.2.1 Offline Evaluations

The offline evaluations, which represent the *Component Level Testing – Technical Performance* aspect of the TRANSTAC evaluation, were designed to test the TRANSTAC systems with exactly the same set of data so comparison among the systems would truly be "apples-to-apples." Identical speech utterances, both English and foreign language, were fed into each developer's system. These utterances were collected from audio recordings from data gathering events. First, an audio file was fed into each system to test the systems' speech-to-text (S to T) capabilities. Then a text format to test their systems' text-to-text (T to T) capabilities was fed into each technology. Since the system outputs include translated text and speech, metrics were extracted through comparison of the system outputs to ground truth. A range of metrics including low-level concept transfer and automated metrics were extracted from the offline outputs [8].

Since this evaluation focused on inputting utterances into each development teams' system, choosing the appropriate test site was trivial. For simplicity, the offline evaluation was conducted at the same site as the live evaluations since there were tighter venue constraints for these tests. Additionally, participant selection and training is very straight forward. An offline evaluation specialist worked with a member of each research team to ensure that each system accepted the offline utterances without incident.

The use-case scenarios under which the utterances (both audible and text speech) were generated stemmed from the supporting data collections (and their respective scenarios) that took place months in advance of the evaluation. This scenario development process began with the IET meeting with the technology's potential end-users, both English-speaking military personnel and foreign language experts, to determine the representative use-cases in which this type of technology would be most beneficial. Once those situations were established, the IET developed scenarios that were used in the data collections. The data collections brought together English and foreign language speakers to talk/role-play through the data collection scenarios that produced 10 to 20 minute data collection dialogues. Each of the audio dialogues were transcribed and translated. A majority of the data was provided to the developers to train their systems while the remainder was held out by the IET to create the evaluation scenarios. Utterances from the evaluation set were selected to be used in the offline evaluation.

3.2.2 Live Evaluations

The live evaluations were performed in two different venues, the lab and the field (both containing facets of *System Level Testing – Technical Performance* and *System Level Testing – Utility Assessment*) and were conducted with structured scenarios. This scenario type provided a set of questions to the English speaker that they needed to find answers, while the foreign language speaker was given the answers to those questions in paragraph format. A dialogue occurred between the two speakers and the number of questions that the English speaker was able to get answered were noted. In addition, surveys were provided to the English and foreign language speakers to gauge their perception of the TRANSTAC systems.

Lab evaluations were designed to test the TRANSTAC systems in an idealistic environment, with no background noise and the participants being stationary. The TRANSTAC systems were placed on a table as opposed to being worn by the speakers. This idealistic environment gave the IET and the developers an idea of the best that the systems can do at this stage in their development.

The purpose of the field evaluations was to test the TRANSTAC systems in a more realistic environment. This included well-controlled background noise, the English-speakers carrying the TRANSTAC systems, and both the English and foreign language speakers being mobile during the evaluation.

Twenty structured scenarios (ten in the lab and ten in the field) were designed to foster the evaluation dialogues. These scenarios were derived from the same held back scenarios that the offline scenarios originate.

The system users for these evaluations were chosen to be potential TRANSTAC system end-users including both English-speaking military personnel and representative foreign language speakers. Training and preparation of these individuals was critical. These individuals had to be both trained on the proper usage of the TRANSTAC systems, but also had to be educated on the procedures and flow of the structured scenarios. This training was done sequentially to enable the IET the ability to isolate and address any areas of concern.

Selecting a site for these evaluations required the consideration of numerous factors including a location that could support the offline, lab, and field evaluations, a spot that could accommodate 50+ personnel, a site that was available for six consecutive days, etc. Ultimately, the NIST campus was selected after extensive exploration.

3.3 Initial Findings using SCORE

The ASSIST Phase I and II and TRANSTAC Phase II evaluations were successful events that provided DARPA with the necessary and detailed results desired by the programs. SCORE is viewed as a contributor to this success due to the extensive nature in which it laid out these evaluations. Following its prescribed steps and addressing each evaluation component ensured that comprehensive and relevant evaluations were generated. The following section will show how SCORE has evolved to produce innovative evaluations as the ASSIST and TRANSTAC programs further advance.

4. EVOLUTION of the FRAMEWORK

Both the ASSIST and TRANSTAC programs have since moved into Phase III. To date, two ASSIST Phase III evaluations have been performed while a single TRANSTAC Phase III evaluation has already occurred with each program having one more Phase III evaluation to go. Since these programmatic goals have changed from the previous phases, the SCORE framework has evolved to produce the desired metrics. One major innovation is the addition of a fourth evaluation goal type, described below:

- *Capability Level Testing – Utility Assessments* – This evaluation group is proposed to assess the utility of an individual capability (where the complete system is made up of multiple capabilities), where utility is defined as the value the application provides to the system end-user (just as it is *System Level Testing – Utility Assessments*). The benefit of this evaluation type is that specific capability utility and usability to the end-user can still be addressed even when the system and user-interface are still under development.

This goal type can be inserted into the tiered approach either after the *Component Level Testing – Technical Performance* or the *System Level Testing – Technical Performance* goal types.

Each of the evaluation elements described in section 2.1 are applied to this new goal type. This new SCORE addition will be presented within the following discussion of the ASSIST Phase III evaluation design whereas further applications of SCORE will be discussed in TRANSTAC Phase III evaluation plan.

4.1 ASSIST – Phase III

As the ASSIST program moved into Phase III, the program evaluation focus was altered to place more emphasis on end-user utility assessments as opposed to technical performance. With the technologies further along in their development cycles (as compared to their status in earlier phases), it was becoming more important to gain insight into the end-users' value of specific capabilities. In addition to emphasizing utility assessments, the program is now more focused on real-time capabilities as opposed to those that support after-mission reporting. Three separate research teams produced Phase III evaluation technologies that included the following capabilities:

- Face recognition/matching
  - Face image collection using commercial off-the-shelf (COTS) hardware
  - Face image matching displayed on COTS wearable interface
- Real-time information collection and sharing
  - Automatic capture of image, audio, and GPS data
  - End-user viewing own-captured data on wearable, COTS visual display (see Figure 2)
  - Transmit/receive image and GPS data to/from other ASSIST units
- After-mission reporting
  - Retrieving mission data at specific times and locations
  - Locating field-marked significant actions
  - Observing soldier state analytics

Figure 2: Soldier using real-time data collection ASSIST system

To satisfy the program goals, only technical performance evaluations were designed for the face recognition/matching technology at both the component and system levels. This took the form of elemental tests, similar to those outlined in section 3.1.1. Additionally, system level utility assessments were collected for the real-time information sharing and after-mission reporting technologies through additional vignettes (comparable to those presented in section 3.1.2.).

However, the need to gather further utility assessments, especially of the face recognition/matching technology which was not evaluated in the vignette tests during this phase, spawned the SCORE evaluation goal type of *Capability Level Testing – Utility Assessment.* This inspired the development of task tests whose intent was to assess end-user utility of specific capabilities within the various ASSIST technologies.

After determining the objective of the task evaluations, the IET continued down the path of identifying the remainder of the SCORE evaluation elements by identifying the necessary measures and metrics. The measures extracted from this test include IET observer notes (made while following the end-users with the technology during the tasks) along with surveys presented to the end-users at the conclusion of each task (similar to those given at the end of the vignette tests). The data collection methods used to gather the observer notes include the use of hand-held PDA note-taking devices while the surveys were administered via PC. The survey results and observer notes were combined to produce the necessary metrics (similar to what was done to produce the metrics from the vignette tests).

These task tests, in addition to the elemental and vignette tests, took place at the same Aberdeen MOUT site that supported the Phase I and II evaluations. Multiple participants were required for the task tests. Soldiers, the ultimate end-users, were selected to use/wear the systems throughout the task tests. For the task tests (like the other test types), training was a critical component. Specific time was set up for the research teams to brief the Soldiers on their technology along with allowing them an opportunity to have hands-on practice with the various systems. Additionally, training time was also allocated for the Soldiers to become competent with the specific task tests (both, the test objectives and flow). Prior to the Soldiers running these tasks,

they were briefed on the specific task objectives (both with respect to using the technology and the tactical goals). Following the briefing, the end-users and the IET practiced each of the task runs (without the technology) to ensure everyone was competent with the tests when it became time to run them with the technology.

The task test scenarios were created in parallel to addressing the numerous steps presented above. Tasks were developed to specifically address all of the Phase III technology capabilities including the following:

- Street observation and interaction – This task was developed to specifically test real-time image sharing across multiple ASSIST systems.
- Presence Patrol – This task was designed to evaluate personnel tracking, GPS positioning and map annotation capabilities.
- Insurgent Surveillance – This test was created to assess the capability of image and map transfer between the laptop-based systems and ground-based wearable ASSIST technologies.
- Insurgent Surveillance and Ambush – This task was created to test the ASSIST technology's ability to calculate soldier state analytics.
- Base/entry checkpoint – This task was developed to test the face recognition/matching system's ability to capture images in the field and present matches in real-time on the system-wearer's personnel interface.

These tasks were designed to be between 10 to 15 minutes in length where each was run twice. The runs were also set up to have three end-users use the relevant ASSIST technology with an emphasis on the specified capabilities. Two Soldiers used the portable, wearable technology while the other user interacted with the laptop-based system.

Addressing each one of the SCORE framework elements with respect to the task tests enhanced the effectiveness of this series of evaluations at the most recent ASSIST events. Comprehensive utility assessments were collected from the task tests which enabled the IET to produce an extensive picture of the current state of the ASSIST technologies when combined with the elemental and vignette test data.

This additional *Capability Level Testing – Utility Assessment* is an advancement in the SCORE framework. Additional improvements will be shown in the following section discussing the latest phase of the DARPA TRANSTAC program.

4.2 TRANSTAC – Phase III

Phase III of the DARPA TRANSTAC program continues to present the same overall evaluation objectives as presented in Phase II. Additionally, this phase brought about additional technical performance and utility assessments of several specific TRANSTAC technologies including a ruggedized, portable hardware platform (known as the Lynx system) and the systems' ability to handle the translation of names, streets, and places (simply stated as "names" throughout the rest of this paper) from a specific foreign language to English.

Keeping these goals in mind, offline and live formats (lab venue, only) were conducted similar to those run in Phase II to accomplish the primary evaluation goals. The SCORE framework played a critical role in defining the testing scopes for evaluating the Lynx system and the systems' capability to address names.

The Lynx evaluation was designed under the *System Level Testing – Technical Performance* and *System Level Testing – Utility Assessment* evaluation goal types. The design of this evaluation closely mirrored that of the live lab evaluations. Recall that the main TRANSTAC systems were evaluated with both speakers sitting at a table interacting with the laptop-based system which was placed on the table (as opposed to being worn). To that end, the Lynx systems were evaluated in a similar manner where they were placed on a table where the English speaker sat on one side of the table and the foreign language speaker on the opposite. The Lynx test tasked the speakers with transferring as many concepts as possible within a ten minute timeframe while adhering to the structured scenario format. For the sake of comparison, the same structured scenarios that were used in the main evaluation were selected for the Lynx evaluation. As in the main test, the evaluation team was able to extract technical performance metrics through the number of concepts transfer. Additionally, the end-users were administered specific surveys to assess their utility of the Lynx technology.

Because the Lynx platform was different from that of the laptop-based systems, additional training was provided to the end-users before this evaluation. This was particularly important so that the end-users did not confuse this system's operation with that of the technology they had used earlier (the main laptop system evaluations were conducted immediately prior to the Lynx system testing).

The names capability was evaluated under the *Component Level Testing – Technical Performance* and *Capability Level Testing – Utility Assessment* evaluation goal types. This test was conducted in both the live lab and offline settings and used the main evaluation laptop-based platforms. The only other similarity to the main live lab evaluations include the fact that the speakers were sitting across from one another at a table and did not have to wear the system.

Since the goal of this evaluation was to isolate the systems' ability to translate names, the SCORE elements directed the IET to design unique scenarios to support both the offline and live lab venues. This specialized scenario design stemmed back to the data collection scenarios. Three unique, names-laden scenarios were created as scripted dialogues and recorded by unique speaker-pairs. These dialogues were crafted such that there was at least one name in each foreign language utterance where it was noted whether this name appeared in the names lexicon (a list of names that the research teams have access) or if it did not along with whether each name was unique (the name can only mean a name) or whether it was a "double" (the name can also mean an object, etc). This recorded data was used to create the offline names evaluation set where all of the recordings were kept by the IET (no names data was released to the developers since the intent was to prevent the out of lexicon names from being known by the researchers ahead of time). The scenarios used in the live names evaluations were identical to those scripted ones used in the names data collections.

The offline names evaluation ran similarly to that of the main offline evaluation. Specific utterances are selected and fed directly into the TRANSTAC systems. However, the measures and metrics from this test focused on how the systems specifically handled the translations of the names.

The live names evaluation ran in a different manner than that of the live main (lab) evaluation. The speakers were provided with the scripted names scenarios (as opposed to the standard structured scenarios) and instructed to read them verbatim. The English speaker began each utterance by stating the number of the utterance they were about to read (which alerts the foreign language speaker to the current utterance) and then spoke the utterance. Since the focus of this evaluation was on the translation of names from the foreign language into English, the English speaker did not speak into the TRANSTAC system. After hearing the English utterance, the foreign language speaker responded with their scripted utterance which they spoke into the TRANSTAC system. If the English speaker was able to understand the name that was communicated, they noted that and moved on to the next utterance. If the English speaker was unable to ascertain a name from the TRANSTAC output, then they were able to rephrase their utterance in any manner they saw fit. Likewise, the foreign language speaker, upon hearing the English speaker rephrase their utterance, rephrased theirs accordingly to convey the desired name. The output of this evaluation produced both technical performance and utility assessment data. This took the form of measuring the number of names successfully transferred and collecting survey responses from the end-users regarding their specific names interactions.

The SCORE framework was successfully employed to address additional evaluation goals including the Lynx system and names translation capabilities. Likewise, the framework further evolved to address progressing needs in the ASSIST program.

5. FUTURE EFFORTS

SCORE is still being used to design the remaining ASSIST and TRANSTAC Phase III evaluations which will both take place before the end of the calendar year. If these programs continue, it is envisioned that SCORE will be used to design their successive evaluations.

The SCORE framework is applicable to domains beyond emerging military technologies and those solely dealing with intelligent systems. Personnel at NIST are applying the SCORE framework to the virtual manufacturing automation competition (VMAC) and the virtual RoboRescue competition (within the domain of urban search and rescue). Their intent is to develop elemental tests and vignette scenarios to test complex system capabilities and their component functions. Likewise, personnel in NIST's construction metrology group have expressed interest in the SCORE framework with respect to designing evaluations within the automated construction domain.

It is envisioned that SCORE will be applied to a broad range of technologies, both to design evaluations of emerging components and systems along with enhancing evaluation procedures of pre-existing technologies. This framework is highly adaptable and capable of meeting most any evaluation requirement.

6. CONCLUSION

SCORE has proven to be an invaluable evaluation design tool of the NIST IET and was the backbone of eight (five for ASSIST and three for TRANSTAC) evaluations. Further, it is expected to play a critical role in the remaining Phase III ASSIST and TRANSTAC evaluations. The NIST IET will continue to apply the SCORE framework in future evaluations (including those outside of the military community) and will support other members in the technology evaluation community who wish to leverage it.

7. DISCLAIMER

Certain commercial products and software are identified in this paper in order to explain our research. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the products and software identified are necessarily the best available for the purpose.

8. ACKNOWLEDGMENTS

The authors would like to acknowledge the DARPA ASSIST/ TRANSTAC program manager, Dr. Mari Maeda, and the members of the NIST IET for their continued support.

9. REFERENCES

[1] Steves, M. and Scholtz, J. 2005. A Framework for Evaluating Collaborative Systems in the Real World. In *Proceedings of the 48th Hawaii International Conference of System Sciences* (HICSS-38), Hawaii.

[2] Schlenoff, C., Steves, M., Weiss, B.A., Shneier, M., and Virts, A. 2006. Applying SCORE to Field-based Performance Evaluations of Soldier Worn Sensor Technologies. *Journal of Field Robotics*, Volume 24, (8-9), pp. 671-698.

[3] Schlenoff, C., Weiss, B.A., Steves, M., Virts, A., Shneier, M. and Linegang, M. 2006. Overview of the First Advanced Technology Evaluations for ASSIST, In *Proceedings of the 2006 Performance Metrics for Intelligent Systems (PerMIS) Conference*, Gaithersburg, Maryland.

[4] Weiss, B.A., Schlenoff, C., Shneier, M., and Virts, A. 2006. ASSIST: Technology Evaluations and Performance Metrics for Soldier-Worn Sensor Systems. In *Proceedings of the 2006 Performance Metrics for Intelligent Systems (PerMIS) Conference*, Gaithersburg, Maryland.

[5] CONOPS, http://en.wiktionary.org/wiki/CONOPS, (Definition retrieved on 07/10/08)

[6] Steves, M. P. 2006. Utility Assessments for ASSIST Systems. In *Proceedings of the 2006 Performance Metrics for Intelligent Systems (PerMIS) Conference*, Gaithersburg, Maryland.

[7] Weiss, B.A., Schlenoff, C., Sanders, G., Steves, M., Condon, S., Phillips, J., and Parvaz, D. 2008. Performance Evaluation of Speech Translation Systems, In *Proceedings of the 6th edition of the Language Resources and Evaluation Conference*. Marrakech, Morocco.

[8] Sanders, G., Bronsart, S., Condon, S., and C. Schlenoff. 2008. Odds of successful transfer of low-level concepts: A key metric for bidirectional speech-to-speech machine translation in DARPA's TRANSTAC program. In *Proceedings of LREC 2008.*

Reliability Estimation and Confidence Regions from Subsystem and Full System Tests via Maximum Likelihood

James C. Spall
The Johns Hopkins University
Applied Physics Laboratory
11100 Johns Hopkins Road
Laurel, Maryland 20723-6099 U.S.A.
james.spall@jhuapl.edu

ABSTRACT
This paper develops a rigorous and practical method for estimating the reliability—with confidence regions—of a complex system based on a combination of full system and subsystem (and/or component or other) tests. It is assumed that the system is composed of multiple processes (e.g., the subsystems and/or components within subsystems), where the subsystems may be arranged in series, parallel (i.e., redundant), combination series/parallel, or other mode. Maximum likelihood estimation (MLE) is used to estimate the overall system reliability based on this fusion of multiple sources of information. Interestingly, for a given number of subsystems and/or components, the performance metric (likelihood function) does *not* change with the system configuration; rather, only the optimization constraints change, leading to an appropriate MLE. The MLE approach is well suited to providing asymptotic or finite-sample confidence bounds through the use of Fisher information or Monte Carlo sampling (bootstrap). The paper establishes formal conditions for the convergence of the MLE to the true system reliability.

Categories and Subject Descriptors
G.3 Probability and statistics
I.3 Simulation and modeling
J.2 Physical sciences and engineering

General Terms
Algorithms, design, experimentation, measurement, performance, reliability, theory.

Keywords
Reliability, parameter estimation, optimization, bootstrap, maximum likelihood, likelihood performance metric, Fisher information matrix, data fusion.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

1. INTRODUCTION
This paper considers the problem of estimating the reliability for a complex system based on a combination of information from tests on the subsystems, components, or other aspects of the system, and, if available, tests on the full system. A key motivation for this setting comes from the fact that it is often difficult or infeasible to directly evaluate the reliability of complex systems through a large number of full system tests alone. Such a difficulty may arise, for example, when the full system is very costly (or dangerous) to operate and/or when each full system test requires the destruction of the system itself. Nevertheless, it is also often the case that there are at least a few tests of the full system available; it is obviously desirable to include such information in the overall reliability assessment. Such full systems tests are often critical to help guard against possible mismodeling of the relationship between the subsystems and full system in calculating overall reliability. This paper develops a method based on principles of maximum likelihood for estimating the overall system reliability from a *combination* of full system and subsystem or other tests.[1]

Certainly, other approaches exist for estimating system reliability when the subsystems are independent (see, e.g., Hwang et al., 1981; Tang et al., 1997; and Ramírez-Márquez and Jiang, 2006). However, these approaches do not allow for easy inclusion of limited full system tests (when available), and do not generalize to include systems where the subsystems may be statistically dependent. (Note that the inequality-based reliability method of Hill and Spall, 2007, *does* allow for such dependence in producing a bound to the full system reliability, but this method requires certain pairwise subsystem tests that may not be feasible in practice.)

A key part of the approach here is the calculation of uncertainty (confidence) bounds on the estimates. We discuss the Fisher information matrix as a basis for asymptotic bounds and also discuss a bootstrap-based method for computing confidence regions when the asymptotic bounds are inappropriate. This

[1]To avoid the cumbersome need to repeatedly refer to tests on subsystems, components, and other aspects of the system as the key source of information other than full system tests, we will usually only refer to subsystem tests; "subsystem tests" in this context should be considered a proxy for all possible test information short of full system tests.

bootstrap approach deals with the inadequacies of traditional methods based on the asymptotic normality of the MLE.

Several other approaches have been proposed to deal with the inadequacy of asymptotic normality in the context of using subsystem tests to estimate full system reliability. For example, Myhre and Saunders (1968) use the asymptotic chi-squared distribution of the log-likelihood ratio to deal with the problem of having confidence intervals outside the unit interval [0, 1]. Easterling (1972) treats the system reliability (derived from subsystem estimates) as an estimate from data having a binomial distribution and then uses standard results on the exact distribution of the binomial estimate to get the confidence interval; this yields an exact solution in the special case where the system is composed of one subsystem (i.e., system = subsystem), and an intuitively appealing approximation when there are two or more subsystems. Coit (1997) considers the case where there are a "large" number of subsystems in either a series or parallel configuration; in the series configuration, the logarithm of the system reliability is approximately normally distributed by central limit theorem effects. Hence, a log-normal distribution is assumed for the system reliability, providing the basis for the confidence interval. Ramírez-Márquez and Jiang (2006) focus on methods for estimating the variance of the reliability estimates, and then use these variance estimates together with normal or binomial approximations to the distribution of the estimates to form confidence intervals. There are also a large number of Bayesian methods for forming estimates and uncertainty bounds (e.g., Tang et al., 1997); we do not consider these here due to the difficulties in specifying reliable prior distributions.

Section 2 of this paper gives the general MLE formulation, showing that one likelihood function (the objective function for optimization) applies across all problem settings, with only the constraints changing to accommodate the different settings. Section 3 considers the case of systems with subsystems that are statistically dependent while the system is operating. Section 4 establishes that the MLE for the full system reliability is stochastically (almost surely) convergent to the true system reliability as the amount of test data gets large. Section 5 discusses the Fisher information matrix and gives explicit forms in the special cases of fully series and fully parallel systems. This matrix is critical in determining whether there is sufficient information to estimate the reliability (the identifiability) and in forming asymptotic confidence regions for the MLE. Section 6 describes a bootstrap-based method for constructing confidence intervals that is useful when the asymptotic bounds do not apply. Section 7 is a numerical study and Section 8 offers some concluding remarks.

2. THE LIKELIHOOD FUNCTION AND MLE FORMULATION

2.1 General Formulation

Consider a system composed of p processes, typically subsystems and/or components of subsystems. The subsystems may be arranged in series, parallel (i.e., redundant), combination series/parallel, or other form (e.g., standby systems), subject to being able to write down a probabilistic characterization of the system that leads to a likelihood function (Leemis, 1995, Chap. 2, includes a thorough discussion of the many types of systems frequently encountered in practice). While we assume that the *test data* for estimating the reliability are statistically independent, we do not, in general, require that the processes be statistically independent *when operating as part of the full system*. (In principle, it is also possible to formulate a likelihood function based on test data that are statistically dependent. That is, the testing outcome for process j may be statistically dependent on the outcome for process i, for some $i \neq j$. We do not pursue that extension here.) In the discussion below, the term "operationally [in]dependent" is used to refer to the case where the processes are statistically independent or dependent (as relevant) when operating as part of the full system.

The general MLE formulation involves a parameter vector $\boldsymbol{\theta}$, representing the parameters to be estimated, together with an associated log-likelihood criterion $\mathcal{L}(\boldsymbol{\theta})$. Let ρ and ρ_j represent the reliabilities (success probabilities) for the full system and for process j, respectively, $j = 1, 2, ..., p$. The vector $\boldsymbol{\theta} = [\rho_1, \rho_2, \ldots, \rho_p]^T$; ρ is not included in $\boldsymbol{\theta}$ because it will be uniquely determined (or bounded) from the ρ_j and possibly other information via relevant constraints. Note that when ρ is uniquely determined by a function of $\boldsymbol{\theta}$, then the estimate $\hat{\rho}$, as determined from applying this function to the MLE of $\boldsymbol{\theta}$ (say $\hat{\boldsymbol{\theta}}$), is the MLE of ρ. This invariance of MLE applies even though the mapping from $\boldsymbol{\theta}$ to ρ is not generally one-to-one and may not be continuous (see, e.g., Zehna, 1966).

Ultimately, we are interested in an estimate (with confidence bounds) for ρ (derived from the MLE for $\boldsymbol{\theta}$). The specific definition of $\boldsymbol{\theta}$ will depend on the details of the system. Interestingly, for a given definition of $\boldsymbol{\theta}$, the definition of $\mathcal{L}(\boldsymbol{\theta})$ will *not* depend on how the subsystems are arranged in the full system (i.e., $\mathcal{L}(\boldsymbol{\theta})$ is the same regardless of whether, say, the subsystems are in series or parallel). However, the MLE *will* change as a function of the system arrangement. This is a consequence of the constraints in the optimization problem that is solved to produce the MLE, as illustrated below.

It is generally assumed, at a minimum, that success/failure data are available on the p processes within the full system. As mentioned above, it is also generally assumed that success/failure data are available directly on the full system. In cases involving dependent subsystems, it may be desirable that the information from the p processes include some data other than direct subsystem success/failure data in order to obtain the information needed for characterizing the nature of the dependence (we say "desirable" because it may be possible to estimate *bounds* to the system reliability in the absence of such information). For example, in the dependent-subsystem case discussed in Section 3, obtaining data on one critical component appearing within multiple subsystems allows for an MLE of ρ; the absence of such data allows the analyst to estimate a *lower bound* to ρ.

We now present the general MLE optimization problem. Let Θ represent the feasible region for the elements of $\boldsymbol{\theta}$. To ensure that relevant logarithms are defined and that the appropriate derivatives exist, it is assumed, at a minimum, that the feasible region Θ includes the restriction that $0 < \rho_j < 1$ for all j (other restrictions may be included as appropriate). The general MLE formulation is:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}\in\Theta} \mathcal{L}(\boldsymbol{\theta}) \quad \text{subject to } f(\boldsymbol{\theta},\rho) \geq 0, \tag{2.1}$$

where $f(\cdot)$ is some function reflecting the constraints associated with the operation of the full system. In some common cases (e.g., fully series and fully parallel cases, with the processes corresponding to the subsystems as in Subsection 2.2), the inequality in the constraint can be replaced with an equality.

Let X represent the number of successes in n independent, identically distributed (i.i.d.) experiments with the full system and X_j represent the number of successes in n_j i.i.d. experiments with process j, j = 1, 2, ..., p. Note that in the discussion below there is no notational distinction between a random variable (vector) and its realization, with the expectation that the distinction should be clear from the context. Let $\boldsymbol{Y}$ represent the full set of data $\{X, X_1, X_2,\ldots, X_p\}$. From the assumption of independence of all test data, the probability mass function, say $p(\boldsymbol{Y}\,|\,\boldsymbol{\theta},\rho)$, is:

$$p(\boldsymbol{Y}\,|\,\boldsymbol{\theta},\rho) = \underbrace{\binom{n}{X}\rho^X(1-\rho)^{(n-X)}}_{\text{system}} \times \underbrace{\binom{n_1}{X_1}\rho_1^{X_1}(1-\rho_1)^{(n_1-X_1)}\cdots\binom{n_p}{X_p}\rho_p^{X_p}(1-\rho_p)^{(n_p-X_p)}}_{p\text{ processes}}, \tag{2.2}$$

leading to the log-likelihood function $\mathcal{L}(\boldsymbol{\theta}) \equiv \log p(\boldsymbol{Y}\,|\,\boldsymbol{\theta},\rho)$:

$$\mathcal{L}(\boldsymbol{\theta}) = X\log\rho + (n-X)\log(1-\rho) + \sum_{j=1}^{p}\left[X_j\log\rho_j + (n_j - X_j)\log(1-\rho_j)\right] + \text{constant}, \tag{2.3}$$

where the constant is not dependent on $\boldsymbol{\theta}$. Note that the generic forms for the above likelihood and log-likelihood apply regardless of the specific layout of the subsystems (series, parallel, combination series/parallel, or operationally dependent). However, the relationship between ρ and the ρ_j differs according to the layout of the processes. In finding the MLE of $\boldsymbol{\theta}$, say $\hat{\boldsymbol{\theta}}$, this relationship manifests itself as constraints in an optimization problem.

Typically, the MLE is determined via finding a root of the score equation $\partial\mathcal{L}(\boldsymbol{\theta})/\partial\boldsymbol{\theta} = \mathbf{0}$ (or a normalized form of this equation in the asymptotic sample size case), where the score vector is

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}} = \begin{bmatrix} \partial\mathcal{L}/\partial\rho_1 \\ \partial\mathcal{L}/\partial\rho_2 \\ \vdots \\ \partial\mathcal{L}/\partial\rho_p \end{bmatrix}. \tag{2.4}$$

The elements of the score vector depend on the constraints. A common special case is where the constraints lead to a unique differentiable function $h(\cdot)$ that relates $\boldsymbol{\theta}$ to ρ: $\rho = h(\boldsymbol{\theta})$. In that case, (2.3) leads to the following form of the score vector in (2.4):

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}} = \frac{X}{\rho}\frac{\partial h}{\partial\boldsymbol{\theta}} - \frac{n-X}{1-\rho}\frac{\partial h}{\partial\boldsymbol{\theta}} + \begin{bmatrix} \dfrac{X_1}{\rho_1} - \dfrac{n_1 - X_1}{1-\rho_1} \\ \vdots \\ \dfrac{X_p}{\rho_p} - \dfrac{n_p - X_p}{1-\rho_p} \end{bmatrix}. \tag{2.5}$$

We will see several illustrations of the form in (2.5) in the examples and theoretical results below.

2.2 Fully Series and Fully Parallel Cases

To illustrate the general formulation of Subsection 2.1, we consider here the two most common "extreme" cases of fully series systems and fully parallel systems with the p processes corresponding to p operationally independent subsystems. The results below can extend readily to common cases of mixed (operationally independent) series/parallel systems (see, e.g., Leemis, 1995, Sect. 2.1, for a description of such general systems). The results here illustrate the setting mentioned in the context of (2.5) above, where there exists a differentiable function $\rho = h(\boldsymbol{\theta})$.

From (2.1), the MLE in the series-subsystem case is found according to

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}\in\Theta} \mathcal{L}(\boldsymbol{\theta}) \quad \text{subject to } \rho = \prod_{j=1}^{p}\rho_j,$$

while the MLE in the parallel-subsystem case is found according to

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}\in\Theta} \mathcal{L}(\boldsymbol{\theta}) \quad \text{subject to } 1-\rho = \prod_{j=1}^{p}(1-\rho_j).$$

In the above series and parallel cases cases, it is straightforward to determine the score vector using (2.5).

Making the substitution $\rho = \prod_{j=1}^{p}\rho_j$ in eqn. (2.5), the j = 1, 2, ..., p elements of the score vector in (2.4) (or (2.5)) for the series case are:

$$\frac{\partial\mathcal{L}}{\partial\rho_j} = \frac{X + X_j}{\rho_j} - \frac{(n-X)\rho}{(1-\rho)\rho_j} - \frac{(n_j - X_j)}{(1-\rho_j)}. \tag{2.6}$$

Likewise, making the substitution $1-\rho = \prod_{j=1}^{p}(1-\rho_j)$ in eqn. (2.5) (i.e., $\rho = h(\boldsymbol{\theta}) = 1 - \prod_{j=1}^{p}(1-\rho_j)$), the elements of the score vector in (2.4) (or (2.5)) for the parallel case are:

$$\frac{\partial\mathcal{L}}{\partial\rho_j} = \frac{X_j}{\rho_j} + \frac{X}{\rho}\prod_{\substack{i=1\\ i\neq j}}^{p}(1-\rho_i) - \frac{(n + n_j - X - X_j)}{(1-\rho_j)}. \tag{2.7}$$

Except for the degenerate settings of $X = n$ in the series case and $X = 0$ in the parallel case, the solution to $\partial\mathcal{L}(\boldsymbol{\theta})/\partial\boldsymbol{\theta} = \mathbf{0}$ must generally be found by numerical search methods. (The two degenerate cases yield $\hat{\rho}_j = (n + X_j)/(n + n_j)$ and $\hat{\rho}_j =$

$X_j/(n+n_j)$, respectively, both of which are the natural—intuitively obvious—solutions.)

3. DEPENDENT SUBSYSTEMS

There are obviously innumerable ways in which subsystems can interact while operating as part of a full system. The reliability analysis for each such system must be handled separately based on the information available. While the reliability analysis with operationally dependent subsystems is usually more difficult than with operationally independent subsystems, the MLE may still be available if the problem can be cast in the form of (2.1) with an appropriate constraint. For example, in systems that may be represented as a series of $m \le p$ (generally dependent) subsystems, the following expression relates the full system reliability to conditional subsystem reliabilities:

$$\begin{aligned}\rho = \rho_1 P(S_2 = 1 \mid S_1 = 1) P\left(S_3 = 1 \middle| \{S_1 = 1\} \cap \{S_2 = 1\}\right) \\ \cdots P\left(S_m = 1 \middle| \bigcap_{j=1}^{m-1} \{S_j = 1\}\right),\end{aligned} \tag{3.1}$$

where S_j = 0 or 1 is the indicator of whether subsystem j is, respectively, a failure or success and $\rho_1 = P(S_1 = 1)$. Hence, the analyst may have the information needed to implement (2.1) with an equality constraint that uniquely defines ρ if data and/or prior information are available to characterize the conditional probabilities on the right-hand side of (3.1).

We illustrate here an application of (3.1) in the relatively common setting where dependence gets introduced through shared components. That is, it is assumed that two or more subsystems share at least one piece of hardware or software while the system is operating (e.g., a battery may provide power to more than one subsystem). Note that this setting differs from that in Jin and Coit (2001), which assumes that multiple subsystems share the same *type* of component, but not the same specific component.

Let us consider the example of $m = p - 1 \ge 2$ subsystems illustrated in Figure 3.1, with C_1, C_2, ..., C_p representing p independent components within the subsystems, but with component C_p appearing in all subsystems. With the possible exception of tests on this shared component (see the discussion on constraints below), it is assumed that testing can only be done at the subsystem level (i.e., it is not feasible to test each component separately).

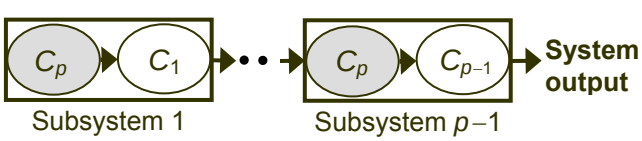


Figure 3.1. Series system with dependent subsystems. Dependence follows from shared component C_p in all $p-1$ subsystems.

It is necessary to formulate the constraint $f(\cdot) \ge 0$ in (2.1) in order to produce the MLE for the setting of Figure 3.1 (recall that the test data are assumed to satisfy the standard independence assumptions associated with the likelihood function in (2.2) as appears in the top line of (2.1)). We now show two methods by which the constraint can be implemented, with the two methods depending on the level of information available. Analogous to the indicator variable S_j = 0 or 1 in (3.1), let C_j = 0 or 1 indicate the failure or success of component j (so the notation C_j represents both a label for the jth component and the associated indicator variable). Let $\rho_j = P(S_j = 1)$, j = 1, 2, ..., $p-1$, and $\rho_p = P(C_p = 1)$.

To derive the constraints via the two methods, we see from (3.1) that

$$\rho = \rho_1 \frac{\rho_2}{\rho_p} \frac{\rho_3}{\rho_p} \cdots \frac{\rho_{p-1}}{\rho_p} = \frac{1}{\rho_p^{p-1}} \prod_{j=1}^{p-1} \rho_j, \tag{3.2}$$

where we have used the independence of the components. Note also that (3.2) is analogous (at the component level) to the standard series subsystem formulation. That is, $\rho = \prod_{j=1}^{p} P(C_j = 1)$, but that this component-series equality is not directly usable in (2.1) due to the above-mentioned restriction that it is not feasible to test each component separately. The form in (3.2), however, *is* implementable subject to obtaining information about the reliability for the shared component C_p.

The preferred method for estimation is based on separate testing of the common component. Here $\boldsymbol{\theta} = [\rho_1, \rho_2, \ldots, \rho_p]^T$, as usual, implying that the estimation involves the $p-1$ subsystems and the shared component. Then, from (3.2), the formulation in (2.1) applies directly with constraint $f(\boldsymbol{\theta}, \rho) \equiv \rho - \rho_p^{1-p} \prod_{j=1}^{p-1} \rho_j = 0$ and associated log-likelihood (2.3) and score vector (2.5) (with $\rho = h(\boldsymbol{\theta}) = \rho_p^{1-p} \prod_{j=1}^{p-1} \rho_j$).

The second method by which the constraint can be formulated applies when it is not possible to isolate the shared component for separate testing. Here, one simply sets $\rho_p = 1$ in (3.2), yielding a *lower bound* to ρ. That is, (2.1) involves the maximization of $\mathcal{L}(\boldsymbol{\theta})$ subject to $\rho \ge \prod_{j=1}^{p-1} \rho_j$ (i.e., $f(\boldsymbol{\theta}, \rho) \equiv \rho - \prod_{j=1}^{p-1} \rho_j \ge 0$), where the definition of $\boldsymbol{\theta}$ is modified to remove ρ_p (so $\boldsymbol{\theta}$ is now a vector of dimension $p-1$). Note that this method is most appropriate when the reliability of C_p is high, especially if p is large (so that the inequality is at least moderately tight).

4. THEORETICAL PROPERTIES

This section summarizes the convergence properties associated with the MLE formulation above. Note that standard i.i.d. MLE theory (e.g., Serfling, 1980, Sect. 4.2) does not apply because of the different success/failure probabilities associated with the different subsystems. Nevertheless, the structure associated with (2.1) and (2.3) allows us to show that the MLE for ρ will converge to the true full system reliability under reasonable conditions.

First, however, we present a result giving conditions under which there is a unique function $h(\boldsymbol{\theta})$ relating θ to ρ.

Lemma 1. Suppose that the constraint in problem statement (2.1) can be represented as an equality $f(\boldsymbol{\theta}, \rho) = 0$ with f being a continuously differentiable function in both $\boldsymbol{\theta} \in \Theta$ and $0 < \rho < 1$. For a fixed $\boldsymbol{\theta}' \in \Theta$, suppose $\partial f(\boldsymbol{\theta}', \rho)/\partial \rho \ne 0$ almost surely (a.s.) at ρ such that $f(\boldsymbol{\theta}', \rho) = 0$. Then there exists an open neighborhood of $\boldsymbol{\theta}'$ and a unique continuously differentiable function h such that for all $\boldsymbol{\theta}$ in this neighborhood, $\rho = h(\boldsymbol{\theta})$ and

$$\frac{\partial h(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\frac{\partial f(\boldsymbol{\theta},\rho)}{\partial \boldsymbol{\theta}}\left(\frac{\partial f(\boldsymbol{\theta},\rho)}{\partial \rho}\right)^{-1} \tag{4.1}$$

Proof. From the fact that the constraint in (2.1) can be represented as an equality $f(\boldsymbol{\theta}, \rho) = 0$ and the fact that $\partial f(\boldsymbol{\theta}',\rho)/\partial\rho \neq 0$ for a given $\boldsymbol{\theta}'$, the implicit function theorem (e.g., Apostol, 1974, Sect. 13.4) ensures that there exists an open neighborhood of $\boldsymbol{\theta}'$ and one, and only one, continuously differentiable function h such that for all $\boldsymbol{\theta}$ in this neighborhood, $\rho = h(\boldsymbol{\theta})$. Further, $\partial h/\partial\boldsymbol{\theta}$ is as shown in (4.1) in this neighborhood (e.g., Trench and Kolman, 1972, p. 483). *Q.E.D.*

We now present the following strong (a.s.) convergence result for the MLE of ρ. Let ρ^* be the true value of the full system reliability and $\boldsymbol{\theta}^* = [\rho_1^*, \rho_2^*, \ldots, \rho_p^*]^T$ be the corresponding true value of $\boldsymbol{\theta}$. Further, let $\boldsymbol{A}_N = \operatorname{diag}\left[\operatorname{var}(\partial\mathcal{L}/\partial\rho_1), \operatorname{var}(\partial\mathcal{L}/\partial\rho_2), \ldots, \operatorname{var}(\partial\mathcal{L}/\partial\rho_p)\right]$ be a diagonal matrix used to normalize for the variability of the elements in $\partial\mathcal{L}/\partial\boldsymbol{\theta}$, where $N = \{n, n_1, n_2, \ldots, n_p\}$ is the collective sample size. Proposition 1 below establishes conditions for $\hat{\rho} = \rho^* + o(1)$ being a root of the normalized score equation, $\boldsymbol{A}_N^{-1}\partial\mathcal{L}/\partial\boldsymbol{\theta} = \mathbf{0}$, where $o(1)$ is a term going to zero as N gets large in the sense described below. As usual when working with score vectors, however, the Proposition does not guarantee that this solution is unique and/or a global maximum of $\mathcal{L}(\boldsymbol{\theta})$.

Proposition 1. For constants $0 < C^- \leq C^+ < 1$, suppose the feasible region Θ is such that $\Theta = \{\boldsymbol{\theta}: C^- \leq \rho_j \leq C^+ \text{ for all } j\}$ and that $C^- \leq \rho \leq C^+$. Further, suppose that the constraint in problem statement (2.1) can be represented as an equality $f(\boldsymbol{\theta}, \rho) = 0$ with f being a continuously differentiable function in both $\boldsymbol{\theta} \in \Theta$ and $C^- \leq \rho \leq C^+$. For some strictly positive constants C and C' and all $\boldsymbol{\theta} \in \Theta$ and $C^- \leq \rho \leq C^+$, suppose $C < |\partial f(\boldsymbol{\theta},\rho)/\partial\rho| \leq C'$ and $C \leq |\partial f(\boldsymbol{\theta},\rho)/\partial\rho_j| \leq C'$ for all j. Then, for the problem described in (2.1) and (2.3), $\hat{\rho} = \rho^* + o(1)$ is an a.s. solution to $\boldsymbol{A}_N^{-1}\partial\mathcal{L}/\partial\boldsymbol{\theta} = \mathbf{0}$ as $n + n_1 \to \infty$, $n + n_2 \to \infty$, ..., $n + n_p \to \infty$, where, for each j, one of the following three possibilities holds: (i) $n_j/n = o(1)$, (ii) $n/n_j = o(1)$, or (iii) $n_j/n = O(1)$ and $n/n_j = O(1)$.

Comment. The multiple limits $n + n_1 \to \infty$, $n + n_2 \to \infty$, ..., $n + n_p \to \infty$ are true if and only if one of the following three (mutually exclusive) possibilities occur: (a) $n \to \infty$ and $n_j < \infty$ for all j, (b) $n < \infty$ and $n_j \to \infty$ for all j, or (c) $n \to \infty$ and $n_j \to \infty$ for at least one j. The proof considers these three cases in turn subject to the additional constraints (i) – (iii) in the Proposition statement.

Proof. Note that the conditions of the Proposition are stronger than those of Lemma 1. Hence, there exists a differentiable function h such that for all $\boldsymbol{\theta}$ in an open neighborhood of $\hat{\boldsymbol{\theta}}$, $\rho = h(\boldsymbol{\theta})$ with derivative given by (4.1). Thus, the score vector is given by (2.5).

As stated in the comment following the Proposition statement, there are three cases to consider: (a), (b), and (c). In case (a) ($n \to \infty$ and $n_j < \infty$ for all j), it is apparent from (4.1), (2.5), and the fact that the definition of Θ implies that the terms inside the square brackets on the right-hand side of (2.5) are bounded (independent of n) and that $\operatorname{var}(\partial\mathcal{L}/\partial\rho_j)$ grows at rate proportional to n for any fixed $\boldsymbol{\theta}$ and ρ. Hence, in solving for the MLE of $\rho = h(\boldsymbol{\theta})$, we know that

$$\lim_{n\to\infty}\frac{1}{n}\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}} = \lim_{n\to\infty}\frac{1}{n}\left(\frac{X}{\rho}\frac{\partial h}{\partial\boldsymbol{\theta}} - \frac{n-X}{1-\rho}\frac{\partial h}{\partial\boldsymbol{\theta}} + O(1)\right) = \mathbf{0} \text{ a.s.}$$

(i.e., $\lim_{n\to\infty}\boldsymbol{A}_N^{-1}\partial\mathcal{L}/\partial\boldsymbol{\theta} = \mathbf{0}$ a.s. according to (2.3) and the assumptions in the Proposition statement), leading to the MLE,

$$\hat{\rho} = \frac{X}{n} + O\left(\frac{1}{n}\right) \to \rho^* \text{ a.s.,} \tag{4.2}$$

where we have used the fact that $|\partial h(\boldsymbol{\theta})/\partial\rho_j| \geq C(C')^{-1} > 0$, as follows by (4.1) and the assumptions $|\partial f(\boldsymbol{\theta},\rho)/\partial\rho| \leq C'$ and $|\partial f(\boldsymbol{\theta},\rho)/\partial\rho_j| \geq C$.

The second case (b) ($n < \infty$ and $n_j \to \infty$ for all j) follows a pattern analogous to the above. Namely, for the jth component, $\operatorname{var}(\partial\mathcal{L}/\partial\rho_j)$ grows at rate proportional to n_j for any fixed $\boldsymbol{\theta}$ and ρ, and, therefore, the MLE of each ρ_j can be found as the solution to

$$\lim_{n_j\to\infty}\frac{1}{n_j}\frac{\partial\mathcal{L}}{\partial\rho_j} = \lim_{n_j\to\infty}\frac{1}{n_j}\left(O(1) + \frac{X_j}{\rho_j} - \frac{n_j - X_j}{1-\rho_j}\right) = \mathbf{0},$$

where the $O(1)$ term follows from $C^- \leq \rho \leq C^+$ and the fact that $|\partial h(\boldsymbol{\theta})/\partial\rho_j| \leq C^{-1}C' < \infty$. This yields

$$\hat{\rho}_j = \frac{X_j}{\rho_j} + O\left(\frac{1}{n_j}\right) \to \rho_j^* \text{ a.s.} \tag{4.3}$$

From Lemma 1 and the fact that $f(\boldsymbol{\theta}^*, \rho^*) = 0$, there exists an open neighborhood of $\boldsymbol{\theta}^*$ and a unique continuously differentiable function h such that for all $\boldsymbol{\theta}$ in this neighborhood, $\rho = h(\boldsymbol{\theta})$ and $\rho^* = h(\boldsymbol{\theta}^*)$. Hence, because (4.3) holds for all j, we know that for all n_j sufficiently large, $\hat{\boldsymbol{\theta}}$ can be made sufficiently close to $\boldsymbol{\theta}^*$ to guarantee that $\hat{\boldsymbol{\theta}}$ lies in this open neighborhood a.s., implying, by the continuity of $h(\boldsymbol{\theta})$ at $\boldsymbol{\theta}^*$, $\hat{\rho} = h(\hat{\boldsymbol{\theta}}) \to \rho^*$ a.s. when $n < \infty$ and $n_j \to \infty$ for all j.

Finally, for the third case (c) ($n \to \infty$ and $n_j \to \infty$ for at least one j), consider a j with $n_j \to \infty$ (the case (a) results above apply for those j with $n_j < \infty$). From (2.5), it is known that $\operatorname{var}(\partial\mathcal{L}/\partial\rho_j)$ grows at rate proportional to $n + n_j$. Hence, the normalized gradient (score vector) with respect to ρ_j is as follows:

$$\begin{aligned}\frac{1}{n+n_j}\frac{\partial\mathcal{L}}{\partial\rho_j} &= \frac{n}{n+n_j}\left(\frac{X/n}{\rho}\frac{\partial h}{\partial\rho_j} - \frac{1-X/n}{1-\rho}\frac{\partial h}{\partial\rho_j}\right) \\ &\quad + \frac{n_j}{n+n_j}\left(\frac{X_j/n_j}{\rho_j} - \frac{1-X_j/n_j}{1-\rho_j}\right).\end{aligned} \tag{4.4}$$

When $n_j/n = o(1)$ (see the assumption in the Proposition statement), the first $(\cdot)$ term on the right-hand side of (4.4) dominates as $n \to \infty$ and $n_j \to \infty$. Because $|\partial h(\boldsymbol{\theta})/\partial\rho_j| \geq C(C')^{-1} > 0$ (see below (4.2)) and because $X/n \to \rho^*$ a.s. as $n \to \infty$, we know that $\left[n + n_j\right]^{-1}\partial\mathcal{L}/\partial\rho_j \to 0$ a.s. as $n + n_j \to \infty$ if $\hat{\rho} \to \rho^*$.

When $n/n_j = o(1)$, the second $(\cdot)$ term in (4.4) dominates as $n \to \infty$ and $n_j \to \infty$ because $\left|\partial h(\boldsymbol{\theta})/\partial \rho_j\right| \le C^{-1}C' < \infty$. By the fact that $X_j/n_j \to \rho_j^*$ a.s. as $n_j \to \infty$, it is known that $\left[n+n_j\right]^{-1}\partial\mathcal{L}/\partial\rho_j \to 0$ a.s. as $n + n_j \to \infty$ if $\hat{\rho}_j \to \rho_j^*$ a.s. If this $n/n_j = o(1)$ setting applies for all j, then, as below (4.3), we know that $\hat{\rho} = h(\hat{\boldsymbol{\theta}}) \to \rho^*$ because $h(\boldsymbol{\theta})$ is continuous at $\boldsymbol{\theta}^*$ and $\rho^* = h(\boldsymbol{\theta}^*)$; if $n/n_j \ne o(1)$ for at least one j, then the case (a) analysis for (4.2) shows that $\hat{\rho} \to \rho^*$ a.s. directly when $n \to \infty$ and $n_j < \infty$ or the analysis to follow (n/n_j and n_j/n are $O(1)$) applies. That is, when n and n_j grow proportionally in the sense that both of n/n_j and n_j/n are $O(1)$, then both $(\cdot)$ terms in the right-hand side of (4.4) are significant as $n \to \infty$ and $n_j \to \infty$. From the above arguments, therefore, we know that $\left[n+n_j\right]^{-1}\partial\mathcal{L}/\partial\rho_j \to 0$ a.s. as $n + n_j \to \infty$ if $[\hat{\rho}_j, \hat{\rho}] \to [\rho_j^*, \rho^*]$ a.s., as desired. Because the case (a) analysis applies to those j such that $n \to \infty$ and $n_j < \infty$, we know that $\hat{\rho} \to \rho^*$ a.s. as $n \to \infty$ and $n_j \to \infty$ for at least one j.

The above establishes that $\hat{\rho} = \rho^* + o(1)$ is an a.s. solution to $\boldsymbol{A}_N^{-1}\partial\mathcal{L}/\partial\boldsymbol{\theta} = \mathbf{0}$, completing the proof of convergence for the three cases associated with $n + n_j \to \infty$ for all j. *Q.E.D.*

5. THE FISHER INFORMATION MATRIX

The Fisher information matrix is helpful in at least two respects in the reliability estimation problem: (i) It can be used to determine when the estimation problem in Section 2 is well posed (i.e., when $\boldsymbol{\theta}$ is identifiable) through an evaluation of the conditions ensuring that the FIM is positive definite (e.g., Goodwin and Payne, 1977, pp. 104 and 139) and (ii) the inverse (average) information matrix is the covariance matrix appearing in the asymptotic distribution of the appropriately normalized MLE. Hence, when combined with the asymptotic normal distribution, the information matrix may be used in constructing confidence regions for the MLE when the sample size is sufficiently large. More generally, the information matrix provides a summary of the amount of information in the data relative to $\boldsymbol{\theta}$ (e.g., Spall, 2003, Sect. 13.3). We restrict our attention below to the fully series and fully parallel subsystems cases (Subsection 2.2), but the analysis can be modified in a straightforward manner for certain other cases, including hybrid series-parallel subsystems cases.

The $p \times p$ Fisher information matrix $\boldsymbol{F}(\boldsymbol{\theta})$ for a twice-differentiable log-likelihood function is given by

$$\boldsymbol{F}(\theta) = E\left(\frac{\partial\mathcal{L}}{\partial\theta}\cdot\frac{\partial\mathcal{L}}{\partial\theta^T}\right) = -E\left(\frac{\partial^2\mathcal{L}}{\partial\boldsymbol{\theta}\,\partial\boldsymbol{\theta}^T}\right), \tag{5.1}$$

where $\partial^2\mathcal{L}/\partial\boldsymbol{\theta}\,\partial\boldsymbol{\theta}^T$ appearing after the last equality above corresponds to the Hessian matrix of the log-likelihood function.

Let us first compute $\boldsymbol{F}(\boldsymbol{\theta})$ for the series case using the Hessian-based form appearing after the second equality in (5.1). Because $\Theta = \{0 < \rho_j < 1 \text{ for all } j\}$, it is known that the Hessian matrix is continuous and, consequently, symmetric. From (2.6), the elements of the negative Hessian for the series case of interest are:

$$-\frac{\partial^2\mathcal{L}}{\partial\rho_j\partial\rho_k} = \begin{cases} \dfrac{X+X_j}{\rho_j^2} + \dfrac{(n-X)\rho^2}{(1-\rho)^2\rho_j^2} + \dfrac{(n_j-X_j)}{(1-\rho_j)^2} & \text{when } j=k, \\[2ex] \dfrac{(n-X)\rho}{(1-\rho)^2\rho_j\rho_k} & \text{when } j \ne k. \end{cases}$$

Then, the corresponding elements of the information matrix $\boldsymbol{F}(\boldsymbol{\theta}) = \left[F_{jk}(\theta)\right]$ are:

$$F_{jk}(\theta) = \begin{cases} \dfrac{n\rho + n_j\rho_j}{\rho_j^2} + \dfrac{n\rho^2}{(1-\rho)\rho_j^2} + \dfrac{n_j}{(1-\rho_j)} & \text{when } j=k, \\[2ex] \dfrac{n\rho}{(1-\rho)\rho_j\rho_k} & \text{when } j \ne k. \end{cases} \tag{5.2}$$

Likewise, the Hessian can be used to compute $\boldsymbol{F}(\boldsymbol{\theta})$ in the parallel subsystem case. From (2.7), the elements of the negative Hessian in the parallel case are:

$$-\frac{\partial^2\mathcal{L}}{\partial\rho_j\partial\rho_k} = \begin{cases} \dfrac{X_j}{\rho_j^2} + \dfrac{(n+n_j-X-X_j)}{(1-\rho_j)^2} + \dfrac{X(1-\rho)^2}{(1-\rho_j)^2\rho^2} & \text{when } j=k, \\[2ex] \dfrac{(1-\rho)(\rho^2 - X\rho + X)}{(1-\rho_j)(1-\rho_k)\rho^2} & \text{when } j \ne k, \end{cases}$$

leading to the following elements of the information matrix:

$$F_{jk}(\theta) = \begin{cases} \dfrac{n_j}{\rho_j} + \dfrac{n(1-\rho) + n_j\rho(1-\rho_j)}{(1-\rho_j)^2\rho} & \text{when } j=k, \\[2ex] \dfrac{(1-\rho)(\rho - n\rho + n)}{(1-\rho_j)(1-\rho_k)\rho} & \text{when } j \ne k. \end{cases} \tag{5.3}$$

One can use the expression above to determine if the information matrix is positive definite, thereby characterizing the identifiability of $\boldsymbol{\theta}$. It is clear that both n and the n_j can contribute to the positive definiteness of $\boldsymbol{F}(\boldsymbol{\theta})$. For example, if the n_j dominate n, then increasing all n_j at the same rate (in the sense that $n_j/n_k = O(1)$ and $n_k/n_j = O(1)$ for all j, k) is sufficient to achieve the positive definiteness for sufficiently large sample sizes. It is also possible to have $n \to \infty$ subject to the n_j growing sufficiently rapidly as well. In a practical application, one will have to assume values for $\boldsymbol{\theta}$ prior to carrying out the estimation in order to evaluate $\boldsymbol{F}(\boldsymbol{\theta})$; these values may be chosen conservatively or as "typical" values in determining identifiability.

The other main interest for application of the information matrix is determining approximate confidence regions. However, one of the complications in using the standard asymptotic normality results is that there are multiple samples sizes, n, n_1,..., n_p. Fortunately, the form for $\boldsymbol{F}(\boldsymbol{\theta})$ provides clarification with respect to this mix of sample sizes. Recall that the standard *generic* form for the asymptotic distribution of MLEs is,

$$\sqrt{\text{sample size}}\,(\text{MLE} - \text{true value}) \xrightarrow{\text{dist}} N(\mathbf{0}, \bar{\boldsymbol{F}}^{-1}) \tag{5.4}$$

where $\xrightarrow{\text{dist}}$ denotes convergence in distribution and $\bar{\boldsymbol{F}}$ is the limit of the mean information matrix (i.e., the limit of the information matrix averaged over the sample size) (e.g., Hoadley,

1971; Rao, 1973, pp. 415–417). In well-posed problems, $\bar{\boldsymbol{F}}$ is a finite-magnitude positive definite matrix. Hence, to within slower growth terms, the magnitude of the *unaveraged* Fisher information matrix (à la $\boldsymbol{F}(\boldsymbol{\theta})$ above) must grow linearly with the increase in the relevant sample size. In the context of the form for $\boldsymbol{F}(\boldsymbol{\theta})$ above, it is clear that both n and the n_j can contribute to the growth in magnitude for $\boldsymbol{F}(\boldsymbol{\theta})$ (for the "proper" asymptotic normality for $\hat{\boldsymbol{\theta}}$). For example, as above, if the n_j dominate n, then increasing all n_j at the same rate (in the sense that $n_j/n_k = O(1)$ and $n_k/n_j = O(1)$ for all j, k) is sufficient to achieve the necessary growth in the magnitude of $\boldsymbol{F}(\boldsymbol{\theta})$. It is also possible to have $n \to \infty$ subject to the n_j growing sufficiently rapidly as well.

6. BOOTSTRAP CONFIDENCE INTERVALS

It is well known that the traditional asymptotic normality-based methods are often inadequate in constructing confidence intervals for reliability estimates. Two factors contribute to this inadequacy: (i) sample sizes that are too small to justify the asymptotic normality and (ii) confidence intervals from the asymptotic normality that fall outside of the interval [0, 1] as a consequence of the need to approximate the true asymmetric distribution with the symmetric normal distribution (this is exacerbated by the fact that practical reliability estimates are often very near unity). We now present a bootstrap-based method for constructing confidence intervals for the full system reliability estimate $\hat{\rho}$ under the assumption that $\hat{\rho}$ is uniquely determined from $\hat{\boldsymbol{\theta}}$. Lemma 1 presented sufficient conditions for such a function via the implicit function theorem.

Bootstrap methods are well-known Monte Carlo procedures for creating important statistical quantities of interest when analytical methods are infeasible (e.g., Efron and Tibshirani, 1986; Ljung, 1999, pp. 304 and 334; and Aronsson et al., 2006). The steps below describe a "parametric bootstrap" approach to constructing confidence intervals for $\hat{\rho}$ (parametric bootstrap methods sample from a specified distribution based on using the estimated parameter values; a standard bootstrap method would sample from the raw data).

Step 0: Treat the MLE $\hat{\boldsymbol{\theta}}$, and associated $\hat{\rho}$, as the true value of $\boldsymbol{\theta}$ and ρ.

Step 1: Generate (by Monte Carlo) a set of bootstrap data of the same collective sample size $N = \{n, n_1, n_2, ..., n_p\}$ as the real data $\boldsymbol{Y}$ using the assumed probability mass function in (2.2) and the value of $\boldsymbol{\theta}$ and ρ from Step 0.

Step 2: Calculate the MLE of $\boldsymbol{\theta}$, say $\hat{\boldsymbol{\theta}}_{\text{boot}}$, from the bootstrap data $\boldsymbol{Y}$ in Step 1, and then calculate the corresponding full system reliability MLE, $\hat{\rho}_{\text{boot}}$.

Step 3: Repeat Steps 1 and 2 a large number of times (perhaps 1000) and rank order the resulting $\hat{\rho}_{\text{boot}}$ values; one- or two-sided confidence intervals are available by determining the appropriate quantiles from the ranked sample of $\hat{\rho}_{\text{boot}}$ values.

7. EXAMPLE: DEPENDENT SUBSYSTEMS

To illustrate the generality of the approach above, let us consider an example from Jin and Coit (2001), where the subsystems are arranged in series, but where they have shared components. The shared components introduce dependencies between the subsystems when the full system is operating. The system of interest is illustrated in Figure 3.1 above. This study is not included here due to space limitations; results are available from the author upon request.

8. CONCLUDING REMARKS

We have described above an MLE-based approach for estimating the reliability of a complex system by combining data from full system reliability tests and subsystem (or other) tests. The method applies in general systems, where the subsystems may be arranged in series, parallel (i.e., redundant), combination series/parallel, or other mode.

This MLE approach provides a means of estimating the reliability of systems with relatively few (or even no) full system tests while maintaining stated requirements for statistical confidence in the full system estimates through the knowledge obtained via subsystem tests. By appropriately formulating constraints in an optimization problem, the approach accommodates general relationships between the subsystems and full system, including statistical dependencies among subsystems operating within the full system. Interestingly, the MLE objective function (i.e., the likelihood or log-likelihood function) has the same general form across all settings; only the constraints in the optimization problem change.

We also provided conditions under which the MLE will be almost surely (a.s.) convergent to the true system reliability. The method includes asymptotic (Fisher information-based) and finite-sample (bootstrap) methods for characterizing the uncertainty via confidence regions.

9. ACKNOWLEDGEMENTS

This work was supported by the JHU/APL IRAD Program. The author thanks Dr. Coire Maranzano for helpful discussions and for verifying some of the calculations.

10. REFERENCES

[1] Apostol, T. M. (1974), *Mathematical Analysis* (2nd ed.), Addison-Wesley, Reading, MA.

[2] Aronsson, M., Arvastson, L. Holst, J., Lindoff, B., and Svensson, A. (2006), "Bootstrap Control," *IEEE Transactions on Automatic Control*, vol. 51(1), pp. 28–37.

[3] Coit, D. W. (1997), "System-Reliability Confidence-Intervals for Complex-Systems with Estimated Component Reliability," *IEEE Transactions on Reliability*, vol. 46(4), pp. 487–493.

[4] Easterling, R. G. (1972), "Approximate Confidence Limits for System Reliability," *Journal of the American Statistical Association*, vol. 67, pp. 220–222.

[5] Efron, B. and Tibshirani, R. (1986), "Bootstrap Methods for Standard Errors, Confidence Intervals, and Other Measures of Statistical Accuracy" (with discussion), *Statistical Science*, vol. 1, pp. 54–77.

[6] Goodwin, G. C. and Payne, R. L. (1977), *Dynamic System Identification: Experiment Design and Data Analysis*, Academic Press, New York.

[7] Hill, S. D. and Spall, J. C. (2007), "Variance of Upper Bounds in Inequality-Based Reliability," *Proceedings of the American Control Conference*, 11–13 July 2007, New York, NY, pp. 2320–2321 (paper ThA07.1).

[8] Hoadley, B. (1971), "Asymptotic Properties of Maximum Likelihood Estimates for the Independent Not Identically Distributed Case," *Annals of Mathematical Statistics*, vol. 42, pp. 1977–1991.

[9] Hwang, C. L., Tillman, F. A., and Lee, M. H. (1981), "System-Reliability Evaluation Techniques for Complex/Large Systems—A Review," *IEEE Transactions on Reliability*, vol. R-30, pp. 416–423.

[10] Jin, T. and Coit, D. (2001), "Variance of System-Reliability Estimates with Arbitrarily Repeated Components," *IEEE Transactions on Reliability*, vol. 50(4), pp. 409–413.

[11] Leemis, L. M. (1995), *Reliability: Probabilistic Models and Statistical Methods*, Prentice-Hall, Englewood Cliffs, NJ.

[12] Ljung, L. (1999), *System Identification—Theory for the User* (2nd ed.), Prentice-Hall PTR, Upper Saddle River, NJ.

[13] Myhre, J. M. and Saunders, S. C. (1968), "On Confidence Limits for the Reliability of Systems," *Annals of Mathematical Statistics*, vol. 39(5), pp. 1463–1472.

[14] Ramírez-Márquez, J. E. and Jiang, W. (2006), "On Improved Confidence Bounds for System Reliability," *IEEE Transactions on Reliability*, vol 55(1), pp. 26–36.

[15] Rao, C. R. (1973), *Linear Statistical Inference and Its Applications* (2nd ed.), Wiley, New York.

[16] Serfling, R. J. (1980), *Approximation Theorems of Mathematical Statistics*, Wiley, New York.

[17] Spall, J. C. (2003), *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*, Wiley, Hoboken, NJ.

[18] Tang, J., Tang, K., and Moskowitz, H. (1997), **"**Exact Bayesian Estimation of System Reliability from Component Test Data," *Naval Research Logistics*, vol. 44(1), pp. 127–146.

[19] Trench, W. F. and Kolman, B. (1972), *Multivariable Calculus with Linear Algebra and Series*, Academic, New York.

[20] Zehna, P. W. (1966), "Invariance of Maximum Likelihood Estimators," *Annals of Mathematical Statistics*, vol. 37, p. 744.

Fuzzy-Logic-Based Approach for Identifying Objects of Interest in the PRIDE Framework

Zeid Kootbally, Craig Schlenoff and Raj Madhavan*
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD, USA
{zeid.kootbally, craig.schlenoff}@nist.gov, raj.madhavan@ieee.org

Sebti Foufou
University of Burgundy
Laboratoire LE2i, UMR CNRS 5158
Dijon, France
sebti.foufou@u-bourgogne.fr

ABSTRACT

On-road autonomous vehicle navigation requires real-time motion planning in the presence of static and moving objects. Based on sensed data of the environment and the current traffic situation, an autonomous vehicle has to plan a path by predicting the future location of objects of interest. In this context, an object of interest is a moving or stationary object in the environment that has a reasonable probability of intersecting the path of the autonomous vehicle within a predetermined time frame. This paper investigates the identification of objects of interest within the PRIDE (PRediction In Dynamic Environments) framework. PRIDE is a multi-resolutional, hierarchical framework that predicts the future location of moving objects for the purposes of path planning and collision avoidance for an autonomous vehicle. Identifying objects of interest is an aspect of situation awareness and is performed in PRIDE using a dangerous zone, i.e., a fuzzy-logic-based approach representing a hazardous space area around an autonomous vehicle. Once objects of interest are identified, the risk of collision between the autonomous vehicle and each object of interest is then evaluated. To illustrate the performance of a dangerous zone within PRIDE, preliminary results are presented using a traffic scenario with the high-fidelity physics-based framework for the Unified System for Automation and Robot Simulation (USARSim).

Keywords

4D/RCS, autonomous vehicle, dangerous zone, fuzzy logic, fuzzy space, long-term prediction, moving object prediction, object of interest, PRIDE, situation awareness

*R & D Staff Member, Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA.

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA.
ACM ISBN 978-1-60558-293-1/08/08.

1. INTRODUCTION

Road traffic driving for autonomous vehicles (AVs) is continuing to gain traction both with researchers and practitioners. Funding for research in AVs has continued to grow over the past few years, and recent high profile funding opportunities have started to push theoretical research efforts into worldwide development of the most advanced projects. A leading example of the state of the art in autonomous driving is the Defense Advanced Research Projects Agency's (DARPA) series of grand challenges. In 2007, DARPA presented the Urban Challenge [1] (Victorville, CA, USA), a research and development program on AVs with the goal of developing technology that will keep warfighters off the battlefield and out of harm's way. This event required each participating team to build an AV capable of executing simulated military supply missions while merging into moving traffic, navigating traffic circles, etc.

As demonstrated by the DARPA challenges, one of the main goal of AVs is to reduce the number of casualties in traffic accidents. The National Highway Traffic Safety Administration (NHTSA) [12] reported 42 642 killed people and 2 575 000 injured in motor vehicle crashes for the year 2006. It was also reported that one cause (after speeding) of these accidents is misjudgment due to carelessness. To reduce fatalities, many efforts have led to the enhancements of road designs, imposition of laws and regulations, and improvement of situation awareness (SAw) of the drivers.

Consequently, much effort has been directed towards trying to understand the "human factors" component in vehicle accidents. As pointed out by Sukthankar [20], a primary challenge to create an AV that can competently drive in traffic is the task of tactical reasoning, i.e., the AV should be able to decide which actions to perform in a particular driving situation, in real-time, given incomplete information about the rapidly changing traffic configuration. Humans are able to understand highly dynamic and complex environments via their cognitive capabilities. One component of these cognitive capabilities is SAw, namely, the human's ability to perceive the environment, comprehend the situa-

tion, project that comprehension into the near future, and determine the best action to execute [5, 6]. Researchers' hypothesis is that an AV with human-like SAw capabilities should improve the mission success of future AV systems [2]. Research has shown that poor SAw is an important cause of driving accidents [22], hence, an AV should have good capability of early recognition of obstacles and danger prediction. Adopting the idea that SAw is a key component in driving safety, the AV community has given considerable amount of attention on this topic.

The research interest of this paper bears upon a level of SAw of how other vehicles in the environment are expected to behave considering their situation. When humans drive, they need to understand how each object in the environment moves according to the situation they find themselves in. To address this need, PRIDE (PRediction in Dynamic Environments), a multi-resolution hierarchical framework has been developed. This framework provides an AV planning system with information that it needs to perform path planning in the presence of moving objects [16]. PRIDE supports the prediction of the future location of moving objects at various levels of resolution, thus providing prediction information at the frequency and level of abstraction necessary for planners at different levels within the hierarchy.

This paper presents a fuzzy-logic-based methodology to identify objects of interest within a dangerous zone for an AV. A dangerous zone is defined as a space with a potential of hazard. Once objects of interest have been identified, the risk of collision is then evaluated for an AV with each object of interest inside the dangerous zone. A simulated scenario using the Unified System for Automation and Robot Simulation (USARSim) [4] shows preliminary results and demonstrates the performance of a dangerous zone within PRIDE for identifying objects of interest.

The remainder of this paper is organized as follows: section 2 gives an overview of the PRIDE framework. Section 3 describes SAw within PRIDE and goes into detail about objects of interest. Section 4 describes the concept of dangerous zone used to identify objects of interests in the environment. Section 5 discusses the performance of a dangerous zone and section 6 concludes this paper.

2. THE PRIDE FRAMEWORK

PRIDE is a multi-resolutional hierarchical framework that provides an AV planning system with information required to perform path planning in the presence of moving objects. This framework supports the prediction of the future location of moving objects at various levels of resolution. PRIDE is based on the 4D/RCS architecture [8], which provides a reference model for unmanned vehicles on how their software components should be identified and organized.

The PRIDE framework provides moving object predictions to planners running at any level of the 4D/RCS hierarchy at an appropriate scale and resolution. The underlying concept of PRIDE lies in the incorporation of multiple prediction algorithms into a single, unifying framework.

At the higher levels of the framework, the prediction of moving objects needs to occur at a much lower frequency and a greater level of inaccuracy is tolerable. At these levels, moving objects are identified as far as the sensors can detect and a long-term (LT) prediction algorithm predicts where those objects will be at various time steps into the future. Higher-level reasoning processes need a global representation of the environment to compute the future location of an AV. PRIDE uses the road network database [14] (RND) to access different information about the road networks, including individual lanes, lane markings, intersections, legal intersection traversability, etc. The lower levels of the framework use estimation theoretic short-term (ST) predictions based on an extended Kalman filter (EKF) to predict the future location of moving objects with an associated confidence measure. Complete details on the LT and ST prediction algorithms can be found in previous efforts [11, 16].

PRIDE currently integrates the Mobility Open Architecture Simulation and Tools (MOAST) framework along with USARSim [17]. This integration provides predictions incorporating the physics, kinematics and dynamics of AVs involved in traffic scenarios. MOAST is a framework that provides a baseline infrastructure for the development, testing, and analysis of autonomous systems[1]. MOAST implements a hierarchical control technique which decomposes the control problem into a hierarchy of controllers with each echelon (or level) of control adding additional capabilities to the system. USARSim is a high-fidelity physics-based simulation system that provides the embodiment and environment for the development and testing of autonomous systems. USARSim utilizes high-quality 3D rendering facilities to create a realistic simulation environment that provides the embodiment of a robotic system. The system architecture on the integration of PRIDE with the MOAST and USARSim frameworks is described in previous work [11].

PRIDE also handles drivers' aggressivity. In this context, the aggressivity represents the style and driving preferences of a driver. For example, one would likely assume that a conservative driver will remain in his lane whenever possible and will keep a gap between his vehicle and the leading vehicle. Conversely, an aggressive driver would have a higher probability of changing lanes and would be more apt to tailgate the leading vehicle. One may also find that the aggressivity of the driver may change over time, e.g., the driver can be very aggressive when trying to get to a certain lane, but become more passive when he gets there. The PRIDE framework addresses all the driver types and situations mentioned above. Experiments and corresponding results performed on aggressivity can be found in previous work [15].

3. SITUATION AWARENESS

To make assumptions of the future positions of moving objects, PRIDE has access to a level of SAw of how other vehicles in the environment are expected to behave considering the road traffic situation. An AV should be able to plan a path while avoiding any collision with obstacles or other moving objects on the road. The AV also requires knowledge of the environment and knowledge on the status of other objects in the environment to be able to drive tactically. The modeling of other vehicles is the most important aspect of tactical driving [20]. It is straightforward to model speed and relative positions, however, it is a challenging task to model the future behavior of the drivers.

SAw was first discussed in connection with pilot perfor-

[1]Autonomous systems in this context refer to embodied intelligent systems that can operate fairly independently from human supervision.

mance in air-to-air combat and was seen as the critical difference between fighter aces and ordinary pilots [7, 10]. Since its original conception, numerous definitions of SAw have been proposed. The work presented in this paper uses the formal definition from Endsley [6] where SAw is described as *[An expert's] perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future.*

3.1 Situation Awareness Model

The model of SAw within the PRIDE framework is being developed based on a three-level model provided by Endsley [6], as sketched in Figure 1.

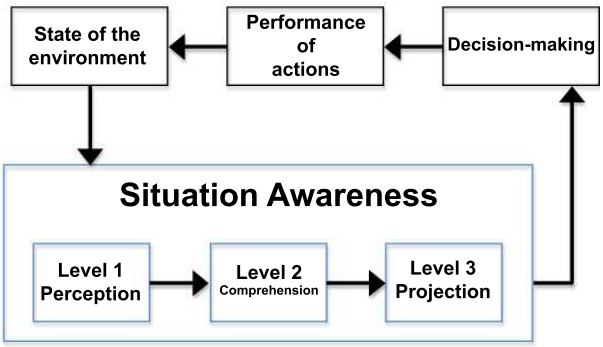


Figure 1: Endsley's model of situation awareness (from [19]).

- Perception: this level of awareness is achieved if AVs are able to perceive different elements (e.g., vehicles, roads) in the environment as well as their characteristics (e.g., size, color, location).
- Comprehension: not only the AVs must perceive relevant information in the environment, they also must combine the perceived data to interpret the situation correctly.
- Projection: at this level, the AVs have the ability to anticipate the actions of other vehicles and predict the future states of the environment.

The perception level for an AV in PRIDE is addressed through the MOAST/USARSim framework [17] and the RND [14]. MOAST queries USARSim to retrieve the characteristics of the AVs, such as the speed, the cartesian coordinates, the orientation, and the speed rotation of the wheels. PRIDE then compares the collected information to the RND to determine the position of the AV in the environment. The query returns the type of the road where the AV is positioned, the ID of the lane, the speed limit, and the traffic flow on the lane.

At the comprehension level, PRIDE combines the elements from the perception level to present a situation for the AV. For example, a vehicle with high speed (as compared to the speed limit of the road) and high acceleration could be considered as aggressive for example. Likewise, a vehicle driving at high speed toward the same uncontrolled intersection as the AV has a higher probability of collision than if its speed was lower. Hereby, PRIDE has the ability to understand the situation by gathering different information from different sources.

At the projection level, the LT prediction algorithm computes the future position of each AV by first computing all realistic action sequences. Then, based on a final cost for performing each action sequence, the LT algorithm chooses the action with the lowest cost, i.e., the one with the highest probability (see [16] for more details). The selected action sequence is based on the actions of other moving objects and on the situation of the AV itself. The output of the LT prediction algorithm is a collision-free path for the AV.

3.2 Object of Interest

The goal of PRIDE is to emulate human drivers' behaviors for AVs. As such, to achieve autonomous driving with human-like SAw capabilities in the presence of moving objects, the AVs have first to identify objects of interest in the environment. This section establishes the idea of identification of objects of interest, which is part of the *state of the environment* step, as depicted in Figure 1.

At its current state, PRIDE first takes into account all moving objects in the environment and then tests if any future collision is likely possible. It is obvious here that there is a need for identifying only specific objects (moving or static) and then evaluating the danger caused by each object. In real world, a driver pays attention to only a few objects around him, and obviously not to all of them. Since time constraints prevent processing all of this information at every time instant, the driver must intelligently select the information most critical to the immediate task. Focusing on some moving vehicles or static obstacles first reduces the computation time for collision, especially for a large number of vehicles and obstacles in the environment. Furthermore, identifying these specific objects constitutes a step further towards the simulation of a typical driving behavior. The AV should first focus on objects of interest in the environment that most constrain its available actions [13]. For example, when approaching an intersection with a STOP sign, the AV can safely ignore the trajectories of the vehicles beyond the intersection, since the STOP sign forces the AV to come to a halt. The AV should also make strong assumptions about objects in the environment. While observing an oncoming vehicle, the AV could note its position and velocity, then "forget" about the oncoming vehicle for some time interval, knowing that the vehicle would not be able to close the distance in that time. The AV focuses on particular objects at particular time in particular situations. These objects are termed "objects of interest" and can be defined as *a moving or stationary object in the environment that has a reasonable probability of intersecting the path of the autonomous vehicle within a predetermined time frame.* The identification of objects of interest is performed with the methodology presented in the next section.

4. METHODOLOGY TO IDENTIFY OBJECTS OF INTEREST

The methodology for the identification of objects of interest consists of two steps:

1. Building a dangerous zone around an AV to identify objects of interest.

2. Evaluating the risk of collision of the AV (called driving risk level) with any object of interest.

4.1 Dangerous Zone

Moving vehicles are subject to physical hazards coming from any direction, e.g., lateral impacts from the non-respect of rights-of-way at intersections or from a non-detected vehicle in the blind spot, and rear-end crashes usually due to inattention, following too closely, or both. Some of these accidents occur when the driver fails to maintain a safe headway from the leading car because of a perceptual inadequacy in estimating headways [21].

To effectively model the importance of an object on the road, PRIDE relates to the concept of dangerous zone (DZ) [18] to identify objects of interest in the space area around the AV. A DZ is defined as a space with a potential of hazard. Within the DZ, objects of interest have a different degree of risk according to different criteria such as the distance between an object of interest and an AV.

In conventional methods, the classical definition of "membership" puts an object either inside our outside a zone. The approach proposed in this paper tries to evaluate the degree of severity of an object within the DZ by classifying this object based on several criteria. One criterion would be for example, the closeness of an object of interest to the AV, which could be interpreted as close, very close, far, very far. As such, the effort of this paper describes a DZ by applying multi-dimension fuzzy sets to model gradual changes in collision severity. The concept of fuzzy space (FS) is used to present the spatial consideration fuzzy sets in two dimensions.

4.1.1 Fuzzy Space

The concept of FS is based on fuzzy logic and fuzzy sets. Fuzzy logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth. In 1965, Zadeh introduced fuzzy sets as an extension of the classical notion of a set to represent uncertain and imprecise knowledge [23]. Fuzzy sets and fuzzy logic are used to heuristically quantify the meaning of linguistic variables, linguistic terms, and linguistic rules that are specified by the expert.

Fuzzy logic uses graded statements rather than ones that are strictly true or false. Fuzzy logic attempts to incorporate the "rule of thumb"[2] approach generally used by human beings for decision making. Thus, fuzzy logic provides an approximate but effective way of describing the behavior of systems that are not easy to describe precisely.

Definitions.

To define a FS, the universe of discourse is in the form of $\mathbb{R}^2$. The work proposed in this paper considers the longitudinal distance x and the lateral distance y as linguistic variables from the relative coordinates of the AV. A typical linguistic variable is expressed as:

Linguistic Variable(*term 1, term 2,..., term n*)

where n is the number of terms in the linguistic variable.

To define a FS, let $\mathrm{X} \subset \mathbb{R}$, and $\mathrm{x}, \mathrm{y} \in \mathrm{X}$. $\mathrm{A_x}$ and $\mathrm{A_y}$ are the fuzzy sets for the degree of risk as defined below:

[2]A method of procedure based on experience and common sense.

$$\mathrm{A_x} = (\mathrm{x}, \mu_{\mathrm{A_x}}) | \mathrm{x} \in \mathrm{X}, \mathrm{X} \rightarrow [0,1]$$
$$\mathrm{A_y} = (\mathrm{y}, \mu_{\mathrm{A_y}}) | \mathrm{y} \in \mathrm{X}, \mathrm{X} \rightarrow [0,1]$$

Figure 2 depicts a trapezoidal membership function for the longitudinal direction as defined by Equation 1.

$$\mu_{\mathrm{A_x}}(\mathrm{x}) = \max(\min(\frac{\mathrm{x}-\mathrm{a}}{\mathrm{b}-\mathrm{a}}, 1, \frac{\mathrm{d}-\mathrm{x}}{\mathrm{d}-\mathrm{c}}), 0) \tag{1}$$

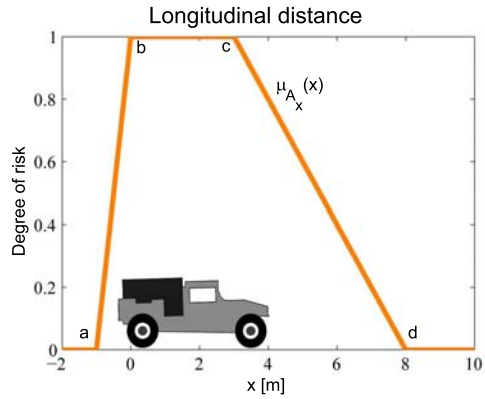


Figure 2: Longitudinal dangerous zone.

The parameters a and d correspond to the "feet" of the function $\mu_{\mathrm{A_x}}(\mathrm{x})$. The parameter d represents the safe distance headway of the AV. This headway is typically defined in terms of time rather than distance, and a commonly recommended minimum safe headway is 2 s. That way, if a lead driver initiates a braking action, the following driver has 2 s to initiate a braking response to the slowing down of the vehicle ahead. Using several parameters (e.g., current velocity of an AV, aggressivity, and weather), PRIDE converts the safe time headway into the corresponding distance headway.

Being able to modify the headway is an interesting point for the LT prediction algorithm. The LT algorithm computes the future location of moving objects at n seconds in the future [16]. So far, the time of prediction was established before running the simulation and could not be changed thereafter. With the ability to change the headway regarding the situation of the AV and the environment, the time of prediction is also modified in real-time. This subjects is further discussed in the rest of this paper.

The parameters b and c represent the range of the membership function for which the degree of risk is the highest, i.e., where the x values are closest to the AV. It is reasonable to use the length of the AV (information from USARSim) to define b and c.

For an object far from the AV, the risk of collision is low. Conversely, for an object closer to the AV, the risk of collision increases to its maximum. The function increases faster in the rear of the AV, thus describing a greater danger for a vehicle too close to the leading AV.

Figure 3 shows the bell shape membership function for the lateral distance as defined by Equation 2.

$$\mu_{\mathrm{A_y}}(\mathrm{y}) = \frac{1}{1 + |\frac{\mathrm{x}-\mathrm{c}}{\mathrm{a}}|^{2\mathrm{b}}} \tag{2}$$

The parameter b is linked to the width of the lane and is defined using the RND. The risk of collision grows for any

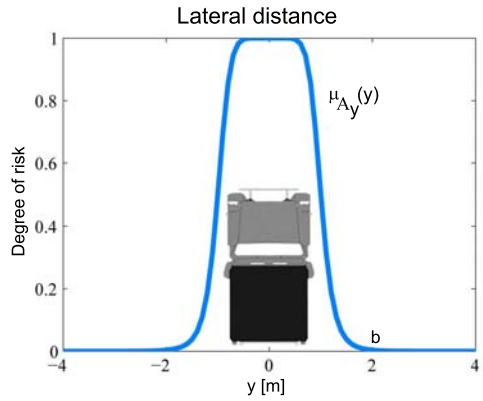


Figure 3: Lateral dangerous zone.

point approaching the AV on its sides. The parameter c locates the center of the curve (0 in this case). The value of a determines the membership values of the extreme points (0,1) of the universe when the crisp value, c, coincides with the center of the universe (0 in this case). The membership function $\mu_{A_y}(y)$ shows an increasing risk of danger for any object coming closer to the sides of the AV. Similar to the definition of the longitudinal distance, the membership function for the lateral distance has the ability to include the dimension of the AV. In this case, the width of the lane is taken into account, and thus the width of the AV within the lane.

Construction of the Fuzzy Space.

Different methods exist for the construction of FSs to describe DZs, such as the minimum intersection, the multiplication intersection, the rotational extension, and the cylindrical extension (see [9] for an exhaustive list).

In this paper, the construction of the FS A_{xy} is performed with the multiplication intersection method. The algebra multiplication $\mu_{A_x}(x)\ \mu_{A_y}(y)$ is preferred to the fuzzy logic multiplication $\min(\mu_{A_x}(x), \mu_{A_y}(y))$. As pointed out by Shahrokhi and Bernard [18], the minimum intersection fuzzy space is not sufficient to demonstrate all DZs. Furthermore, the algebra multiplication is more efficient for risk detection. The multiplication intersection fuzzy space A_{xy} for the longitudinal and lateral distances is defined by Equation 3 and is depicted in Figure 4.

$$A_{xy} = \mu_{A_x}\mu_{A_y} \tag{3}$$

It is important to understand that the fuzzy sets defined previously are not a standard for all road structures. The fuzzy sets are tuned according to the type of the road. The membership functions defined by Equations 1 and 2 are appropriate for a straight road. For a AV approaching an intersection for example, the fuzzy space can be spread over a larger area and can be represented by a semi-spherical shape incorporating some parts of the intersection.

4.2 Fuzzy Expert System

As discussed previously, the membership functions depend on different parameters, e.g., the aggressivity of the driver to compute the safe distance headway d in Equation 1 (Figure 2). The designer has to intelligently choose relevant parameters so that the FS could adapt to different situations. A fuzzy expert system is used with these parameters to compute the appropriate FS.

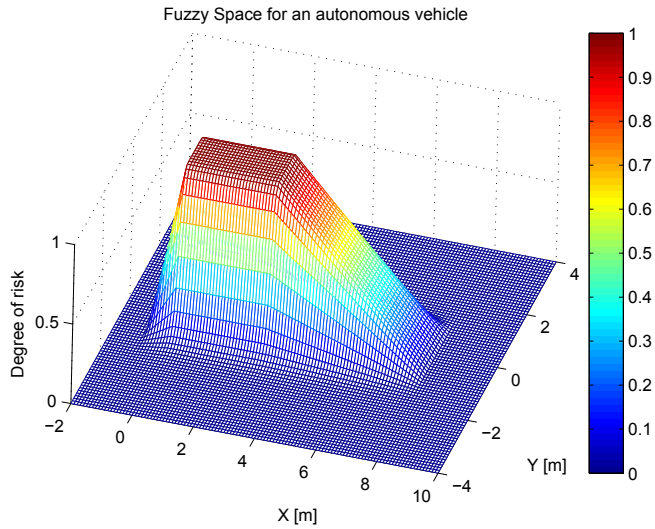


Figure 4: Fuzzy space A_{xy}.

Fuzzy expert systems are rule based controllers where the inference mechanism is grounded on fuzzy logic. The general architecture of a fuzzy expert system is depicted in Figure 5.

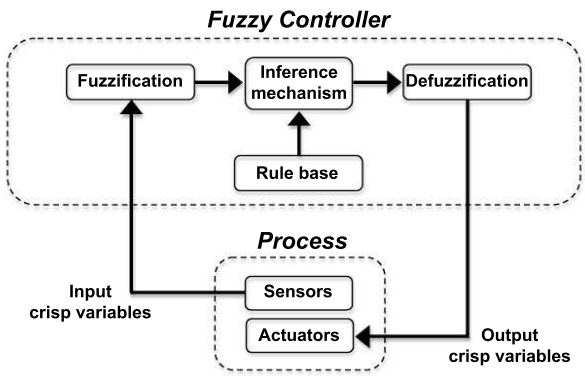


Figure 5: Architecture of a fuzzy expert system.

- The *fuzzification* module takes real input values (crisp values) and maps them to the terms by assigning a degree of membership. For continuous variable, the degree of membership is expressed by a membership function. There is a degree of membership for each linguistic term that applies to the linguistic input variable.
- The *rule base* holds the knowledge in the form of a set of rules, of how best to control the system. In general, fuzzy controllers are based on control rules of the type "IF `condition`" THEN "`control`" where `condition` and `control` are always fuzzy propositions (formula of fuzzy logic) of the type "x is A", where x is a linguistic variable and A is a linguistic term.

`condition` tells when the rules should be applied and `control` describes the action to apply.

- The *inference mechanism* is the kernel of the fuzzy controller. The inference mechanism evaluates which control rules are relevant at the current time and then decides the fuzzy commands to apply to the process.
- The *defuzzification* module is needed to translate the fuzzy output of a fuzzy controller to a numerical representation. Intuitively, defuzzification can be done using an averaging technique. The work described in this paper uses the center of gravity method [9], which is the same method employed to calculate the center of gravity of a mass.

4.2.1 Linguistic Variables

In this paper, PRIDE uses the aggressivity, the speed, the weather, and the acceleration as input linguistic variables and the safe distance headway as the output. The linguistic variables and the linguistic terms are presented in Table 1.

	Variables				
	Input				*Output*
	Speed	Aggressivity	Weather	Acceleration	Headway
Terms	Zero	Passive	Rainy	Zero	Very Low
	Small	Normal	Snowy	Small	Low
	Medium	Aggressive	Stormy	Medium	Medium
	Big		Sunny	Big	Big
					Very Big

Table 1: Linguistic variables used by the fuzzy expert system.

Once the linguistic variables are established, a set of rules for the inference mechanism has to be defined. Example of rules are shown below:

1. IF "`Speed is Small`" AND "`Aggressivity is Normal`" AND "`Weather is Sunny`" AND "`Acceleration is Zero`" THEN "`Headway is Low`".
2. IF "`Speed is Medium`" AND "`Aggressivity is Normal`" AND "`Weather is Snowy`" AND "`Acceleration is Small`" THEN "`Headway is Medium`".

4.3 Evaluation of Driving Risk Level

Any object of interest is likely to lead to a potential collision. To evaluate the driving risk level (DRL) of the AV, the distribution of the objects of interest within the DZ is taken into account. Each object of interest is represented by its position O_{x_i,y_i} in the environment. The DRL for each object of interest O_{x_i,y_i} is computed by maximizing the FS $A_{x_i y_i}$ as shown by Equation 4 [3].

$$\mathrm{DRL} = \max(A_{x_i y_i}) \tag{4}$$

Since the time of prediction coincides with the headway, the LT cost-based approach computes the cost for an AV to perform an action sequence. When an object of interest is identified in the DZ, the LT algorithm computes the cost of collision of the AV with the object of interest. At this point, this cost is modified by the value of the DRL computed using Equation 4.

5. PRELIMINARY RESULTS AND DISCUSSION

This section describes a traffic scenario and demonstrates the performance of the DZ for an AV. In the following scenario, the dimension of the AV is Length× Width=3.686 m×1.799 m (from USARSim). According to the RND, the width of the lane is 3.75 m. Finally, the weather is set to "snowy".

The chosen scenario is a lane-change maneuver over an obstacle. The AV and the static obstacle are in lane L1 as shown in Figure 6.

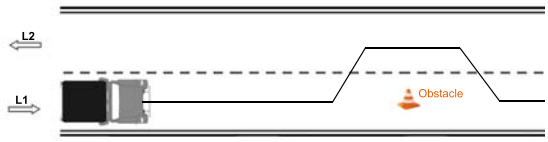


Figure 6: Vehicle avoiding a static obstacle.

Figure 7 shows the current positions of the AV and the static obstacle. The negative values for the Y coordinates are due to the coordinates of this particular road network in USARSim.

During its trajectory, the AV starts to switch to the left lane L2 at X=103.6 m and Y=-220 m. At this time, the distance between the AV and the static obstacle is approximately 5.5 m. The average speed of the AV on this track was 4 m/s. Since the weather is snowy for this scenario, the headway is greater (about 6.6 m) than it would be for a sunny weather.

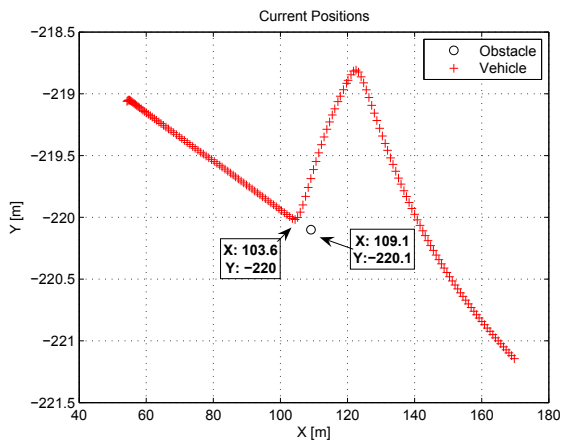


Figure 7: Current positions of the autonomous vehicle and the static obstacle.

Figure 8 depicts the variation of the DRL computed using Equation 4 with the static obstacle as object of interest. The negative values refer to the distance while the AV drives toward the obstacle (before reaching the obstacle), and the positive values indicate the distance when the AV drives away from the obstacle.

It can be seen that the DRL is null while the AV is far away from the obstacle, before and after passing the obstacle. During this time period, no object of interest is detected

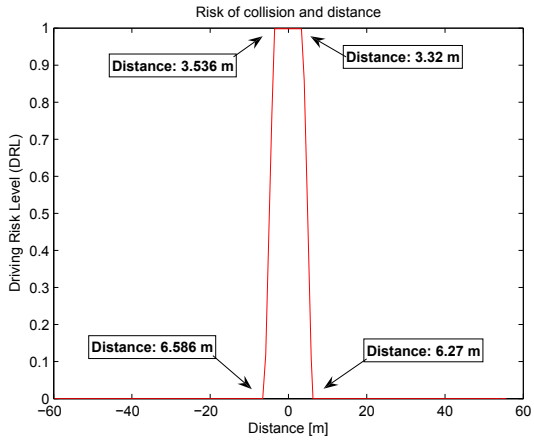


Figure 8: Driving risk level compared to the distance from the obstacle.

by the AV. However, at a distance of 6.586 m from the obstacle, the DRL starts to increase when the object of interest is identified within the DZ of the AV. The closer the AV is to the obstacle, the faster the DRL increases, meaning the higher is the risk of collision. The AV starts to gradually move to L2 as soon as the value of the DRL is high enough for a possible danger of collision. The DRL starts to increase at a distance of 6.5 m from the obstacle, however, the AV starts to switch to L2 only at a distance of 5.5 m from the obstacle, i.e., when the DRL is in [0.6 - 0.7] (Figure 8). The DRL value within this range modifies the cost associated to the straight path of the AV and thus the LT algorithm chooses a less expensive action, i.e., swerving to L2 in that case.

The DRL reaches its highest level (0.9982) for the AV being at around 3.5 m from the obstacle. While the AV is moving to L2, the value of the DRL decreases and reaches 0 at 6.27 m away from the obstacle. At this point, since no object of interest is identified within the DZ, the LT algorithm modifies the cost associated to the AV driving in lane L2. A penalty is given to the AV for not being in the right-most lane, hence the lane switching to L1 at X=120 m and Y=-218.7 m as depicted in Figure 7.

6. CONCLUSION

The work presented in this paper enhances situation awareness within the PRIDE framework by identifying objects of interest in the environment. Autonomous driving requires human-like situation awareness capabilities. Consequently, autonomous vehicles (AVs) must consider objects of interest in the environment in order to plan a collision-free trajectory. Identifying objects of interest can be assimilated to a driver who only focuses on some objects that most constrain his available actions.

The identification of objects of interest is performed by a dangerous zones (DZs). In this context, a DZ is a fuzzy space which represents a hazard area for an AV. The DZ is built by first assembling relevant parameters, which are then processed through a fuzzy expert system to adapt the fuzzy space to different situations. Any object that falls inside the DZ is identified as object of interest. Once the objects of interest are identified, the risk of collision of the AV is evaluated.

The fuzzy space has the advantages in considering the dimension of the AVs, thus improving collision avoidance. Another advantage is the modifications of the time of prediction for the LT prediction algorithm in real-time. This second point is useful to emulate driving tasks taking into account the current forecast and the variation of the aggressivity for example. Lastly, by first identifying objects of interest, and only then evaluating the danger pertinent to these objects, the time of computation of the LT algorithm is theoretically reduced, as compared with the former version of the LT algorithm.

The concept of DZ has demonstrated reasonable results with a new way to identify any danger in the environment. However, the preliminary results were obtained for a single AV on a simple straight road with a static obstacle. Identifying objects of interest in more complex traffic situations is a challenge and should be developed in the near future. Since the concept of DZ was first introduced in industrial systems, PRIDE already has the capacity of considering DZs before moving towards simulation in industrial facilities.

7. REFERENCES

[1] DARPA Urban Challenge 2007. http://www.darpa.mil/GRANDCHALLENGE/, 2007.

[2] J. A. Adams. Unmanned Vehicle Situation Awareness: A Path Forward. In *Proceedings of the 2007 Human Systems Integration Symposium*, 19–21 March 2007.

[3] H. Cai and Y. Lin. A Preliminary Study on a Fuzzy Driving Risk Model. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 2, pages 1785–1790, 10–12 October 2005.

[4] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper. USARSim: a Robot Simulator for Research and Education. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1400–1405, 10–14 April 2007.

[5] M. R. Endsley. Design and Evaluation for Situation Awareness Enhancement. In *Proceedings of the Human Factors Society 32 nd Annual Meeting*, volume 1, pages 97–101, 1988.

[6] M. R. Endsley. Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 37(1):32–64, March 1995.

[7] K. Harwood, B. Barnett, and C. Wickens. Situational Awareness: A Conceptual and Methodological Framework. In F.E. McIntire, editor, *Proceedings of the 11 th Biennial Psychology in the Department of Defense Symposium*, pages 23–27, 1988.

[8] J. Albus et al. 4D/RCS Version 2.0: A Reference Model Architecture for Unmanned Vehicle Systems. Technical Report NISTIR 6910, National Institute of Standards and Technology, 2002.

[9] R. Jager. *Fuzzy Logic in Control*. PhD thesis, Technische Universiteit Delft, The Netherlands, June 1995.

[10] S. Kass, D. Hershler, and M. Companion. Are They Shooting at Me: An Approach to Training Situational Awareness. In *Proceedings of Human Factors Society*

34 th Annual Meeting, pages 1352–1356, 1990.

[11] Z. Kootbally, C. Schlenoff, and R. Madhavan. A Brief History of PRIDE. In *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, NIST SP 1073, pages 40–47, August 2007.

[12] NHTSA. Traffic Safety Facts 2006: A Compilation of Motor Vehicle Crash Data from the Fatality Analysis Reporting System and the General Estimates System. Technical Report DOT HS 810 818, U.S. Department of Transportation, National Center for Statistics and Analysis, U.S. Department of Transportation, Washington, DC 20590, September 2007.

[13] D. A. Reece. *Selective Perception for Robot Driving*. PhD thesis, Carnegie Mellon University, May 1992.

[14] C. Schlenoff, S. Balakirsky, A. Barbera, C. Scrapper, and J. Ajot. The NIST Road Network Database: Version 1.0. Technical Report NISTIR 7136, National Institute of Standards and Technology, July 2004.

[15] C. Schlenoff, Z. Kootbally, and R. Madhavan. Driver Aggressivity Analysis within the Prediction In Dynamic Environments (PRIDE) Framework. In *Proceedings of the 2007 SPIE Defense and Security Symposium*, volume 6561, page 65611O, May 2007.

[16] C. Schlenoff, R. Madhavan, and Z. Kootbally. PRIDE: A Hierarchical, Integrated Prediction Framework for Autonomous On-Road Driving. In *Proceedings of the 2006 International Conference on Robotic Applications (ICRA)*, pages 2348–2353, 15–19 May 2006.

[17] C. Scrapper, S. Balakirsky, and E. Messina. MOAST and USARSim: A Combined Framework for the Development and Testing of Autonomous Systems. In *Proceedings of the 2006 SPIE Defense and Security Symposium*, volume 6230, 17–21 April 2006.

[18] M. Shahrokhi and A. Bernard. A Fuzzy Approach for Definition of Dangerous Zone in Industrial Systems. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, volume 7, pages 6318–6324, 10–13 October 2004.

[19] P. R. Smart, A. Bahrami, D. Braines, D. McRae-Spencer, J. Yuan, and N. R. Shadbolt. Semantic Technologies and Enhanced Situation Awareness. In *1st Annual Conference of the International Technology Alliance (ACITA)*, 25–27 September 2007.

[20] R. Sukthankar. *Situation Awareness for Tactical Driving*. PhD thesis, Robotics Institute, Carnegie Mellon University, January 1997.

[21] M. Taieb and D. Shinar. Minimum and Comfortable Driving Headways: Reality versus Perception. *Human Factors*, 43:159–172, 2001.

[22] J. R. Treat, N. S. Tumbas, S. T. McDonald, D. Shinar, R. D. Hume, R. E. Mayer, R. L. Stanisfer, and N. J. Castellan. Tri-Level Study of the Causes of Traffic Accidents. Technical report, National Highway Traffic Safety Administration, 1979.

[23] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

Identifying Objects in Range Data Based on Similarity Transformation Invariant Shape Signatures

Xiaolan Li[1,2] Afzal Godil[1] Asim Wagan[1]
[1] National Institute of Standard and Technology, Gaithersburg, MD 20899, U.S.A.
[2] Zhejiang Gongshang University, Hangzhou, Zhejiang 310018, China
Contact: lixlan@nist.gov

ABSTRACT
Identification and recognition of three dimensional (3D) objects in range data is a challenging problem. We propose a novel method to fulfill the task through two steps: 1) construct the feature signatures for the objects in the scene and the models in a 3D database; 2) based on the feature signature, find out the most similar model which decides the class of the corresponding object in the scene. We also evaluate the accuracy, robustness of the recognition method with several configurations. Our experimental results validate the effectiveness of our method.

Categories and Subject Descriptors
I.2.10 [**Vision and Scene Understanding**]: Shape, Representations, data structures, and transforms.

General Terms
Algorithms, Performance, Reliability.

Keywords
Object recognition, accuracy evaluation.

1. INTRODUCTION
With the development of sensor technology, laser scanners along with digital color cameras, sonar sensors and other sensors share the role of being "perception organs" of a robot. The action of the robot is highly dependent on the information obtained from the data collected by the "perception organs". Here is a search and rescue example. When there is a gas leak in a dark factory, the robot enters as a rescuer. However, color cameras cannot capture images because of the darkness. Laser scanning becomes the preferred method for the acquisition of the information. Having obtained the range data, the robot first differentiates people from other objects, and then, carries on different strategies according to the recognition results: helping the people, and avoiding other objects.

Besides the advantage in applications, such as robot localization and strategy choice, object recognition in a cluttered scene is an interesting and challenging problem in its own right. The problem is defined as follows: given a 3D point cloud produced by a laser scanner observing a 3D scene, the goal is to identify objects in the scene by comparing them to a set of candidate objects. This is closely related to 3D shape retrieval.

The main difference between 3D shape retrieval and the recognition problem here is that, for range data, only part of the object is captured by the scanner because of the limitation of the view angle and the occlusion. The situation of occlusion is complicated. For simplicity, we only focus on the case that the occlusion does not destroy the silhouette of the objects. As a result, a complete outline is preserved, which is used as a source to construct the shape representation of the object in the scene.

For the range data, we start by segmenting it into several regions. Then each region's data is projected into a plane perpendicular to the view direction of the scanner to get a silhouette. As for the candidate objects in a 3D shape repository, their silhouettes are captured from several views. Then Fourier Mellin Transform is performed on those images to extract similarity transform invariant 2D features. After that, a comparison is done between regional range data and the models in the database to get the most similar one. The regional range data is labeled after the chosen model.

The rest of this paper is organized as follows. Section 2 describes some of the related work to our proposed approach for identifying 3D objects in range data. The procedures of feature extraction and similarity comparison are described in Section 3. In Section 4, we provide the recognition results for simulated range data and evaluate the accuracy and robustness of the approach. Section 5 provides some conclusions.

2. RELATED WORKS
There exists extensive literature addressing 3D object recognition [Bustos05]. For simple scenes, it is straightforward to use several basic geometries to represent them, such as generalized cylinders [Binford71], superquadrics [Solina90], geons [Wu94], and so on. Unfortunately, this kind of representation is too abstract to describe complicated real 3D objects. Other sophisticated methods have been put forth, including visual similarity-based [Chen03] [Vranic03], geometric similarity-based [Osada02] [Papadakis07], topologic similarity-based [Biasotti04], and local region similarity-based [Frome04]. Our approach is closely related to the visual similarity-based methods. In this section, we give a brief description about this kind of approach.

Orthogonal projected silhouette image is the most popular when researchers are thinking about using the collection of 2D images to represent the 3D shape [Chen03] [Vranic03]. Chen et al. [Chen03] first captured 100 silhouettes with 10 different configurations of cameras mounted on 10 dodecahedrons. Then 35 Zernike moments coefficients and 10 Fourier coefficients calculated from one image are concatenated as one descriptor. After that, the similarities between objects are measured using a

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

(a) A scene. **(b) The corresponding range data.** **(c) segmented scene.**

Figure 1. The scene.

particular metric. Vranic [Vranic03] only use 6 silhouettes, which is comparably less than that in [Chen03], because of the pre-alignment procedure (PCA). Actually, the recognition power of visual similarity-based methods is strongly dependent on the 2D shape descriptors and the view direction used to capture the image.

Several approaches exist to describe the shape of an image, including geometric moments, complex moments, Legendre moments, Zernike moments, Fourier descriptors, etc. [Liao96]. Fourier and Zernike, which are a contour shape descriptor and a region shape descriptor respectively, are superior to the others according to the research of Zhang et al [Zhang02]. Nevertheless, they do not completely satisfy the invariance requirement with respect to similarity transformations (i.e. rotation, translation and scale). Under the Fourier-Mellin transform framework, which is widely used in image reconstruction and image retrieval, complete invariant descriptors can be derived [Yu07].

In this paper, we investigate a new method for 3D object recognition based on orthogonally projected silhouettes under the Fourier-Mellin transform framework.

3. IDENTIFICATION PROCEDURE

The whole identification procedure is divided into 2 main parts: constructing the 3D descriptors, and computing the similarity between the range data and the candidate objects in the database. However, a scene usually includes several objects. In order to figure out what each object is, we should segment the range data.

Actually, for the range data captured from a certain view direction, it can be regarded as an image whose resolution is equal to the scanner's resolution and whose pixel values record the depth from the surface of the objects to the scanner. Thus thousands of image segmentation methods can be used [Sezgin04]. A threshold approach is applied here to segment the range data. Figure 1(a) shows a scene, while figure 1(b) is the corresponding range data, and figure 1(c) displays the segmented result in which different color refers different object.

3.1. 3D descriptor construction

For each object in the scene, after the segmentation phrase, a silhouette is obtained. Three steps of Fourier-Mellin transform are performed on the silhouette to extract a similarity invariant feature vector, which is shown in figure 2.

1) A 2D FFT

$$F(u,v) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)e^{-j2\pi(ux+vy)}dxdy$$

is applied to the silhouettes.

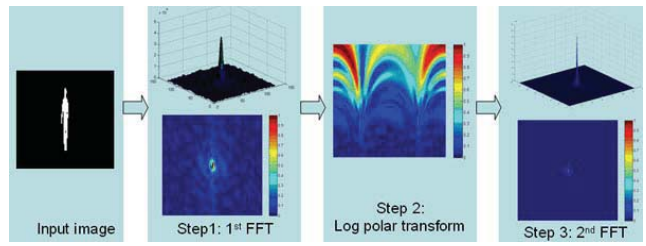


Figure 2. The procedure of FMT.

2) A log polar transform is performed on the images composed of the magnitudes of the Fourier coefficients. In this step, the resolution of the image can be changed, which defines the size of the final descriptor. The resolution is denoted as $M*M$.

3) Another 2D FFT is carried out on the log polar images to obtain the Fourier Mellin coefficients.

Because of the symmetry property of FFT, we choose the magnitudes of the Fourier Mellin coefficients located in the first quadrant as the descriptor, that is

$$FV = (c_1, c_2, \ldots, c_K) \quad (1)$$

where $K=M*M/4$.

To decrease the size of the feature vector, there is another solution to construct it. The coefficients in the first quadrant are summed up along x and y directions. Therefore, the form of feature vector keeps the same form as in equation (1), in which $K=M*2$.

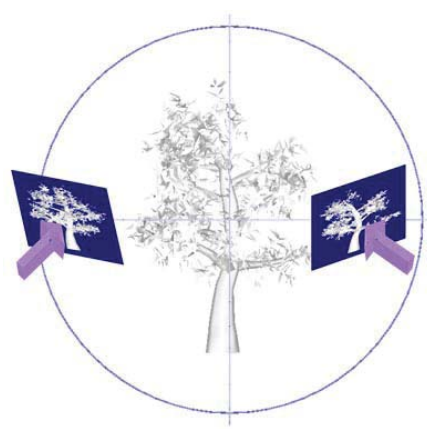
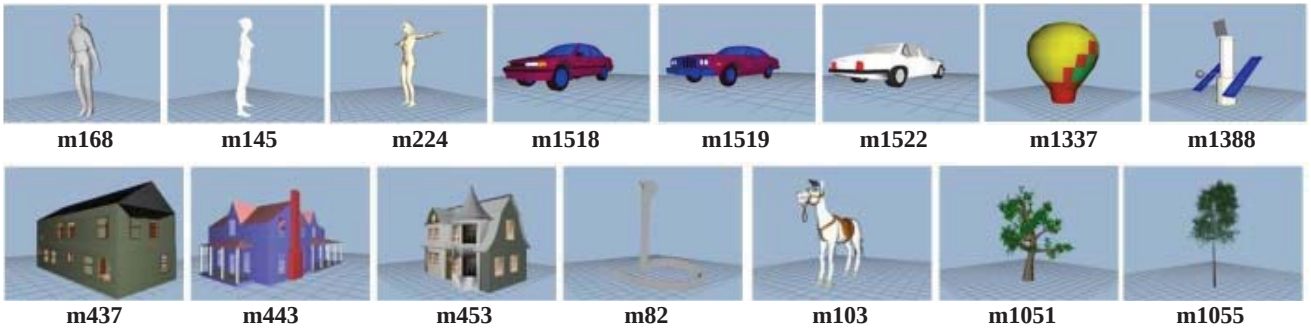


Figure 3. The demonstration for capturing several silhouettes from defined positions on the surface of bounding sphere.



In the reference data set, m168, m145, m224 belong to class "human"; m1518, m1519, m1522 belong to class "sedan"; m437, m443, m453 belong to class "two story house"; the others belong to different classes.

Figure 4. The reference data set.

For the candidate models, the construction of descriptor is similar to that of the objects in the scene. Nevertheless, more than one such descriptor is needed to give a complete description for the model. Several cameras are placed on the surface of a sphere to fulfill this goal, whose center is the center of the model and whose radius equals to the model's max radius (shown in figure 3). The positions are defined by the longitude and latitude:

$$(\theta_i, \varphi_j), 0 \le \theta_i < 2\pi, 0 \le \varphi_j < \pi \ , \qquad (2)$$

where i,j=1,2,…,N. As a result, the descriptor for one candidate model is denoted as an array:

$$FA = \begin{bmatrix} f_1^1 & f_2^1 & \cdots & f_K^1 \\ f_1^2 & f_2^2 & \cdots & f_K^2 \\ \vdots & \vdots & \vdots & \vdots \\ f_1^{N*N} & f_2^{N*N} & \cdots & f_K^{N*N} \end{bmatrix} \qquad (3)$$

3.2. Similarity computation

To compare descriptor FV eq. (1) with descriptor FA eq. (3), L1 distance measurement is used. The similarity is evaluated based on it:

$$sim_{l_1} = \min_j \left(\sum_{i=1}^{K} \left| f_i^j - c_i \right| \right), \qquad (4)$$

where j=1,2,…,N. The smaller the value is for equation (4), the more similar the object is to the candidate model.

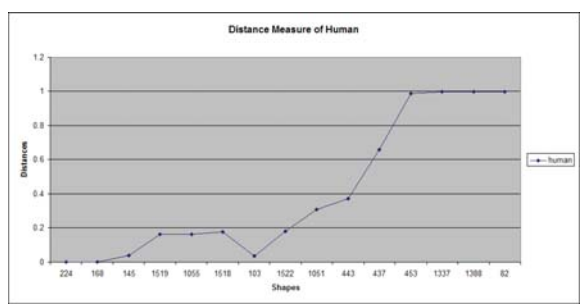
4. RESULTS AND EVALUATION

The recognition power of the method is tested on a set of reference 3D shapes made up of 15 models taken from the PSB 3D model library [Shilane04]. Figure 4 shows all of them. The point clouds used in our experiments were generated using a simulation program with resolution 256 by 256, which is regarded as query data. It is composed of one car (m1518), one person (m168) and one two-story house (m443) from the reference 3D model sets, which are all scaled to the normal size as in the real world in order to be reasonable.

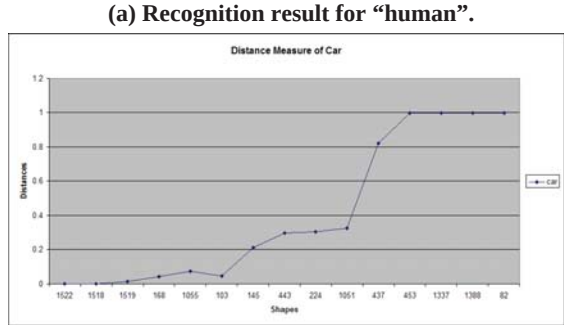
The recognition results are shown in figure 5 with N=10, M=64. The X axis represents all the reference models in the data set, while the Y axis indicates the distance between the query object and the models.

How the noise level affects the recognition results is discussed here. Take one object –house – in the query scene as an example, we add Gaussian noise to it and get the result with different noise levels. The recognition curves for different noise level are shown in figure 5. From the plot, we know that the noise will change the distance value a little bit, but the appearance of the curve keeps similar to the original one (the dark blue curve), which shows the robustness of our method.

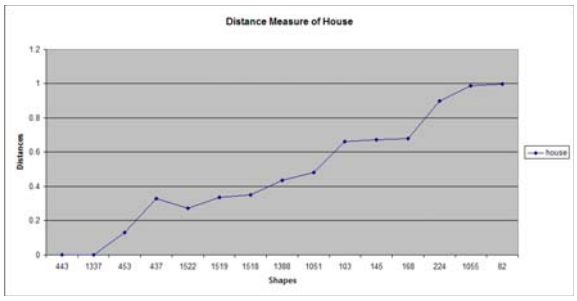
Since the range data can be extracted from different directions, the robustness of the algorithm related to the extracting directions should be evaluated. The 3D range scanner is placed on the surface of the bounding sphere of the scene. And the positions are defined by the longitude and latitude (eq. 2) with N=15, which means the amount of the range data from different directions is 225. The configuration guaranteed the differences between the shooting direction of the camera and the scanning direction of the range scanner. Taking the model of a person (m168) as an example, figure 6 shows the effect of the scanning directions, in which the x axis shows the name of the model and the y axis shows how many times the object is recognized as the model. It shows the accuracy is 87%.



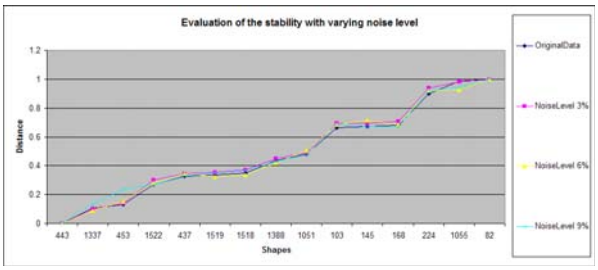
(a) Recognition result for "human".



(b) Recognition result for "Car".



(c) Recognition result for "House".



(d) Recognition result with different noise level for "House".

Figure 5. The recognition results (a)(b)(c) for all the objects in the scene. The X axis records the shapes from the reference set. (d) represents the recognition results for "House" with different noise level.

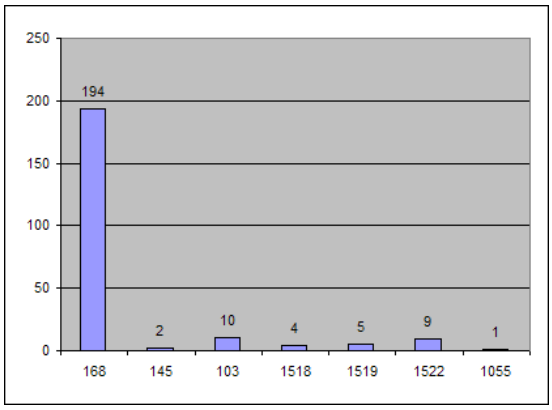


Figure 6. The effect of the shooting directions.

5. CONCLUSION

In this paper, we presented a novel procedure for 3D object recognition using the Fourier-Mellin transform. Although FMT is widely used in 2D image retrieval and reconstruction, it has not been used in 3D shape recognition. We have applied it to 3D shape recognition, and the experimental results show its effectiveness. Furthermore, how the noise level and the scanning direction affect the recognition result is investigated. Nevertheless, to guarantee the recognition accuracy, the completeness of the silhouette should be kept, which is rare in a cluttered and clustered scene. In future work, more stable local feature signatures will be introduced to alleviate the effect of cluttering and clustering.

ACKNOWLEDGEMENT

We would like to thank the SIMA program and the IDUS program for supporting this work.

REFERENCES

[Biasotti04] S. Biasotti. Reeb graph representation of surfaces with boundary. Proc. of the Shape Modeling International 2004 (SMI'04), 2004.

[Binford71] T.O. Binford. Visual perception by computer. IEEE Conf. on Systems and Control, Miami, FL, 1971.

[Bustos05] B. Bustos, D. A. Keim, D. Saupe, T. Schreck, D. V. Vranic. Feature-based similarity search in 3D object databases. ACM Computing Surveys (CSUR), 37(4):345-387, 2005.

[Chen03] D.Y. Chen, M. Ouhyoung, X.P. Tian, Y.T. Shen. On visual similarity based 3D model retrieval. Computer Graphics Forum (Eurographics '03), 03: 223-232, 2003.

[Frome04] A. Frome, D. Huber, R. Kolluri, T. Bulow, J. Malik. Recognizing objects in range data using regional point descriptors. Proc. of European Conference on Computer Vision (ECCV), Prague, Czech Republic, 2004.

[Liao96] S. X. Liao, M. Pawlak. On image analysis by moments. IEEE Trans. Pattern Analysis and Machine Intelligent, 18(1996): 254-266, 1996.

[Osada02] R. Osada, T. Funkhouser, B. Chazelle, D. Dobkin. Shape distributions. ACM Transaction on Graphics, 21(4):807-832, 2002.

[Papadakis07] P. Papadakis, I. Pratikakis, S. Perantonis, T. Theoharis. Efficient 3D shape matching and retrieval using a concrete radicalized spherical projection representation. Pattern Recognition, 40(2007):2437-2452, 2007.

[Sezgin04] M. Sezgin, B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging, 13(1), 146-165, Jan. 2004.

[Shilane04] P. Shilane, P. Min, M.Kazhdan, T. Funkhouser. The Princeton shape benchmark. *Proc. of the Shape Modeling International 2004 (SMI'04)*, 04(00): 167-178, 2004.

[Solina90] F. Solina, R. Bajcsy. Recovery of parametric models from range images: the case for superquadrics with global deformations. IEEE Trans. On Pattern Analysis and Machine Intelligence (PAMI), Feb. 1990.

[Vranic03] D.V. Vranic. 3D model retrieval. Ph. D. Dissertation, 2003.

[Wu94] K. Wu, M. Levine, Recovering parametric geons from multiview range data. CVPR, June, 1994.

[Yu07] H. Yu, M. Bennamoun. Complete invariants for robust face recognition. Pattern Recognition, 40(2007):1579-1591, 2007.

[Zhang02] D. S. Zhang, G. Lu. An integrated approach to shape based image retrieval. Proc. of 5th Asian Conf. on Computer Vision (ACCV), 02: 652-657, January 2002.

Stepfield Pallets: Repeatable Terrain for Evaluating Robot Mobility

A. Jacoff, A. Downs, A. Virts, E. Messina
Intelligent Systems Division, National Institute of Standards and Technology (NIST)
Gaithersburg, MD 20899-8230, USA
Email: {adam.jacoff, anthony.downs, ann.virts, elena.messina}@nist.gov

ABSTRACT
Stepfield pallets are a fabricated and repeatable terrain for evaluating robot mobility. They were developed to provide emergency responders and robot developers a common mobility challenge that could be easily replicated to capture statistically significant robot performance data. Stepfield pallets have provided robot mobility challenges for the international RoboCupRescue Robot League competitions since 2005 and have proliferated widely for qualification and practice. They are currently being proposed as a standard test apparatus to evaluate robot mobility. This paper describes the origin and design of stepfield pallets, and discusses their use in several proposed standard test methods for response robots.

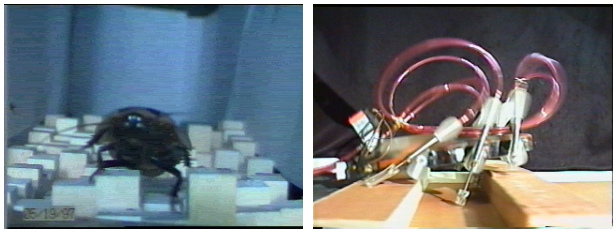
Keywords
Stepfield, step field, robot mobility, robot test method, broken terrain, rough terrain, artificial rubble, urban search and rescue, RoboCupRescue.

1. INTRODUCTION
The National Institute of Standards and Technology (NIST) is developing standard test methods for emergency response robots as part of an ongoing effort sponsored by the Department of Homeland Security and the National Institute of Justice. A series of workshops with subject matter experts from urban search and rescue (US&R) task forces and other emergency response organizations defined thirteen robot categories and over a hundred specific robot performance requirements [1, 2]. Many of these requirements address robot mobility in complex terrains, which necessitated repeatable test methods to capture statistically significant robot performance data. Our approach toward developing mobility test methods has relied upon well-defined apparatuses to differentiate robot capabilities, and typically use the time to negotiate a specified obstacle or path, or the total distance traversed, to measure performance. These mobility tests are always conducted with a remote operator station, out of sight and sound of the robot but within communications range, to emphasize the overall system performance.

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

Stepfield pallets were developed to represent complex terrain or rubble that is describable, reproducible, and repeatable for robot testing. Each pallet consists of a grid of square wood posts cut to assorted cubic unit lengths (a unit is the actual post width when using metric dimensional lumber) and assembled into either symmetric or random topographies tending toward flat, perpendicular hill, or diagonal hill patterns. Multiple stepfield pallets have been assembled into configurations such as the stepfield "dash," a sequential series of five specific pallets in a straight line that has proliferated widely for qualification in the international RoboCupRescue Robot League competitions [3]. A stepfield "figure-8" has also been fabricated using thirteen stepfield pallets and surrounding walls to provide a well-defined continuous path with turns for robot endurance tests. Dozens of stepfield pallets have been configured into a "field" apparatus that allows unconstrained negotiation of stepfield terrain features. Robot developers and purchasers can replicate these common configurations to compare robot performance, improve designs, and support operator training.

2. BIOLOGICALLY INSPIRED TESTS
The concept of using stepfield pallets to evaluate robot mobility in rough terrain derived from researchers investigating biologically inspired mechanisms and control systems. Researchers at the PolyPEDAL Laboratory at the University of California at Berkeley who evaluated cockroaches as effective legged mobility systems noted that they appeared to negotiate relatively rough terrain for their size with almost no loss in speed when compared to negotiation of flat surfaces [4]. They constructed terrain approximating a fractal surface consisting of 1 cm (0.4 in) posts of random height with a variance of 0.5 cm (0.2 in), the height of the insect's center of mass. Extremes of height and depth of the terrain surface reached three times the height of the insect's center of mass (see Figure 1A). Researchers in the Biomimetics Robotics Laboratory at Stanford University built a 16 cm (6 in) hexapod robot that emulated the movement of cockroaches. They used strips of wood equal to the robot's "hip" height as obstacles along the robot's path (see Figure 1B) [5].



QuickTime™ and a decompressor are needed to see this picture

QuickTime™ and a decompressor are needed to see this picture.

Figure 1: A) A cockroach running over a fabricated fractal surface at the University of California – Berkeley. (Photo courtesy of Full et al.) B) A 16 cm (≈ 6 in) long hexapedal robot named Sprawlita modeled after cockroach mobility being tested on hip-height obstacles at Stanford University. (Photo courtesy of Clark et al.) C & D) A 53 cm (≈ 21 in) long hexapedal robot named RHex, with 17.5 cm (≈ 7 in) legs, being tested on a scaled up experimental "broken terrain" made of wood posts at the University of Michigan and McGill University. (Photos courtesy of Saranli et al.)

Researchers at the University of Michigan and McGill University scaled up these experimental terrains using an array of wood posts cut to various lengths to test their RHex hexapedal robot, which itself was inspired from the research noted above [6]. They performed extensive mobility experiments in this "broken terrain" and others in an effort to quantify their robot's mobility. Their experimental terrain used clusters of four posts of similar heights ranging from 10 cm (≈ 4 in) to 30 cm (≈ 12 in) with random height variations up to approximately 20 cm (8 in).

Stepfield pallets essentially combine the ideas noted above to form uneven terrains with elevated ridges as obstacles rising to heights roughly relative to the robot dimensions. Several inexpensive materials were considered before choosing the same simple wood posts used by Saranli et al. to form the square flat surfaces. Wood posts provide ruggedness and reasonable cost, thought they limit the scale of the discretized surfaces to typically available post sizes. However, they are easily fabricated and inexpensive enough to allow researchers and emergency responders to assemble many stepfield pallets into large test apparatuses for practice, evaluation, and training.

Three different scales of pallets provide proportional testing for a variety of robot sizes (see Figure 2). Given that emergency response robots can be wheeled, tracked, or legged, stepfield pallet sizes and the associated ridge heights are more generally correlated to overall robot dimensions rather than to "hip" height, axle height, or other single dimension. To be appropriately proportional to the stepfield terrain, a given robot's footprint should be no larger than 1/4 to 1/3 the area of a single stepfield pallet.

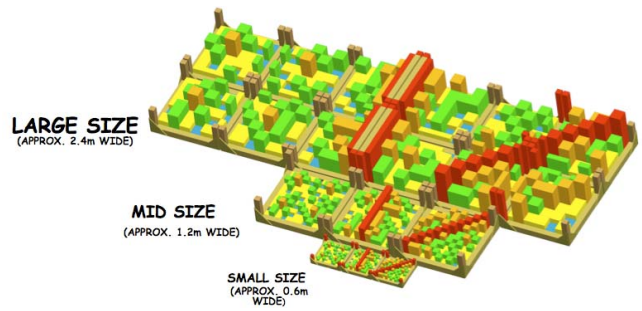


Figure 2: Three different scales of random stepfields for testing different size robots. In each scale, similar pallet topographies are shown using flat, perpendicular hill, and diagonal hill pallets. The hill pallets have posts that randomly tend toward the elevated ridges (shown in red) while following a few rules for continuity.

3. STEPFIELD PALLET FABRICATION

Stepfield pallets are fabricated with a 10x10 grid of square wood posts standing on end to form the terrain with a containment perimeter on all sides made of similar wood post material. The pallet base is made from oriented strand board (OSB) plywood with a thickness of at least 16 mm (≈ 0.675 in) to support the weight of the stepfield when lifted. The containment perimeter is fastened to the plywood base using screws, but the interior grid of cut posts are free to jostle against one another and can be removed and reconfigured as necessary. Blocks can be fastened underneath the plywood base to allow forklift access for easier reconfiguration of test apparatuses made from many stepfield pallets.

The interior posts that form the terrain for each stepfield pallet are cut into four different cubic unit lengths based on defined step heights for each scale of stepfield. Each successive scale increases the step size by a factor of two in length, width, and height. The three different scales of stepfield pallets are described below. Note that each pallet's containment perimeter adds two unit lengths in each direction to the overall assembly:

- Small size stepfield pallets have overall dimensions of 60 cm (≈ 24 in) on a side and are made of a 10x10 grid of 5 cm (≈ 2 in) cubic steps plus a containment perimeter. Each step is made of a single square wood post cut into unit lengths of 5 cm (≈ 2 in), 10 cm (≈ 4 in), 15 cm (≈ 6 in), and 20 cm (≈ 8 in) according to the layouts in Figure 3.

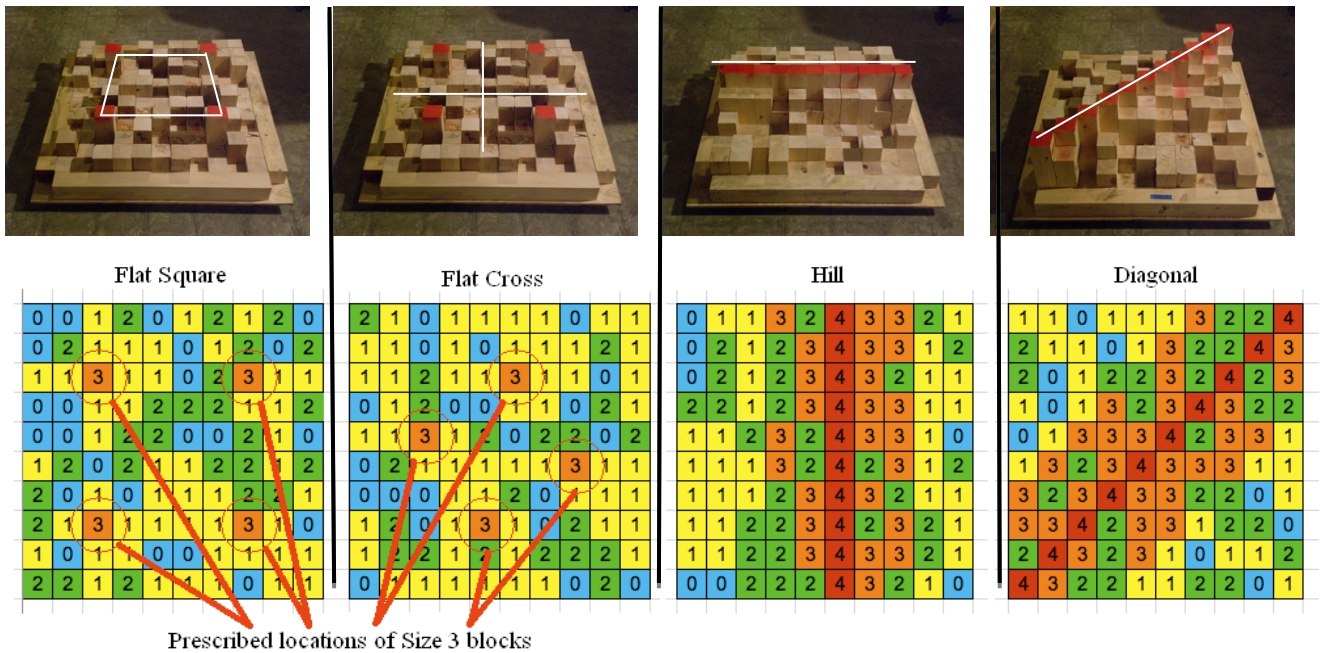


Figure 3: Images and sample designs of random stepfield pallets showing cubic unit post heights. Post heights specified as 0 units are filled with 5 cm (~2 in) posts to maintain grid spacing. (Left-Right) A) Flat square stepfield pallet. B) Flat cross stepfield pallet. C) Perpendicular hill stepfield. D) Diagonal hill stepfield pallet.

- Medium size stepfield pallets have overall dimensions of 120 cm (≈ 48 in) on a side and are made of a 10x10 grid of 10 cm (≈ 4 in) cubic steps. Each step is made of a single square wood post cut into unit lengths of 10 cm (≈ 4 in), 20 cm (≈ 8 in), 30 cm (≈ 12 in), and 40 cm (≈ 16 in) according to the layouts in Figure 3.

- Large size stepfields have overall dimensions of 240 cm (≈ 96 in) on a side and are made of four individual 120 cm (≈ 48 in) pallets with a terrain pattern that spans all four pallets. The terrain is made of a 10x10 grid of 20 cm (≈ 8 in) cubic steps. Each step is made with a cluster of four 10 cm (≈ 4 in) posts cut into unit lengths of 20 cm (≈ 8 in), 40 cm (≈ 16 in), 60 cm (≈ 24 in), and 80 cm (≈ 32 in) according to the layouts in Figure 3.

These dimensions work well with metric wood found in Europe, Asia, and elsewhere. But typical wood posts sizes in the United States are actually ≈ 4 cm (1.5 in) and ≈ 9 cm (3.5 in) square. This reduces the overall dimensions of the stepfield terrains though the ridge elevations can still be cut to the metric dimensions. Non-metric stepfield pallets are good for practice, but the standard apparatus will likely be the metric version.

3.1 Random Pallet Designs

To make random stepfield pallets easy to proliferate, only four different general topographies have been generated: flat square, flat cross, perpendicular hill, and diagonal hill (see Figure 3). Within a given stepfield pallet, the individual post heights are randomly generated using a few rules applied and are shown graphically in the cubic unit lengths mentioned previously. The following rules are used to maintain some continuity in slopes tending toward the elevated ridges:

1. There cannot be a step height difference of more than 2 cubic units between any two adjacent steps.
2. For the two generally flat pallet configurations, there are 4 locations that are 3 cubic units tall. These are the tallest steps on the pallet. The rest of the steps are generated randomly while following rule 1.
3. For the perpendicular hill configurations, the ridge is made of 4 cubic unit step heights and extends across the entire pallet. Two rows on either side of the ridge can range between 2 to 3 cubic units. The remaining rows range between 0 to 2 while following rule 1.
4. For the diagonal hill configurations, the ridge is made of 4 cubic unit step heights and extends across the entire pallet. Three diagonal rows on either side of the ridge can range between 2 to 3 cubic units. The remaining rows range between 0 to 2 while following rule 1.

3.2 Half-Cubic Stepfield Pallets

All of the stepfield pallets discussed so far have been full-cubic stepfields. These are often called red stepfields due to

their placement in the RoboCupRescue competition's red arena for advanced mobility. Half-cubic stepfields have also been fabricated which conform to the same general overall dimensions but have half step heights.

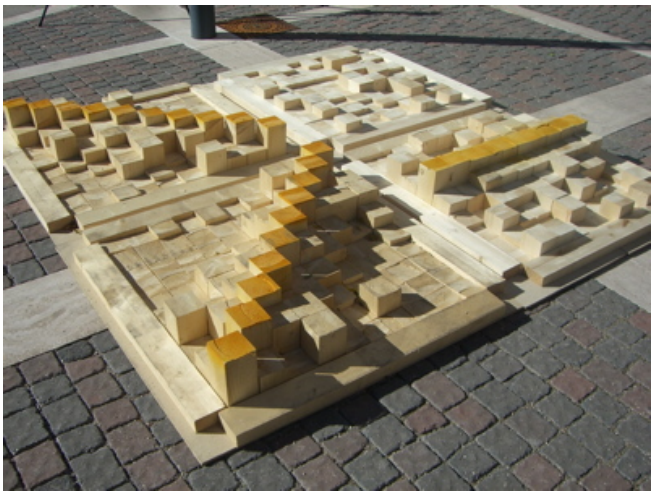


Figure 4: An arrangement of half-cubic (orange) stepfield pallets.

These so-called orange stepfields provide for complexity in robot orientation without challenging mobility for properly scaled robots. They are used to make robotic tasks such as mapping, directed perception, and mobile manipulation more challenging than on simple flat flooring.

4. ROBOT TEST METHODS

A suite of mobility and other test methods for emergency response robots are emerging through the ASTM International standards committee on Homeland Security Applications; Operational Equipment; Robots (E54.08.01) [7]. Stepfields play a prominent role in several of the test methods both to provide challenging and repeatable terrain to evaluate robot mobility and to provide complex flooring for robotic tasks other than mobility. These test methods are always conducted with a remote operator station, out of sight and sound of the robot but within communications range, to emphasize the overall system performance. Several of the test methods that use stepfield pallets are discussed below.

4.1 Stepfield Dash

The Stepfield Dash is a sequential series of five specific pallet types in a straight line: flat square, perpendicular hill, flat cross, diagonal hill, flat square (see Figure 5). It has proliferated widely for qualification in the international RoboCupRescue Robot League competitions, which requires new teams to show a video of the robot traversing this test method along with submission of a team description paper.

The Stepfield Dash was first introduced into the competition arenas at the 2005 German Open. Robots initially had difficulty traversing them but the researchers recognized the challenge. Later that year, at the 2005 RoboCupRescue Championship in Osaka, Japan, several Stepfield Dashes were sequenced together to additionally require turning on pallets. Only a few robots could finish the course. The robot with the fastest time won the best-in-class mobility award. Since then, stepfield pallets have been one of the main mobility challenges in the competition.

Figure 5: A, B) The Stepfield Dash is used for robot qualification for the RoboCupRescue Robot League competitions. C) The 2005 Championship arenas in Osaka, Japan.

4.2 Endurance

The Endurance test method was designed to measure the performance of robots traversing various terrain types within a constrained figure-8 course [8]. The advanced mobility configuration includes a sequential series of stepfield pallets bounded by walls to provide a continuous path with turns (see Figure 6). Robots are timed for average lap speeds and the total numbers of pallets traversed during a single battery cycle.

Figure 6: A) The Endurance test method has one lobe of the figure-8 with perpendicular hill pallets separated by generally flat pallets. B) The other lobe has diagonal hill pallets separated by generally flat pallets. The center path of the figure-8 uses flat pallets (not shown).

4.3 Confined Space

The Confined Space test method is essentially a covered Stepfield Dash (see Figure 7). The tops of the confined spaces are made of inverted stepfield pallets with slight modifications; only the tallest step height posts are used to minimize weight. They are screwed into the OSB plywood base to hold them in place. The minimum gap between the upper and lower stepfield posts can be adjusted in 10 cm (≈ 4 in) increments. Robots are timed to completely traverse the test and the minimum vertical gap is recorded.

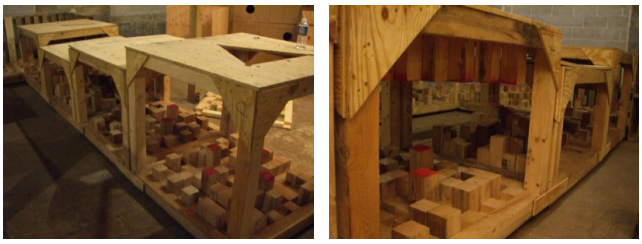


Figure 7: The Confined Space test method uses random stepfield pallets as stalagmites and stalactites.

4.4 Manipulator Test Methods

Half-cubic (orange) stepfield pallets are used as one of three flooring options in the Manipulator Dexerity test methods (see Figure 8).

Figure 8: The Manipulator Dexterity: Perception test method is shown with half-cubic stepfield terrain for added complexity.

The other two flooring options are flat flooring, typically used to capture baseline performance data, and 15 degree pitch/roll ramps to tilt the robot while interacting with the environment. The half-cubic stepfields provide complexity in robot orientation without aggressively challenging mobility.

5. SYMMETRIC STEPFIELDS

All proposed test methods go through a rigorous process to become a standard. One major hurdle is capturing the necessary repeatability data for the test method itself. This requires testing several representative robots for 10 trials each. If the performance range of the proposed test method, or repeatability, produces the same result for a given robot the test method can be considered valid, though it may require a more granular scale. In the case of the random stepfield pallets, the opposite is true. For example, robot performance across multiple stepfield pallets in the Endurance figure-8 with random stepfields can vary widely for each lap. Randomly placed posts on any given stepfield pallet can cause problems for a particular robot's mobility, or for the remote operator's obstacle avoidance capability, even if it can traverse all the other stepfield pallets with relative ease.

A standard test method must also reproduce the same measurable robot performance at different test facilities, which is called reproducibility. A given robot must perform similarly on a particular test method at Facility A, demonstrated by a statistically significant data collection, as it does on the same test method fabricated and tested at Facility B. Although the random stepfield pallets can be replicated with enough precision, particular variants of individual random stepfield pallets have proven inconsistent for some robots. This may ultimately make the standardization of random versions of stepfield pallets difficult to achieve. Nonetheless, they can still provide a good evaluation and training tool for robots prior to deployment.

Symmetric stepfield pallets provide similar mobility obstacles for robots of an appropriate size, but they present the same challenges no matter which direction the robot approaches (see Figure 9). When assembled into a larger test method apparatus, a robot that can traverse a diagonal hill, for example, should always be able to traverse any other diagonal hill pallet, even if it approaches from the opposite direction. Symmetric stepfield pallets were included as mobility obstacles in the 2008 RoboCupRescue Robot League Championship held in Suzhou, China [9]. Over 100 competition missions were conducted, though they do not count toward results regarding test method repeatability since RoboCupRescue teams have other goals and distractions. It appears anecdotally that the symmetric stepfields produce more repeatable robot performance than random stepfield pallets. More testing is required and the designs may evolve, but symmetric stepfields may provide the right balance of rigorous mobility challenges in a repeatable test method apparatus.

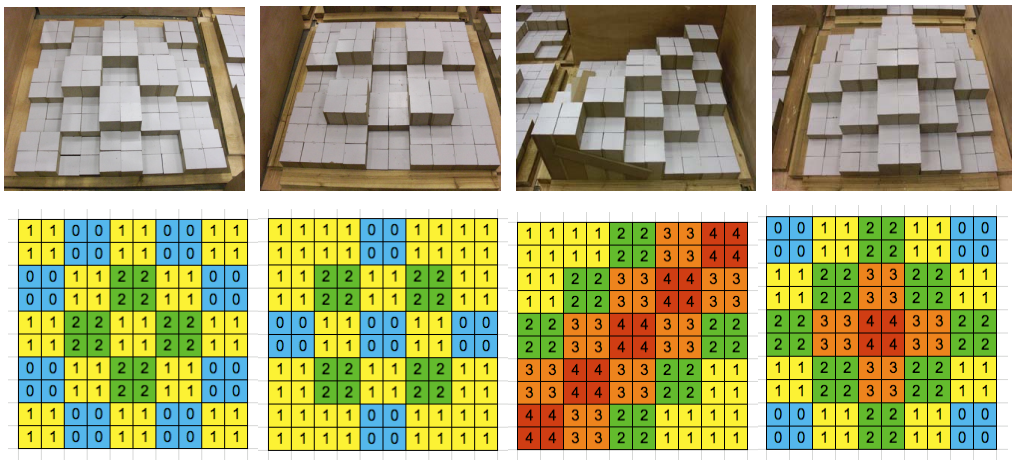


Figure 9: Symmetric stepfield pallets fabricated in Suzhou, China.
(Left-Right) A) Flat cross pallet. B) Flat square pallet. C) Diagonal hill pallet. D) New center peak pallet.

6. CONCLUSIONS

Stepfield pallets, in both their full-cubic and half-cubic variants, have become a central part of several proposed NIST/ASTM test methods for response robots. They have also proliferated around the world as mobility challenges for international competitions. While they provide repeatable terrain to challenge and evaluate robot mobility and a remote operator's situational awareness, they have proven to be difficult to standardize as a mobility test method. Random stepfields can continue to play a key role in helping robots move from the laboratory toward practice deployments and even to support operator training. But as a standard test method apparatus, a more repeatable version of the stepfields is necessary. More experimentation is required, but initial tests using symmetric stepfield pallets suggest that repeatability in robot performance and reproducibility of the test apparatus can be achieved.

7. ACKNOWLEDGMENTS

The authors would like to thank all the emergency responders involved in the NIST/DHS Response Robot Evaluation Exercises and other supporting events. Their initial set of robot requirements provided tangible goals to work toward and their ongoing feedback on test method designs has been much appreciated. Similarly, thanks to all the robot developers who have provided robot practice and training opportunities within prototype test methods and ongoing collaborations toward developing meaningful and challenging test methods for the community. And finally, we would like to thank the participants of the RoboCupRescue Robot League competitions for their enthusiastic robot testing, proliferation of apparatuses, and constructive ideas for new test methods.

8. REFERENCES

[1] Messina, E., "Performance Standards for Urban Search and Rescue Robots," ASTM International Standardization News, August 2006.

[2] Statement of Requirements for Urban Search and Rescue Robot Performance Standards, May 2005. http://www.isd.mel.nist.gov/US&R_Robot_Standards/Requirements%20Report%20(prelim).pdf As accessed August 2008.

[3] RoboCupRescue Robot League competitions. http://www.isd.mel.nist.gov/projects/USAR/competitions.htm As accessed August 2008.

[4] Full, R. J., Autumn, K., Chung, J. I., Ahn, A., "Rapid negotiation of rough terrain by the deathhead cockroach," American Zoologist, 38:81A, 1998.

[5] Clark, J., Cham, J., Bailey, S., Froehlich, E., Nahata, P., Full, R., Cutkosky, M., "Biomimetic Design and Fabrication of a Hexapedal Running Robot," Proceedings of the IEEE 2001 International Conference on Robotics and Automation, May 2001.

[6] U. Saranli, M. Buehler, and D. E. Koditschek, "RHex: A Simple and Highly Mobile Hexapod Robot," Int. J. Robotics Research, 20(7):616-631, July 2001.

[7] ASTM International Standards Committee on Homeland Security Applications; Operational Equipment; Robots (E54.08.01). http://www.astm.org/COMMIT/SUBCOMMIT/E5408.htm As accessed August 2008.

[8] Response Robots Pocket Guide: Test Methods and Robots, 25 July 2008. http://www.isd.mel.nist.gov/US&R_Robot_Stndards/montgomery_county/pocketguide.htm As accessed August 2008.

[9] RoboCup World Championship, Suzhou, China, 14-20 July 2008. http://www.robocup-cn.org./en/index.php As accessed August 2008.

Potential Scaling Effects for Asynchronous Video in Multirobot Search

Prasanna Velagapudi
Paul Scerri, Katia Sycara
Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
001-412-268-8825
pkv@cs.cmu.edu

Huadong Wang, Michael Lewis
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA 15260
001-412-624-9426
ml@sis.pitt.edu

ABSTRACT
Camera guided teleoperation has long been the preferred mode for controlling remote robots with other modes such as asynchronous control only used when unavoidable. Because controlling multiple robots places additional demands on the operator we hypothesized that removing the forced pace for reviewing camera video might reduce workload and improve performance. In an initial experiment we evaluated the usefulness of asynchronous operation for a four robot search task. Participants operated four robot teams performing a simulated urban search and rescue (USAR) task using conventional streaming video plus a map interface or an experimental interface without streaming video but with the ability to store panoramic images on the map to be viewed at leisure. Search performance was somewhat better using the conventional interface; however, ancillary measures suggest that the asynchronous interface succeeded in reducing temporal demands for switching between robots. Because the advantages hypothesized for the asynchronous interface are due to reduction in time stress and workload, the four robot condition may have simply been too easy to observe this advantage. This view is at least partially supported by the reduced switching found in the panoramic condition. We have recently collected data for the streaming video condition for 4, 8, and 12 robots. In this data we found a marked deterioration in performance for the 12 robot condition, suggesting that at this level of difficulty asynchronous video might have an advantage. In this paper we present data for the four robot case comparison and discuss the implications of the recent data from larger teams.

Categories and Subject Descriptors
I.2.9 [**Artificial Intelligence**]: Robotics—*operator interfaces*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

General Terms
Human Factors, Measurement, Experimentation

Keywords
Human-robot interaction, metrics, evaluation, multi-robot system

1. INTRODUCTION
Practical applications of robotics can be classified by two distinct modes of operation. Terrestrial robotics in tasks such as surveillance, bomb disposal, or pipe inspection has used synchronous realtime control relying on intensive operator interaction usually through some form of teleoperation. Interplanetary and other long distance robotics subject to lags and intermittency in communications have used asynchronous control relying on labor intensive planning of waypoints and activities that are subsequently executed by the robot. In both cases planning and decision making are performed primarily by humans with robots exercising reactive control through obstacle avoidance and safeguards. The near universal choice of synchronous control for situations with reliable, low latency communication suggests a commonly held belief that experientially direct control is more efficient and less error prone. When this implicit position is rarely discussed it is usually justified in terms of "naturalness" or "presence" afforded by control relying on teleoperation. Fong and Thorpe [10] observe that direct control while watching a video feed from vehicle mounted cameras remains the most common form of interaction. The ability to leverage experience with controls for traditionally piloted vehicles appears to heavily influence the appeal for this interaction style.

Control based on platform mounted cameras, however, is no panacea. Wickins & Hollands [31] identify 5 viewpoints used in control, three of them, immersed, tethered, and "plan view" can be associated with the moving platform while 3rd person (tethered) and plan views require fixed cameras. In the immersed or egocentric view (A) the operator views the scene from a camera mounted on the platform. The field of view provided by the video feed is often much narrower than human vision, leading to the experience of viewing the world through a soda straw from a foot or so above the ground. This perceptual impairment leaves

the operator prone to numerous, well-known operational errors, including disorientation, degradation of situation awareness, failure to recognize hazards, and simply overlooking relevant information [8, 15]. A sloped surface, for example, gives the illusion of being flat when viewed from a camera mounted on a platform traversing that surface [11]. For fixed cameras the operator's ability to survey a scene is limited by the mobility of the robot and his ability to retain viewed regions of the scene in memory as the robot is maneuvered to obtain views of adjacent regions. A pan-tilt-zoom (ptz) camera resolves some of these problems but introduces new ones involving discrepancies between the robots heading and the camera view that frequently lead to operational mishaps [6]. A tethered "camera" (B,C) provides an oblique view of the scene showing both the platform and its 3D environment. A 3rd person fixed view (C) is akin to an operator's view controlling slot cars and has been shown effective in avoiding roll-overs and other teleoperation accidents [4] but can't be used anywhere an operator's view might be obstructed such as within buildings or in rugged terrain. The tethered view (B) in which a camera "follows" an avatar (think Mario Brothers©) is widely favored in virtual environments [16,26] for its ability to show the object being controlled in relation to its environment by showing both the platform and an approximation of the scene that might be viewed from a camera mounted on it. This can be simulated for robotic platforms by mounting a camera on a flexible pole giving the operator a partial view of his platform in the environment [32]. Because of restriction in field of view and the necessity of pointing the camera downward, however, this strategy is of little use for surveying a scene although it can provide a view of the robot's periphery and nearby obstacles that could not be seen otherwise. The exocentric views show a 2 dimensional version of the scene such as might be provided by an overhead camera and cannot be obtained from an onboard camera. This type of "overhead" view can, however, be approximated by a map. For robots equipped with laser range finders, generating a map and localizing the robot on that map provides a method for approximating an exocentric view of the platform. If this view rotates with the robot (heading up) it is a type D plan view. If it remains fixed (North up) it is of type E.

An early comparison at Sandia Laboratory between viewpoints for robot control [15] investigating accidents focused on the most common of these: (A) egocentric from onboard camera and (C) 3rd person. The finding was that all accidents involving rollover occurred under egocentric control while 3rd person control led to bumping and other events resulting from obstructed or distanced views. In current experimental work in remotely controlled robots for urban search and rescue (USAR) robots are typically equipped with both a ptz video camera for viewing the environment and a laser range finder for building a map and localizing the robot on that map. The video feed and map are usually presented in separate windows on the user interface and intended to be used in conjunction. While Casper and Murphy [5] reporting on experiences in searching for victims at the World Trade Center observed that it was very difficult for an operator to handle both navigation and exploration of the environment from video information alone, Yanco et al. [32] found that first responders using a robot to find victims in a mock environment made little use of the generated map. One possible explanation is that video is simply more attention grabbing than other presentations leading operators to control primarily from the camera while ignoring other information available on their interface. A number of recent studies conducted by Goodrich, Neilsen, and colleagues [2,6,19,23,32] have attempted to remedy this through an ecological interface that fuses information by embedding the video display within the map. The resulting interface takes the 2D map and extrudes the identified surfaces to derive a 3D version resembling a world filled with cubicles. The robot is located on this map with the video window placed in front of it at the location being viewed. This strategy uses the egocentric camera view and the overhead view from the map to create a synthetic tethered view of the sort found most effective in virtual environments and games [16,26]. The anticipated advantages, however, have been difficult to demonstrate with ecological and conventional interfaces trading advantages across measures. Of particular interest have been comparisons between control based exclusively on maps or videos. In complex environments with little opportunity for preview maps were superior in assisting operators to escape from a maze [19]

When considering such potential advantages and disadvantages of viewpoints it is important to realize that there are two, not one, important subtasks that are likely to engage operators [26]. The escape task and the accidents reviewed at Sandia involved Navigation, the act of explicitly moving the robot to different locations in the environment. In many applications search, the process of acquiring a specific viewpoint—or set of viewpoints—containing a particular object may be of greater concern. While both navigation and search require the robot to move, an important distinction is the focus of the movement. Navigation occurs with respect to the environment at large, while search references a specific object or point within that environment. Switching between these two subtasks may play a major role in undermining situation awareness in teleoperated environments. For example, since search activities move the robot with respect to an object, viewers may lose track of their global position within the environment. Additional maneuvering may be necessary to reorient the operator before navigation can be effectively resumed. Because search relies on moving a viewpoint through the environment to find and better view target objects, it is an inherently egocentric task. This is not necessarily the case for navigation which does not need to identify objects but only to avoid them.

Search, particularly multi-robot search, presents the additional problem of assuring that areas the robot has traversed have been thoroughly searched for targets. This requirement directly conflicts with the navigation task which requires the camera to be pointed in the direction of travel in order to detect and avoid objects and steer toward its goal. When the operator attempts to compromise by choosing a path to traverse and then panning the camera to search as the robot moves he runs both the risk of hitting objects while he is looking away and missing targets as he attends to navigation. For multirobot control these difficulties are accentuated by the need to switch attention among robots multiplying the likelihood that a view containing a target will be missed. In earlier studies [29,30] we have demonstrated that success in search is directly related to the frequency with which the operator shifts attention between robots over a variety of conditions. An additional issue is the operator's confidence that an area has been effectively searched. In our natural environment we move and glance about to construct a representation of our environment that is informed by planning and proprioception that

knit together the sequence of views. In controlling a robot we are deprived of these natural bridging cues and have difficulty recognizing as we pan and tilt whether we are resampling old views or missing new ones. The extent of this effect was demonstrated by Pausch [21] who found that participants searching for an object in a virtual room using a headmounted display were twice as fast as when they used a simulated handheld camera. Since even the handheld camera provides many ecological cues we should expect viewing from a moving platform through a ptz camera to be substantially worse.

1.1 Asynchronous Imagery

To combat these problems of attentive sampling among cameras, incomplete coverage of searched areas, and difficulties in associating camera views with map locations we are investigating the potential of asynchronous control techniques previously used out of necessity in NASA applications as a solution to multi-robot search problems. Due to limited bandwidth and communication lags in interplanetary robotics camera views are closely planned and executed. Rather than transmitting live video and moving the camera about the scene, photographs are taken from a single spot with plans to capture as much of the surrounding scene as possible. These photographs taken with either an omnidirectional overhead camera (camera faces upward to a convex mirror reflecting 360◦) and dewarped [20,21] or stitched together from multiple pictures from a ptz camera [22] provide a panorama guaranteeing complete coverage of the scene from a particular point. If these points are well chosen, a collection of panoramas can cover an area to be searched with greater certainty than imagery captured with a ptz camera during navigation. For the operator searching within a saved panorama the experience is similar to controlling a ptz camera in the actual scene, a property that has been used to improve teleoperation in a low bandwidth high latency application [23].

1.2 Controlling Multiple Robots

Because some functions of a MrS such as identifying victims among rubble depend on human input, evaluating the operator's span of control as the number of controlled entities scale is critical for designing feasible human-automation control systems.

Controlling multiple robots substantially increases the complexity of the operator's task because attention must constantly be shifted among robots in order to maintain situation awareness (SA) and exert control. In the simplest case an operator controls multiple independent robots interacting with each as needed. A search task in which each robot searches its own region would be of this category although minimal coordination might be required to avoid overlaps and prevent gaps in coverage. Control performance at such tasks can be characterized by the average demand of each robot on human attention [4]. Under these conditions increasing robot autonomy should allow robots to be neglected for longer periods of time making it possible for a single operator to control more robots.

Established methods of estimating MrS control difficulty, the neglect tolerance model, NT, [4] and the Fan-out measure [5] are predicated on the independence of robots and tasks. In the NT model the period following the end of human intervention but preceding a decline in performance below a threshold is considered time during which the operator is free to perform other tasks. If the operator services other robots over this period the measure provides an estimate of the number of robots that might be controlled. Fan-out refers to maximum number of robots that can advantageously controlled under particular conditions. Fan-out can be determined empirically as in [5] by adding robots and measuring performance until a plateau without further improvement is reached or indirectly by predicting the maximum number of robots using parameters from the NT model [4].

Multi-robot control appears to impact the human operator's workload in three distinct ways: (1) building and maintaining awareness, (2) making decisions, and (3) controlling the system. Increasing the autonomy level in robotic system, whether providing decision support or individual robot autonomy, allows us to shift the decision-making and robot control workload from the human to the robotic system. On the other hand, increased robot autonomy may cause an increase in perception and decision-making workloads. Thus, there is a trade-off between the autonomy level of the robotic system and the level of human intervention.

Current estimates of human span of control limitations are severe. Miller [17], for example, showed that under expected target densities, a controller who is required to authorize weapon release for a target identified by a UCAV, could control no more than 13 UAVs even in the absence of other tasks. A similar breakpoint of 12 was found by [7] for retargeting Tomahawk missiles. Smaller numbers (3-9) [6] have typically been found for ground robots depending upon task and level of automation. By most estimates, however, a team of four robots should be well within an operator's span of control. Two experiments are reported in this paper. The first experiment compares a user interface supplying streaming video with one in which operators must rely on static panoramas for performing a USAR task with a four robot team. The second experiment investigates performance at the same task for the streaming video interface as the number of robots is increased from 4 to 8 to 12 to identify the operators' limits of control.

2. METHOD

2.1 USARSim and MrCS

The reported experiments were performed using the USARSim robotic simulation with simulated UGVs performing Urban Search and Rescue (USAR) tasks. USARSim is a high-fidelity simulation of urban search and rescue (USAR) robots and environments we developed as a research tool for the study of HRI and multi-robot coordination. USARSim supports HRI in ways lower fidelity simulations cannot by accurately rendering user interface elements (particularly camera video), accurately representing robot automation and behavior, and accurately representing the remote environment that links the operator's awareness with the robot's behaviors. USARSim can be downloaded from www.sourceforge.net/projects/usarsim and serves as the basis for the Virtual Robots Competition of the RoboCup Rescue League. The current version of USARSim includes detailed replicas of NIST USAR Arenas, as well as large-scale indoor and outdoor hypothetical disaster scenarios, and a large outdoor area along the Chesapeake Bay. USARSim complements these maps with high fidelity models of commercial (pioneer P2-DX, P2-AT, iRobot's ATRV Jr., Foster-Miller's Talon, and Telerob's Telemax) and experimental (PER from CMU, Zerg from University of Freiburg, Kurt 3D from University

of Osnabruk) robots, including a snake (Soyu from Tohoku University), air (Air-robot helicopter) and a large Ackerman-steered UGV (Hummer) and sensor models for many common robotic sensing packages. USARSim uses Epic Games' UnrealEngine2 [14] to provide a high fidelity simulator at low cost. Validation studies showing agreement for a variety of feature extraction techniques between USARSim images and camera video are reported in Carpin et al. [3]. Other sensors including sonar and audio are also accurately modeled. Validation data showing close agreement in detection of walls and associated Hough transforms for a simulated Hokuyo laser range finder are described in [13]. The current UnrealEngine2 integrates MathEngine's Karma physics engine [14] to support high fidelity rigid body simulation. Validation studies showing close agreement in behavior between USARSim models and real robots being modeled are reported in [13,22,27,33].

MrCS (Multi-robot Control System), a multirobot communications and control infrastructure with accompanying user interface developed for experiments in multirobot control and RoboCup competition [1] was used. MrCS provides facilities for starting and controlling robots in the simulation, displaying camera and laser output, and supporting inter-robot communication through Machinetta a distributed mutiagent system. Figure 1 shows the elements of the MrCS involved in this experiment. In the standard MrCS the operator selects the robot to be controlled from the colored thumbnails at the top of the screen that show a slowly updating view from the robot's camera. Streaming video from the in focus robot which the operator now controls is displayed on the Image Viewer. To view more of the scene the operator uses pan/tilt sliders (not shown) to control the camera. Robots are tasked by assigning waypoints on a heading-up map on the Mission Panel (not shown) or through a

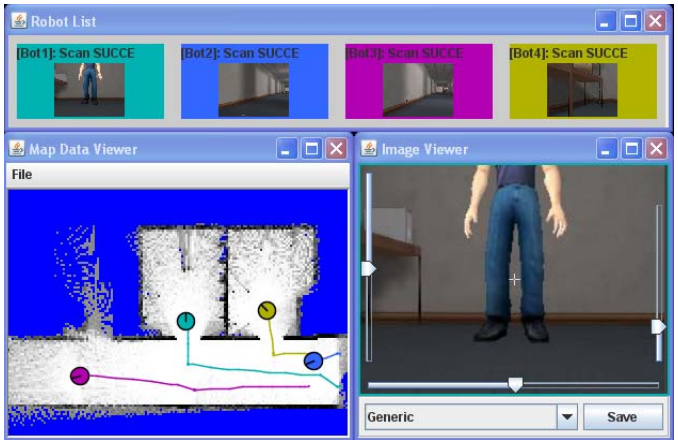


Figure 1. MrCS components for Streaming Video Mode

teleoperation widget (not shown). The current locations and paths of the robots are shown on the Map Data Viewer.

2.1.1 Panorama Mode

Our USAR application which requires finding victims and locating them on a map needs to use both map and camera views [32]. In Panorama mode the operator directs navigation from the map being generated with panoramas being taken at the last

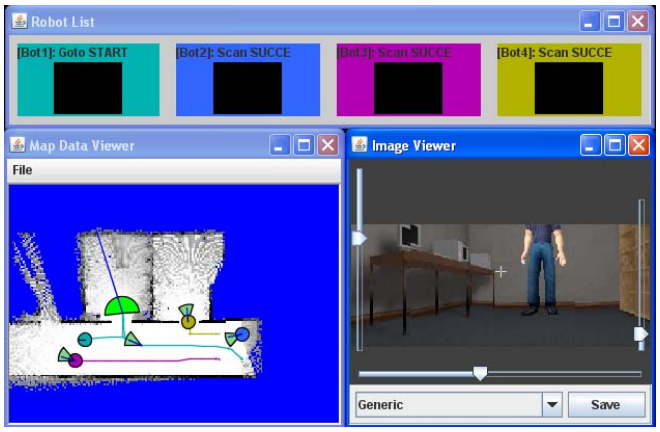


Figure 2. MrCS components for Asynchronous Panorama Mode

waypoint of a series. The panoramas are stored and accessed through icons showing their locations on the map. The operator can find victims by asynchronously panning through these stored panoramas as time becomes available. When a victim is spotted the operator uses landmarks from the image and corresponding points on the map to record the victim's location. By changing the task from a forced paced one with camera views that must be controlled and searched on multiple robots continuously to a self paced task in which only navigation needs to be controlled in realtime we hoped to provide a control interface that would allow more thorough search with lowered mental workload. The reductions in bandwidth and communications requirements [1] are yet another advantage offered by this approach.

Although the experimental panoramic interface (Fig. 2) looks much the same it behaves quite differently. Robots are again selected for control from the colored thumbnails which now lack images. Panoramic images are acquired at the terminal point of waypoint sequences. Icons conveying the robot's location and orientation at these points are placed on the map for accessing the panoramas. The operator can then view stored panoramas by selecting an icon and dragging a mouse over the Image Viewer to move the image around or using the mouse's scroll wheel to zoom in and out of the image. The associated icon on the Map Data Viewer changes orientation in accordance with the part of the scene being viewed.

2.2 Experiment 1 Asynchronous vs. Streaming Video

2.2.1 Method

Two equivalent search environments previously used in the 2006 RoboCup Rescue Virtual Robots competition [1] were selected for use in the experiment. Each environment was a maze like hall with many rooms and obstacles, such as chairs, desks, cabinets, and bricks. Victims were evenly distributed within the environments. A third simpler environment was used for training. The experiment followed a repeated measures design with participants searching for victims using both panorama and streaming video modes. Presentation orders for mode were counterbalanced. Test environments were presented in a fixed order confounding differences between the environments with learning effects. Because the environments were closely matched we will discuss these differences as transfer of training effects.

2.2.1.1 Participants and Procedure

21 paid participants, 9 male and 12 female old recruited from the University of Pittsburgh community. None had prior experience with robot control although most (15) were frequent computer users. Six of the participants (28%) reported playing computer games for more than one hour per week.

2.2.1.2 Procedure

After collecting demographic data the participant read standard instructions on how to control robots via MrCS. In the following 15~20 minute training session, the participant practiced control operations for panorama and streaming video modes (both were enabled) and tried to find at least one victim in the training environment under the guidance of the experimenter. Participants then began two testing sessions in which they performed the search task using the panorama and streaming video modes.

2.2.2 Results

Only one participant failed to find any victims under the most lenient criterion of marking the victim within 2 m of the actual location. This occurred in the panorama mode on the initial trial. Overall participants were successful in searching the environment in either mode finding as many as 9 on a trial. The average across conditions using the 2 m radius was 4.5 falling to 4.1 for a 1.5 m radius, 3.4 at 1 m and 2.7 when they were required to mark victims within .75 m. Repeated measures ANOVAs found differences in accuracy of location favoring the streaming video mode at the 1.5 m radius $F(1,19) = 8.038$, $p=.01$), and 2.0 radius $F(1,19)=9.54$, $p=.006$. Figure 3 shows these differences.

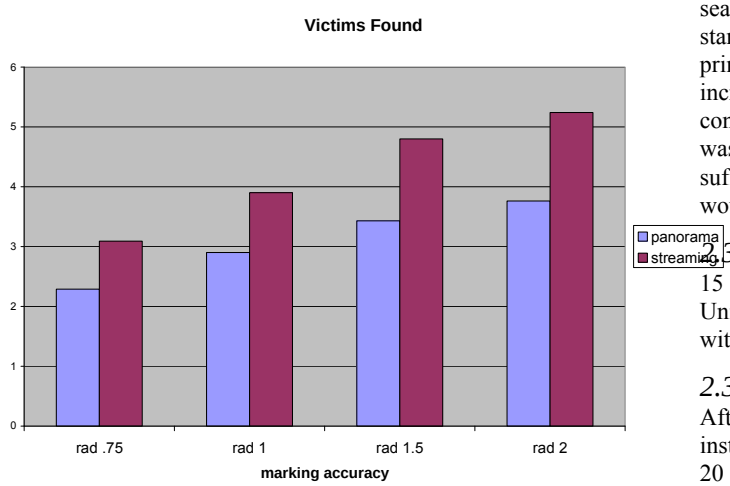


Figure 3. Effects of display modes

No significant order effect (learning) was observed. As in earlier studies we found a positive relation, $F(1,19)=3.86$, $p=.064$, between the number of times the operator switched between robots and the victims that were found (Figure 4). In accord with our hypothesis that this is due to the forced pace of performing the task using streaming video no relation was found between the frequency of switching and victims for the panorama mode.

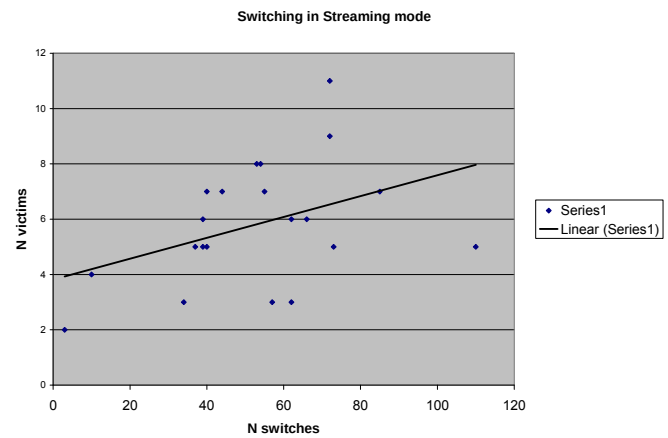


Figure 4. Finding victims was related to switches only in Streaming mode

2.3 Experiment 2 Scaling in N Robots

2.3.1 Method

A large USAR environment previously used in the 2006 RoboCup Rescue Virtual Robots competition [1] was selected for use in the experiment. The environment was a maze like hall with many rooms and obstacles, such as chairs, desks, cabinets, and bricks. Victims were evenly distributed within the environment. A second simpler environment was used for training. The experiment followed a repeated measures design with participants searching for victims using 4, 8, and finally 12 robots. Robot starting points were varied over the three trials. Because our primary concern was with changes in performance as N robots increased, trials were presented in a fixed order. This design confounding learning effects and starting points with N of robots was adopted because the randomly selected starting points were sufficiently comparable not to bias results and any learning effect would attenuate rather than accentuate the expected decrements.

2.3.1.1 Participants

15 paid participants, 8 male and 7 female were recruited from the University of Pittsburgh community. None had prior experience with robot control although most were frequent computer users.

2.3.1.2 Procedure

After collecting demographic data the participant read standard instructions on how to control robots via MrCS. In the following 20 minute training session, the participant practiced control operations and tried to find at least one victim in the training environment under the guidance of the experimenter. Participants then began three testing sessions (15 minute each) in which they performed the search task using 4, 8, and finally 12 robots. After each task, the participants were asked to file the NASA-TXL workload survey immediately.

2.3.2 Results

Overall participants were successful in searching the environment in all conditions finding as many as 12 victims on a trial. The average number of victims found was 4.80 using 4 robots, 7.06 for 8 robots, but only 4.73 when using 12 robots. A paired t-test shows that in the 8 robots condition (R8) participants explored larger regions, $t(15) = -10.44$, $p = 0.000$, while finding

more victims, t (15) = -3.201, p = 0.003, than using a 4 robot team (R4). On the other hand, adding four addition robots degraded performance with participants in the 8 robot condition (R8) exploring larger regions, t (15) = -1.19, p = 0.059, as well as finding more victims, t (15) = -3.014, p = 0.005, than they did using a 12 robot team (R12).

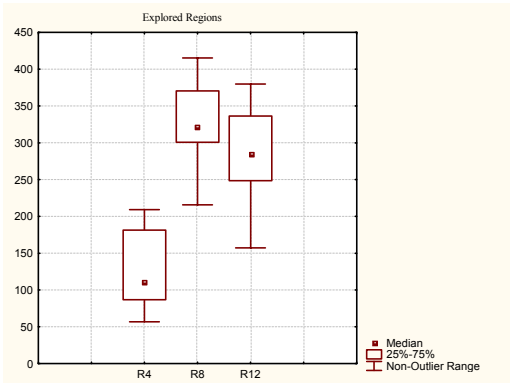


Figure 5. Explored Regions

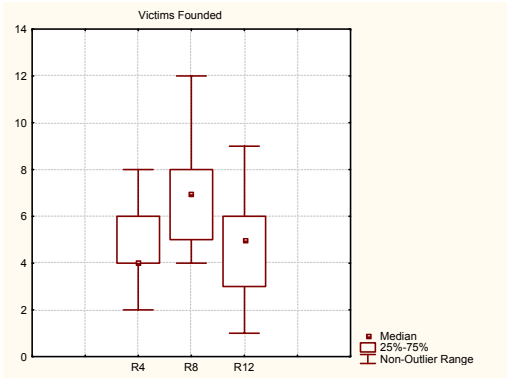


Figure 6. Victims Found

Figure 7 shows that as the number of robots is increased, fewer victims were found per robot. This measure should remain constant if robots were controlled to the same level of effectiveness. However, these differences were not significant. With increasing numbers of robots operators tended to neglect some robots either entirely or after an initial movement as shown in Table 2. A paired t-test indicates that participants neglected more robots in the 12 robot condition, t (15) = -1.922, p = 0.064, than under 4 robots team (R4).

More robots were neglected after an initial move in the 8 robot (R8) condition t (15) = -2.092, p = 0.046, than for 4 robots (R4); and still more comparing a 12 robot team (R12) to the 8 robot (R8) condition t (15) = -3.761, p = 0.001.

With increasing numbers of robots operators tended to neglect some robots either entirely or after an initial movement as shown in Table 2. A paired t-test indicates that participants neglected more robots in the 12 robot condition, t (15) = -1.922, p = 0.064, than under 4 robots team (R4).

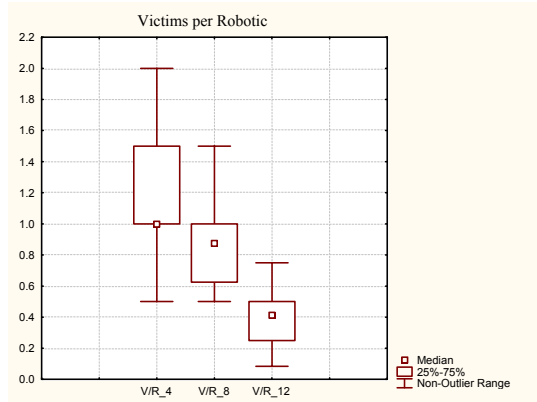


Figure 7. Victims Found per Robot

more robots in the 12 robot condition, t (15) = -1.922, p = 0.064, than under 4 robots team (R4).

More robots were neglected after an initial move in the 8 robot (R8) condition t (15) = -2.092, p = 0.046, than for 4 robots (R4); and still more comparing a 12 robot team (R12) to the 8 robot (R8) condition t (15) = -3.761, p = 0.001.

Table 1. Neglected Robots

Number of Robots	R4	R8	R12
Totally	0.00	0.13	0.80
After the Initial Move	0.00	0.33	1.87

As in earlier studies we found a positive relation between the number of times the operator switched between robots and the victims that were found. Higher switching rates are an indicator of shortened ITs or more efficient use of NT to service additional robots and hence should improve team performance. Figure 6 shows the number of switches observed under each of the three conditions. There were significant differences in number of switches between robots for the 4 robot and 8 robot conditions (t= -2.914, P<0.007) and the 4 robot and 12 robot conditions (t= -2.620, P<0.014). Similar results were found for numbers of missions (waypoint assignments) between the 4 robot and 8 robot condition (t= -3.079, P<0.005) and the 4 robot and 12 robot condition (t= -2.118, P<0.043).

The result of the workload assessment indicates that workload increased with increasing numbers of robots to be controlled. This difference in workload was significant between the 4 robot and 12 robot conditions (P<0.0146).

Our results show that we have successfully bounded the number of directly controllable robots for a realistic USAR task at between 8-12. This can be seen for both the number of victims found and the regions explored which improve between the four robot and eight robot conditions but decline again between 8 and 12 robots. Determining Fan-out empirically as in [20] the Fan-out plateau (point at which performance is no longer improved by

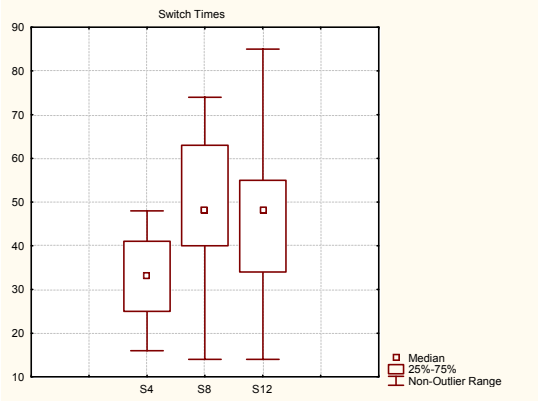


Figure 8. Number of Switches

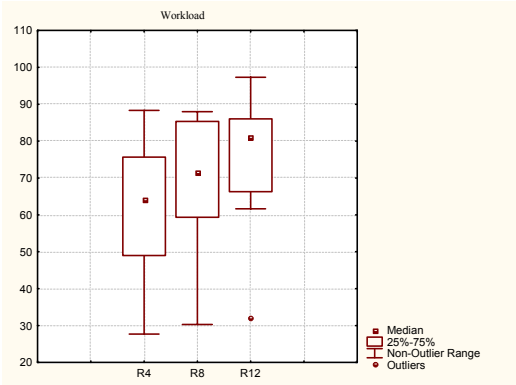


Figure 9. NASA-TLX Measurement of Mental workload

adding additional robots) lies somewhere within this region. The point at which operator capabilities become saturated can be

estimated more closely by observing the number of robots that are either completely neglected or neglected after the first move. This number is approximately 2.7 for the 12 robot condition suggesting that the actual limit for this experiment is approximately 9 robots. An examination of the number of switches between robots supports this estimate because the number of switches is essentially the same in both the 8 and 12 robot conditions. This means that operators have reached their limit for interactions and are neglecting the robots for slightly longer times to accommodate the additional robot they are actively controlling.

With this initial experiment we are establishing a baseline for exploring the effects of robot team size on human performance. Even these early results suggest that the navigation component of the operator's task and the perceptual tasks involved in search may be somewhat differently impacted by increases in team size. We hope to be able to use such results to guide system designers in allocating functions statically or dynamically between the operator and robot (team) autonomy.

3. DISCUSSION

Our original motivation for developing a panorama mode for MrCS was to address restrictions posed by a communications server added to the 2007 RoboCup Rescue competition to simulate bandwidth limitations and drop-outs due to attenuation from distance and obstacles. Although the panorama mode was designed to drastically reduce bandwidth and allow operation despite intermittent communications our system was so effective we decided to test it under conditions most favorable to a conventional interface. Our first experiment showed that under conditions allowing uninterrupted, noise free, streaming video a conventional interface lead to somewhat better (5 vs. 4 victims) search performance. The switching results, however, suggest that asynchronous panoramas do overcome the forced pace switching needed to avoid missing unattended targets in realtime interfaces. We would expect this advantage to grow as the number of robots increases with performance approaching or surpassing streaming video at some point.

The second experiment demonstrates that for this USAR task control of four robots lies well within the cognitive capabilities of an operator. Our data show that the operators had sufficient reserve capacity to effectively control twice as many robots. These results suggest that if advantages are to be found for panorama-based search they are likely to be found in the 8-12 robot range.

It is also important to note that while victims were marked more accurately in the streaming video mode, the greater accuracy may simply reflect advantages for marking locations rather than actual improvement in search performance. Because participants in the streaming video mode were able to drive their robots in close proximity to victims, their ability to mark locations with greater accuracy may have simply resulted from being nearer their targets than in the panorama condition where the target must be marked from the location at which the panorama had been taken. Just as [19] have demonstrated that maps may be better than cameras for navigation we hope that asynchronous video and related strategies may play a role in improving multirobot search capabilities. Coupled with the ability to control robots under poor communication conditions such as are expected in USAR and other field work we believe that interface innovations of this sort have an important role to play in making control of robot teams a reality.

4. ACKNOWLEDGMENTS

This research has been sponsored in part by AFOSR FA9550- 07-1-0039, AFOSR FA9620-01-1-0542, ONR Grant N00014-03-1-0580 and L3-Communications (4500257512)

5. REFERENCES

[1] S. Balakirsky, S. Carpin, A. Kleiner, M. Lewis, A. Visser, J. Wang, and V. Zipara, "Toward hetereogeneous robot teams for disaster mitigation: Results and performance metrics from RoboCup Rescue," *Journal of Field Robotics*, in press.

[2] D. J. Bruemmer, D. A. Few, M. C. Walton, R. L. Boring, J. L. Marble, C. W. Nielsen, and J. Garner. Turn off the television: Real-world robotic exploration experiments with a virtual 3-D display. *Proc. HICSS*, Kona, HI, 2005.

[3] S. Carpin, T. Stoyanov, Y. Nevatia, M. Lewis and J. Wang. Quantitative assessments of USARSim accuracy". *Proceedings of PerMIS 2006* [ir-13] S. Carpin, J. Wang, M. Lewis, A. Birk and A. Jacoff. High fidelity tools for rescue robotics: Results and perspectives, *Robocup 2005 Symposium, 2005.*

[4] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, C. Scrapper. (2006b). Bridging the gap between simulation and reality in urban search and rescue. *Robocup 2006: Robot Soccer World Cup X*, Springer, Lecture Notes in Artificial Intelligence, 2006.

[5] J. Casper and R. R. Murphy. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *IEEE Transactions on Systems, Man, and Cybernetics Part B*, 33(3):367–385, June, 2003

[6] J. W. Crandall, M. A. Goodrich, D. R. Olsen, and C. W. Nielsen. Validating human-robot interaction schemes in multitasking environments. IEEE Transactions on Systems, Man, and Cybernetics, Part A, 35(4):438–449, 2005.

[7] Cummings, M. and Guerlain, S. An interactive decision support tool for real-time in-flight replanning of autonomous vehicles, AIAA Unmanned Unlimited Systems, Technologies, and Operations, 2004.

[8] R. Darken, K. Kempster, and B. Peterson, "Effects of streaming video quality of service on spatial comprehension in a reconnaissance task," in *Proc. Meeting I/ITSEC*, Orlando, FL, 2001.

[9] M. Fiala, "Pano-presence for teleoperation," *Proc. Intelligent Robots and Systems (IROS 2005)*, 3798-3802, 2005.

[10] T. Fong and C. Thorpe, "Vehicle teleoperation interfaces," *Autonomous. Robots*, no. 11, pp. 9–18, 2001.

[11] M. Lewis and J. Wang, "Gravity referenced attitude display for mobile robots : Making sense of what we see, " *Transactions on Systems, Man and Cybernetics Part A*, 37(1), 94-105, 2007.

[12] M. Lewis, J. Wang, and S. Hughes, "USARsim : Simulation for the Study of Human-Robot Interaction," *Journal of Cognitive Engineering and Decision Making*, 1(1), 98-120, 2007.

[13] M. Lewis, S. Hughes, J. Wang, M. Koes, and S. Carpin. Validating USARsim for use in HRI research, *Proceedings of the 49th Annual Meeting of the Human Factors and Ergonomics Society*, Orlando, FL, 2005, 457-461

[14] Mathengine, *MathEngine Karma User Guide*, http://udn.epicgames.com/Two/KarmaReference/KarmaUserGuide.pdf, accessed May 3, 2005.

[15] D. E. McGovern, "Experiences and Results in *Teleoperation of Land Vehicles*," Sandia Nat. Labs., Albuquerque, NM, Tech. Rep. SAND 90-0299, 1990.

[16] P. Milgram and J. Ballantyne, "Real world teleoperation via virtual environment modeling," in *Proc. Int. Conf. Artif. Reality Tele-Existence*, Tokyo, 1997.

[17] Miller, C. Modeling human workload limitations on multiple UAV control, Proceedings of the Human Factors and Ergonomics Society 47th Annual Meeting, 526-527, 2004.

[18] J.R. Murphy, "Application of Panospheric Imaging to a Teleoperated Lunar Rover," in *Proceedings of the 1995 International Conference on Systems, Man, and Cybernetics*, 1995.

[19] C. Nielsen and M. Goodrich. Comparing the usefulness of video and map information in navigation tasks. In *Proceedings of the 2006 Human-Robot Interaction Conference*, Salt Lake City, Utah, March 2006.

[20] D.R Olsen and S.B. Wood, Fan-out: measuring human control of multiple robots, in Proceedings of the SIGCHI conference on Human factors in computing systems. 2004, ACM Press: Vienna, Austria. p. 231-238.

[21] R. Pausch, M. A. Shackelford, and D. Proffitt, "A user study comparing head-mounted and stationary displays," *IEEE Symposium on Research Frontiers in Virtual Reality*, October 25-26, 1993, San Jose, CA.

[22] C. Pepper, S. Balakirsky, and C. Scrapper. Robot Simulation Physics Validation, *Proceedings of PerMIS'07*, 2007.

[23] B. W. Ricks, C. W. Nielsen, and M. A. Goodrich. Ecological displays for robot interaction: A new perspective. In *International Conference on Intelligent Robots and Systems IEEE/RSJ*, Sendai, Japan, 2004.

[24] P. Scerri, Y. Xu, E. Liao, G. Lai, M. Lewis, and K. Sycara, "Coordinating large groups of wide area search munitions," In D. Grundel, R. Murphey, and P. Pandalos (Ed.), *Recent Developments in Cooperative Control and Optimization*, Singapore: World Scientific, 451-480.

[25] N. Shiroma, N. Sato, Y. Chiu, and F. Matsuno, "Study on effective camera images for mobile robot teleoperation," In *Proceedings of the 2004 IEEE International Workshop on Robot and Human Interactive Communication*, Kurashiki, Okayama Japan, 2004

[26] D. S. Tan, G. G. Robertson, and M. Czerwinski, "Exploring 3D navigation: Combining speed-coupled flying with orbiting," in CHI 2001 *Conf. Human Factors Comput. Syst*., Seattle, WA, 2001.

[27] B. Taylor, S. Balakirsky, E. Messina and R. Quinn. Design and Validation of a Whegs Robot in USARSim, *Proceedings of PerMIS'07, 2007*.

[28] R. Volpe, "Navigation results from desert field tests of the Rocky 7 Mars rover prototype," in *The International Journal of Robotics Research*, 18, pp.669-683, 1999.

[29] J. Wang, and M. Lewis, "Human control of cooperating robot teams", *2007 Human-Robot Interaction Conference*, ACM, 2007a.

[30] J. Wang, and M. Lewis, "Assessing coordination overhead in control of robot teams," *Proceedings of the 2007 International Conference on Systems, Man, and Cybernetics*, 2007b

[31] C. Wickens and J. Hollands, *Engineering Psychology and Human Performance*, Prentice Hall, 1999.

[32] H. Yanco, M. Baker, R. Casey, B. Keyes, P. Thoren, J. Drury, D. Few, C. Nielsen, D. Bruemmer. "Analysis of human-robot interaction for urban search and rescue, *Proceedings of PERMIS, 2006*.

[33] M. Zaratti, M. Fratarcangeli and L. Iocchi. A 3D Simulator of Multiple Legged Robots based on USARSim. *Robocup 2006: Robot Soccer World Cup X*, Springer, LNAI, 2006.

The Rat's Life Benchmark: Competing Cognitive Robots

Olivier Michel
Cyberbotics Ltd.
PSE C - EPFL
CH-1015 Lausanne, Switzerland
Olivier.Michel@cyberbotics.com

Fabien Rohrer
Cyberbotics Ltd.
PSE C - EPFL
CH-1015 Lausanne, Switzerland
Fabien.Rohrer@cyberbotics.com

ABSTRACT

This paper describes Rat's Life: a complete cognitive robotics benchmark that was carefully designed to be easily reproducible in a research lab with limited resources. It relies on two e-puck robots, some LEGO bricks and the Webots robot simulation software. This benchmark is a survival game where two robots compete against each other for resources in an unknown maze. Like the rats in cognitive animal experimentation, the e-puck robots look for feeders which allow them to live longer than their opponent. Once a feeder is reached by a robot, the robot draws energy from it and the feeder becomes unavailable for a while. Hence, the robot has to further explore the maze, searching for other feeders while remembering the way back to the first ones. This allows them to be able to refuel easily again and again and hopefully live longer than their opponent.

Keywords

benchmark, SLAM, navigation, autonomy, vision

1. WHY WE NEED COGNITIVE ROBOTICS BENCHMARKS

1.1 Introduction

Most scientific publications in the area of robotics research face tremendous challenges: comparing the achieved result with other similar research results and hence convincing the reader of the quality of the research work. These challenges are very difficult because roboticists lack common tools allowing them to evaluate the absolute performance of their systems or compare their results with others. As a result, such publications often fail at providing verifiable results, either because the studied system is unique and difficult to replicate or they don't provide enough experimental details so that the reader could replicate the system accurately.

Nevertheless, some of these publications become the de facto state of the art and this makes it extremely difficult to further explore these research areas, and hence to demonstrate advances in robotics research.

This matter of fact is unfortunately impairing the credibility of robotics research. A number of robotics researchers proposed to develop series of benchmarks to provide a means of evaluation and comparison of robotics research results [1, 3, 4, 9, 19].

Although a few robotics benchmarks already exist, the only robotics benchmarks that are widely known and practiced are implemented as robot competitions.

1.2 Existing Robot Competitions and Benchmarks

Several popular robot competitions are organized on a regular basis, usually once a year. The Robocup soccer [6] and FIRA [16] is a robot soccer tournament with several categories (small size league, middle size league, standard platform league, simulation league, etc.). The Robocup Rescue is based on the Urban Search And Rescue (USAR) benchmark developed by the NIST [7] where robots have to search and rescue the victims of a disaster in a urban environment. MicroMouse [11], involves wheeled robots solving a maze. The FIRST [12], Eurobot [13] and Robolympics [14] are robot competitions more focused on education than research in various disciplines (often inspired from sports). The AAAI Robot Competition [15] proposes different scenarios each year during the AAAI conference, but often lacks a clear performance metrics. The DARPA Grand Challenge and Urban Challenge [17] and the European Land-Robot Trial [18] focus on unmanned ground and sometimes aerial vehicles racing against each other.

Although some of these competitions clearly focus on education and are more intended to students and children rather than researchers (FIRST, Eurobot), others competitions (Robocup, FIRA, AAAI) are more intended to researchers. Such competitions are useful as they can provide elements of comparison between different research results. However one of the major problem is that the rules often change across the different editions of the same competition. Hence it is difficult to compare the progress achieved over time. Also these competitions are very specific to particular problems, like Robocup is focused mostly on robot soccer and has arguably a limited interest for cognitive robotics [5]. Moreover, in most cases, and especially in the Robocup case, installing a contest setup is expensive and takes a lot of resources (many robots, robot environment setup, room,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

maintenance, controlled lighting conditions, etc.).

1.3 Going Further with Cognitive Robotics Benchmarks

Among all the benchmarks we reviewed which are mostly robot competitions, none of them provides both stable rules with advanced cognitive robotics challenges and an easy setup. This paper proposes a new robotics benchmark called "Rat's Life" that addresses a number of cognitive robotics challenges while being cheap and very easy to setup for any research lab. The aim of this benchmark is to foster advanced robotics and AI research.

Comparing to soccer playing contests (RoboCup, FIRA), the Rat's Life benchmark is more bio-inspired as it focuses on foraging and survival. Also, it is more likely to contribute to scientific advances in Learning and Self Localization And Mapping (SLAM) as mazes are initially unknown to the robots. Moreover, it allows the researchers to focus on a single agent (competiting against another) rather than a whole team of agents, making the problem somehow simpler to handle. Finally, it is cheaper.

2. BENCHMARK REQUIREMENTS

In order to be useful a benchmark has to be practiced by a large number of the best researchers trying to push further the current state of the art. This can be achieved by proposing a scientifically and practically appealing series of benchmarks that will convince researchers to invest their time with these tools. Hence the Rat's Life benchmark is trying to achieve a number of objectives:

2.1 Scientifically appealing

To be scientifically interesting, a benchmark has to address a number of difficult challenges in robotics. The Rat's Life benchmark focuses on cognitive robotics and addresses advanced research topics such as image processing, learning, navigation in an unknown environment, landmark recognition, SLAM, autonomy management, game strategies, etc.

2.2 Cheap and easy to setup

The benchmark should be easily practicable by any researcher. Hence it has to be cheap and easy to setup. All the components should be easily available. The Rat's Life benchmark costs no more than EUR 2000, for two e-puck robots and many LEGO components (including a LEGO NXT unit and four LEGO distance sensors). It requires only a table to setup a LEGO maze of 114x114 cm.

2.3 Accurate

Accuracy is a very important aspect of a benchmark. The environment, robots and evaluation rules should be defined very carefully in an exhaustive manner. This way, the benchmark is accurately replicable and hence different results obtained with different instances of the setup in different research lab can be compared to each others.

2.4 Comparable

Finally, a benchmark is useful if users can compare their own results to others and thus try to improve the state of the art. Hence a benchmark should keep a data base of the solutions contributed by different researchers, including binary and source code of the robot controller programs. These different solutions should be ranked using a common performance metrics, so that we can compare them to each other.

3. STANDARD COMPONENTS

The Rat's Life benchmark is based on three standard affordable components: the e-puck mobile robot, LEGO bricks and the Webots robot simulation software (free version).

3.1 The e-puck mobile robot

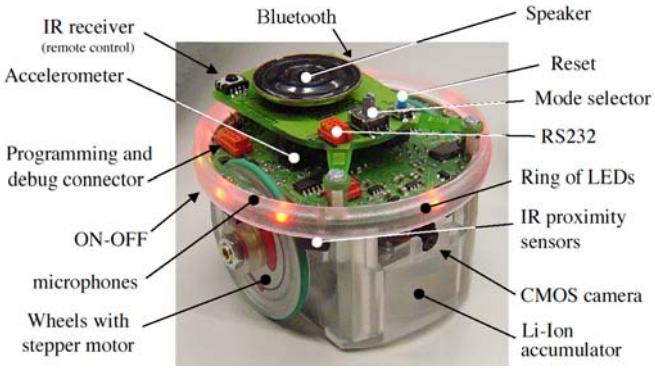


Figure 1: The e-puck robot

The e-puck mini mobile robot was originally developed at the EPFL for teaching purposes by the designers of the successful Khepera robot. The e-puck hardware and software is fully open source, providing low level access to every electronic device and offering unlimited extension possibilities. The robot is already equipped with a large number of sensors and actuators (figure 1). It is well supported by the Webots simulation software with simulation models, remote control and cross-compilation facilities. The official e-puck web site [23] gathers a large quantity of information about the robot, extension modules, software libraries, users mailing lists, etc. The robot is commercially available from Cyberbotics [21] for about EUR 570.

3.2 LEGO bricks

The LEGO bricks are used to create an environment for the e-puck robot. This environment is actually a maze which contains "feeder" devices (see next sections) as well as visual landmarks made up of patterns of colored LEGO brick in the walls of the maze (see figure 2). These landmarks are useful hints helping the robot to navigate in the maze. Since LEGO models are easily demountable, the maze is easily reconfigurable so that the users can create different instances of the maze according to the specifications of the benchmark.

All the maze, landmarks and the feeder devices are properly defined in a LEGO CAD file in LXF format using the LEGO digital designer software freely available from the LEGO factory web site [20]. The corresponding LXF files are freely available on the Rat's Life web site [10].

Thanks to the LEGO factory system, users can very easily order a box containing all the LEGO bricks necessary to build the environment of the robots.

3.3 The Webots robot simulation software

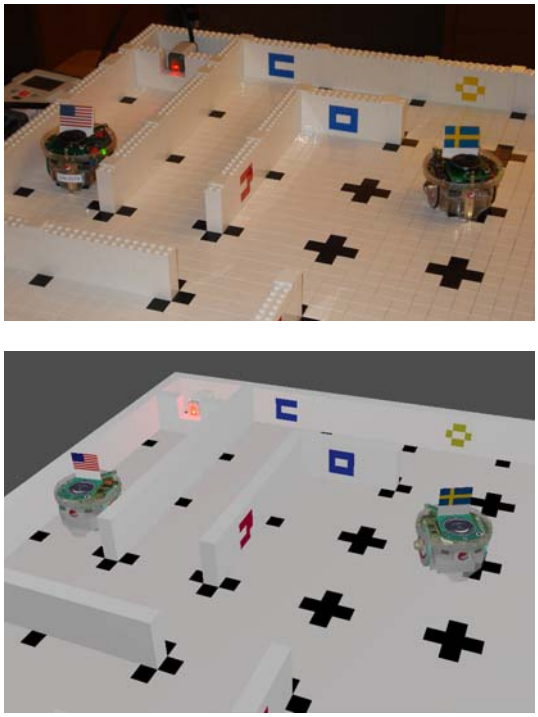


Figure 2: The Rat's Life maze: LEGO bricks, e-puck robots and a feeder device (left) and its simulated counterpart (right)

Webots [8] is a commercial software for fast prototyping and simulation of mobile robots. It was originally developed at the Swiss Federal Institute of Technology in Lausanne (EPFL) from 1996 and has been continuously developed, documented and supported since 1998 by Cyberbotics Ltd. Over 500 universities and industrial research centers worldwide are using this software for research and educational purposes. Webots has already been used to organize robot programming contests (ALife contest and Roboka contest). Although Webots is a commercial software, a demo version is freely available from Cyberbotics's web site [21]. This demo version includes the complete Rat's Life simulation. So, anyone can download, install and practice the simulation of the Rat's Life benchmark at no cost.

4. RAT'S LIFE BENCHMARK DESCRIPTION

This paper doesn't claim to be a technical reference for the Rat's Life benchmark. Such a technical reference is available on the Rat's Life web site [10].

4.1 Software-only Benchmark

The Rat's Life benchmark defines precisely all the hardware necessary to run the benchmark (including the robots and their environment). Hence the users of the benchmarks don't have to develop any hardware. Instead, they can focus on robot control software development only. This is similar to the Robocup standard league where the robot platforms (Aibo robots) and the environment is fully defined and the competitors are limited to develop control software only. This has the disadvantage of preventing hardware research and is constraining the contest to the defined hardware only.

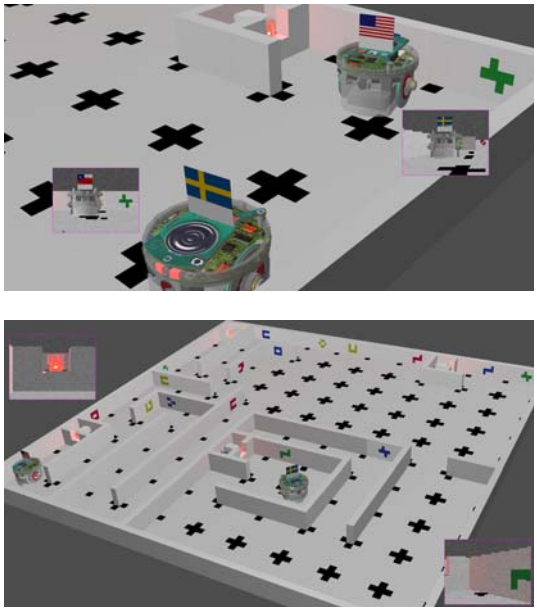


Figure 3: Closeup of the Rat's Life simulated robots in Webots (left) and general overview (right)

However, it has the great advantage of letting the users focus on the most challenging part of cognitive robotics, i.e., the control software.

4.2 Configuration of the Maze

For each evaluation, the maze is randomly chosen among a series of 10 different configurations of the maze. In each configuration, the walls, landmarks and feeder are placed at different locations to form a different maze. Each configuration also has 10 different possible initial positions and orientations for the two robots. One of them is chosen randomly as well. This makes 100 possible initial configurations. This random configuration of the maze prevents the robots from having a prior knowledge of the maze, and forces them to discover their environment by exploring it. This yields to much more interesting robot behaviors. A possible configuration is depicted on figure 3 (right).

4.3 Virtual Ecosystem

The Rat's Life benchmark is a competition where two e-puck robots compete against each other for resources in a LEGO maze. Resources are actually a simulation of energy sources implemented as four feeder devices. These feeder devices are depicted on figure 4. They are made up of LEGO NXT distance sensors which are controlled by a LEGO NXT control brick. They display a red light when they are full of virtual energy. The e-puck robots can see this colored light through their camera and have to move forward to enter the detection area of the distance sensor. Once the sensor detects the robot, it turns its light off to simulate the fact that the feeder is now empty. Then, the robot is credited an amount of virtual energy corresponding to the virtual energy that was stored in the feeder. This virtual energy will be consumed as the robot is functioning and could be interpreted as the metabolism of the rat robot. The feeder will remain empty (i.e., off) for a while. Hence the robot has to find another feeder with a red light on to get more

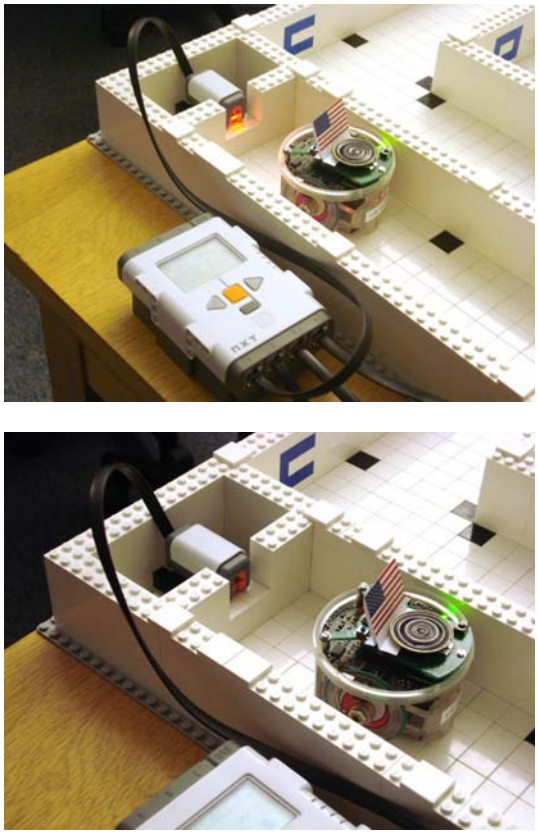


Figure 4: A full feeder facing an e-puck robot (left) and an empty one (right)

energy before its energy level reaches 0. When a robot runs out of virtual energy (i.e., its energy level reaches 0), the other robot wins.

4.3.1 Biological Comparison

This scenario is comparable to an ecosystem where the energy is produced by feeders and consumed by robots. The feeders could be seen as plants, slowy growing from the energy of the sun, water and ground and producing fruits whereas the robots could be seen as rats, foraging fruits. Since the fruits produced by a single plant are not sufficient to feed a rat, the rat has to move around to find more plants.

4.3.2 Electronics Comparison

Although the e-puck robots and feeder devices used in the Rat's contest are electronic devices, the energy is actually simulated for convenience reasons. However, it could be possible to deal with real electrical energy: the feeder would correspond to photovoltaic solar docking stations accumulating electrical energy over time and the robots could recharge their actual battery from these stations. However, such a system would be more complex to setup from a practical and technological point of view. This is why we decided to use virtual energy instead.

4.4 Robotics and AI Challenges

Solving this benchmark in an efficient way requires the following cognitive capabilities:

- Recognize a feeder (especially a full one) from a camera image.
- Navigate to the feeder and dock to it to grab energy.
- Navigate randomly in the maze while avoiding to get stuck.
- Remember the path to a previously found feeder and get back to it.
- Optimize energy management.
- Try to prevent the other robot from getting energy.

This translates into a number of control software techniques, namely image processing, motor control, odometry, landmark based navigation, SLAM, autonomy management, game theory. Most of these techniques are still open research areas where new progress will benefit directly to robotics and AI applications. Both bio-inspired (neural networks, generic algorithms, learning) and traditional approaches (control theory, environment mapping) are concerned as no assumption is made on the technologies used to implement the controllers. Moreover, because of its similarities with experiments with rodents, the Rat's Life contest may be a very interesting benchmark for testing different bio-inspired models, such as place cells, grid cells, spatial learning, conditioning, etc.
The best robots are expected to be able to somehow fully memorize the maze they explore with the help of the landmarks, to rapidly find their way to the feeders, to maintain an estimation of the status of every feeder and to develop a strategy to prevent the opponent from recharging.

4.5 Online Contest

4.5.1 Real World and Simulation

The Rat's Life contest is defined both as a real environment and a simulation. However, the same control programs, written in C or Java programming language, can run on both the simulation and the real robots. To run the control program on the real robots, there are actually two options. The user can either execute the controller program on a computer remote controlling the robot or cross-compile it and execute it on the real robot. In the first case, the program running on the computer remote controls the real robot by reading the sensor values from and sending the motor commands to the Bluetooth connection with the robot. In the second case, the control program is executed directly on the real robot. All the necessary software tools for remote control and cross-compilation are integrated within the Webots software, making the transfer from the simulation to the real robot a very easy process. This way, the same controller program can control both the real and the simulated robot.

4.5.2 Participation to the Contest

In order to participate in the online contest, the competitors can download the free version of Webots from Cyberbotics' web site [21]. They can program the simulated e-puck robots to perform in the simulated maze. Then, they have to register a contestant account on the contest's web site [10]. Once open, this account allows the competitors to upload the controller programs they developed with the free version of Webots. Participation to the contest is totally free of charge.

4.5.3 Ranking System

Every business day (i.e., Monday to Friday) at 12 PM (GMT) a competition round is started in simulation and can be watched online from the Rat's Life web site [10]. A hall of fame displays a table of all the competitors registered in the data base and who submitted a robot controller program. If there are N competitors in the hall of fame, then $N-1$ matches are played. The first match of a round opposes the last entry, i.e., number N at the bottom of the hall of fame, to the last but one entry, i.e., number $N-1$. If the robot number N wins, then the position of these two robots in the hall of fame are switched. Otherwise no change occurs in the hall of fame. This procedure is repeated with the new robot number $N-1$ (which may have recently changed due to the result of the match) and robot number $N-2$. If robot number $N-1$ wins, then it switches its position with robot number $N-2$, otherwise nothing occurs. This is repeated with robots number $N-3$, $N-4$, etc. until robots number 2 and 1, thus totaling a number of $N-1$ matches.
This ranking algorithm is similar to the bubble sort. It makes it possible for a newcomer appearing initially at the bottom of the ranking, to progress until the top of the ranking in one round. However, any existing entry cannot loose more than one position in the ranking during one round. This prevents a rapid elimination of a good competitor (which could have been caused by a buggy update of the controller program for example).

4.5.4 And The Winner Is...

The contest is open for a fixed period of time. During this period of time, new contestants can register and enter the contest. The contestants can submit new versions of their controller program any time until the closing date. Once the closing date is reached, new entry and submissions of new versions are disabled. Then, seven final rounds are run. The final ranking for each competitor is computed as the average ranking over these seven final rounds. The competitor ranked at the top position is declared to be the "winner of the simulated Rat's Life benchmark" and its authors receive a prize for this: a Webots PRO pack. Moreover the top 3 competitors are selected for a real world series of 3 rounds (i.e., 6 matches). The winner of these real world rounds is declared to be the "winner of the real world Rat's Life benchmark" and receives a prize: an e-puck robot.
The real world rounds should however occur during an international conferences or robotics competition to ensure that a large number of people, including a scientific committee, attends the event and can check that nobody is cheating the benchmark.
The contest will run continuously over years so that we can measure the progress and performances of the robot controllers over a fairly long period of robotics and AI research.

5. EVOLUTION OF THE COMPETITION OVER TIME

5.1 Movie Database

Observing the evolution of the competition over days was very interesting and we decided to store all the simulation movies in a data base to be able to analyse this evolution afterwards. The movie database contains more than 2500 movies (totalling more than 50 GB of data) and is freely available online at http://www.cyberbotics.com/ratslife/movies/.

5.2 Participation to the Contest

The competition started on January 7th, 2008 with one, then two competitors. Table 1 summarizes the evolution of the participation to the contest over time.

Table 1: Evolution of the participation to the Rat's Life contest

Date	Number of Competitors
January 31st	6
February 29th	15
March 31st	23
April 30th	26
May 31st	36
June 30th	to be completed

5.3 Evolution of Behaviors

The behavior of the robot controllers evolves in a similar fashion as a genetic algorithm. Competitors are uploading new versions of their robot controller fairly frequently. Sometimes, a new version brings a significant performance breakthrough and drives the competitors' robot on the top of the hall of fame. This performance breakthough is then immediately analysed by the other competitors seeing the simulation movies. They take inspiration from it to improve their own robot controller and submit their new improved version to the contest. Rapidly, a large number of competitors can replicate this winning behavior on their robot and most of the robots in the contest adopt this new efficient behavior. This co-evolution dynamics is similar to what happens in genetic algoritm where the behavior of efficient individuals spreads rapidly across the population. If we could plot a fitness function over time, such breakthroughs would correspond to suddent increase of the fitness value making steps in the evolutionary process.

5.4 Major breakthroughs

During the contest, several major performance breakthroughs could be observed simply by analysing the behavior of the robots in the simulation movies. One could identify five major breakthroughs which happened chronologically one after the other, bringing each time an improved performance:

5.4.1 Random Walkers

The random walkers came actually from the very first version of the sample source code included with the contest software development kit, made available to all the competitors. This simple control algorithm similar to Braitenberg vehicles [2] let the robots move randomly while avoiding the obstacles. By chance some of them met a feeder from time time, but this behavior is very inefficient are rely mostly on luck. Also, this very first version was not very efficient at navigating and often caused the robot to get stuck in some unexpected situations, like facing a corner.

5.4.2 Vision-Enabled Random Walkers

Last round completed Jun-05-2008 - 37 active competitors out of 139

Rank	Country	Flag	Rat's name	Author	Ready	Last update	Progress	Movies
1			Ratatouille	Manuel	✔	Jun-05-2008 23:50	-	
2			golum	Laurent	✔	Jun-03-2008 13:25	+2	
3			Krysicka	Martin Pilat	✔	May-06-2008 12:26	-1	
4			AMT	Ares Macrotechnology	✔	May-24-2008 20:08	-1	
5			Bins.GPR	Ricardo Luis Binsfeld	✔	Apr-17-2008 17:05	-	
6			Ratchou	Xavier	✔	Jun-03-2008 21:01	+3	
7			muchachada wii	muchachada wii	✔	May-31-2008 13:35	-1	
8			Tony	Anthony Morse	✔	Jun-05-2008 16:02	-1	
9			Wayne	Wayne	✔	Jun-03-2008 01:51	-1	
10			Comstar	Comstar	✔	May-03-2008 19:38	+1	
11			hippohotamus	Hip Rohotamus	✔	Mar-30-2008	-1	

Figure 5: Rat's Life Hall Of Fame, on June 6th, 2008

The so called vision-enabled random walkers are an improved version of the original random walker making an extensive use of vision to recognize the feeders and adjust the trajectory of the robot to reach the feeder instead of simply moving randomly. This results in slightly more efficient robots who won't pass in front of a feeder without getting energy from it. A vision-enabled random walker is included in the sample code currently distributed to the competitors. This sample version has however been largely improved by different competitors over time.

5.4.3 *Right Hand Explorers*

One of the problems with the random walkers is that a Braitenberg vehicle behavior is not very efficient at exploring extensively a maze and hence at finding the feeders. Maze exploration algorithms exist and are much more efficient. The right hand algorithm is one of the simplest and best known maze exploration algorithms. It consists in simply following the first wall found on the right hand side of the robot (this also works with the left hand side of course). Using this algorithm combined with some vision to reach efficiently the feeders, a significant performance breakthrough was reached. The first right hand explorer appeared on February 22nd, with a robot named Tony (which reached rank #1 of the hall of fame on February 22th very rapidly) and was rapidly copied by many other competitors as this behavior is both easy to understand and to re-program.

5.4.4 *Energy-aware robots*

Getting the energy from the feeder as soon as you find the feeder is nice, but there is an even better strategy: Once a robot finds a feeder, it can simply stop and sit in front of the feeder, thus preventing the other robot from reaching this feeder. In the meanwhile the robot sitting in front of the feeder should watch its energy level and decide to move to the feeder once its energy level reached a very low value, just enough to make that move to the feeder and refuel. During this waiting time, the other robot may be struggling to find a feeder and possibly loose the game if it runs out of energy. This kind of energy-aware robots appeared on February 28th, with a robot named Ratchou (which reached rank #1 thanks to this breakthrough). Similarly to the right hand explorer, it was rapidly copied by other competitors as it was easy to understand and to re-program as well.

5.4.5 *SLAMers*

SLAM stands for Self Localization And Mapping. Comparing to other techniques mentioned above, it involves a much more complicated algorithm and requires an efficient image processing. SLAMer robots actually seems to use the right hand algorithm on a first stage to explore extensively the maze, but they build dynamically a map of this maze while exploring it and eventually don't use the right hand algorithm at all. Their internal representation of the environment contains the walls, the feeders and likely the landmarks. This map is then used by the robot to get back to previously found feeders. It turned out to be very efficient and clearly outperformed the simpler reactive controllers. The first SLAMer robot is Ratatouille who implemented a first version of visual SLAM-based navigation on April 6th and reach rank #1. This first version was however probably not well tuned (or somehow buggy) and it happened to loose in rare occasions against lucky and efficient right-hand explorers. However, the author of Ratatouille continued to improve the performance of his SLAMer robot and finally sat steadily on the very top of the hall of fame for more than two months. The other competitors, including Tony among others, tried hard to implement such an efficient SLAM-based navigation controller, they were not very successful until June 5th. At this point a competitor with a robot controller named gollum developed a pretty efficient SLAMer robot able to challenge Ratatouille. Golum reached rank #2 on June 5th and had a really disputed and very interesting match against Ratatouille, but was not successful.

(to be continued)

5.4.6 *Super-SLAMers*

The author of Ratatouille actually never stopped from April 6th to improve his SLAM-based robot controller. A major improvement was probably the estimation of the status of the feeders, combined with an estimation of the time needed to travel the maze to reach the feeder. From the most recent simulation movies, Ratatouille seems to be able to anticipate that a mapped feeder will become available again: when the feeder is still red, Ratatouille starts to navigate towards this feeder and about one second before it reaches the feeder, the feeder becomes green again. This makes Ratatouille the most efficient robot controller currently on the Rat's Life benchmark (see figure 5). At this point, it is difficult to imagine a better behavior than the one exhibited by Ratatouille.

6. TRANSFER TO THE REAL WORLD

At the time of writting, the competition occured only in simulation. We plan to run a series of rounds in the real world, using the same robot controller as the ones used in simulation for the top three robots. We are currently working to fix wireless communication speed issues to allow a reliable and real time remote control of real robots allowing to test the best Rat's Life controllers in the real world. We believe that some calibrations of the sensors, motors and image processing algorithms might be necessary to be able to transfer the best control algorithms from the simulation to the real world. This issue will be discussed in an upcoming publication.

7. CONCLUSION

Thanks to the Rat's Life benchmark, it becomes possible to evaluate the performance of various approaches to robot control for navigation in an unknown environment, including various SLAM and bio-inspired models. The performance evaluation allow us to make a ranking between the different control programs submitted, but also to compare the progresses achieved over a short period of time of research on this problem. However, this period of time could be extented and we could, for example, compare the top 5 controller programs developed in 2008 to the top 5 controller programs developed in 2012 to evaluate how much the state of the art progressed.

The control program resulting from the best robot controllers could be adapted to real world robotics applications in the areas of surveillance, mobile manipulators, UAV, cleaning, toys, etc. Also, interesting scientific comparisons with biological intelligence could be drawn by opposing the best robot controllers to a real rat (or a rat-controlled robot) in a similar problem. Similarly, we could also oppose the best robot controllers to a human (possibly a child) remote controlling the robot with a joystick and with limited sensory information coming only from the robot sensors (mainly the camera).

We hope that this initiative is a step towards a more general usage of benchmarks in robotics research. By its modest requirements, simplicity, but nevertheless interesting challenges it proposes, the Rat's Life benchmark has the potential to become a successful reference benchmark in cognitive robotics and hence open the doors to more complex and advanced series of cognitive robotics benchmarks.

Acknowledgements

The authors of this paper would like to express their acknowledgments to the IST Cognitive Systems Unit of the European Commission which funds the ICEA European project [22] within which the Rat's Life benchmark and contest were developed. ICEA stands for Integrating Cognition, Emotion and Autonomy and is focused on brain-inspired cognitive architectures, robotics and embodied cognition, bringing together cognitive scientists, neuroscientists, psychologists, computational modelers, roboticists and control engineers. It aims at developing a cognitive systems architecture integrating cognitive, emotional and bioregulatory (self-maintenance) processes, based on the architecture and physiology of the mammalian brain.

8. REFERENCES

[1] Baltes, J.: "A benchmark suite for mobile robots", IROS 2000 conference proceeding (2000), 1101-1106 vol.2, ISBN: 0-7803-6348-5

[2] Braitenberg, V.: "Vehicles: Experiments in Synthetic Psychology" Cambridge, MA: MIT Press (1984)

[3] Serri, A.: "A Lego robot for experimental benchmarking of robust exploration algorithms", Circuits and Systems (2004), iii - 163-6 vol.3, ISBN: 0-7803-8346-X

[4] Eaton, M., Collins, J. J., and Sheehan, L.: "Toward a benchmarking framework for research into bio-inspired hardware-software artefacts", Artificial Life and Robotics, Springer Japan, 40-45 Vol.5 number 1 / March 2001, ISSN: 1433-5298

[5] Collins, J.J., Eaton, M., Mansfield, M., Haskett, D., OâĂŹSullivan, S.: "Developing a benchmarking framework for map building paradigms", Proceedings of the 9th International Symposium on Artificial Life and Robotics, January 2004, Oita, Japan, pp. 614-617

[6] Kitano, H., Asada, M., Kuniyoshi, Y., et al: "RoboCup: the robot world cup initiative", IJCAI-95 workshop on entertainment and AI/ALife (1995)

[7] Jacoff, A., Weiss, B., Messina, E.: "Evolution of a Performance Metric for Urban Search and Rescue Robots", Proceedings of the 2003 Performance Metrics for Intelligent Systems (PerMIS) Workshop, Gaithersburg, MD, September 16-18, 2003

[8] Michel, O. "Webots: Professional Mobile Robot Simulation", Journal of Advanced Robotics Systems, 39-42 Vol.1 number 1 / 2004 http://www.ars-journal.com/International-Journal-of-Advanced-Robotic-Systems/Volume-1/39-42.pdf

[9] Dillmann, R.: "Benchmarks for Robotics Research", EURON, April 2004. http://www.cas.kth.se/euron/eurondeliverables/ka1-10-benchmarking.pdf

[10] Rat's Life contest. http://www.ratslife.org

[11] Micromouse contest. http://www.micromouseinfo.com

[12] FIRST Robotics competition. http://www.usfirst.org

[13] Eurobot Robotics competition. http://www.eurobot.org

[14] ROBOlympics. http://www.robolympics.net

[15] AAAI Robot Competition and exhibition. http://palantir.cs.colby.edu/aaai07

[16] The Federation of International Robot-soccer Association (FIRA). http://www.fira.net

[17] DARPA Grand Challenge. http://www.darpa.mil/grandchallenge

[18] European Land-Robot Trial ELROB. http://www.elrob.org/

[19] EURON Benchmarking Activities. http://www.euron.org/activities/benchmarks/index.html

[20] LEGO factory. http://factory.lego.com

[21] Cyberbotics Ltd. http://www.cyberbotics.com

[22] ICEA European Project (IST 027819) http://www.iceaproject.eu

[23] e-puck web site. http://www.e-puck.org

The iCub humanoid robot: an open platform for research in embodied cognition

(Special Session on EU-projects)

Giorgio Metta
Giulio Sandini
Italian Institute of Technology and University of Genoa
Via Morego, 30
16163 Genoa, Italy
+39 010 71781411
{giorgio.metta, giulio.sandini}@iit.it

David Vernon
Khalifa University
P.O. Box 573
Sharjah
david@vernon.eu

Lorenzo Natale
Francesco Nori
Italian Institute of Technology
Via Morego, 30
16163, Genoa, Italy
+39 010 71781420
{lorenzo.natale, francesco.nori}@iit.it

ABSTRACT

We report about the iCub, a humanoid robot for research in embodied cognition. At 104 cm tall, the iCub has the size of a three and half year old child. It will be able to crawl on all fours and sit up to manipulate objects. Its hands have been designed to support sophisticate manipulation skills. The iCub is distributed as Open Source following the GPL/FDL licenses. The entire design is available for download from the project homepage and repository (http://www.robotcub.org). In the following, we will concentrate on the description of the hardware and software systems. The scientific objectives of the project and its philosophical underpinning are described extensively elsewhere [1].

Categories and Subject Descriptors

I.2.9 [**Artificial Intelligence**]: Robotics – *Commercial robots and applications, Kinematics and dynamics, Manipulators, Operator interfaces, Sensors.*

General Terms

Experimentation, Standardization.

Keywords

Humanoid robotics, cognitive systems, open source, software modularity.

1. INTRODUCTION

RobotCub is a collaborative project funded by the European Commission under the sixth framework programme (FP6) by Unit E5: Cognitive Systems, Interaction and Robotics. It has the two-fold goal of: i) creating an open hardware/software humanoid robotic platform for research in embodied cognition, and ii) advancing our understanding of natural and artificial cognitive systems by exploiting this platform in the study of the development of cognitive capabilities.

The RobotCub stance on cognition posits that manipulation plays a fundamental role in the development of cognitive capability [1-4]. As many of these basic skills are not ready-made at birth, but developed during ontogenesis [5], RobotCub aims at testing and developing this paradigm through the creation of a child-like humanoid robot: the iCub. This "baby" robot will act in cognitive scenarios, performing tasks useful for learning while interacting with the environment and humans. The small (104cm tall), compact size (approximately 22kg and fitting within the volume of a child) and high number (53) of degrees of freedom combined with the Open Source approach distinguish RobotCub from other humanoid robotics projects developed worldwide.

In this paper, we focus on the description of the iCub, both in terms of hardware and software. In particular, we will briefly discuss the rationale of the hardware design, the modularity and reuse of software components, and the consequences of the Open Source distribution policy. RobotCub has, in parallel, the goal of advancing the science and engineering of cognitive systems. This part of the research has been discussed elsewhere in greater detail [4, 6] and will not be reported here.

The hardware of iCub has been specifically optimized and designed somewhat holistically: modularity in this case had to be traded for functionality and overall size. Software, on the other hand, has been designed with modularity and component reuse in mind [7]. Both the hardware and software of the iCub have been released under the GPL and FDL licenses. The mechanical drawings, the electronics, schematics and documentation are available from the RobotCub website. The iCub software is available as well from the same repository together with a basic dynamical simulator of the robot.

Additional initiatives are aiming at promoting the iCub as the platform of choice for research in embodied cognition. Fifteen

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

robots are expected to be delivered by the end of the project (end of 2009) as part of RobotCub and of other EU funded projects.

2. THE ICUB

The iCub has been designed to allow manipulation and mobility. For this reason 30 degrees of freedom (DOF) have been allocated to the upper part of the body. The hands, in particular, have 9 DOF each with three independent fingers and the fourth and fifth to be used for additional stability and support (only one DOF). They are tendon driven, with most of the motors located in the forearm. The legs have 6 DOF each and are strong enough to allow bipedal locomotion.

From the sensory point of view, the iCub is equipped with digital cameras, gyroscopes and accelerometers, microphones, and force/torque sensors. A distributed sensorized skin is under development using capacitive sensor technology.

Each joint is instrumented with positional sensors, in most cases using absolute position encoders. A set of DSP-based control cards, designed to fit the iCub, take care of the low-level control loop in real-time. The DSPs talk to each other via CAN bus. Four CAN bus lines connect the various segments of the robot.

All sensory and motor-state information is transferred to an embedded Pentium based PC104 card that handles acquisition and reformatting of the various data streams. Time consuming computation is typically carried out externally on a cluster of machines. The communication with the robot occurs via a Gbit Ethernet connection.

The overall weight of the iCub is 22kg. The umbilical cord contains both an Ethernet cable and power to the robot. At this stage there is no plan for making the iCub fully autonomous in terms of power supply and computation (e.g. by including batteries and/or additional processing power on board).

The mechanics and electronics were optimized for size, starting from an evaluation and estimation of torques in the most demanding situations (e.g. crawling). Motors and gears were appropriately sized according to the requirements of a set of typical tasks. The kinematics was also defined following similar criteria. The controllers were designed to fit the available space. Figure 5 shows the prototype of the iCub.

2.1 Mechanics

The kinematic specifications of the body of the iCub, the definition of the number of DOF, their actual locations as well as the actual size of the limbs and torso were based on ergonomic data and x-ray images.

The possibility of achieving certain motor tasks is favored by a suitable kinematics and, in particular, this translates into the determination of the range of movement and the number of controllable joints (where clearly replicating the human body in detail is impossible with current technology). Kinematics is also influenced by the overall size of the robot which was imposed *a priori*. The size is that of a 3.5 years old child (approximately 100cm high). This size can be achieved with current technology. QRIO[1] is an example of a robot of an even smaller size although with less degrees of freedom. In particular, our task specifications, especially manipulation, require at least the same kinematics of QRIO with the addition of the hands and moving eyes. Also, we considered the workspace and dexterity of the arms and thus a three degree of freedom shoulder was included. This was elaborated into a proper list of joints, ranges, and sensory requirements at the joint level.

Considering dynamics, the most demanding requirements appear in the interaction with the environment. Impact forces, for instance, have to be considered for locomotion behaviors, but also and more importantly, developing cognitive behaviors such as manipulation might require exploring the environment erratically. As a consequence, it is likely that high impact forces need to be sustained by the robot mechanical structure. This requires strong joints, gearboxes, and more in general powerful actuators and appropriate elasticity (for absorbing impacts). In order to evaluate the range of the required forces and stiffness, various behaviors were simulated in a dynamical model of the robot. These simulations provided the initial data for the design of the robot. The simulations were run using Webots[2] and were later cross-checked by traditional static analysis.

At a more general level, we evaluated the available technology, compared to the experience within the project consortium and the targeted size of the robot: it was decided that electric motors were the most suitable technology for the iCub, given also that it had to be ready according to the very tight schedule of the overall project. Other technologies (e.g. hydraulic, pneumatic) were left for a “technology watch” activity and were not considered further for the design of the iCub.

From the kinematic and dynamic analysis, the total number of degrees of freedom for the upper body was set to 38 (7 for each arm, 9 for each hand, and 6 for the head). For the legs the simulations indicated that for crawling, sitting and squatting a 5 DOF leg is adequate. However, it was decided to incorporate an additional DOF at the ankle to support standing and walking. Therefore each leg has 6 DOF: these include 3 DOF at the hip, 1 DOF at the knee and 2 DOF at the ankle (flexion/extension and abduction/adduction). The foot twist rotation was not implemented. Crawling simulation analysis also showed that for effective crawling a 2 DOF waist/torso is adequate. However, to support manipulation a 3 DOF waist was incorporated. A 3 DOF waist provides increased range and flexibility of motion for the upper body resulting in a larger workspace for manipulation (e.g. when sitting).

The neck has a total of 3 DOF and provides full head movement. The eyes have further 3 DOF to support both tracking and vergence behaviors.

The actuation solution adopted for the iCub is based on a combination of a harmonic drive reduction system (CSD series, 100:1 ratio for all the major joints) and a brushless frameless motor (BLM) from the Kollmorgen frameless RBE series (Figure 1). The harmonic drive gears provide zero backlash, high reduction ratios on small space with low weight while the brushless motors exhibit the desired properties of robustness, high power density, and high torque and speed bandwidths (especially when compared with conventional DC brushed motors). The use of frameless motors permits integration of the motor and gears in an endoskeletal structure that minimizes size,

[1] http://www.sony.net/Fun/design/history/product/2000/sdr-4x.html

[2] http://www.cyberbotics.com/products/webots/webots5.pdf

weight and dimensions. Smaller motors (brushed-DC type) were used for the hands and head joints.

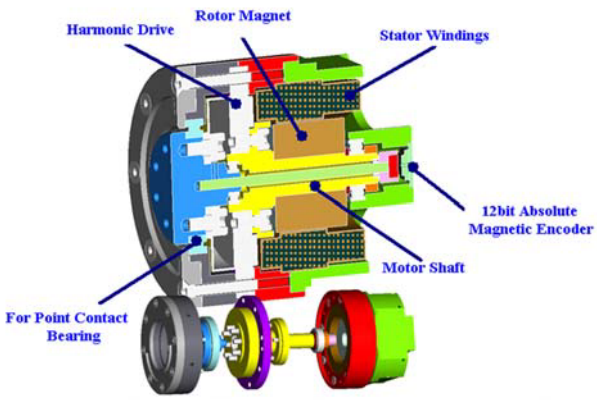


Figure 1: section of the standard brushless motor group of the iCub. Positioning of the motor and gears can be noted (as indicated). Figure from [8]. Note the compact assembly of the frameless motor and harmonic drive gear.

An example on the use of this structure is depicted in Figure 2, which shows the shoulder of the iCub with details of the motor enclosure and tendon-driven pulley mechanisms.

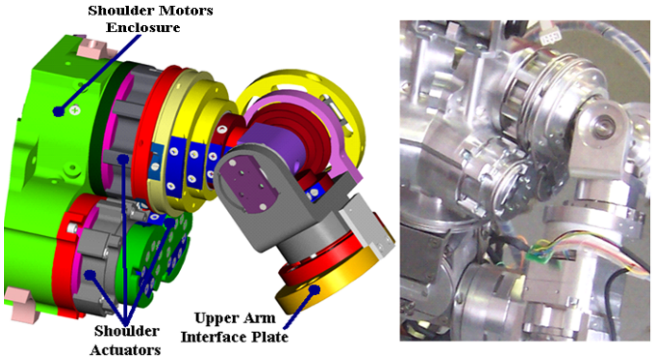


Figure 2: the shoulder of the iCub. Left: CAD schematics. Right: the implementation. Note the three DOF of the shoulder with intersecting axes of rotation and the placement of the actuators in the chest as indicated. 1.75mm steel cables join the movement of the motors with the pulleys actuating the joints.

Certain features of the iCub are unique. Tendon driven joints are the norm both for the hand and the shoulder, but also in the waist and ankle. This reduces the size of the robot but introduces elasticity that has to be considered in designing control strategies where high forces might be generated.

The hand, for example, is fully tendon-driven. Seven motors are placed remotely in the forearm and all tendons are routed through the wrist mechanism (a 2 DOF differential joint). The thumb, index, and middle finger are driven by a looped tendon in the proximal joint. Motion of the fingers is driven by tendons routed via idle pulleys on the shafts of the connecting joints. The flexing of the fingers is directly controlled by the tendons while the extension is based on a spring return mechanism. This arrangement saves one cable per finger. The last two fingers are coupled together and pulled by a single motor which flexes 6 joints simultaneously. Two more motors, mounted directly inside the hand, are used for adduction/abduction movements of the thumb and all fingers except the middle one which is fixed with respect to the palm. In summary, eight DOF out of a total of nine are allocated to the first three fingers, allowing considerable dexterity. The last two fingers provide additional support to grasping.

Joint angles are sensed using a custom designed Hall-effect-magnet pair. In addition room for the electronics and tactile sensors has been planned. The tactile sensors are under development [9].

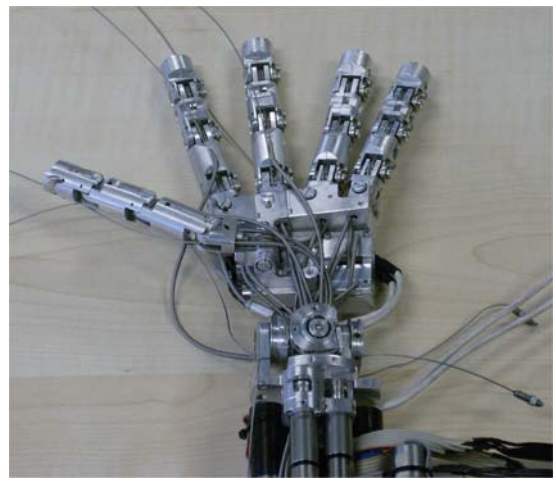


Figure 3: the hand of the iCub, showing the routing of the tendons through the wrist and some of the DOF before full assembly is completed (the palm is missing). Tendons are made of Teflon-coated cables sliding inside Teflon coated flexible steel tubes.

The overall size of the palm has been restricted to 50mm in length; it is 34mm wide at the wrist and 60mm at the fingers. The hand is only 25mm thick.

2.2 Electronics

The generation of motor control signals and sensory data acquisition is fully embedded into the iCub electronics. Further control layers are implemented externally. The interface between the iCub and the outside world occurs through a Gbit Ethernet cable. The robot contains motor amplifiers, a set of DSP controllers, a PC104-based CPU, and analog to digital conversion cards.

The low-level controller cards are of two types for the brushless and the brushed-DC motors respectively. They are based on the same DSP (Freescale 56F807). The controller of the brushless motors is made of two parts (logic and power) and can deliver a current of 6A continuous (20A peak) at 48V. All supply voltages are generated internally. The CAN bus is employed to communicate with the PC104 CPU. Logic and power are

58x42mm each and can control up to two motors. The power stage mounts also a metal heatsink that is then connected to the external shell of the robot for dissipation.

Similarly the controller of the brushed-DC motors is made of two parts. One card acts as power supply; the other contains the CPU and amplifiers to control up to four motors. In this case the maximum continuous current is limited to 1A at 12V.

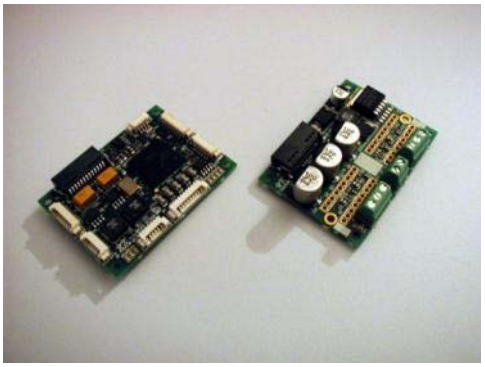


Figure 4: the brushless motor control logic and power amplifier of the iCub. Transistors and heatsinks are not shown. The size of the two PCBs is 58x42mm.

More development is in progress to interface tactile and force/torque sensors as discussed in [8].

2.3 Sensors

Given the size of the iCub, sensors were evaluated for performance but also weight and interface standards. The following table contains the list of available sensors and their status of maturity (i.e. integration into the robot hardware):

Component	Model/type	Notes
Cameras	PointGrey Dragonfly 2 640x480 30fps	Firewire cameras, support also higher resolution
Microphones	MICRO POM-2746L	Condenser electrect type
Inertial sensors	XSense MTx	3 gyroscopes, 3 linear accelerometers, compass
Force/torque sensors	Custom	Mechanically compatible with the ATI Mini-45
Position sensors	AS5045	12bit, absolute magnetic encoder
Position sensors	Faulhaber	Integrated position sensing for DC motors
Position sensors	Honeywell SS495A	Finger position sensing
Tactile sensors	Custom	Based on the AD7147, capacitive sensing

All sensors are fully integrated apart from the force/torque sensor whose control electronics is still under development and the skin whose entire technology is under testing. More information can be found in [8, 9].

3. SOFTWARE

The iCub software was developed on top of Yarp [7]. RobotCub supported a major overhaul of the Yarp libraries to adapt to a more demanding collaborative environment. Better engineered software and interface definitions are now available in Yarp.

Yarp is a set of libraries that support modularity by abstracting two common difficulties in robotics: namely, modularity in algorithms and in interfacing with the hardware. Robotics is perhaps one of the most demanding application environments for software recycling where hardware changes often, different specialized OSs are typically encountered in a context with a strong demand for efficiency. The Yarp libraries assume that an appropriate real-time layer is in charge of the low-level control of the robot and instead takes care of defining a soft real-time communication layer and hardware interface that is suited for cluster computation.

Yarp takes care also of providing independence from the operating system and the development environment. The main tools in this respect are ACE [10] and CMake[3]. The former is an OS-independent communication library that hides the quirks of interprocess communication across different OSs. CMake is a cross-platform make-like description language and tool to generate appropriate platform specific project files.

Yarp abstractions are defined in terms of protocols. The main Yarp protocol addresses inter-process communication issues. The abstraction is implemented by the *port* C++ class. *Ports* follow the observer pattern by decoupling producers and consumers. They can deliver messages of any size, across a network using a number of underlying protocols (including shared memory when possible). In doing so, *ports* decouple as much as possible (as function of a certain number of user-defined parameters) the behavior of the two sides of the communication channels. *Ports* can be commanded at run time to connect and disconnect.

The second abstraction of Yarp is about hardware devices. The Yarp approach is to define interfaces for classes of devices to wrap native code APIs (often provided by the hardware manufactures). Change in hardware will likely require only a change in the API calls (and linking against the appropriate library). This easily encapsulates hardware dependencies but leaves dependencies in the source code. The latter can be removed by providing a “factory” for creating objects at run time (on demand).

The combination of the *port* and *device* abstractions leads to remotable device drivers which can be accesses across a network: e.g. a grabber can send images to a multitude of listeners for parallel processing.

Overall, Yarp’s philosophy is to be lightweight and to be “gentle” with existing approaches and libraries. This naturally excludes hard real-time issues that have to be necessarily addressed elsewhere, likely at the OS level.

[3] http://www.cmake.org

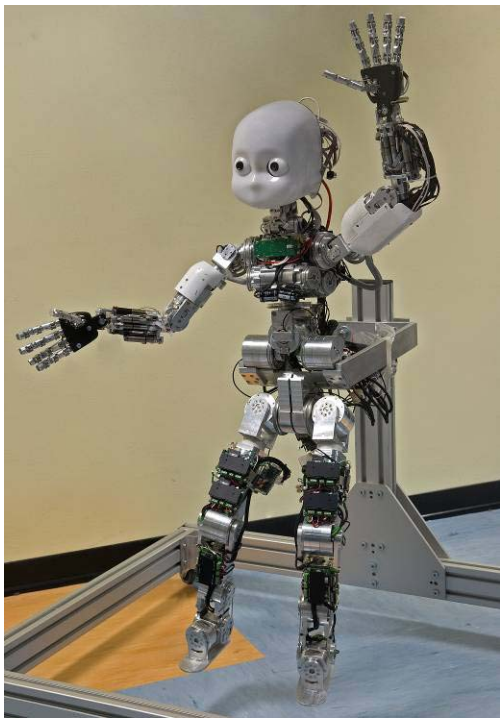


Figure 5: the complete iCub prototype.

3.1 Yarp example

For the purposes of YARP, communication takes place through connections between named entities called *ports*. These form a directed graph, the YARP Network, where ports are the nodes, and connections are the edges. Each port is assigned a unique name, such as "/icub/camera/right". Every port is registered by name with a "name server". The goal is to ensure that if you know the name of a port, that is all you need in order to be able to communicate with it from any machine. The YARP name server converts from symbolic names to all the details necessary to make a connection with a specific resource. The YARP name server is designed to be easily used by clients who are not themselves using the YARP libraries or executables.

The purpose of ports is to move data from one thread to another (or several others) across process and machine boundaries. The flow of data can be manipulated and monitored externally (e.g. from the command-line) at run-time. It can also be accessed without using the YARP libraries or executables, since the relevant protocols are documented.

A port can send data to any number of other ports. A port can receive data from any number of other ports. Connections between ports can be freely added or removed, and may use different underlying transports. The use of several different transports and protocols allows us to exploit their best characteristics. TCP is reliable; it can be used to guarantee the reception of a message. UDP can be faster than TCP, but without guarantees. Multicast is efficient for distributing the same information to large numbers of targets. Shared memory can be employed for local connections.

Figure 6 shows a very simple network of ports for a visual tracking application. Machine 1, in this example, grabs images which are sent to another application (the tracker proper). The output of the tracker consists of two parts: the image coordinates of the tracked object and an image with a graphic overlay showing how good the tracker is doing. The output is sent to a control process on another machine (Machine 2) and for visualization to yet another machine. Different protocols can be used for reasons of efficiency.

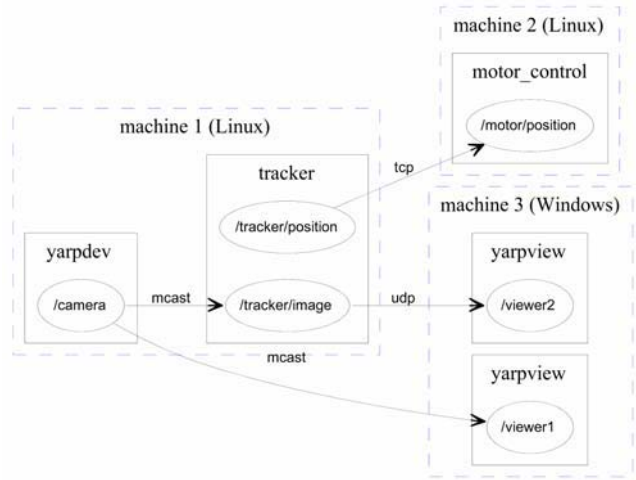


Figure 6: example of a YARP network for a simple visual control loop.

4. OPEN SOURCE ROBOTICS

RobotCub is Open Source both for software and hardware. While the phrase "Open Source software – OSS" is clear, "Open Source hardware" might sound strange, but in fact it is a plain transfer of the open source philosophy to the entire design of the RobotCub platform. The design of the robot started from the preparation of specifications (e.g. estimation of torque, speed, etc.), a typical 3D CAD modeling, and eventually in the preparation of the executive files which are used to fabricate parts and for assembly. Without good documentation it is very complicated to build and assemble a full robot. This means that documentation (as for software) is particularly important.

The CAD files, in some sense, can be seen as the source code, since they are the "preferred form of the work for making modifications to it", in the language of the GPL. They get "compiled" into 2D drawings which represent the executive drawings that can be used by any professional and reasonably well-equipped machine shop either to program CNC machines or to manually prepare the mechanical parts. This compilation process is not fully automated and requires substantial human intervention. There is a clear dependency of the 2D drawings on the original 3D CAD model. To enable the same type of virtuous development cycle as occurs in open source software, the 3D CAD is required, since changes happen in 3D first and get propagated to 2D later. In addition, assembly diagrams, part lists, and all the material produced during the design stage should be included to guarantee that the same information is available to new developers.

One difference between software and the hardware design is that there are currently no effective formats for interchange of 3D models. Proprietary systems such as SolidWorks and Pro/E can import and export a range of formats, but going from one to another is lossy, destroying the information needed for production

and leaving just the basic geometrical shape. So in practice, designs are tied to tools produced by a particular vendor, and interoperability between hardware design tools is limited. In RobotCub we were forced to choose a specific set of tools for mechanical and electronic CAD and future upgrades will have to strictly adhere to these standards. Due to the absence of open source professional design tools, RobotCub uses proprietary products. This is an unfortunate situation, but there is no practical alternative at the moment. The “C++” and “gcc” of CAD do not exist yet.

As a practical matter, the simple duplication of RobotCub parts does not require the use of any of these tools since we provide all executive drawings and production files (e.g. Gerber files for the PCBs). For modification, the design tools are somewhat expensive (although educational discounts or educational releases exist). Free of charge viewers are currently available for all file types in question.

For RobotCub, we decided to license all the CAD sources under the GPL which seems appropriate given their nature. Associated documentation will be licensed under the FDL. These will be made available through the usual source code distribution channels (e.g. repositories, websites).

5. CONCLUSION

The design process of RobotCub has been a distributed effort as for many open source projects. Various groups developed various subcomponents and contributed in different ways to the design of the robot including mechanics, electronics, sensors, etc. In particular, a whole design cycle was carried out for the subparts (e.g. head, hand, legs) and the prototypes that have been built and debugged. The final CAD and 2D drawings were discussed and then moved to the integration stage. Clearly, communication was crucial at the initial design stage to guarantee a uniform design and a global optimization.

The distributed design broke down at the integration stage where the industrial partner (Telerobot Srl. – Genoa) stepped in to carry out integration, verification and consistency checks. The design and fabrication of the control electronics was also subcontracted to a specialized company. It is important to stress the collaboration with industry for a project of this size and with these goals and requirements. For many reasons building a complete platform involves techniques and management that is better executed following industrial standards. One example that applies to RobotCub is the standardization of the documentation.

A further strategy used in RobotCub is that of building early. Each subsystem was built and copied as soon as possible. In several cases debugging happened either because the copies of the robot did not work as expected or because easy-to-fix problems were spotted. Sometimes the documentation had to be improved. Unfortunately, this strategy was applied less extensively to some of the subparts which are or were still under design and debugging. The design stage will be completed with the realization of the fifteen copies of the iCub.

This will further test the documentation and in general the reliability of the overall platform including software, debugging tools, electronics, etc. The first release of the iCub will be consolidated after this final fabrication stage.

The actual design of the robot had to incorporate manipulation by providing sophisticated hands, a flexible oculomotor system, and a reasonable bimanual workspace. On top of this, the robot has to support global body movements such as crawling, sitting, etc. These many constraints were considered in preparing the specifications of the robot and later on during the whole design process.

Both the iCub design and its software architecture are distributed as Open Source. This is not enough to guarantee success. Additional initiatives are required. RobotCub is giving away six copies of the iCub to the winners of an Open Call for proposals to use the iCub (recently concluded). In addition a structure called the Research and Training Site (RTS) has being created to support visiting researchers to work on the iCub prototypes in Genoa.

6. ACKNOWLEDGMENTS

The authors would like to thank the RobotCub Consortium. The authors were supported by European Union grant RobotCub (IST-2004-004370) and by euCognition (FP6 Project 26408). Paul Fitzpatrick is gratefully acknowledged for the continuous support to Yarp.

7. REFERENCES

[1] L. Fadiga, L. Craighero, and E. Olivier, "Human motor cortex excitability during the perception of others' action," *Current Biology*, vol. 14 pp. 331-333, 2005.

[2] L. Fadiga, L. Craighero, G. Buccino, and G. Rizzolatti, "Speech listening specifically modulates the excitability of tongue muscles: a TMS study," *European Journal of Neuroscience*, vol. 15, pp. 399-402, 2002.

[3] G. Rizzolatti and L. Fadiga, "Grasping objects and grasping action meanings: the dual role of monkey rostroventral premotor cortex (area F5)," in *Sensory Guidance of Movement, Novartis Foundation Symposium*, G. R. Bock and J. A. Goode, Eds. Chichester: John Wiley and Sons, 1998, pp. 81-103.

[4] D. Vernon, G. Metta, and G. Sandini, "A Survey of Cognition and Cognitive Architectures: Implications for the Autonomous Development of Mental Capabilities in Computational Systems," *IEEE Transactions on Evolutionary Computation, special issue on AMD*, vol. 11, 2007.

[5] C. von Hofsten, "On the development of perception and action," in *Handbook of Developmental Psychology*, J. Valsiner and K. J. Connolly, Eds. London: Sage, 2003, pp. 114-140.

[6] G. Sandini, G. Metta, and D. Vernon, "RobotCub: An Open Framework for Research in Embodied Cognition," presented at IEEE-RAS/RJS International Conference on Humanoid Robotics, Santa Monica, CA, 2004.

[7] P. Fitzpatrick, G. Metta, and L. Natale, "Towards Long-Lived Robot Genes," *Journal of Robotics and Autonomous Systems, Special Issue on Humanoid Technologies*, vol. 56, 2008.

[8] N. G. Tsagarakis, G. Metta, G. Sandini, D. Vernon, R. Beira, F. Becchi, L. Righetti, J. Santos-Victor, A. J. Ijspeert, M. C. Carrozza, and D. G. Caldwell, "iCub – The Design and Realization of an Open Humanoid

Platform for Cognitive and Neuroscience Research," *Advanced Robotics*, vol. 21, 2007.

[9] M. Maggiali, G. Cannata, P. Maiolino, G. Metta, M. Randazzo, and G. Sandini, "Embedded Distributed Capacitive Tactile Sensor," presented at The 11th Mechatronics Forum Biennial International Conference 2008, University of Limerick, Ireland, 2008.

[10] S. D. Huston, J. C. E. Johnson, and U. Syyid, *The ACE Programmer's Guide*: Addison-Wesley, 2003.

An Open-Source Simulator for Cognitive Robotics Research: The Prototype of the iCub Humanoid Robot Simulator

V. Tikhanoff, A. Cangelosi
University of Plymouth
Plymouth PL4 8AA
UK

vadim.tikhanoff@plymouth.ac.uk; acangelosi@plymouth.ac.uk

P. Fitzpatrick
Lira-lab University of Genova
Viale F. Causa, 13
Genova Italy

paulfitz@liralab.it

G. Metta, L. Natale, F. Nori
Italian Institute of Technology
Via Morego 30
Genova Italy

giorgio.metta@iit.it
lorenzo.natale@iit.it
francesco.nori@iit.it

ABSTRACT
This paper presents the prototype of a new computer simulator for the humanoid robot iCub. The iCub is a new open-source humanoid robot developed as a result of the "RobotCub" project, a collaborative European project aiming at developing a new open-source cognitive robotics platform. The iCub simulator has been developed as part of a joint effort with the European project "ITALK" on the integration and transfer of action and language knowledge in cognitive robots. This is available open-source to all researchers interested in cognitive robotics experiments with the iCub humanoid platform.

Keywords
Open-Source, Simulator, iCub humanoid robot, cognitive robotics.

1. INTRODUCTION
Computer simulations play an important role in robotics research. Despite the fact that the use of a simulation might not provide a full model of the complexity present in the real environment and might not assure a fully reliable transferability of the controller from the simulation environment to the real one, robotic simulations are of great interest for cognitive scientists [18]. There are several advantages of robotics simulations for researchers in cognitive sciences. The first is that simulating robots with realistic physical interactions permit to study the behavior of several types of embodied agents without facing the problem of building in advance, and maintaining, a complex hardware device. The computer simulator can be used as a tool for testing algorithms in order to quickly check for any major problems prior to use of the physical robot. Moreover, simulators also allow researchers to experiment with robots with varying morphological characteristics without the need to necessarily develop the corresponding features in hardware [1]. This advantage, in turn, permits the discovery of properties of the behavior of an agent that emerges from the interaction between the robot's controller, its body and the environment. Another advantage is that robotic simulations make it possible to apply particular algorithms for creating robots' controllers, such as evolutionary or reinforcement learning algorithms [12]. The use of robotics simulation permits to drastically reduce the time of the experiments such as in evolutionary robotics. In addition, it makes it possible to explore research topics like the co-evolution of the morphology and the control system [1]. A simulator for the iCub robot magnifies the value a research group can extract from the physical robot, by making it more practical to share a single robot between several researchers. The fact that the simulator is free and open makes it a simple way for people interested in the robot to begin learning about its capabilities and design, with an easy "upgrade" path to the actual robot due to the protocol-level compatibility of the simulator and the physical robot. And for those without the means to purchase or build a humanoid robot, such small laboratories or hobbyists, the simulator at least opens a door to participation in this area of research.

The iCub simulator is currently being used by both the RobotCub and the ITALK project partners for preliminary experiments on the simulator robot, and subsequent testing with the physical robots.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

2. ICUB SIMULATOR DEVELOPMENT
The iCub simulator has been designed to reproduce, as accurately as possible, the physics and the dynamics of the robot and its environment. The simulated iCub robot is composed of multiple rigid bodies connected via joint structures. It has been constructed collecting data directly from the robot design specifications in order to achieve an exact replication (e.g. height, mass, Degrees of Freedom) of the first iCub prototype developed at the Italian Institute of Technology in Genoa. The environment parameters on gravity, objects mass, friction and joints are based on known environment conditions.

2.1 Open-Source Approach
The iCub simulator presented here has been created using open source libraries in order to make it possible to distribute the

simulator freely to any researcher without requesting the purchase of restricted or expensive proprietary licenses.

The very first iCub simulator prototype was developed using the commercial Webots package [10,17], a professional robotic simulator which is widely used in academia and research. The Webots package is primarily designed for industrial simulations but used as a reliable tool for robotic research. Although a powerful software, the main disadvantages of the Webots package are its price, the computational heaviness of the package itself and the fact that, depending on the type of license, there are limitations on the source code available in order to modify some properties of the actual simulator. Therefore the potential open source distribution of such a first prototype was quite limited.

Other open source simulators suitable for robotics research also exist. Amongst others we can find the Player/Gazebo project [6, 7], Simbad [16], Darwin2K [8], EvoRobot [4,12] and the OpenSim [14]. The Simbad, Darwin2K and Evorobot simulators have a strong focus on evolutionary algorithms and they have been mainly developed for scientific educational purposes. They are built to study AI algorithms and machine learning for multi-robot platforms. Gazebo is a powerful and complex multi-robot simulation in a 3D environment. OpenSim is a general multi-robot platform built in a similar way as the Gazebo package. However such systems use the same third party software and libraries.

Although the proposed iCub simulator is not the only open source robotics platform, it is one of the few that attempts to create a 3D dynamic robot environment capable of recreating complex worlds and fully based on non-proprietary open source libraries.

2.2 Physics Engine

The iCub simulator uses ODE [13] (Open Dynamic Engine) for simulating rigid bodies and the collision detection algorithms to compute the physical interaction with objects. The same physics library was used for the Gazebo project and the Webots commercial package. ODE is a widely used physics engine in the open source community, whether for research, authoring tools, gaming etc. It consists of a high performance library for simulating rigid body dynamics using a simple C/C++ API. ODE was selected as the preferred open source library for the iCub simulator because of the availability of many advanced joint types, rigid bodies (with many parameters such as mass, friction, sensors...), terrains and meshes for complex object creation.

2.3 Rendering Engine

Although ODE is a good and reliable physics engine, computing all the physical interaction of a complex system can take a good deal of processing power. Since ODE uses a simple rendering engine based on OpenGL, it has limitations for the rendering of complex environments comprising many objects and bodies. This can significantly affect the simulation speed of complex robotic simulation experiments. It was therefore decided to use OpenGL directly combined with SDL [15], an open source cross platform multimedia library. This makes it possible to render the scene with much more ease and to carry out computationally-efficient simulation experiments.

2.4 YARP Protocol

As the aim was to create an exact replica of the physical iCub robot, the same software infrastructure and inter-process communication will have to be used as those used to control the physical robot. iCub uses YARP [5, 9] (Yet Another Robot Platform) as its software architecture. YARP is an open-source software tool for applications that are real-time, computation-intensive, and involve interfacing with diverse and changing hardware. The simulator and the actual robot have the same interface either when viewed via the device API or across network and are interchangeable from a user perspective. The simulator, like the real robot, can be controlled directly via sockets and a simple text-mode protocol; use of the YARP library is not a requirement. This can provide a starting point for integrating the simulator with existing controllers in esoteric languages or complicated environments.

2.5 Architecture

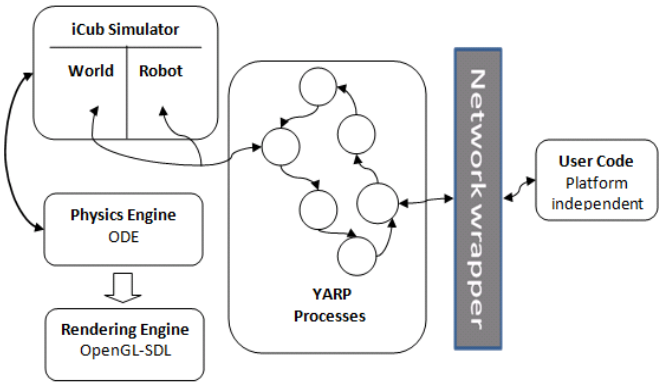
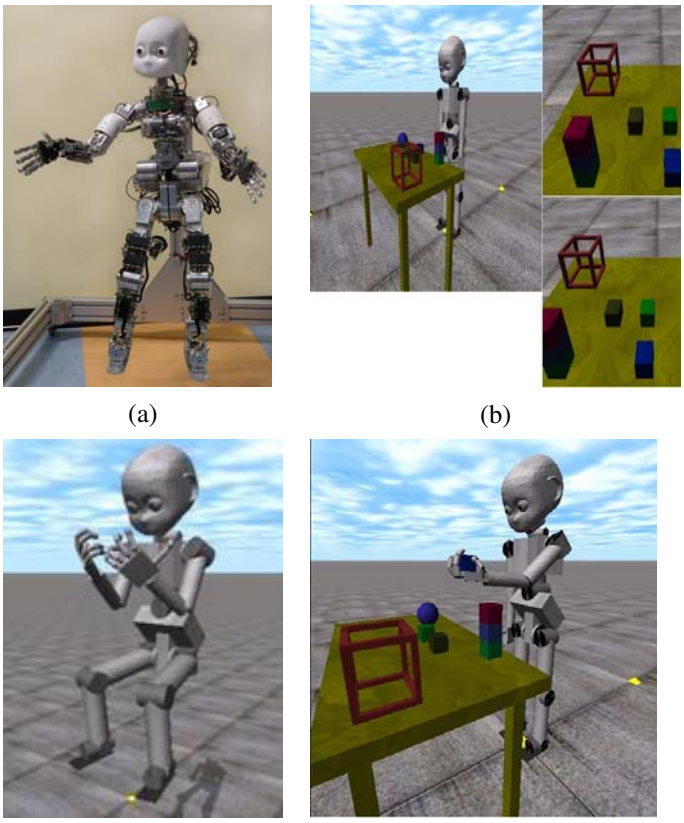


Figure 1. This figure shows the architecture of the simulator with YARP support. The User code can send and receive information to both the simulated robot itself (motors/sensors/cameras) and the world (manipulate the world). Network wrappers allow device remotization. The Network Wrapper exports the YARP interface so that it can be accessed remotely by another machine.

2.6 iCub Body Model

The iCub simulator has been created using the data from the physical robot in order to have an exact replica of it. As for the physical iCub, the total height is around 105cm, weighs approximately 20.3kg and has a total of 53 degrees of freedom (DoF). These include 12 controlled DoFs for the legs, 3 controlled DoFs for the torso, 32 for the arms and six for the head.

The robot body model consists of multiple rigid bodies attached through a number of different joints. All the sensors were implemented in the simulation on the actual body, such as touch sensors and force/torque sensors. As many factors impact on the torque values during manipulations, the simulator might not guarantee to be perfectly correct. However the simulated robot torque parameters and their verification in static or motion are a good basis and can be proven to be reliable [11].



(a) (b)

(c) (d)

Figure 2. Photo of real iCub (a), of simulated iCub and the binocular view (b) The simulated iCub moving all four limbs as part of a demo (c) and the simulated iCub looking at and manipulating an object in its environment.(d)

All the commands sent to and from the robot are based on YARP instructions. For the vision we use cameras located at the eyes of the robot which in turn can be sent to any workstation using YARP in order to do develop vision analysis algorithms.

The system has full interaction with the world/environment. The objects within this world can be dynamically created, modified and queried by simple instruction resembling those that YARP uses in order to control the robot.

3. ICUB SIMULATOR AND COGNITIVE ROBOTICS RESEARCH PROJECTS

This section presents a research initiative on embodied cognition and developmental cognitive robotics based on the two EU projects RobotCub (robotcub.org) and ITALK (italkproject.org). Various robotics experiments in these two projects will rely on the use of the iCub simulator.

The RobotCub project aims at the development of an open humanoid robotic platform (iCub) and simultaneously the advancement of our understanding of cognitive systems by exploiting this platform in the study of the development of cognitive capabilities in humanoid robots. The idea is that, by creating a common platform, this will enable many laboratories to join this effort without having to invest themselves in developing yet another robotic platform. The second aim of RobotCub is to investigate the development of these cognitive skills in natural and artificial cognitive systems. The project will carry out a plan of empirical research including neuroscience, developmental psychology, and robotics. This plan is centered on manipulation behavior, ranging from the direct aspects of reaching and grasping for objects to the use of gestures for communication. Aspects that will be touched along the way are—for instance—looking and overt attention, reaching, the detection and discovery of affordances, learning through imitation, and interaction. The emergent approach naturally encompasses the study of ontogenic development and, in fact, a comparatively large effort will be devoted to its study. The RobotCub project roadmap for this investigation includes the study of the starting point in terms of core abilities, the motivation of the system to explore and gather data, and new studies on a few research areas such as looking, reaching and manipulation, posture, locomotion, and social interaction. For each of these areas, issues of prospective use of information, motivation, and the mechanisms of exploration have to be experimentally investigated. The RobotCub agenda aims at covering—through targeted empirical investigation—most if not all of these issues.

The ITALK project intends to develop cognitive robotic agents, based among others on the iCub humanoid platform, that learn to handle and manipulate objects and tools autonomously, to cooperate and communicate with other robots and humans, and to adapt their abilities to changing internal, environmental, and social conditions. The main theoretical hypothesis behind the project is that the parallel development of action, conceptualisation and social interaction permits the bootstrapping of language capabilities, which on their part enhance cognitive development. This is possible through the integration and transfer of knowledge and cognitive processes involved in sensorimotor learning and the construction of action categories, imitation and other forms of social learning, the acquisition of grounded conceptual representations and the development of the grammatical structure of language.

The undergoing research falls into five main research themes: (i) action development, (ii) conceptualisation, (iii) social interaction, (iv) language emergence, and (v) integration and bootstrapping of cognition.

The study of the development of complex action manipulation capabilities will –in contrast to existing approaches– be based on synchronous development of motor, social and linguistic skills. For this it is fundamental to identify the characteristics of action development that are compatible with this scenario and reject those that are mere engineering solutions. Two core properties of biological motor control systems will be considered: compositionality, the construction of hierarchically ordered gesturing and manipulation, and generalization. We will study how action development can be guided by individual exploration by the robot and by imitating humans.

A fundamental skill of any cognitive system is the ability to produce a variety of behaviours and to display the behaviour that is appropriate to the current individual, social, cultural and environmental circumstances. This will require agents (1) to reason about past, present and future events, (2) to mediate their motor actions based on this reasoning process and (3) to communicate using a communication system that shares

properties with natural language. For this agents will need to develop and maintain internal categorical states, i.e. ways to store and classify sensory information. We term such internal states *embodied concepts* and we understand them as representations grounded in sensory-motor experiences that identify crucial aspects of the environment and/or of the agent/environmental interaction.

Another essential component of the ITALK research project is to look at the role of social learning and social interaction to support the development of a shared linguistic communication system. In particular, new research will consider (i) the role of imitation and human-robot interaction for the acquisition of shared communication systems based on deixis, gestures and reference, (ii) the role of users' expectation in human-robot interaction and (iii) the emulation of actions and gestures in the learning of multimodal task-oriented behavior. Such research will be based on a series of human-robot interaction (HRI) experiments and on observational studies on parent-child dyads which will inform robot-robot and human-robot experiments. We expect to extend the expertise and methodologies in dialog systems for HRI studies to new studies on social interaction and communication where the robot's linguistic communication system develops through interaction with its environment and other robots and humans.

The ITALK project will follow a cognitive linguistics approach. As it is centred on the interaction between action and language development, it provides the ideal testbed to investigate the emergence of linguistic constructions in close interaction with the development of action, social and grounded conceptual capabilities. We will focus on the emergence of linguistic structure. Among the research issues include (i) generalisation as the basis of the emergence of symbolic systems, (ii) the role of speech and "acoustic packaging": speech or sound signals which serve as a cue to aid the learning of action sequences, (iii) the role of constructional grounding: the acquisition of linguistic construction and how one construction become favoured over another, (iv) the ontogenetic emergence of compositional lexicons, and (v) evolutionary studies on language emergence.

In the coming years the ITALK project aims to achieve a series of scientific and technological objectives such as providing new scientific explanations of the integration of action, social and linguistic skills and in particular on the hypothesis that action, social and linguistic knowledge co-develop and further bootstrap cognitive development. Another main aim deals with developing sets of methods for analyzing the interaction of language, action and cognition in humans and artificial cognitive agents using robot learning experiments, computer simulations, cognitive linguistic analysis, and experimental investigations from developmental linguistics, the neuroscience of language and action, and human-robot interaction experiments. Furthermore the project will develop innovative and cognitively plausible engineering principles, techniques and approaches for the design of communicative and linguistic capabilities in cognitive robots able to interact with their physical and social world and to manipulate entities, artefacts and other agents including humans.

All of the above aims would be demonstrated through the use of robotic experiments on the acquisition of object manipulation, social skills and linguistic capabilities in simulated and physical cognitive robots. In particular, robotic agents will be able to (a) acquire complex object manipulation capabilities through social interaction; (b) develop an ability to create and use embodied concepts; (c) develop social skills that allow flexible interaction with other agents or people; (d) develop linguistic abilities to communicate about their interaction with the world.

4. CONCLUSION

The current version of the iCub simulator has been used for preliminary testing by partners in the RobotCub and ITALK projects. In addition to being used for experiments on the development of controllers for the iCub robot, some groups have used the simulator to create a mental model [2] used by the robot to represent the current state of the environment.

Future plans on the simulator development will mostly involve the design of functionalities to model and interact with the physical environment. For example, this will allow the users to modify the objects in the world where the iCub resides, in order to allow different types of experiments. Finally, further work will focus on the systematic testing and replication of simulation studies with the physical robot.

5. ACKNOWLEDGEMENTS

This work was supported by the European Commission FP6 Project RobotCub (IST-004370) and FP7 Project ITALK (ICT-214668) within the Cognitive Systems and Robotics unit (FP7 ICT Challenge 2). The authors also acknowledge the support of the FP6 euCognition Coordinated Action project funded under the same Cognitive Systems and Robotics unit.

6. REFERENCES

[1] J.C. Bongard & R. Pfeifer (2003). Evolving complete agents using artificial ontogeny. In Hara, F. & R. Pfeifer, Eds., Morpho-functional Machines: The New Species (Designing Embodied Intelligence) Springer-Verlag, pp. 237-258

[2] PF Dominey (2007) Sharing Intentional Plans for Imitation and Cooperation: Integrating Clues from Child Developments and Neurophysiology into Robotics, Proceedings of the AISB 2007 Workshop on Imitation.

[3] European commission, unit e5, home page. http://cordis.europa.eu/ist/cognition/index.html, 2007.

[4] Evorobot http://laral.istc.cnr.it/evorobot/simulator.html

[5] P. Fitzpatrick, G. Metta, L. Natale: Towards Long-lived Robot Genes, Robotics and Autonomous Systems, 56(1):29-45, 2008

[6] B. Gerkey, R. T. Vaughan and A. Howard. The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. Proceedings of the 11th International Conference on Advanced Robotics, 2003.

[7] N. Koenig and A. Howard. Design and Use Paradigms for Gazebo, An Open-Source Multi-Robot Simulator. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan, 2004.

[8] C. Leger, Darwin2K: An Evolutionary Approach to Automated Design for Robotics. Kluwer Academic Publishers, Aug 2000.

[9] G. Metta, P. Fitzpatrick & L. Natale. YARP: Yet Another Robot Platform. International Journal on Advanced Robotics Systems, 3(1):43–48, 2006.

[10] O. Michel, “Webots: Professional mobile robot simulation,” International Journal of Advanced Robotic Systems, vol. 1, no. 1, pp.39–42, 2004.

[11] N. Nava,V. Tikhanoff, G. Metta, G Sandini, Kinematic and Dynamic Simulations for The Design of RoboCub Upper-Body Structure ESDA 2008

[12] S. Nolfi & D. Floreano (2000). Evolutionary Robotics: The Biology, Intelligence and Technology of Self-Organizing Machines Cambridge, MA: MIT Press/Bradford Books

[13] Open Dynamics Engine http://opende.sourceforge.net/.

[14 OpenSim Simulator http://opensimulator.sourceforge.net/.

[15 SDL – Simple DirectMedia Layer http://www.libsdl.org/

[16] Simbad Simulator - http://simbad.sourceforge.net

[17] Webots - http://www.cyberbotics.com/

[18] Ziemke T. (2003). On the role of robot simulations in embodied cognitive science, *AISB Journal*, 1(4), 389-99

Symbiotic Robot Organisms: REPLICATOR and SYMBRION Projects

Special Session on EU-projects

Serge Kernbach*
Eugen Meister
Florian Schlachter
Kristof Jebens
University of Stuttgart
Universitätsstr. 38,
D-70569 Stuttgart,
Germany

Marc Szymanski
Jens Liedke
Davide Laneri
Lutz Winkler
University of Karlsruhe
Engler-Bunte-Ring 8
D-76131 Karlsruhe,
Germany

Thomas Schmickl
Ronald Thenius
Karl-Franzens-University
Graz
Universitatsplatz 2
A-8010 Graz, Austria

Paolo Corradi
Leonardo Ricotti
Scuola Superiore
Sant'Anna
Viale Rinaldo Piaggio
34
56025 - Pontedera
(PI), Italy

ABSTRACT

Cooperation and competition among stand - alone swarm agents can increase the collective fitness of the whole system. An interesting form of collective system is demonstrated by some bacteria and fungi, which can build symbiotic organisms. Symbiotic communities can enable new functional capabilities which allow all members to survive better in their environment. In this article we show an overview of two large European projects dealing with new collective robotic systems which utilize principles derived from natural symbiosis. The paper provides also an overview of typical hardware, software and methodological challenges arose along these projects, as well as some prototypes and on-going experiments available on this stage.

Keywords: *collective robotics, swarms, artificial evolution, reconfigurable systems*

1. INTRODUCTION

Nature shows several interesting examples for cooperation of individuals. Most prominent examples of cooperation are found in social insects [1], where specialized reproductive schemes (in most cases just a few out of thousands of colony members are able to reproduce) and the close relationships of colony members favoured the emergence of highly cooperative behaviours [2]. However, also non-eusocial forms of cooperative communities evolved, like the collective hunting in predatory mamals [3] (e.g., lions, whales, ...) or the trophallactic altruism in vampire bats. Such cooperative behaviours are mostly explained by reciprocal advantages due to the cooperative behaviours and/or by the close relationship among the community members. In contrast to that, cooperation sometimes arises also among individuals that are not just very distant in a gene pool, sometimes they do not even share the same gene pool: Cooperative behaviours between members of different species is called 'Symbiosis'. A non-exhaustive list of prominent examples are the pollination of plants by flying insects (or birds), the cooperation between ants and aphids. Also lichens, which are a close integration of fungi and algae and the cooperation between plant roods and fungi represent symbiotic interactions.

A common pattern in all these above-mentioned forms of cooperation is that single individuals perform behaviours, which - on the first sight - are more supportive for the collective of the group than for themselves. However, as these behaviours have emerged through natural selection, we can assume that these cooperative behaviours have their ultimate reasoning in a sometimes delayed and often non-obvious individual egoistic advantage.

Symbiotic forms of organization emerge new functional capabilities which allow aggregated organisms to achieve better fitness in the environment. When the need of aggregation is over, symbiotic organism can dis-aggregate and exists further as stand-alone agents, thus an adaptive and dynamical form of cooperation is often advantageous.

Lately, technical systems mimic natural collective systems in improving functionality of artificial swarm agents. Collective, networked or swarm robotics are scientific domains, dealing with a cooperation in robotics [4]. Current research in these domains is mostly concentrated on cooperation and competition among stand-alone robots to increase their common fitness [5]. However, robots can build a principally new kind of collective systems, when to allow them to aggregate into a multi-robot organism-like-forms. This "robot organism" can perform such activities that cannot be achieved by other kind of robotic systems and so to achieve better functional fitness.

To demonstrate this idea, we consider a collective energy foraging scenario for micro-robots Jasmine [6]. Swarm robots can autonomously find an energy source and recharge.

*Corresp. author: Serge.Kernbach@ipvs.uni-stuttgart.de

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS 08, August 19-21, 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

The clever collective strategy can essentially improve the efficiency of energy foraging, but nevertheless a functional fitness of a swarm is limited. For instance, if the recharging station is separated from a working area by a small barrier, robots can never reach the energy source. However, if robots aggregate into more complex high-level organism which can pass the barrier, they will reach the docking station. In this way a cooperative organization of robotic system allows an essential increase of functional capabilities for the whole group. The large integrated project "REPLICATOR" (www.replicatores.eu), funded by the European commission, within the work programme "Cognitive systems, interaction and robotics", deals with such issues as reconfigurability of sensors and actuators, adaptive control and learning strategies as well as working in real environments.

The cooperative (swarm-based or symbiotic) organization of the robotic system provides essential plasticity of used hardware and software platforms. The robot organism will be capable of continuously changing its own structure and functionality. Such an evolve-ability opens many questions about principles and aspects of long- and short-term artificial evolution and controllability of artificial evolutionary processes. The large integrated project "SYMBRION" (www.symbrion.eu), funded by European commission, within the work programme "Future and Emergent Technologies", is focused on evolve-ability, dependability and artificial evolution for such robot organisms based on bio-inspired and computational paradigms. Both projects are open-science and open-source.

Both projects, consortia and the European commission are closely cooperating to achieve the targeted goals. It is expected that results of both projects create new technology for making artificial robotic organisms self-configured, self-healing, self-optimizing and self-protecting from a hardware and software point of view. This leads not only to extremely adaptive, evolve-able and scalable robotic systems, but also enables the robot organisms to reprogram themselves without human supervision, to develop their own cognitive structures and, finally, to allow new functionalities to emerge.

The rest of this paper is organized in the following way: In Section 2 we discuss a new paradigm of symbiotic systems. Section 3 gives an example of the energy foraging scenario. Sections 4 and 5 briefly mention the hardware and software challenges, where as Section 6 introduces several ideas towards evolve-ability of the robot organisms. Finally, in Section 7 we conclude this work.

2. NEW PARADIGM IN COLLECTIVE ROBOTIC SYSTEMS

Collective intelligence is often associated with macroscopic capabilities of coordination among robots, collective decision making, labor division and tasks allocation in the group [7]. The main idea behind this is that robots are achieving better performance when working collectively and so are capable of performing such activities which are not possible for individual robots. The background of collective intelligence is related to the capability of swarm agents to interact jointly in one medium. There are three different cases of such interactions:

1. In the first case agents communicate through a digital channel, capable for semantic messages exchange. Due to information exchange, agents build different types of common knowledge [8]. This common knowledge in fact underlie collective intelligence.

2. The second case appears when macroscopic capabilities are defined by environmental feedback. The system builds a closed macroscopic feedback-loop, which works in a collective way as a distributed control mechanisms. In this case there is no need of complex communication, agents interact only by kinetic means. This case if interaction is often denoted as a spatial reasoning, or spatial computing.

3. The third case of interactions we encounter in nature, when some bacteria and fungi (e.g. dictyostelium discoideum) can aggregate into a multi-cellular organism when this provides better chances of survival [9]. In this way, they interact not only through information exchange or spatial interactions, they build the closest physical, chemical and mechanical interconnections, through the agents still remain independent from each other.

The first two cases of interactions are objects of extensive research in many domains: robotics, multi-agents systems, bio-inspired and adaptive community and so on. However the practical research in the last case represents essential technological difficulties and therefore is not investigated enough. Despite the similarities between a robot swarm and multi-robot organism, such as a large number of robots, focus on collective/emergent behavior, a transition between them is a quite difficult step due to mechanical, electrical and, primarily, conceptual issues [10]. In the following sections we introduces corresponding challenges in more detail.

Now, we believe that research around the third case of interactions is concentrated on four important questions:

1. Reconfigurability, adaptability and learn-ability of the symbiotic systems. These issues include flexible and multi-functional sensors and actuators, distributed computation, scalability, modelling, control and other issues, which are closely related to the reconfigurable robotic research. The REPLICATOR project is focused on these points.

2. Evolve-ability of the symbiotic systems, which includes principles and aspects of long- and short-term artificial evolution and adaptivity as well as exploring and analogies to biological systems. The SYMBRION project is focused on these points.

3, 4 Embodiment of evolutionary systems for different environments and medias as well as investigation of information properties of such systems. These points are covered by other research initiatives and projects.

In this way, the next step in a further research within the collective robotic community can consist in investigation multi-robot organisms or, in other words, a transition from robot swarm to a multi-robot organisms. All further sections are devoted to demonstrate diverse aspects of such a transition.

3. EXAMPLE: ENERGY FORAGING SCENARIO

In this section we will demonstrate the advantages of symbiotic organization of autonomous robotic systems. We choose for this purpose an example of energy homeostasis, because it is applicable to both living and robotic organisms and so we can draw several analogies between them.

The distinctive property of any living organism is the energy homeostasis and, closely connected, foraging behavior and strategies [3]. The robots, equipped with on-board

recharging electronics, can also possess its own energy homeostasis. In this way, when swarm robots get "hungry", they can collectively look for energy resources and execute different strategies in a cooperative energy foraging [11]. In critical cases robots can even decide to perform individual foraging, competing with other robots for resources.

The need of energy is a perfect example of natural fitness. If robots that are performing individual strategies find enough energy, they can survive in the environment. In turn, this means that these strategies were sufficient enough to balance these robots energetic budgets. Simultaneously, other energetically die if their behavioral strategy was poor. Based on such energy foraging, many of evolutionary approaches for different robotic species can be developed, compared and tested.

However, if there are many robots foraging in the environment, several undesired effects can emerge: (1) the docking station can become a "bottleneck" resource that essentially decreases the swarm efficiency; (2) robots with a high-energy level can occupy the docking station and block low-energetic robots. These robots can energetically die (and so decrease the swarm efficiency); (3) many robots can create a "crowd" around a docking station and essentially hinder a docking approach. This can increase the total recharging time and makes worse the energetic balance of the whole swarm.

Robots, in pursuing their energetic homeostasis, have only two possible decisions to make: (1) to execute a current collective task or (2) to move for recharging. In balancing these two behaviours, a cooperative strategy may find the right timing and the right combination between these individual decisions of all robots. Lately, several strategies of energy foraging for a robot swarm up to 70 swarm agents are implemented, see Fig. 1. These cover different bio-inspired approaches [12], [13] and hand-coded strategies [14].

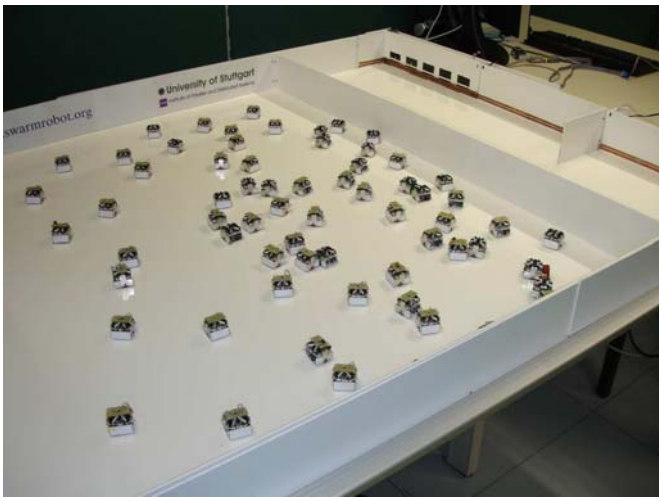
In one of these experiments, a few robots died close to the docking station and blocked the recharge area (we "simulated" this in the Fig. 1(b)). Robots that were in front of this barrier (away from the docking station) finally also died. This is the limit of functional fitness of swarm robots. There is no strategy, that allow swarm robots to overpass the barrier. Only when swarm robots would collectively emerge new functionality, like "pass the barrier", they would solve the "barrier problem".

Thus, an ideal solution for the "barrier problem" can be the aggregation of many single robots into one cooperative multi-robot organism. This way, they can reach the docking stations by "growing legs" and stepping over the barrier. In that case, the robots are helping each other in a cooperative manner, see Fig. 1(c).

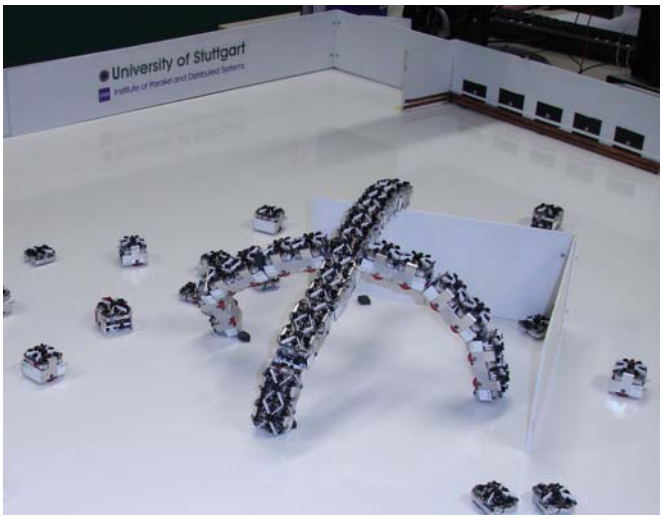
Obviously, such a robotic behaviour is extremely challenging from many viewpoints: Cooperative (symbiotic) robot systems have many similarities with known robotic research as e.g. mechanical self-assembling [15] or reconfigurable robotics [16]. However, the symbiotic form, show in Fig. 1, essentially differs from this robotic research, namely: (1) **Robots should be capable for autonomous aggregation and disaggregation;** (2) **Robots in the disaggregate state should possess individual locomotion**; (3) **There is no central control neither for disaggregated state (swarm) nor for the aggregate state (organism)**; (4) **Stand-alone robots should profit from the aggregation into organism**.

The swarm-based approaches, which is underlying the ag-

(a)



(b)



(c)

Figure 1: (a) Docking of a few robots for recharging. Shown is the two-line approach: the first line - recharging robots, the second line - robots waiting for recharging; (b) The "barrier problem" - robots are separated form docking stations by a barrier; (c) A possible solution to the "barrier problem": swarm robots form a symbiotic multi-robot organism and collective pass the barrier.

gregation processes, differs primarily from aggregated systems which are studied in the field of "reconfigurable robotics". In the following we consider on-going work with aggregated (symbiotic) robot organisms.

4. HARDWARE CHALLENGES

The main feature of a modular robot consists in being composed of several potentially independent modules, with limited complexity and capabilities, which are able to connect to each other in different configurations, in order to form a robot with greater capabilities. The global knowledge and the functionalities of the assembled robot generate by sharing of information and of resources between modules and by the fundamental capability of self-reconfiguration, in order to meet the demands of different tasks or different working environments. As a consequence, the overall functionalities and capabilities of a robotic modular organism are deeply related to the hardware structure and functions of its basic composing modules.

At the current stage of development of the projects (both projects started in 2008), the development of the hardware represents one of the hardest issues. In general, the concept of hardware design is as follows:

1. Independence for separate robots, this includes capabilities for communication, computation and sensing as a stand-alone robot, as well as individual locomotion and energy management.

2. Large computational power of the organism, required for performing on-line and on-board evolutionary approaches.

3. Heterogeneity of individual robots, which allows their later specialization within the organism.

4. Rich sensing and communication capabilities of the organism. The more robots are joined in the organism, the more functional diversity the organism can demonstrate.

5. Possible higher independency from human in term of energy, support and maintenance.

The consortia considered many state-of-the-art reconfigurable solutions, such as superBot [17], M-Tran [18], PolyBot [19], molecube [20], HYDRA/ATRON [21] and others, even visited some of these labs for exchange of experience. Currently, we follow three different developmental lines, which will be later fused into one or two first prototypes.

4.1 Mechanical Challenges

The mechanical design of a robot, which is working together with other robots inside a swarm, differs in several points from the design of a robot being a part of an organism. In the first case, criteria like small size, simple kinematics, simple casing, high mobility and low price define the design of the robot. On the other hand, a robot inside of a self-reconfigurable organism needs docking elements, high-power motors to produce enough torque, depending on the design of the organism one or more independent degrees of freedom and a casing with high stiffness to handle reaction between robots. Within the REPLICATOR and SYMBRION projects, one of the challenges will be to combine the characteristics of both kinds of robots into one.

In the beginning, there seems to be a few major problems that need to be solved. First of all the robot must have a docking element capable of handling the stress of several robots docked to it while applying all their forces (e.g. gravity, reaction, inertia force). Additionally, the docking element needs to assure the automatic coupling of several electric contacts needed for information exchange and power distribution between the robots inside an organism. Beside technical requirements the docking element should support the self aggregation of the robots. No matter how the position of two robots to each other is, the docking procedure should work. Therefore the docking element needs to balance misalignment and displacement to a certain degree. To increase the amount of possible structures for the organism and to simplify docking for the robots, all docking elements will be unisex and there will be at least four docking elements on each robot. Another problem is the mobility of the robot requested by the swarm based requirements. In order to guarantee local communication between robots, a reasonable velocity (i.e. a contacting rate) is needed. The kind of suitable locomotion is under evaluation.

The general approach in the state of art of modular robotics is the development of "cube-like" robotic modules with internal motors, batteries and control. The docking ports are usually placed on the sides and both locomotion and lifting abilities are provided mechanically separating the module in two blocks, able to bend reciprocally. This bending allows the lifting of attached modules, but often represents the only locomotion strategy for the robot, that can be quite slow and complex to control in accuracy and resolution of movement. Hence, as a new feature in modular robotics, we are currently considering to introduce higher locomotion capabilities, for instance integrating wheels in the modules, giving more independency to each module. The aim is to fabricate modules that are firstly conceived as independent robots rather than "just" modules to be assembled in a robotic organism. The increase in independency for what concerns locomotion allows in this way single modules, now robotic units, to move and explore the environment, rapidly acquiring information about the environment. Subsequently, they can rapidly reach their neighbours and, as a last process, engage assembly. Furthermore, wheeled modules could be used by the robotic organism as "wheeled feet" in order to have a faster global locomotion. As advanced feature, the wheels themselves could be an actuation mechanism (i.e., a rotational degree of freedom), considering to integrate into the wheels the docking mechanism. A wheeled-locomotion approach is characterized by a very high-energy efficiency on smooth surfaces, but it could show limitation on sandy or pebbly surfaces and even in facing small obstacles (like electrical cables, grass, etc.). The first concept in order to solve this issue consists in moving from a basic mini-rover configuration with four wheels to a caterpillar-based robotic unit, able to provide locomotion even on challenging surfaces and environments.

A differential drive is easy to implement and to control. However, not every movement is possible. The docking of two robots in the orientation of their wheel axis is only possible with a non trivial motion sequence. A non-holonomic drive is capable of positioning the robot everywhere and in any orientation to another robot, but is difficult to implement in design as well as in control. With at least two degrees of freedom a movement by crawling is also possible. Unfortunately, this is done by use of the main actuators which consume a lot of power. The optimal solution depends therefore on the scenario for the robots. At the moment, a crawling like locomotion is likely for the replicator robot while a mixture between non-holonomic and differen-

tial drive is more suitable for the SYMBRION robot.

These are only two challenges out of many which need to be solved within the REPLICATOR and SYMBRION projects, but in the end we will know much more about suitable design of self aggregating robots.

4.2 Electronic Challenges

The electronic design is a huge challenge due to strong restrictions of the size of the robot and the complexity of the hardware design. Each stand-alone robot is equipped with two processors, one main microcontroller (MCU) and one shadow microprocessor (CPU, see Fig. 2). The breakdown in microcontroller and microprocessor was deliberately intended to separate computational tasks within the single cell.

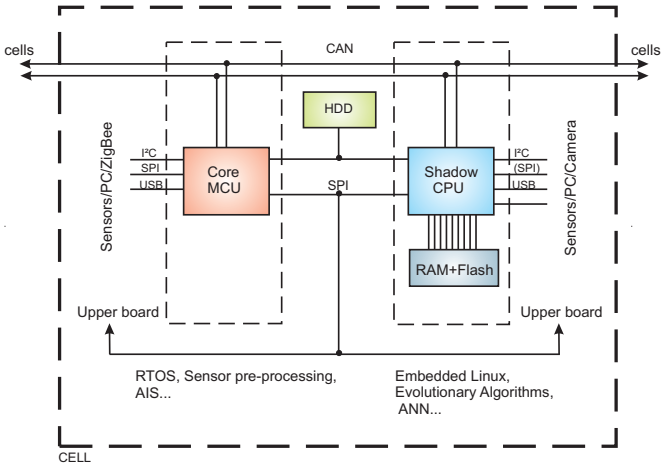


Figure 2: Functional electronic design of modules.

The core microcontroller performs basic functionality (e.g. sensor pre-processing, running artificial immune system) and keeps the robot alive. The shadow microprocessor is mainly responsible for bio-inspired approaches like the genetic algorithms, sensor fusion, ANNs etc. and is more powerful in comparison to the core processor. Due to higher computational power results in higher energy consumption, the shadow processor is able to run at different power-down modes when computational power isn't needed.

One of the biggest challenges during the electronic design is the development of the shadow processor module in a tiny size as well as finding solutions for shared resources like memory, power and communication capabilities.

5. SOFTWARE CHALLENGES

Beside the hardware challenges, the project is faced with many software requirements. Because robots can either run independently, as a swarm, as an organism, or even as a swarm of organisms, the interaction has to be managed in an organized and efficient way.

The different layers of software development are shown in Fig. 3. On the bottom layer there are two different processors, which are able to communicate with each other and have to be coordinated at the robot level. To cope with the additional difficulties a swarm or an organism causes, a middleware-like system is necessary. On top of this abstraction layer high-level control mechanisms and distributed applications can be integrated.

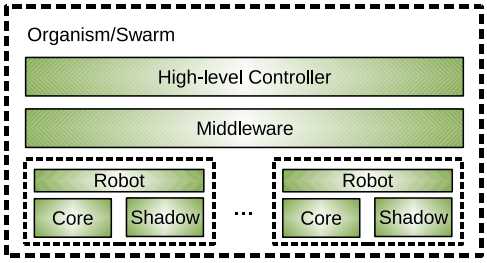


Figure 3: Different layers for software development.

5.1 Robot OS

On the robot level it is necessary to adopt mechanisms to coordinate data and data flow in a suitable way, and at the same time it is essential to predict and organize the behaviour of the multitasking system. It is also critical for the project to deal with the event driven architecture of the robot especially in respect to the real-time sensor system. For this reason a Real-Time Operating System (RTOS) builds the base of the software structure. The software architecture of the robot is modular and provides basic functionalities such as creation and termination of tasks and threads, inter-task communication, inter-processor communication, external event management, and memory allocation. Additionally the robot runtime system has to provide the middleware system with appropriate interfaces.

5.2 Middleware

Once the robots are aggregated into a more advanced multi-cellular organism it is essential to have an efficient controlling and coordinating mechanism. Therefore a middleware layer is introduced which defines unified interfaces and communication services according to the individual robot capabilities. The distributed middleware controls synchronization processes between nodes, configures and handle the communication bus (CAN) and manages distributed memory and energy resources. Furthermore it has to provide the robots with an abstraction layer between the operational system functions and the high-level controller domain.

5.3 High-level control concepts

Our projects will evaluate and test a variety of different control-concepts for the single robots as well as for the aggregated high-level robotic organism. Example can be given by artificial immune and artificial neuronal networks, different learning mechanisms as well as classical model-based controllers. In the following we describe one of these controllers - a bio-inspired controller concept which we call "Hormone-Driven Robot Controller" (HDRC). A data-structure that will hold configuration information for the robot, especially for the used software controllers of the robotic node, is called "Genome" in our constortia's terminology. This Genome will contain also a set of rules that link the degradation and the secretion of hormones to the local levels of other hormones. The secretion of hormones can be triggered by other hormones or by receptors that get activated by receiving environmental stimuli. Hormones can alter the sensitivity of receptors, trigger activities, modulate certain controllers or even activate/deactivate whole (sub-)controllers. This way we expect that a variety of systems can easily evolve:

1. **Homeostatic systems:** These hormone systems can

allow the organism to regulate a variety of internal properties around a homeostatic set point. For example, an robot "hungry" for light but located in the dark, can increase its motion level, thus will increase its chance to find a light spot.

2. Adaptive behaviours: Hormones can reflect a change of state of the environment, thus they can modulate controllers to respond to these environmental changes.

3. Target-oriented behaviours: Hormones with very fast dynamics (short-term acting hormones) can be even used to steer robots autonomously towards certain targets or to actively avoid areas or objects. This can be used in the previously mentioned foraging-for-energy scenarios.

3. Signal propagation and timing: As hormones are passed also among the robotic nodes in an aggregated organism, hormones can be used for signal propagation along the body of this organism. In nature, such systems show frequently the ability to perform rather well working timing tasks (see for example the synchronization of fireflies in [2]). We can expect to evolve such signal-propagation mechanisms also in our aggregated robotic organisms, possibly synchronizing the movement patterns of legs or other body parts.

To allow the HDRC to evolve the above-mentioned tasks and to evolve the needed functionality to perform such tasks, we have to implement a separate hormone controller. This controller is created from the evolved information in the Genome and frequently simulates hormone secretions (additions), degradations and diffusion within each robotic node. Using the available communication capabilities of the robots, the hormones are exchanged also between the robotic nodes, thus allowing a diffusion of virtual hormones within the whole higher-level organism.

Fig. 4 shows two distinct ways how the HDRC can be used in two different swarm states:

State 1: Fig. 4a shows that each robot is contains several virtual compartments, which associated with different real robot "body parts". In the case depicted here, each robot contains 2 lateral, one frontal and one terminal compartment. In the center, there is a fifth (central) compartment located. Sensors can trigger excretion of hormones into their corresponding local compartment and actuators can be modulated/affected only by hormone concentrations which are present in their corresponding compartment. Hormones are diffused to neighboring compartments and to to the central compartment. In the depicted case, a light sensor senses an obstacle to the left of the robot. It triggers the secretion (addition) of a hormone into this segment, which enhances the speed of the associated left motor. By diffusion, the same hormone reaches also the right compartment, where it can decrease the rotation speed of the right motor: The robot turns to the right. A central luminance sensor (central compartment) can trigger the secretion of another hormone, which generally increases motor speed on both sides: The robot drives (forages) faster in brighter illuminated areas.

State 2: Fig. 4b shows a totally different usage of the HDRC: One robot started to call other robots for aggregation. It secrets a specific "head-marking" hormone. This hormone is secreted only in the first robot that starts the aggregation. Due to the diffusion process and the increasing chain length, the concentration of this hormone decreases, as the robotic organism gets larger. By using this gradient as a source of information for the "tail robots", the organism can be limited to certain sizes and there is always a gradient

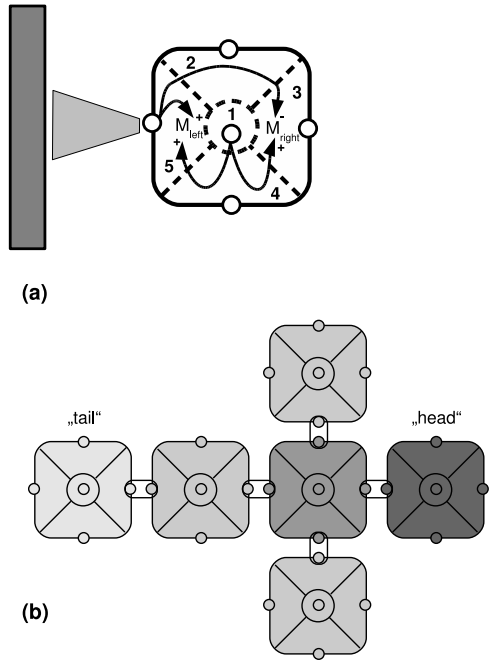


Figure 4: (a) Schematic drawing of a hormone-driven robot controller (HDRC) performing an obstacle-avoidance behaviour. Receptors can trigger hormone secretions. these hormones can differ through several virtual body compartments inside of a single robot. Hormones can switch on or off actuators, modulate actuator function or interfere with other hormones; (b) Several robotic nodes are coupled together to a higher-level organism. Simulated hormones are floating through the "body", forming hormone gradients. In the picture, a dedicated "head"-hormone is shown. These hormone gradients can support the formation of aggregated organisms out from the "fuzzy" swarm state, which is formed by many free-driving or free-walking robots.

inside of the organism that points towards the "head". This gradient information can also be important for coordinating body movements.

5.4 Simulation Framework

To test, compare and verify different robot designs, different organism configuration and the controllers in a quick and cheap way, a simulation environment needs to be implemented. The simulation should offer an easy and fast way to create a test environment and to design some basic robot architectures to test the availabilities the robot might have. Later the simulation can be used to test different organism configurations and to verify the different controllers. Furthermore, the simulation can be applied in long term scenarios to explore biologic mechanisms like evolutionary and genetic algorithms, collective and symbiotic behaviour and neuronal networks.

In the REPLICATOR and SYMBRION projects we will use Delta-3D for the simulation framework, see Fig. 5. It offers lots of interfaces and has already successfully been used in other simulations. The aim is to simulate the physics of the single robots, as well as that one of a whole organism.

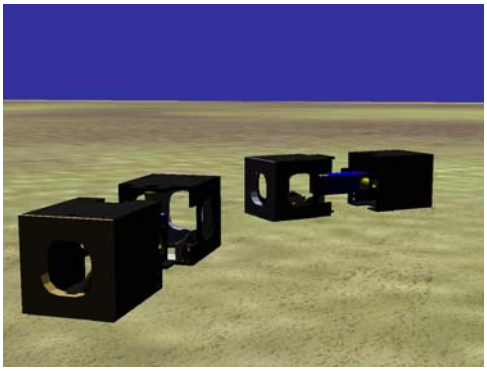


Figure 5: Screenshot of the Simulation Environment Using Delta 3D.

3D-models from CAD-programs can easily be loaded into the environment and without any effort a robot model can be created in the simulation. Different robot configurations like form, size and the position and orientation of the joints of the robot can be easily created and tested within the environment. Additionally the simulation needs to integrate the controller into the environment without modifying it with the aim that the robot will behave in the simulation nearly the same as in the real environment. Therefore a hardware abstraction layer (HAL) needs to be strictly specified, which will offer the same functionality for the actual robot and the simulated one. A robot controller will only use the functionality of the HAL, so that a robot, developed in a simulation environment, will run on a real robot and vice versa.

As the requirements to the simulation will grow with the complexity of the controller, running the simulation on just one computer or even just on a single core won't be sufficient enough. Therefore a distributed simulation will be mandatory. For this purpose Delta-3D offers an interface to the High Level Architecture (HLA), a general purpose architecture for distributed computing. Using this architecture the simulation can be executed on several computers having different platforms located within a local area network or connected through the internet as well.

6. TOWARDS EVOLVE-ABILITY OF THE ROBOT ORGANISM

Within the projects, the creation of evolvable or otherwise adaptive software and hardware is the main focus. However, from the conceptual point of view, achievement of evolve-ability for the robot organism is planned in two complementary ways, which we call bio-inspired (or bio-mimicking) and engendering-based approaches.

6.1 Bio-inspired/bio-mimicking approach

Any bio-inspired approach is based on analogies to living organisms and is carried out by the biological partners in our consortia. Our bio-inspired control algorithms use neither any global point of information nor any form of complex knowledge. Our algorithms are stable to a wide range of environmental conditions and are extremely robust. Therefore, the bio-inspired strategies in projects are going to draw advantage from the well-known robustness/simplicity as well as from the plasticity/adaptability derived from natural systems. Our goal is to create stable, robust and adaptable robotic organisms. Here we will investigate a variety of concepts, such as:

1. Genome: All robotic organisms will carry one or several Genomes. A Genome is a collection of genes, which carry information about controller structure and controller dynamics. A gene can be a simple part of a blueprint, which "depicts" a part of the final controller. But a gene can also work as a rule, which is used to "construct" parts of the final controller. In the latter case, there can be interferences between different genes, thus competition or cooperation can arise also on the genetic level. A self-organized process can be established which will be able to create a flexible, but robust controller structure.

2. Controller: We will investigate several controller types, ranging from rules-based controllers, to Evolvable Artificial Neural Networks (EANN), to hormone-based controllers and to even hand-coded controllers that execute hand-optimized (modular) parts of the whole organism's behavioural repertoire.

3. Sexuality/Reproduction: We plan to enhance and to speed-up the dynamics of artificial evolution by implementing virtual-reproduction of robots. A separate process will allow to remove controllers from the least fit robots and to re-initialize them with mixtures (interbreeds) of the controllers of more fit robots. We will also investigate the advantages of sexual reproduction in such scenarios.

4. Embryology: To allow well-ordered controllers to emerge from the information stored in the Genome, we will mimic embryological processes, driven by a virtual hormone system.

6.2 Engineering-based approach

The engineering-based approach is complementary to the bio-inspired one and focuses in such issues as learning, distributed decision making, navigation and so on. Generally, consortium focuses on three following approaches (these approaches are closely connected so that finally it will be a kind of hybrid framework):

1. On-line learning. On-line learning is based on the behavior level and uses automatically generated feedback. The feedback comes from internal, external and virtual sensors. Some direct feedback can be sensed through vision-based subsystem, by using FRID-based identification or localization technologies, by using smart laser scanner, sound, light, humidity, temperature, internal energy sensor and other sensors. It is intended to use middleware and sensor-fusion approach to generate complex non-direct feedbacks through virtual sensors. Since off-line mechanisms can hardly be applied to real robots, the challenge of the proposed approach is to perform non-supervised learning without any off-line mechanisms (or at least with a minimum of them). This can be achieved by combining evolving computation with rewards/feedback/fitness calculated on-line. Therefore the whole approach can be named "on-line learning".

2. Evolutionary computation. High computational power of the system allows running on-line and on-board such well-known approaches as genetic programming (GP) (e.g. [22]), Genetic Algorithms (GA) (e.g. [23]). To avoid the problems posed by a huge search space, we intend to integrate limitations, originating from hardware platform. Another set of problems we are aware of are the fitness functions required for these algorithms. These fitness functions are very difficult to calculate based only on local sensor

data. Moreover these functions are evaluated extremely delayed because the organism mostly assess their fitness after accomplishing the task.

3. Approaches from the domain of Distributed Artificial Intelligence (DAI). On-line learning as well as GA/GP include diverse aspects of DAI such as a distributed knowledge management, semantic information processing, navigation and actuation in the environment, planning, sensor fusion and others. Development and implementation of these approaches is an important step towards evolve-ability of the robot organisms.

7. CONCLUSION

In this short paper we made on overview of two large European projects, dealing with a new paradigm in collective systems, where the swarm robots get capable of self-assembling into a single symbiotic multi-robot organism. We introduced an energy foraging scenario for both robot species and demonstrated that a transition between collective and symbiotic robot forms represents a very hard problem. It involves not only hardware and software issues, but also very basic questions being also open not only in biological but also in engineering sense. We demonstrated the main hardware and software challenges and the road-map how to achieve the evolve-ability of the robot organisms.

8. ACKNOWLEDGEMENT

REPLICATOR and SYMBRION projects are funded by European Commission within the 7th framework programm. We want to thank "euCognition"[1] for supporting your involvement in the conference.

9. REFERENCES

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm intelligence: from natural to artificial systems*. Oxford University Press, New York, 1999.

[2] S. Camazine, J-L. Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-Organization in Biological Systems*. Princeton University Press, Princeton, NJ, USA, 2003.

[3] D.W. Stephens and J.R. Krebs. *Foraging Theory*. Princeton University Press, 1987.

[4] E. Sahin. *Swarm Robotics: From sources of inspiration to domains of application*. Springer-Verlag, Heidelberg, Germany, 2004.

[5] S. Kornienko, R. Thenius, O. Kornienko, and T. Schmickl. Re-embodiment of honeybee aggregation behavior in artificial micro-robotic system. *Adaptive Behavior (accepted for publication)*.

[6] S. Kornienko, O. Kornienko, and P. Levi. Ir-based communication and perception in microrobotic swarms. In *Proc. of the IROS 2005, Edmonton, Canada*, 2005.

[7] G. Weiss. *Multiagent systems. A modern approach to distributed artificial intelligence*. MIT Press, 1999.

[8] J.Y. Halpern and Y. Mosesi. Knowledge and common knowledge in a distributed environment. *J. of the Association for Computer Machinery*, 37(3):549–587, 1990.

[9] H. Haken. *Synergetics: An introduction, third edition*. Springer-Verlag, New York, 1983.

[10] S. Kornienko, O. Kornienko, A. Nagarathinam, and P. Levi. From real robot swarm to evolutionary multi-robot organism. In *Proc. of the CEC2007, Singapure*, 2007.

[11] S. Kornienko and O. Kornienko. Collective energy homeostasis in a large-scale micro-robotic. *Robotics and Autonomous Systems (under resubmitting)*.

[12] D. Häbe. *Bio-inspired approach towards collective decision making in robotic swarms*. Master Thesis, University of Stuttgart, Germany, 2007.

[13] T. Kancheva. *Adaptive role dynamics in energy foraging behavior of a real micro-robotic swarm*. Master Thesis, University of Stuttgart, Germany, 2007.

[14] A. Attarzadeh. *Development of advanced power management for autonomous micro-robots*. Master Thesis, University of Stuttgart, Germany, 2006.

[15] A. Ishiguro and T. Maegawa. Self-assembly through the interplay between control and mechanical systems. In *Proc. of IEEE/RSJ06 Int. Conf. on Intelligent Robots and Systems*, pages 631–638, Beijing, China, 2006. IEEE.

[16] S. Murata, K. Kakomura, and H. Kurokawa. Docking experiments of a modular robot by visual feedback. In *Proc. of IEEE/RSJ06 Int. Conf. on Intelligent Robots and Systems*, pages 625–630, Beijing, China, 2006. IEEE.

[17] Wei-Min Shen, Maks Krivokon, Harris Chiu, Jacob Everist, Michael Rubenstein, and Jagadesh Venkatesh. Multimode locomotion for reconfigurable robots. *Autonomous Robots*, 20(2):165–177, 2006.

[18] Haruhisa Kurokawa, Kohji Tomita, Akiya Kamimura, Shigeru Kokaji, Takashi Hasuo, and Satoshi Murata. Distributed self-reconfiguration of m-tran iii modular robotic system. *Int. J. Rob. Res.*, 27(3-4):373–386, 2008.

[19] Alex Golovinsky, Mark Yim, Ying Zhang, Craig Eldershaw, and David Duff. Polybot and polykinetic/spl trade/ system: a modular robotic platform for education. In *ICRA*, pages 1381–1386. IEEE, 2004.

[20] V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson. Self-reproducing machines. *Nature*, 435(7039):163–164, 2005.

[21] David Johan Christensen, Esben Hallundbok Ostergaard, and Henrik Hautop Lund. Metamodule control for the atron self-reconfigurable robotic system. In *Proc. of the IAS-8*, pages 685–692, Amsterdam, 2004.

[22] J. Koza. *Genetic programming: on the programming of computers by means of natural selection*. MIT Press, Cambridge, Massacgusetts, London, England, 1992.

[23] M. Srinivas and Lalit M. Patnaik. Genetic algorithms: A survey. *Computer*, 27(6):17–26, 1994.

[1] http://cordis.europa.eu/ist/cognition/index.html

Virtual Agent Modeling in the RASCALLI Platform

(Special Session on EU-projects)

Christian Eis
Research Studios Austria
Vienna, Austria
christian.eis@researchstudio.at

Marcin Skowron
Austrian Research Institute for Artificial Intelligence
Vienna, Austria
marcin.skowron@ofai.at

Brigitte Krenn
Austrian Research Institute for Artificial Intelligence,
Research Studios Austria
Vienna, Austria
brigitte.krenn@ofai.at

ABSTRACT

The RASCALLI platform is both a runtime and a development environment for virtual systems augmented with cognition. It provides a framework for the implementation and execution of modular software agents. Due to the underlying software architecture and the modularity of the agents, it allows the parallel execution and evaluation of multiple agents. These agents might be all of the same kind or of vastly different kinds or they might differ only in specific (cognitive) aspects, so that the performance of these aspects can be effectively compared and evaluated.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Intelligent agents; D.2.11 [**Software Architectures**]: Domain-specific architectures; K.6.3 [**Software Management**]: Software development

General Terms

Design, Experimentation, Measurement

Keywords

Cognitive agents, Agent modeling, Agent evaluation, Open source software

1. INTRODUCTION

This paper gives an overview of the architecture and functionality of the RASCALLI platform, developed as part of the RASCALLI project[1]. In this project, the platform is used as the underlying software environment for the development and execution of so called RASCALLI (Responsive Artificial Situated Cognitive Agents that Live and Learn on the Internet). It provides the facilities for user-agent and agent-agent communication and serves as a testbed for the evaluation of various incarnations of the agents that use different sets of action-perception tools, action selection mechanisms and knowledge resources. The platform supports a modular development style, where agents are assembled from small re-usable building blocks. Agents of different kinds and configurations can run simultaneously within a single platform environment. This enables the evaluation and comparison of different agents as well as the evaluation of whole agent communities.

The paper is organized as follows: Section 2 describes the general objectives of the RASCALLI project to the extent that they are relevant for the design and implementation of the RASCALLI platform. Section 3 gives a brief overview of related platforms and methodologies. The RASCALLI platform itself is then described in section 4, and section 5 gives an overview of the agent components implemented in the RASCALLI project. Finally, section 6 explains how the RASCALLI platform can be used for agent evaluation.

2. PROJECT OBJECTIVES

The project RASCALLI aims at the development of virtual agents that perform tasks related to accessing and processing information from the Internet and domain-specific knowledge bases. RASCALLI agents, further referred to as Rascalli, represent a growing class of cooperative agents that do not have a physical presence, but nevertheless are equipped with major ingredients of cognition including situated correlates of physical embodiment to become adaptive, cooperative and self improving in a virtual environment, given certain tasks.

The project objectives cover the following topics: development of a computational framework for the realization of cognitive agents providing intelligent assistance capabilities; cognitive architecture and modeling; perception and action; reasoning; learning; communication; agent-to-agent and agent-to-user interfaces. The Rascalli answer questions of their users, learn the users' preferences and interests, and use this knowledge to present the users with new, interesting information. The agents exist in an environment consisting of external knowledge sources on the Internet, such as search engines and RSS feeds, and domain-specific knowledge bases. They communicate with their users as well as with other Rascalli.

Rascalli are developed in a modular fashion, which allows individual agents to be built from different sets of components. This enables e.g. the creation of agents which are "ex-

[1]European Commission Cognitive Systems Project FP6-IST-027596-2004 RASCALLI.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

perts" in different knowledge domains. During its life-time, each agent adapts to its user's interests and thus further specializes in a certain sub-domain and evolves in accordance with the user's preferences. This results in a community of specialized agents, which can communicate with each other to provide requested information to their users.

The modular approach enables the evaluation and comparison of different sets of components, including different cognitive aspects (e.g. different learning strategies) by executing multiple Rascalli in the same environment and evaluating their performance (e.g. by means of user tests).

For the realization of these objectives, system integration turned out to be a major roadblock, due to the following reasons:

- Even though Java was chosen as the main implementation language for the project, some project partners have no or little experience with Java development.
- In order to avoid re-implementation, we had to integrate existing components from previous projects. These components are based on a wide range of technologies, such as different programming languages (e.g. Perl, Java, Lisp) and even native binaries for different operating sytems (Linux and Windows).
- Initial attempts at providing a development environment that integrates all of these components and can be replicated on each developer's machine proved to be difficult to use and keep up-to-date.

Based on the project objectives and constraints outlined above, we arrived at the following **set of requirements** for our software platform:

- Support the execution of various agents, belonging to different users.
- Support agent-to-agent and agent-to-user communication.
- Allow developers to implement diverse agents based on shared components (this also means that multiple versions of each component can exist at the same time).
- Integrate external and legacy components with minimal effort.
- Build agents in a modular, component-based fashion.
- Build the platform itself in a component-based, extensible fashion.

3. RELATED WORK

Research into related work has been conducted in multiple directions, including multi-agent platforms, as well as platforms and development methodologies for cognitive systems. While many such platforms and methodologies exist, none of them meets the requirements set for the RASCALLI platform.

3.1 Multi-Agent Platforms

We have investigated FIPA[2] compliant agent platforms, such as JADE[3] [1], which is a Java-based middleware for multi-agent systems. However, these systems mostly focus on distributed systems and communication issues. Also, the RASCALLI platform is not a multi-agent system in the traditional sense, where agents are independent components of a larger application, working for a common goal. Instead, Rascalli are complete individual entities that simply happen to share the same environment and may communicate with each other, if they wish. Furthermore, none of the investigated agent platforms supports the development style targeted by the RASCALLI platform, where multiple agent architectures and agent definitions, as well as multiple versions of agent components co-exist in a single platform instance.

It might be interesting future work to implement or integrate some of the FIPA standards (such as the Agent Communication Language) with the RASCALLI platform.

3.2 Agent Development Methodologies and Frameworks

As an example of an agent development methodology, Behavior Oriented Design (BOD)[4] [2] supports the implementation of agents based on an iterative development process and a modular design. However, it does not provide much of a runtime environment. Therefore, BOD does not really compare to the RASCALLI platform, even though both advocate a modular approach. It would, however, be interesting to implement an agent architecture based on BOD within the RASCALLI platform and thus use the platform as a runtime environment for BOD agents.

As for development frameworks, AKIRA[5] [7] aims to create a C++ development framework to build cognitive architectures and complex artificial intelligent agents. However, like the FIPA multi-agent platforms, it targets different requirements than the RASCALLI platform. It might be interesting to consider exploiting some of AKIRA's concepts within the RASCALLI platform.

4. RASCALLI PLATFORM

This section provides an overview of the RASCALLI platform. We start with a set of features supported by the platform in order to fulfill the requirements listed above. Then we describe the software architecture of the platform and explain how this architecture supports the various platform features.

4.1 Platform Features

Multi-Agent: The platform supports the concurrent execution of multiple agents, including agents of the same kind, as well as agent of different kinds, ranging from very similar to vastly different.

Multi-User: Each agent has a single user (but one user may have several agents).

[2]http://www.fipa.org/
[3]http://jade.tilab.com/
[4]http://www.cs.bath.ac.uk/ai/AmonI-sw.html
[5]https://sourceforge.net/projects/a-k-i-r-a/

Agent Layer	Agent architectures, components, definitions and instances
Framework Layer	Technical services and utilities (e.g. networking support, RDF support, logging support)
Infrastructure Layer	Basic tools and components (e.g. Java, OSGi, Maven)

Figure 1: Platform Layers

Communication: Agent-to-agent communication can be implemented on the Java level, since all agents run within a single runtime environment. Alternatively, agents can communicate via instant messaging (this opens the possibility to use multiple, distributed platform instances). Several channels for agent-to-user communication have been implemented (currently a proprietary protocol for the 3D client interface, Jabber instant messaging and email).

Component-Based Architecture: All the parts that are required to set up a specific agent are developed in a component-based fashion so that individual Rascalli can be assembled from these components in a Lego-like manner.

Extensibility: The platform itself is built in a component-based fashion and can therefore be easily extended, even at runtime.

Distributed, Concurrent Development: A single platform instance is shared by all developers for implementing different agents, with minimal interference between individual developers.

Multi-Version: In order to support the concurrent development of different kinds of agents based on shared components, the platform supports the execution of multiple versions of the same components at the same time.

System Integration: External (and possibly legacy) components need to be integrated only once and are then available to all agents running in the platform in an easy-to-use manner. Since only one instance of the platform is required, there is no need to duplicate the entire software environment on multiple developer machines.

4.2 Platform Architecture

The RASCALLI platform is implemented as a layered architecture (Fig. 1).

4.2.1 Infrastructure Layer

The Infrastructure Layer contains basic tools and components used in the RASCALLI project. Specifically, these are Java, Maven[6] and OSGi[7]. In addition, this layer contains custom-made development and administration tools for the RASCALLI platform, such as user interfaces for agent configuration and deployment tools.

The most important feature of this layer is the use of OSGi, which implements a dynamic component model on top of Java. This means that components can be installed,

[6]http://maven.apache.org/
[7]http://www.osgi.org/

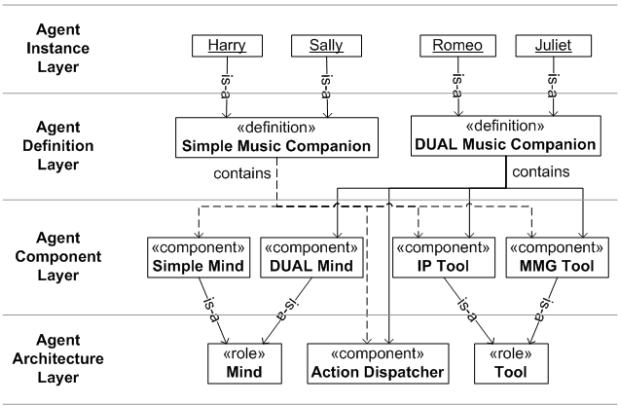


Figure 2: Agent Layer

started, stopped and uninstalled at runtime. Furthermore, dependencies between components are managed by OSGi in a fashion that allows the execution of multiple versions of a single component at the same time. Finally, OSGi provides a framework for service-based architectures, where each component can provide services to other components, based on Java interface specifications.

The use of OSGi thus enables the platform features 'multi-version' and 'extensibility', and supports the implementation of a 'component-based architecture' in the upper two platform layers.

'Distributed, concurrent development' is enabled by Maven and some custom-made components on this layer.

4.2.2 Framework Layer

The Framework Layer comprises general platform services and utilities employed by the Rascalli, including communication (user to agent, agent to agent), event handling, RDF handling, technology integration (Perl, web services, etc.), and various other platform services.

The services on this layer implement the 'multi-agent', 'multi-user' and 'communication' features of the platform. Furthermore, this is the place where 'system integration' takes place. External components are integrated and made available to the components of the Agent Layer as OSGi services, which can then be accessed on the Java level.

4.2.3 Agent Layer

The Agent Layer is the application layer of the platform and contains the implementation of the actual agents. It is designed to support the development and execution of multiple agents of different kinds as required by the project objectives. This layer consists of the following sub-layers:

Agent Architecture Layer: An agent architecture is a blueprint defining the architectural core of a particular type of Rascalli. More precisely, it sets the roles of agent components and provides means for defining and assembling a specific agent. The architecture can also contain implementations of common components shared by all agent definitions.

Agent Component Layer: Contains implementations of the roles defined on the Agent Architecture Layer.

Agent Definition Layer: An agent definition is an assembly of specific components of the Agent Component

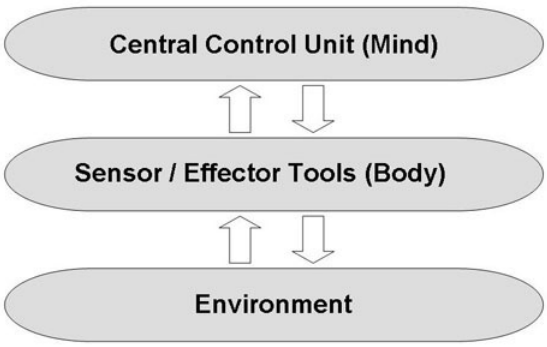


Figure 3: MBE agent architecture

Layer of a specific agent architecture. Different agent definitions for the same agent architecture might contain different components for certain roles.

Agent Instance Layer: Contains the individual agent instances. Each Rascallo is an instantiation of a specific agent definition.

Fig. 2 gives a (simplified) example of an agent architecture (the Mind-Body-Environment architecture, see section 5). The Agent Architecture Layer defines two roles, Mind and Tool, and implements an agent component (Action Dispatcher) shared by all agent definitions. The Agent Component Layer contains two implementations of the Mind role, as well as two Tools. Based on this architecture, two agent definitions combine each of the Mind implementations with the available Tools and the Action Dispatcher into different kinds of agents. Finally, a number of agent instances are shown on the Agent Instance Layer.

5. AGENT COMPONENTS

This section gives an overview of the agent components which have been implemented in the RASCALLI project (for more detailed information see [10]). RASCALLI agents are currently based on a reactive agent architecture with a perception-action loop. This agent architecture is called Mind-Body-Environment architecture, because each agent has a central control unit (Mind), which perceives and acts on the Environment via a set of sensor and effector tools (Body), as shown in Fig. 3.

The Environment comprises the agent's user, other agents, Internet services and domain-specific knowledge bases. The agent perceives its environment (via a set of perception sensors, implemented as software tools) as a set of virtual entities with their own characteristics and properties. These include strings of written language originating from the user or extracted from HTML documents, markup tags, information about the accessibility of various Internet and local tools and resources, user feedback, etc. Therefore, the agent has to deal with a dynamic environment i.e. evolving content of the websites, permanent or temporary inaccessibility of Internet services, appearance of new content or services, changes in the user preferences and interests, as well as the natural language input from the user and the web-pages. Similarly, all actions of an agent in its environment are performed on the above introduced set of entities.

The Body contains components called Tools, which serve as sensors (Perception Layer) and effectors (Actuator Layer). A specific agent definition may contain an arbitrary subset of the available Tools, but of course, the chosen Mind component must be able to deal with the selected Tools. Some of the available Tools are:

- **T-IP4Dual, T-IP4Simple and T-IP4Adaptive**, Input Processing Tools transform natural language and user feedback inputs into categorized information useable by the respective Mind components, and serve as a Perception Layer of an agent.

- **T-MMG**, for generating multi-modal output to the user. The Multi-Modal Generation Component provides a middle-ware functionality between generated agent output and the user interfaces. The generation component implements a template-based approach (in form of Velocity templates) by encoding vocabulary, phrases, gestures etc., which can be combined with the output of the RASCALLI Tools and context data. The use of Velocity[8], a template generation engine, allows templates to be designed and refined separately from the application code.

- **T-QA**, a general purpose open-domain question answering system (based on the work described in [8], [9]) is used in the RASCALLI platform to provide answers to the user factoid-type questions expressed in natural language.

- **T-Nalqi**, a natural language database query interface. The Tool is used in the RASCALLI platform for querying the databases accessible to the Rascalli, in a search for instances and concepts that can provide answers to the user questions. The component analyses natural language questions posed by the user and retrieves answers from the system's domain-specific databases.

- **T-RSS**, provides a mechanism for Rascalli to retrieve current information that might be of interest for the user.

- **T-Wikipedia**, an interface and analysis Tool for Wikipedia.

The Mind component performs action selection, based on the current input from the environment and the agent internal state. It can also make use of supporting services, for example a user/agent modeling service. The following Mind components are being developed in the RASCALLI project:

- **Simple Mind**, which performs action selection with a simple rule-based mechanism. These rules match to specific cues in the input data arriving from sensor channels. 'Simple Mind' extracts relevant information and passes this information on to the appropriate effector tool. Even though seemingly non-trivial behavior can be accomplished through a series of interactions of the Simple Mind and the available tools, the Simple Mind does not contain any cognitive aspects such as memory or learning.

[8] http://velocity.apache.org

- **DUAL Mind**, which incorporates the DUAL/AMBR ([5], [6]) cognitive architecture for action selection. It includes a long term memory (LTM) where general and episodic knowledge is stored, and a working memory (WM) constituted by the active part of LTM, perceptual input and goals. The DUAL mind operates only on represented knowledge and has only a mediated connection to the body and the environment. Thus it contains a partial, selected representation of the environment at an abstract conceptual level and experiential memories related to specific episodes like organization of the interaction of an agent with its environment.

- **Adaptive Mind**, a machine learning based classification driven action selection mechanism. Based on the available knowledge, including the perception of an input situation, the agent finds a set of actions that can be applied to a given input situation. The selection of a particular action is based on the similarities with other actions the agent had successfully performed in the past, i.e. received positive feedback from the user. The action selection classifiers are implemented as Maximum Entropy models [4]. The data used for training the classifiers represents an input situation in terms of entities from the environment perceived by the agent, an action applied to this situation and the feedback obtained from the user.

RASCALLI agents come with a variety of user interfaces comprising a 3D client featuring an embodied conversational character [3] (ECA-UI, see Fig. 4 and 5), a Jabber instant messaging integration, a web-based user interface (Web-UI) and two domain-specific, special purpose interfaces which allow the user the explore in a playful way the domain-specific knowledgebases accessible to the currently implemented Rascalli. The Web-UI[9] mainly serves for user registration, download of the 3D-client and specification of Internet resources (URLs and RSS-feeds) that are considered by the user as important to be monitored by the agent. The agents use the Web-UI to present list-like information to the user, and more generally all information not well suited for presentation by means of synthesized speech. The ECA-UI, on the contrary, specializes on virtual human-to-human dialogue. In order to avoid the bottelneck imposed by speech recognition, user input is restrained to utterances typed into a small text window and to pressing buttons in order to praise or scold the agent. The user is expected to ask domain-specific questions but may also engage in a chatterbot-type of conversation with the ECA. To do this the Rascalli make use of ALICE chat bot technology.[10]

The knowledge sources are a music database featuring songs and albums of more than 60,000 singers/musicians and a database providing background information on musical artists such as their family relations, religion, track record, band membership etc.[11]

Based on these components, multiple agent definitions have been conceptualized and developed, including the initial implementation of an agent with a basic set of tools (sandbox for testing the system components and their integration), an agent utilizing an implementation of the DUAL cognitive architecture as central control unit, and a Smart Music Companion. Using those various incarnations of the agents implemented in the RASCALLI platform the performance (in terms of user satisfaction) can be compared and evaluated.

[9] http://intralife.researchstudio.at/rascalli/
[10] http://www.alicebot.org/downloads/programs.html
[11] See http://rascalli.researchstudio.at/ and http://rascalli.dfki.de/ontology/ for browsing the data.

6. AGENT EVALUATION

In the following, we first give an overview of evaluation scenarios supported by the RASCALLI platform in general and the currently implemented Rascalli in particular. We then give a more detailed account of an evaluation scenario the goal of which is to investigate how the use of cognitive aspects in an agent can improve the user experience.

6.1 Evaluation Scenarios

We distinguish three kinds of scenarios:

1. Automated performance measures: The high modularity of the platform eases the integration of automated performance measures with existing and future agent architectures. For example, one could easily measure the time an agent needs to fulfill a given task, as well as the accuracy with which an agent performs certain tasks.

2. Data mining from user activity logs: All user activities are logged in the platform including user id, agent id, timestamp and activity type. These data can then be evaluated employing data mining techniques and other quantitative evaluation methods, in order to identify prevalent usage patterns of individual users and across different users, as well as different kinds of agents.

3. User testing: We distinguish two kinds of user testings. The one are studies where users interact with an agent for a short time to fulfill a certain, narrowly defined task. The other one are studies where users interact with their agents more freely for a longer period of time. While with the former the focus lies on testing specific aspects of the system and the related human-computer interaction, the latter address more general questions about what makes an agent a suitable companion for its user, which interfaces support which tasks and how the interaction of the user with the agent and his/her attitude towards, his/her liking and understanding of the companion changes over time.

As there are currently no automated performance measures built into the RASCALLI platform, and the platform has not yet been made accessible to a broader public and thus we still lack suffient amounts of usage data for applying data mining techniques, we will concentrate in the following on user testing. Due to its modularity, the platform allows us to experiment with agents being placed in the same environment, but assembled from different internal building blocks and therefore are equipped with different perception and action capabilities as well as decision making strategies.

A male (Fig. 4) and a female (Fig. 5), human-like version of the 3D character have been implemented. Both characters are built on the basis of the same body model and are equipped with a similar set of gestures and facial expressions.

Figure 4: ECA user interface (male)

Figure 5: ECA user interface (female)

They are also comparable as regards their appearance, both are designed to fit a pop-rock scenario. The identity of the characters except for gender related optical features allows us to study effects that may be attributed to different perception of gender by simply switching between the male and female character whereas all other parameters in the system are kept the same. The RASCALLI system enables us, without further programming, to experiment with all kinds of settings where we let a user interact with or train his/her agent making use of either the male or the female version of the 3D character and then switch to the opposite sex.

The availability of two conceptually different, complementary UIs for user-agent communication, i.e., the Web-UI for standard windows and menu-based human-computer interaction and the ECA-UI for interaction closely related to human multimodal dialogue, is a valuable prerequisite to investigate user preferences for different modes of interaction and social implications of human-computer interfaces.

At the time of writing two user studies are completed: the one (S1) concerning the liking and evaluation of individual users engaging in a prescripted question-answer communication with the male Rascalli character, the other one (S2) is a study concerning the usability of the interface to the music data. Two further user studies are under preparation. The one (S3) is a replication of the question-answer study employing the female character instead of the male one. Whereas the previously mentioned studies are all one-time encounters with the system, the second one (S4) of the studies under preparation is a longer-term study where the users interact with and train their agents over a longer period of time, with user assessments at the beginning, at several intermediate stages and at the end of the agent-user collaboration period. Amongst others, we are interested in changes over time of the users' liking of their agents, their expectations on the performance of their agents, their accounts of trust, and their strategies to adapt to their agents in order to achieve/satisfy their information requirements in collaboration with their agents.

In particular, we prepare two variants (narrow and broad) of the longer-term study. In the narrow study (S4.1), the task is to train the agent as good as possible to monitor RSS-feeds for a specific topic of interest. To do so, the users are asked to identifiy and sharpen a topic of interest such as artist, group, genre, song, album etc. by exploring the music-related knowledge of their agent utilizing the agent's domain-specific interfaces. In addition, they are asked to inform the agent about preferred Internet sources by specifying respective RSS feeds through the Web-UI. The agent will then monitor the feeds and alert the user when encountering relevant information making use of a Jabber client.

In the broad study (S4.2) the users are left much more unguided. Other than in S4.1 where the users are confined to the two domain-specific interfaces and the Web-UI, in S4.2 they have access to all UIs. The users are more generally informed that the agents have some domian-specific knowledge about popular music and besides can access the Internet. Depending on what information the users frequently access, which Internet sources they specify, and which feedback (praise, scolding) they give to their agents, the agents will adapt to the users' interests and aim at providing more related information. The more effort the users invest in training their agents, the better the agents should become in providing the users with new information relevant to the users' interests.

6.2 Evaluation Example: Use of Cognitive Aspects in an RSS Feed Filtering Scenario

We make use of a simple scenario for RSS feed filtering to illustrate the platform's capability to augment the agents' abilities by incorporating cognitive aspects into processing. Since the different agents can exist at the same time in the same environment and thus be subject to the same external influences, they can be reliably compared and evaluated.

Recall: In the RSS Feed Filtering scenario, the agents' task is to provide their users with (potentially) interesting information gathered from music-related RSS feeds on the Internet. The users train their agents according to their interests, thus creating an agent profile containing relevant keywords such as artist names, song names and genre names. The users also provide RSS feed URLs to their agents, which the agents then query for new information.

The research question for this simple scenario is whether certain properties of an agent improve the overall user satisfaction (as measured by the rate of false positives and negatives). Therefore, in addition to the basic keyword-based filtering of RSS feeds, the following elements can be added to a particular agent:

- The ability to find similar agents (agents with a similar

agent profile) and consider those agents' RSS feeds in addition to the ones provided by its own user. This essentially extends the agent's search space.

- The ability to find items (artists, songs, etc.) similar to the ones specified in the agent profile. These items are then added to the filter keywords, thus extending the agent's search criteria.

The services required to perform these tasks (finding similar agents, finding similar artists, etc.) are provided by the RASCALLI framework and made available to the agents as additional Tools in the action and perception layers.

This setup allows us to create and compare four agent definitions, containing neither of the two features, one of them, or both. For the evaluation, different users are equipped with one of the available types of agents. The users can then identify false positives and negatives.

Obviously, this example could be implemented without a complex system such as the RASCALLI platform. However, even in this simple case the platform has some advantages to offer:

- Components can be shared between agents. For example, the service to find similar agents is an external web service, which is made available within the platform as a simple OSGi/Java service.

- The platform provides a single runtime environment for all the agents (there would be multiple agents of each kind), so that inter-agent communication (for exchanging RSS feed URLs) can be easily implemented.

7. CONCLUSIONS

The RASCALLI platform meets a unique set of requirements, that is not targeted by any of the investigated platforms or methodologies. In particular, it supports the development and execution of multiple agents of different kinds. Furthermore, it supports the evaluation and comparison of such different agents within a single environment.

Future work includes the completion of the development environment (e.g. the integration with development tools such as the Eclipse IDE), as well as the possible integration of certain aspects of other projects, such as BOD, AKIRA or JADE.

The RASCALLI platform and selected system components will be made available to the research community by the end of 2008 via the project homepage.[12] In particular, the project partners Research Studios Austria (SAT) and the Austrian Research Institute for Artificial Intelligence (OFAI) will provide an open source version of the platform and Tools, respectively. In addition, SAT will make available their other system components and user interfaces as managed services with limited support and data volume free of charge for the research community. The German Research Center for Artificial Intelligence (DFKI) will make available their data sources via an open research license and provide their system components as webservices. The companies Ontotext and Radon Labs will make available their SwiftOWLIM semantic repository and the 3D client, respectively. The New Bulgarian University (NBU) will conribute their DUAL-inspired implementation of an agent's mind.

[12] http://www.ofai.at/rascalli

8. ACKNOWLEDGMENTS

This research is supported by the EC Cognitive Systems Project FP6-IST-027596-2004 RASCALLI, by the national FFG program 'Advanced Knowledge Technologies: Grounding, Fusion, Applications' (project SELP), and by the Federal Ministry of Economics and Labour of the Republic of Austria. The work presented here builds on joint work within the RASCALLI project. In particular the authors would like to thank our collegues from the following institutions: SAT, OFAI, Radon, NBU, Ontotext and DFKI. Finally, our participation in the PerMIS'08 workshop is supported by the European Network for the Advancement of Artificial Cognitive Systems (euCognition, European Commission, Unit E5 - Cognition, FP6 Project 26408).

9. REFERENCES

[1] F. Bellifemine, A. Poggi, and G. Rimassa. JADE: a FIPA2000 compliant agent development environment. In *AGENTS '01: Proceedings of the fifth international conference on Autonomous agents*, pages 216–217, New York, NY, USA, 2001. ACM.

[2] J. Bryson. *Intelligence by Design: Principles of Modularity and Coordination for Engineering Complex Adaptive Agents*. PhD thesis, MIT, Department of EECS, Cambridge, MA, 2001. AI Technical Report 2001-003.

[3] J. Cassell, J. Sullivan, S. Prevost, and E. Churchill, editors. *Embodied Conversational Agents*. MIT Press, 2000.

[4] E. Jaynes. Information theory and statistical mechanics. *Physical Review*, 106(4):620–630, 1957.

[5] B. Kokinov. A hybrid model of reasoning by analogy. In K. Holyoak and J. Barnden, editors, *Analogical Connections*, volume 2 of *Advances in Connectionist and Neural Computation Theory*, pages 247–320. Ablex, 1994.

[6] B. Kokinov and A. Petrov. Integration of memory and reasoning in analogy-making: the AMBR model. In D. Gentner, K. Holyoak, and B. Kokinov, editors, *Analogy: Perspectives from Cognitive Science*. MIT Press, in press.

[7] G. Pezzulo and G. Calvi. Designing modular architectures in the framework AKIRA. *Multiagent Grid Syst.*, 3(1):65–86, 2007.

[8] M. Skowron. *A Web Based Approach to Factoid and Commonsense Knowledge Retrieval*. PhD thesis, Hokkaido University, Sapporo, Japan, 2005.

[9] M. Skowron and K. Araki. Effectiveness of combined features for machine learning based question classification. *Special Issue of the Journal of the Natural Language Processing Society Japan on Question Answering and Automatic Summarization*, 12(6):63–83, 2005.

[10] M. Skowron, J. Irran, and B. Krenn. Computational framework for and the realization of cognitive agents providing intelligent assistance capabilities. In *Proceedings of the Cognitive Robotics Workshop at the 18th European Conference on Artificial Intelligence*, 2008.

Evaluation Criteria for Human-Automation Performance Metrics

Birsen Donmez
MIT
Dept. of Aero-Astro
Cambridge, MA, USA
1(617)258-5046
bdonmez@mit.edu

Patricia E. Pina
MIT
Dept. of Aero-Astro
Cambridge, MA, USA
1(617)258-5046
ppina@mit.edu

M. L. Cummings
MIT
Dept. of Aero-Astro
Cambridge, MA, USA
1(617)252-1512
missyc@mit.edu

ABSTRACT
Previous research has identified broad metric classes for human-automation performance to facilitate metric selection, as well as understanding and comparison of research results. However, there is still lack of an objective method for selecting the most efficient set of metrics. This research identifies and presents a list of evaluation criteria that can help determine the quality of a metric in terms of experimental constraints, comprehensive understanding, construct validity, statistical efficiency, and measurement technique efficiency. Future research will build on these evaluation criteria and existing generic metric classes to develop a cost-benefit analysis approach that can be used for metric selection.

Categories and Subject Descriptors
C.4 [**Performance of Systems**]: Measurement Techniques
J.7 [**Computers in Other Systems**]

General Terms
Measurement, Performance, Experimentation, Human Factors, Standardization, Theory.

Keywords
Metric Quality, Human Supervisory Control, Validity, Statistics, Experiments.

1. INTRODUCTION
Human-automation teams are common in many domains, such as military operations, process control, and medicine. With intelligent automation, these teams operate under a supervisory control paradigm. "Supervisory control means that one or more human operators are intermittently programming and continually receiving information from a computer that itself closes an autonomous control loop through artificial effectors and sensors to the controlled process or task environment [1]." Example applications include robotics for surgery, rock sampling for geology research, and military surveillance with unmanned vehicles.

A popular metric used to evaluate human-automation performance in supervisory control is mission effectiveness [2, 3]. Mission effectiveness focuses on performance as it relates to the final output produced by the human-automation team. However, this metric fails to provide insights into the process that leads to the final mission-related output. A suboptimal process can lead to a successful completion of a mission, e.g., when humans adapt to compensate for design deficiencies. Hence, focusing on just the mission effectiveness makes it difficult to extract information to detect design flaws and to design systems that can consistently support successful mission completion.

Measuring multiple human-computer system aspects, such as the situational awareness of the human, can be valuable in diagnosing performance successes and failures, and identifying effective training and design interventions. However, choosing an efficient set of metrics for a given experiment still remains a challenge. Many researchers select their metrics based on their past experience. Another approach to metric selection is to collect as many measures as possible to supposedly gain a comprehensive understanding of the human-automation team performance. These methods can lead to insufficient metrics, expensive experimentation and analysis, and the possibility of inflated type I errors. There appears to be a lack of a principled approach to evaluate and select the most efficient set of metrics among the large number of available metrics.

Different frameworks of metric classes are found in the literature in terms of human-autonomous vehicle interaction [4-7]. These frameworks define metric taxonomies and categorize existing metrics into high level metric classes that assess different aspects of the human-automation team performance and are generalizable across different missions. Such frameworks can help experimenters identify system aspects that are relevant to measure. However, these frameworks do not include evaluation criteria to select specific metrics from different classes. Each metric set has advantages, limitations, and costs, thus the added value of different sets for a given context needs to be assessed to select the set that maximizes value and minimizes cost.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

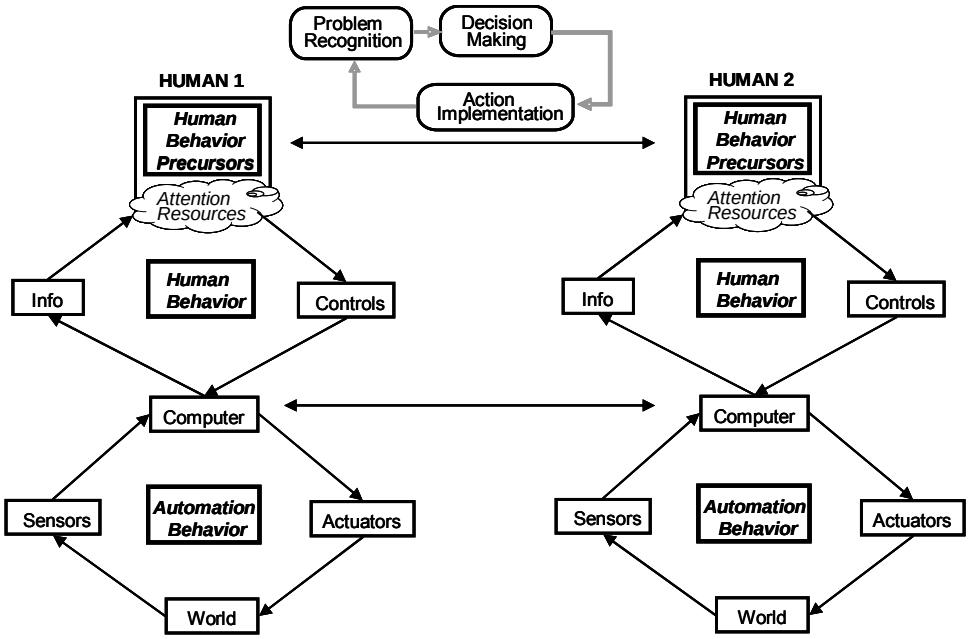


Figure 1. Conceptual model of human-supervisory control (modified from Pina et al. [5]).

This paper presents a brief overview of existing generalizable metric frameworks and then defines a set of evaluation criteria for metric selection. These criteria and the generic metric classes constitute the basis for the future development of a cost-benefit methodology to select supervisory control metrics.

2. GENERALIZABLE METRIC CLASSES

For human-autonomous vehicle interaction, different frameworks of metric classes have been developed by researchers to facilitate metric selection, and understanding and comparison of research results. Olsen and Goodrich proposed four metric classes to measure the effectiveness of robots: task efficiency, neglect tolerance, robot attention demand, and interaction effort [4]. This set of metrics measures the individual performance of a robot, but fails to explicitly measure human performance.

Human cognitive limitations often constitute a primary bottleneck for human-automation team performance [8]. Therefore, a metric framework that can be generalized across different missions conducted by human-automation teams should include cognitive metrics to understand what drives human behavior and cognition.

In line with the idea of integrating human and automation performance metrics, Steinfeld et al. suggested identifying common metrics in terms of three aspects: human, robot, and the system [7]. Regarding human performance, the authors discussed three main metric categories: situation awareness, workload, and accuracy of mental models of device operations. This work constitutes an important effort towards developing a metric toolkit; however, this framework suffers from a lack of metrics to evaluate collaboration effectiveness among humans and among robots.

Pina et al. [5] defined a more comprehensive framework for human-automation team performance based on a high-level conceptual model of human supervisory control. Figure 1 represents this conceptual model for a team of two humans collaborating, with each controlling an autonomous platform. The platforms also collaborate autonomously. These collaboration layers are depicted by arrows between each collaborating unit. The operators receive feedback about automation and mission performance, and adjust automation behavior through controls if required. The automation interacts with the real world through actuators and collects feedback about mission performance through sensors.

Based on this model, Pina et al. [5] defined five generalizable metric classes: mission effectiveness, automation behavior efficiency, human behavior efficiency, human behavior precursors, and collaborative metrics (Table 1). Mission effectiveness includes the popular metrics and measures concerning how well the mission goals are achieved. Automation and human behavior efficiency measure the actions and decisions made by the individual components of the team. Human behavior precursors measure a human's initial state, including attitudes and cognitive constructs that can be the cause of and can influence a given behavior. Collaborative metrics address three different aspects of team collaboration: collaboration between the human and the automation collaboration between the humans that are in the team, and autonomous collaboration between different platforms.

These metric classes can help researchers select metrics that can result in a comprehensive understanding of the human-automation performance, covering issues ranging from automation capabilities to human cognitive abilities. However, there still is a lack of an objective methodology to select a collection of metrics

that most efficiently measure a system's human-automation performance. The following section presents a preliminary list of evaluation criteria that can help researchers evaluate the quality of a set of metrics.

Table 1. Human supervisory control metric classes and subclasses [9]

1) Mission Effectiveness (e.g., key mission performance parameters)
2) Automation Behavior Efficiency (e.g., usability, adequacy, autonomy, reliability)
3) Human Behavior Efficiency
   a) Attention allocation efficiency (e.g., scan patterns, prioritization)
   b) Information processing efficiency (e.g., decision making)
4) Human Behavior Precursors
   a) Cognitive precursors (e.g., situational awareness, mental workload)
   b) Physiological precursors (e.g., physical comfort, fatigue)
5) Collaborative Metrics
   a) Human/automation collaboration (e.g., trust, mental models)
   b) Human/human collaboration (e.g., coordination efficiency, team mental model)
   c) Automation/automation collaboration (e.g., platforms' reaction time to situational events that require autonomous collaboration)

3. METRIC EVALUATION CRITERIA

The proposed metric evaluation criteria for human supervisory control systems consist of five general categories that are listed in Table 2. These categories focus both on the metrics, which are constructs, and on the associated measures, which are mechanisms for expressing construct sizes. There can be multiple ways of measuring a metric. For example, situational awareness, which is a metric, can be measured based on objective or subjective measures [10]. Different measures for the same metric can generate different benefits and costs. Therefore, the criteria presented in this section evaluate a metric set by considering the metrics (e.g., situational awareness), the associated measures (e.g., subjective responses), and the measuring techniques (e.g., questionnaires given at the end of experimentation).

These proposed criteria target human supervisory control systems, with influence from the fields of systems engineering, statistics, human factors, and psychology. These fields have their own flavors of experimental metric selection including formal design of experiment approaches such as response surface methods and factor analyses, but often which metric to select and how many are left to heuristics developed through experience.

Table 2. Metric evaluation criteria

1) Experimental Constraints (e.g., time required to analyze a metric)
2) Comprehensive Understanding (e.g., causal relations with other metrics)
3) Construct Validity (e.g., power to discriminate between similar constructs)
4) Statistical Efficiency (e.g., effect size)
5) Measurement Technique Efficiency (e.g., intrusiveness to subjects)

3.1 Experimental Constraints

Time and monetary cost associated with measuring and analyzing a specific metric constitute the main practical considerations for metric selection. Time allocated for gathering and analyzing a metric also comes with a monetary cost due to man-hours, such as time allocated for test bed configurations. Availability of temporary and monetary resources depends on the individual project; however, resources will always be a limiting factor in all projects.

The stage of system development and the testing environment are additional factors that can guide metric selection. Early phases of system development require more controlled experimentation in order to evaluate theoretical concepts that can guide system design. Later phases of system development require a less controlled evaluation of the system in actual operation. For example, research in early phases of development can assess human behavior for different automation levels, whereas research in later phases can assess the human behavior in actual operation in response to the implemented automation level.

The type of testing environment depends on available resources, safety considerations, and the stage of research development. For example, simulation environments can enable researchers to have high experimental control, and manipulate and evaluate different system design concepts accordingly. In simulation environments, researchers can create off-nominal situations and measure operator responses to such situations without exposing them to risk. However, simulation creates an artificial setting and field testing is required to assess system performance in actual use. The types of measures that can be collected are constrained by the testing environment. For example, the responses to rare events are more applicable for research conducted in simulated environments, whereas observational measures can provide better value in field testing.

3.2 Comprehensive Understanding

It is important to maximize the understanding gained from a research study. However, due to the limited resources available, it is not possible to collect all required metrics. Therefore, each metric should be evaluated based on how much it explains the phenomenon of interest and how much it helps explain the underlying reasons for what other metrics measure.

The most important aspect of a study is finding an answer to the primary research question. The proximity of a metric to answering the primary research question defines the importance of that metric. For example, a workload metric may not tell much

without a mission effectiveness metric. However, this does not mean that the workload metric fails to provide additional insights into the human-automation performance. Another characteristic of a metric that is important to consider is the amount of additional understanding gained using a specific metric when a set of metrics are already collected. For example, rather than having two metrics that measure mission effectiveness, having one metric that measures mission effectiveness and another metric that measures human behavior can provide a better understanding on the team performance.

In addition to providing additional understanding, another desired metric quality is its causal relations with other metrics. A better understanding can be gained, if a metric can help explain the underlying reasons to what the other metrics measure. For example, operator response to an event, hence human behavior, will often be dependent on the conditions and/or operator's state when the event occurs. The response to an event can be described in terms of three set of variables [11]: a pre-event phase that defines how the operator adapts to the environment; an event-response phase that describes the operator's behavior in accommodating the event; and an outcome phase that describes the outcome of the response process. The underlying reasons for the operator's behavior and the final outcome for an event can be better understood if the initial conditions and operator's state when the event occurs is also measured. When used as covariates in statistical analysis, the initial conditions of the environment and the operator can help explain the variability in other metrics of interest. Thus, in addition to human behavior, experimenters are encouraged to measure human behavior precursors and automation behavior in order to assess the operator state and environmental conditions which may influence human behavior.

3.3 Construct Validity

Construct validity refers to how well the associated measure captures the metric or construct of interest. For example, subjective measures for situational awareness ask subjects to rate the amount of situational awareness they had on a given scenario or task. These measures are proposed to help in understanding subjects' situational awareness [10, 12]. However, self-ratings assess meta-comprehension rather than comprehension of the situation: it is unclear whether or not operators are aware of their lack of situational awareness. Therefore, subjective responses on situational awareness are not valid to assess the actual situational awareness but rather the awareness of lack of situational awareness.

Good construct validity requires a measure to have high sensitivity to changes in the targeted construct. That is, the measure should reflect the change as the construct moves from low to high levels [13]. For example, the primary task performance starts to break down only when the workload reaches higher levels [13, 14]. Therefore, primary task performance measures are not sensitive to changes in the workload at lower workload levels, since with sufficient spare processing capacity the operators are able to compensate for the increase in workload.

A measure with high construct validity should also be able to discriminate between similar constructs. The power to discriminate between similar constructs is especially important for abstract constructs that are hard to measure and difficult to define, such as human workload or attentiveness. An example measure that fails to discriminate two related metrics is galvanic skin response. Galvanic skin response is the change in electrical conductance of the skin attributable to the stimulation of the sympathetic nervous system and the production of sweat. Perspiration causes an increase in skin conductance, thus galvanic skin response has been proposed and used to measure workload and stress levels (e.g., Levin et al. [15]). However, even if workload and stress are related, they still are two separate metrics. Therefore, galvanic skin response cannot alone suggest a change in workload.

Good construct validity also requires the selected measure to have high inter- and intra-subject reliability. Inter-subject reliability requires the measure to assess the same construct for every subject, whereas intra-subject reliability requires the measure to assess the same construct if the measure were repeatedly collected from the same subject under identical conditions.

Intra- and inter-subject reliability is especially of concern for subjective measures. For example, self-ratings are widely utilized for mental workload assessment [16, 17]. This technique requires operators to rate the workload or effort experienced while performing a task or a mission. Self-ratings are easy to administer, non-intrusive, and not expensive. However, different individuals may have different interpretations of workload, leading to decreased inter-subject reliability. For example, some participants may not be able to separate mental workload from physical workload [18], and some participants may report their peak workload whereas others may report their average workload. Another example of low inter-subject reliability is for subjective measures of situational awareness. Vidulich & Hughes [10] found that about half of their participants rated situational awareness by gauging the amount of information to which they attended; while the other half of the participants rated their SA by gauging the amount of information they thought they had overlooked. Participants may also have recall problems if the subjective ratings are collected at the end of a test period, raising concerns on the intra-subject reliability of subjective measures.

High correlation between different measures, even if they are intended to assess different metrics, is another limiting factor for metric selection. A high correlation can be indicative of the fact that multiple measures assess the same metric or the same phenomenon. Hence, including multiple measures that are highly correlated with each other can result in wasted resources.

3.4 Statistical Efficiency

There are three metric qualities that should be considered to ensure statistical efficiency: total number of measures collected, frequency of observations, and effect size.

Analyzing multiple measures that are correlated with each other would inflate type I error. That is, as more dependent variables are analyzed, finding a significant effect when there is none becomes more likely. The inflation of type I error due to multiple dependent variables can be handled with multivariate analysis techniques, such as Multivariate Analysis of Variance (MANOVA) [19]. It should be noted that multivariate analyses are harder to conduct as researchers are more prone to include irrelevant variables in multivariate analyses, possibly hiding the few significant differences among many insignificant ones. The best way to avoid failure to identify significant differences is to

design an effective experiment with the most parsimonious metric/measure set that is expected to produce differences, and excluding others that are not expected to show differences among many treatments.

Another metric characteristic that needs to be considered is the number of observations required for statistical analysis. Supervisory control applications require humans to be monitors of automated systems, with intermittent interaction. Because humans are poor monitors by nature [20], human monitoring efficiency is an important metric to measure in many applications. The problem with assessing monitoring efficiency is that, in most domains, errors or critical signals are very rare, and operators can go through an entire career without encountering them. For that reason, in order to have a realistic experiment, such rare events cannot be included in a study with sufficient frequency. Therefore, if a metric requires response to rare events, the associated number of observations may not enable the researchers to extract meaningful information from this metric. Moreover, small frequency of observed events cannot be statistically analyzed unless data is obtained from a very large number of subjects, such as in medical studies on rare diseases. Conducting such large scale supervisory control experiments is generally cost-prohibitive.

The number of subjects recruited for a study is especially limited when participants are domain experts such as pilots. The power to identify a significant difference, when there is one, depends on the differences in the means of factor levels and the standard errors of these means. Standard errors of the means are determined by the number of subjects. One way to compensate for limited number of subjects in a study is to use more sensitive measures that will provide a large separation between different conditions, that is, a high effect size. Experimental power can also be increased by reducing error variance by collecting repeated measures on subjects, focusing on sub-populations (e.g., experienced pilots), and/or increasing the magnitude of manipulation for independent variables (low and high intensity rather than low and medium intensity). However, it should also be noted that increased control on the experiment, such as using sub-populations, can lead to less generalizable results, and there is a tradeoff between the two.

3.5 Measurement Technique Efficiency

The data collection technique associated with a specific metric should not be intrusive to the subjects or to the nature of the task. For example, eye trackers are used for capturing operator's visual attention [21, 22]. In particular, head-mounted eye trackers can be uncomfortable for the subjects, and hence influence their responses. Wearing an eye-tracker can also lead to an unrealistic situation that is not representative of the task performed in the real world.

Eye trackers are an example of how a measurement instrument can interfere with the nature of the task. The measuring technique itself can also interfere with the realism of the study. For example, off-line query methods are used to measure operator's situational awareness [23]. These methods are based on briefly halting the experiment at randomly selected intervals, blanking the displays, and administering a battery of queries to the operators. This situational awareness measure then assesses global situational awareness metric by calculating the accuracy of operator's responses. The collection of the measure requires the interruption of the task in a way that is unrepresentative of the reality generating an artificial setting. The interruption may also interfere with other metrics such as operator's performance and workload, as well as other temporal-based metrics.

4. DISCUSSION

Supervisory control of automation is a complex phenomenon with high levels of uncertainty, time-pressure, and a dynamically-changing environment. The performance of human-automation teams depend on multiple components such as human behavior, automation behavior, human cognitive and physical capabilities, team interactions, etc. Because of the complex nature of supervisory control, there are many different metrics that can be utilized to assess performance. However, it is not feasible to collect all possible metrics. Moreover, collecting multiple metrics that are correlated can lead to statistical problems such as inflated type I errors.

This paper presented a preliminary list of evaluation criteria for determining a set of metrics for a given research question. These criteria were populated under five major headings: experimental constraints, comprehensive understanding, construct validity, statistical efficiency, and measurement technique efficiency. It should be noted that there are interactions between these major categories. For example, the intrusiveness of a given measuring technique can affect the construct validity for a different metric. In one such case, if the situational awareness is measured by halting the experiment and querying the operator, then the construct validity for the mission effectiveness or human behavior metrics become questionable. Therefore, the evaluation criteria presented in this paper should be applied to a collection of metrics rather than each individual metric alone, taking the interactions between different metrics into consideration. The list of evaluation criteria presented in this paper is a guideline for metric selection. It should be noted that there is not a single set of metrics that are the most efficient across all applications. The specific research aspects such as available resources and the questions of interest will ultimately determine the relative metric quality. Future research will identify a methodology based on a cost-benefit analysis approach, which will objectively identify the best set of metrics for classifications of research studies.

5. ACKNOWLEDGMENTS

This research is funded by the US Army Aberdeen Test Center.

6. REFERENCES

[1] Sheridan, T.B., *Telerobotics, automation, and human supervisory control*. 1992, Cambridge, MA: The MIT Press.

[2] Scholtz, J., et al., *Evaluation of human-robot interaction awareness in search and rescue*, in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2004: New Orleans.

[3] Cooke, N.J., et al., *Advances in measuring team cognition*, in *Team cognition: understanding the factors that drive process and performance*, E. Salas and S.M. Fiore, Editors. 2004, American Psychological Association: Washington, D. C. p. 83-106.

[4] Olsen, R.O. and M.A. Goodrich, *Metrics for evaluating human-robot interactions*, in *Proceedings of NIST*

Performance Metrics for Intelligent Systems Workshop. 2003.

[5] Pina, P.E., et al., *Identifying generalizable metric classes to evaluate human-robot teams*, in *Proceedings of Metrics for Human-Robot Interaction Workshop at the 3rd Annual Conference on Human-Robot Interaction*. 2008: Amsterdam, The Netherlands.

[6] Crandall, J.W. and M.L. Cummings, *Identifying predictive metrics for supervisory control of multiple robots.* IEEE Transactions on Robotics - Special Issue on Human-Robot Interaction, 2007. **23**(5): p. 942-951.

[7] Steinfeld, A., et al., *Common metrics for human-robot interaction*, in *Proceedings of the 1st Annual IEEE/ACM Conference on Human Robot Interaction (Salt Lake City, Utah)*. 2006, ACM Press: New York, NY.

[8] Wickens, C.D., et al., *An Introduction to Human Factors Engineering*. 2nd ed. 2004, Upper Saddle River, New Jersey: Pearson Education, Inc.

[9] Pina, P.E., B. Donmez, and M.L. Cummings, *Selecting metrics to evaluate human supervisory control applications*. 2008, MIT Humans and Automation Laboratory: Cambridge, MA.

[10] Vidulich, M.A. and E.R. Hughes, *Testing a subjective metric of situation awareness*, in *Proceedings of the Human Factors Society 35th Annual Meeting*. 1991, The Human Factors and Ergonomics Society: Santa Monica, CA. p. 1307-1311.

[11] Donmez, B., L. Boyle, and J.D. Lee, *The impact of distraction mitigation strategies on driving performance.* Human Factors, 2006. **48**(4): p. 785-804.

[12] Taylor, R.M., *Situational awareness rating technique (SART): the development of a tool for aircrew systems design*, in *Proceedings of the NATO Advisory Group for Aerospace Research and Development (AGARD) Situational Awareness in Aerospace Operations Symposium (AGARD-CP-478)*. 1989. p. 17.

[13] Eggemeier, F.T., C.A. Shingledecker, and M.S. Crabtree, *Workload measurement in system design and evalution*, in *Proceeding of the Human Factors Society 29th Annual Meeting*. 1985: Baltimore, MD. p. 215-219.

[14] Eggemeier, F.T., M.S. Crabtree, and P.A. LaPoint, *The effect of delayed report on subjective ratings of mental workload*, in *Proceedings of the Human Factors Society 27th Annual Meeting*. 1983: Norfolk, VA. p. 139-143.

[15] Levin, S., et al., *Tracking workload in the emergency department.* Human Factors, 2006. **48**(3): p. 526-539.

[16] Wierwille, W.W. and J.G. Casali, *A validated rating scale for global mental workload measurement applications*, in *Proceedings of the Human Factors Society 27th Annual Meeting*. 1983: Santa Monica, CA. p. 129-133.

[17] Hart, S.G. and L.E. Staveland, *The subjective workload assessment technique: a scaling procedure for measuring mental workload*, in *Human Mental Workload*, P. Hancock and N. Meshkati, Editors. 1988, North Holland B. V.: Amsterdam, The Netherlands. p. 139-183.

[18] O'Donnell, R.D. and F.T. Eggemeier, *Workload assessment methodology*, in *Handbook of perception and human performance: vol. II. Cognitive processes and performance*, K.R. Boff, L. Kaufmann, and J.P. Thomas, Editors. 1986, Wiley Interscience: New York. p. 42-1 - 42-49.

[19] Johnson, R.A. and D.W. Wichern, *Applied multivariate statistical analysis*. Fifth ed. 2002, NJ: Pearson Education.

[20] Sheridan, T.B., *Humans and automation: system design and research issues*. 2002, New York, NY: John Wiley & Sons Inc.

[21] Janzen, M.E. and K.J. Vicente, *Attention allocation within the abstraction hierarchy.* International Journal of Human-Computer Studies, 1998. **48**: p. 521-545.

[22] Donmez, B., L. Boyle, and J.D. Lee, *Safety implications of providing real-time feedback to distracted drivers.* Accident Analysis & Prevention, 2007. **39**(3): p. 581-590.

[23] Endsley, M.R., B. Bolte, and D.G. Jones, *Designing for situation awareness: an approach to user-centered design*. 2003, Boca Raton, FL: CRC Press, Taylor & Francis Group.

Assessing Measures of Coordination Demand Based on Interaction Durations

Michael Lewis
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA 15260 USA
ml@sis.pitt.edu

Jijun Wang
Quantum Leap Innovations, Inc.
3 Innovation Way, Suite 100
Newark, DE 19711 USA
jijunwang.cn@gmail.co

ABSTRACT
Controlling multiple robots substantially increases the complexity of the operator's task because attention must constantly be shifted among robots in order to maintain situation awareness (SA) and exert control. In the simplest case an operator controls multiple independent robots interacting with each as needed. Control performance at such tasks can be characterized by the average demand of each robot on human attention. In this paper we present several approaches to measuring, coordination demand, CD, the *added* difficulty posed by having to coordinate as well as operate multiple robots. Our initial experiment compares "equivalent" conditions with and without coordination. Two subsequent experiments attempt to manipulate and measure coordination demand directly using an extension of the Neglect Tolerance model.

Categories and Subject Descriptors
I.2.9 [**Artificial Intelligence**]: Robotics—*operator interfaces*

General Terms
Human Factors, Measurement, Experimentation

Keywords
Human-robot interaction, metrics, evaluation, multi-robot System

1. INTRODUCTION
Borrowing concepts and notation from computational complexity, control of robots by issuing waypoints, could be considered O(n) because demand increases linearly with the number of robots to be serviced. Another form of control such as designating a search region by drawing a box on a GUI (Graphical User Interface), being independent of the number of robots, would be O(1). From this perspective the most complex tasks faced in controlling teams are likely to be those that involve choosing and coordinating subgroups of robots. Simply choosing a subteam to perform a task (the iterated role assignment problem), for example, has been shown to be O(mn) [6]. The three experiments presented in this paper develop methods to assess the operator effort required to coordinate robots in tasks representative of expected application areas.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. *PerMIS'08*, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

1.1 Coordination Demand
Despite the apparent analogy between command complexity and the workload imposed by a command task there is no guarantee that human operators will experience difficulty in the same way. The performance of human-robot teams is complex and multifaceted reflecting the capabilities of the robots, the operator(s), and the quality of their interactions. Recent efforts to define common metrics for human-robot interaction [11] have favored sets of metric classes to measure the effectiveness of the system's constituents and their interactions as well as the system's overall performance. In this paper we present new measures of the demand coordination places on operators of multirobot systems and three experiments evaluating our approach and the usefulness of these measures.

Controlling multiple robots substantially increases the complexity of the operator's task because attention must constantly be shifted among robots in order to maintain situation awareness (SA) and exert control. In the simplest case an operator controls multiple independent robots interacting with each as needed. A search task in which each robot searches its own region would be of this category although minimal coordination might be required to avoid overlaps and prevent gaps in coverage. Control performance at such tasks can be characterized by the average demand of each robot on human attention [5]. Under these conditions increasing robot autonomy should allow robots to be neglected for longer periods of time making it possible for a single operator to control more robots.

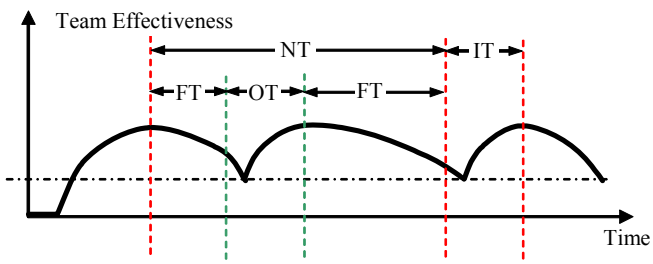
For more strongly cooperative tasks and larger teams, individual autonomy alone is unlikely to suffice. The round-robin control strategy used for controlling individual robots would force an operator to plan and predict actions needed for multiple joint activities and be highly susceptible to errors in prediction, synchronization or execution. Estimating the cost of this coordination, however, proves a difficult problem. Established methods of estimating multirobot system, MRS, control difficulty, neglect tolerance, and fan-out [5] are predicated on the independence of robots and tasks. In neglect tolerance, the period following the end of human intervention but preceding a decline in performance below a threshold is considered time during which the operator is free to perform other tasks. If the operator services other robots over this period, the measure provides an estimate of the number of robots that might be controlled. Fan-out, when measured

empirically, works from the opposite direction, adding robots and measuring performance until a plateau without further improvement is reached. Both approaches presume that operating an additional robot imposes an additive demand. These measures are particularly attractive because they are based on readily observable aspects of behavior: the time an operator is engaged controlling the robot, interaction time (IT), and the time an operator is not engaged in controlling the robot, neglect time (NT).

2. COORDINATION DEMAND

To separate coordination demand (CD) from the demands of interacting with independent robots we have extended Crandall et al.'s (2005) neglect tolerance model by introducing the notion of occupied time (OT) as illustrated in Figure 1.

The neglect tolerance model describes an operator's interaction with multiple robots as a sequence of control episodes in which an operator interacts with a robot for period IT raising its performance above some upper threshold after which the robot is neglected for the period NT until its performance deteriorates below a lower threshold when the operator must again interact with it. To accommodate



NT: Neglect Time; IT: Interaction Time;
FT: Free Time, time off task; OT: Occupied Time
IT+OT: time on task

Figure 1. Extended neglect tolerance model

dependent tasks we introduce occupied time, OT, to describe the time spent controlling other robots in order to synchronize their actions with those of the target robot. The episode depicted in Figure 1 starts just after the first robot is serviced. The ensuing free time (FT) preceding the interaction with a second dependent robot, the OT for robot-1 (that would contribute to IT for robot-2), and the FT following interaction with robot-2 but preceding the next interaction with robot-1 together constitute the neglect time for robot-1. Coordination demand, CD, is then defined as:

$$CD = 1 - \frac{\sum FT}{NT} = \frac{\sum OT}{NT} \qquad (1)$$

Where, CD for a robot is the ratio between the time required to control cooperating robots and the time still available after controlling the target robot, i.e.; the portion of a robot's free time that must be devoted to controlling cooperating robots. Note that OT_n associated with $robot_n$ is less than or equal to NT_n because OT_n covers only that portion of NT_n needed for synchronization.

Most MRS research has investigated homogeneous robot teams where additional robots provide redundant (independent) capabilities. Differences in capabilities such as mobility or payload, however, may lead to more advantageous opportunities for cooperation among heterogeneous robots. These differences among robots in roles and other characteristics affecting IT, NT, and OT introduce additional complexity to assessing CD. Where tight cooperation is required as in box-pushing, task requirements dictate both the choice of robots and the interdependence of their actions. In the more general case requirements for cooperation can be relaxed allowing the operator to choose the subteams of robots to be operated in a cooperative manner as well as the next robot to be operated. This general case of heterogeneous robots cooperating as needed characterizes the types of field applications our research is intended to support. To accommodate this more general case, the Neglect Tolerance model must be further extended to measure coordination between different robot types and for particular patterns of activity.

The resulting expression [13] measures the way in which the team's capabilities or resources are combined to accomplish the task without reference to the operation or neglect of particular robots. So, for example, it would not distinguish between a situation in which one robot of type, X, was never operated while another was used frequently from a situation in which both robots of type, X, were used more evenly. The incorporation of action patterns further extends the generality of the approach to accommodate patterns of cooperation that occur in episodes such as dependencies between loading and transporting robots. When an empty transporter arrives, its brief IT would lead to extended OTs as the loaders do their work. When the transporter has been filled the dependency would be reversed.

3. SIMULATION ENVIRONMENT

The reported experiments were performed using the USARSim robotic simulation with 2-6 simulated robots performing Urban Search and Rescue (USAR), experiments 1 & 3, or box pushing (experiment 2) tasks. USARSim is a high-fidelity simulation of USAR robots and environments developed as a research tool for the study of Human Robot Interaction (HRI) and multi-robot coordination. Validation studies showing agreement for a variety of feature extraction techniques between USARSim images and camera video are reported in [3], showing close agreement in detection of walls and associated Hough transforms for a simulated Hokuyo laser range finder [2] and close agreement in behavior between USARSim models and the robots being modeled [4,8,9,12,15].

3.1 MrCS – The Multirobot Control System

A multirobot control system (MrCS) was developed to conduct these experiments. The system was designed to be scalable to allow of control different numbers of robots, reconfigurable to accommodate different human-robot interfaces, and reusable to facilitate testing different control algorithms.

The user interface of MrCS is shown in Figure 2. The interface is reconfigurable to allow the user to resize the components or change the layout. Shown in the figure is a configuration that used in the RoboCup 2006 competition in which a single operator controls six robots. On the upper and center portions of the left-hand side are the robot list and team map panels, which show the operator an overview of the team. The destination of each of robot is displayed on the map to help the user keep track of current plans. Using this display, the operator is also able to control regional priorities by drawing rectangles on the map. On the center and lower portions of the right-hand side are the camera view and mission control panels,

which allow the operator to maintain situation awareness of an individual robot and to edit its exploration plan. On the mission panel, the map and all nearby robots and their destinations are represented to provide partial team awareness so that the operator can switch between contexts while moving control from one robot to another. The lower portion of the left-hand side is a teleoperation panel that allows the operator to teleoperate a robot.

3.2 Experiments

One approach to investigating coordination demand is to design experiments that allow comparisons between "equivalent" conditions

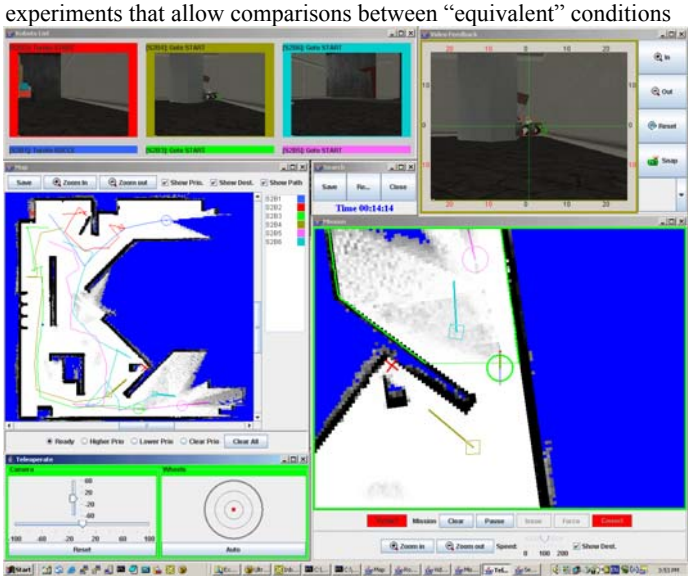


Figure 2. MrCS GUI

with and without coordination demands. The first experiment and one comparison within the third experiment follow this approach. The first experiment compares search performance between a team of autonomously coordinating robots, manually (waypoint) controlled robots, and mixed initiative teams with autonomously coordinated robots that accepted operator inputs. The impact of coordination demand was observable through the difference in performance between the manually controlled teams and the mixed initiative ones. The fully automated teams provided a control ensuring that the benefits in the mixed initiative condition were not due solely to the superior performance of the automation.

While experiment 1 examines coordination demand indirectly by comparing performance between conditions in which it was filled either manually or through automation, experiments 2 & 3 attempt to manipulate and measure coordination demand directly. In experiment 2 robots perform a box pushing task in which CD is varied by control mode and robot heterogeneity. By making the actions of each robot entirely dependent on the other, this choice of task eliminates the problem of distinguishing between interactions intended to control a target robot and those needed to coordinate with another. The third experiment attempts to manipulate coordination demand in a loosely coordinated task by varying the proximity needed to perform a joint task in two conditions and by automating coordination within subteams in the third. Because robots must cooperate in pairs and interaction for control needs to be distinguished from interaction for coordination for this task, CD is computed between robot types (equation 2) rather than directly between robots (equation 1) as done in experiment 2.

All three experiments used paid participants from the University of Pittsburgh and lasted approximately one and a half hours. All used repeated measures designs and followed a standard sequence starting with collection of demographic data. Standard instructions for the experiment were presented followed by a 10 minute training session during which the participant was allowed to practice using the MrCS. Participants then began their first trial followed by a second with a short break in between. Experiments 2 and 3 included a third trial with break. At the conclusion of the experiment participants completed a questionnaire.

4. EXPERIMENT *1*

Participants were asked to control 3 P2DX robots simulated in USARsim to search for victims in a damaged building. Each robot was equipped with a pan-tilt camera with 45 degrees Field of View (FOV) and a front laser scanner with 180 degree FOV and resolution of 1 degree. When a victim was identified, the participant marked its location on NIST Reference Test Arena, Yellow Arena [7]. Two similar testing arenas were built using the same elements with different layouts. In each arena, 14 victims were evenly distributed in the world. We added mirrors, blinds, curtains, semitransparent boards, and wire grid to add difficulty in situation perception. Bricks, pipes, a ramp, chairs, and other debris were put in the arena to challenge mobility and SA in robot control.

Presentation of mixed initiative and manual conditions were counterbalanced. Under mixed initiative, the robots analyzed their laser range data to find possible exploration paths. They cooperated with one another to choose execution paths that avoided duplicating efforts. While the robots autonomously explored the world, the operator was free to intervene with any individual robot by issuing new waypoints, teleoperating, or panning/tilting its camera. The robot returned back to auto mode once the operator's command was completed or stopped. While under manual control robots could not autonomously generate paths and there was no cooperation among robots. The operator controlled a robot by giving it a series of waypoints, directly teleoperating it, or panning/tilting its camera. As a control for the effects of autonomy on performance we conducted "full autonomy" testing as well. Because MrCS doesn't support victim recognition, based on our observation of the participants' victim identification behaviors, we defined detection to have occurred for victims that appeared on camera for at least 2 seconds and occupied at least 1/9 of the thumbnail view. Because of the high fidelity of the simulation, and the randomness of paths picked through the cooperation algorithms, robots explored different regions on every test. Additional variations in performance occurred due to mishaps such as a robot getting stuck in a corner or bumping into an obstacle causing its camera to point to the ceiling so no victims could be found. Sixteen trials were conducted in each area to collect data comparable to that obtained from human participants.

4.1 Results

All 14 participants found at least 5 of a possible 14 (36%) victims in each of the arenas. These data indicate that participants exploring less than 90% of the area consistently discovered 5 to 8 victims while those covering greater than 90% discovered between half (7) and all (14) of the victims. Within participant comparisons found wider regions were explored in mixed-initiative mode, $t(13) = 3.50$, $p < .004$, as well as a marginal advantage for mixed-initiative mode, $t(13) = 1.85$, $p = .088$, in number of victims found. Comparing with

"full autonomy", under mixed-initiative conditions two-tailed t-tests found no difference ($p = 0.58$) in the explored regions.

No difference was found between area explored in autonomous or mixed initiative searches, however, autonomously coordinating robots explored significantly, $t(44) = 4.27$, $p < .001$, more regions than under the manual control condition (see Figure 3). Participants found more victims under both mixed-initiative and manual control conditions than under full autonomy with $t(44) = 6.66$, $p < .001$, and $t(44) = 4.14$, $p < .001$, respectively (see Figure 8). The median number of victims found under full autonomy was five. Comparing the mixed-initiative with the manual control, most participants (79%) rated team autonomy as providing either significant or minor help.

Human Interactions

Participants intervened to control the robots by switching focus to an individual robot and then issuing commands. Measuring the distribution of attention among robots as the standard deviation of the total time spent with each robot, no difference ($p = .232$) was found between mixed initiative and manual control modes. However, we found that under mixed initiative, the same participant switched robots significantly more often than under manual mode ($p = .027$).

Across participants, the frequency of shifting control among robots explained a significant proportion of the variance in number of victims found for both mixed initiative, $R^2 = .54$, $F(1, 11) = 12.98$, $p = .004$, and manual, $R^2 = .37$, $F(1, 11) = 6.37$, $p < .03$, modes.

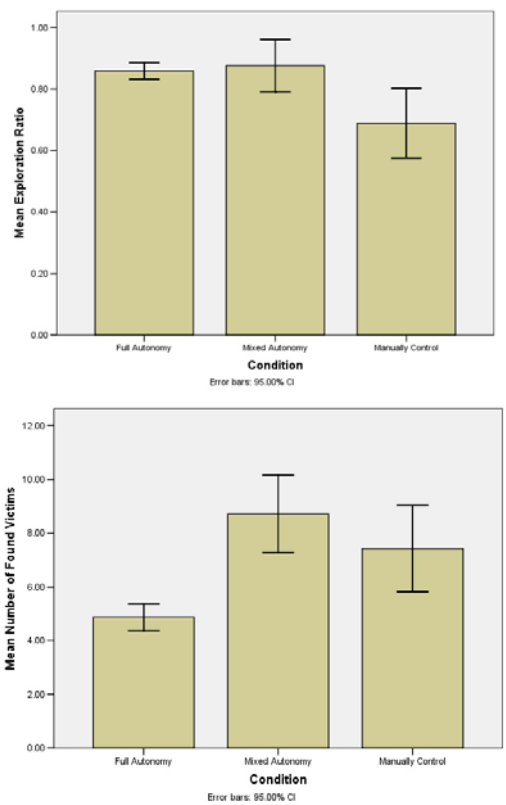


Figure 3. Victims Found and Regions Explored

In this experiment, cooperation was limited to deconfliction of plans so that robots did not re-explore the same regions or interfere with one another. The experiment found that even this limited degree of autonomous cooperation helped in the control of multiple robots. The results showed that cooperative autonomy among robots helped the operators explore more areas and find more victims. The fully autonomous control condition demonstrates that this improvement was not due solely to autonomous task performance as found in [10] but rather resulted from mixed initiative cooperation with the robotic team.

5. EXPERIMENT 2

Finding a metric for cooperation demand (CD) is difficult because there is no widely accepted standard. In this experiment, we investigated CD (as defined in Section II) by comparing performance across three conditions selected to differ substantially in their coordination demands. When an operator teleoperates the robots one by one to push the box forward, he must continuously interact with one of the robots because neglecting both would immediately stop the box. Because the task allows no free time (FT) we expect CD to be 1. However, when the user is able to issue waypoints to both robots, the operator may have FT before she must coordinate these robots again because the robots can be instructed to move simultaneously. In this case CD should be less than 1. Intermediate levels of CD should be found in comparing control of homogeneous robots with heterogeneous robots. Higher CD should be found in the heterogeneous group since the unbalanced pushes from the robots would require more frequent coordination. In the present experiment, we compared computed CDs between these three conditions.

Figure 4. Box pushing task

Figure 4 shows our experiment setting simulated in USARSim [7]. The controlled robots were either two Pioneer P2AT robots or one Pioneer P2AT and one less capable three wheeled Pioneer P2DX robot. Each robot was equipped with a GPS, a laser scanner, and a RFID reader. On the box, we mounted two RFID tags to enable the robots to sense the box's position and orientation. When a robot pushes the box, both the box and robot's orientation and speed will change. Furthermore, because of irregularities in initial conditions and accuracy of the physical simulation the robot and box are unlikely to move precisely as the operator expected. In addition, delays in receiving sensor data and executing commands were modeled presenting participants with a problem very similar to coordinating physical robots.

We introduced a simple matching task as a secondary task to allow us to estimate the FT available to the operator. Participants were

asked to perform this secondary task as possible when they were not occupied controlling a robot. Every operator action and periodic timestamped samples of the box's moving speed were recorded for computing CD .

A within subject design was used to control for individual differences in operators' control skills and ability to use the interface. To avoid having abnormal control behavior, such as a robot bypassing the box bias the CD comparison, we added safeguards to the control system to stop the robot when it tilted the box.

5.1 Participants and Procedure

14 paid participants, 18-57 years old were recruited from the University of Pittsburgh community. None had prior experience with robot control although most were frequent computer users. Participants performed three testing sessions in counterbalanced order. In two of the sessions, the participants controlled two P2AT robots using teleoperation alone or a mixture of teleoperation and waypoint control. In the third session, the participants were asked to control heterogeneous robots (one P2AT and one P2DX) using a mixture of teleoperation and waypoint control. The participants were allowed eight minutes to push the box to the destination in each session. At the conclusion of the experiment participants completed a questionnaire.

5.2 Results

Figure 5 shows a time distribution of robot control commands recorded in the experiment. As we expected no free time was recorded for robots in the teleoperation condition and the longest free times were found in controlling homogeneous robots with waypoints. The box speed shown on Figure 5 is the moving speed along the hallway that reflects the interaction effectiveness (IE) of the control mode. The IE curves in this picture show the delay effect and the frequent bumping that occurred in controlling heterogeneous robots revealing the poorest cooperation performance.

None of the 14 participants were able to perform the secondary task while teleoperating the robots. Hence, we uniformly find TAD=1 and CD=1 for both robots under this condition. Within participants comparison found that under waypoint control the team attention demand in heterogeneous robots is significantly higher than the demand in controlling homogeneous robots, $t(13)=2.213$, $p=0.045$ (Figure 5). No significant differences were found between the homogeneous P2AT robots in terms of the individual cooperation demand ($P=0.2$). Since the robots are identical, we compared the average CD of the left and right robots with the CDs measured under heterogeneous condition. Two-tailed t-test shows that when a participant controlled a P2AT robot, lower CD was required in homogeneous condition than in the heterogeneous condition, $t(13)=-2.365$. $p=0.034$. The CD required in controlling the P2DX under heterogeneous demand (CD) condition is marginally higher than the CD required in controlling homogenous P2ATs, $t(13)=-1.868$, $p=0.084$ (Figure 5). Surprisingly, no significant difference was found in CDs between controlling P2AT and P2DX under heterogeneous condition ($p=0.79$). This can be explained by the three observed robot control strategies: 1) the participant always issued new waypoints to both robots when adjusting the box's movement, therefore similar CDs were found between the robots; 2) the participant tried to give short paths to the faster robot (P2DX) to balance the different speeds of the two robots, thus we found higher CD in P2AT; 3) the participant gave the same length paths to both robots and the slower robot needed more interactions because it trended to lag behind the faster robot, so lower CD for the P2AT was found for the participant. Among the 14 participants, 5 of them (36%) showed higher CD for the P2DX contrary to our expectations.

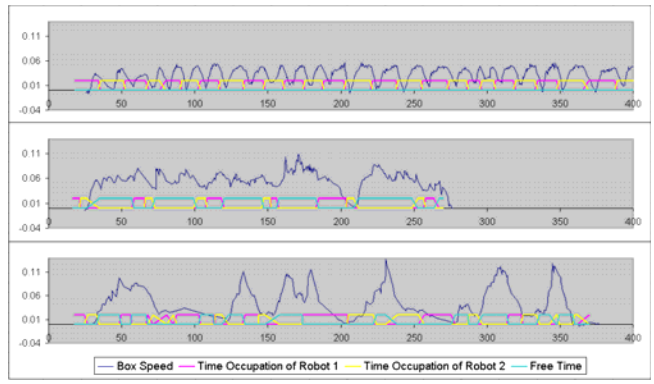


Figure 5 The time distribution curves for teleoperation (upper) and waypoint control (middle) for homogeneous robots, and waypoint control (bottom) for heterogeneous robots

6. EXPERIMENT 3

To test the usefulness of the extended CD measureme for a weakly cooperative MRS, we conducted an experiment assessing coordination demand using an Urban Search And Rescue (USAR) task requiring high human involvement and of a complexity suitable to exercise heterogeneous robot control. In the experiment, participants were asked to control *explorer* robots equipped with a laser range finder but no camera and *inspector* robots with only cameras. Finding and marking a victim required using the inspector's camera to find a victim to be marked on the map generated by the explorer. The capability of the robots and the cooperation autonomy level were used to adjust the coordination demand of the task..

6.1 Experimental design

Three simulated Pioneer P2AT robots and 3 Zergs [1], a small experimental robot were used. Each P2AT was equipped with a front laser scanner with 180 degree FOV and resolution of 1 degree. The Zerg was mounted with a pan-tilt camera with 45 degree FOV. The robots were capable of localization and able to communicate with other robots and control station. The P2AT served as an explorer to build the map while the Zerg could be used as an inspector to find victims using its camera. To accomplish the task the participant must coordinate these two types robot to ensure that when an inspector robot finds a victim, it is within a region mapped by an explorer robot so the position can be marked.

Three conditions were designed to vary the coordination demand on the operator. Under condition 1, the explorer had 20 meters detection range allowing inspector robots considerable latitude in their search. Under condition 2, scanner range was reduced to 5 meters requiring closer proximity to keep the inspector within mapped areas. Under condition 3, explorer and inspector robots were paired as subteams in which the explorer robot with a sensor range of 5 meters followed its inspector robot to map areas being searched. We hypothesized that CDs for explorer and inspector robots would be more even distributed under condition-2 (short range sensor) because explorers would need to move more frequently in response to inspectors' searches than in condition-1 in which CD should be more asymmetric with explorers exerting greater demand on inspectors.

We also hypothesized that lower CD would lead to higher team performance. Three equivalent damaged buildings were constructed from the same elements using different layouts. Each environment was a maze like building with obstacles, such as chairs, desks, cabinets, and bricks with 10 evenly distributed victims. A fourth environment was constructed for training. Figure 6 shows the simulated robots and environment. A within subjects design with counterbalanced presentation was used to compare the cooperative performance across the three conditions.

6.2 Results

Overall performance was measured by the number of victims found, the explored areas, and the participants' self-assessments. To examine cooperative behavior in finer detail, CDs were computed from logged data for each type robot under the three conditions. We compared the measured CDs between condition 1 (20 meters sensing range) and condition 2 (5 meters sensing range), as well as condition 2 and condition 3 (subteam). To further analyze the cooperation behaviors, we evaluated the total attention demand in robot control and control action pattern as well. Finally, we introduce control episodes showing how CDs can be used to identify and diagnose abnormal control behaviors.

6.2.1 *Overall performance*

Examination of data showed two participants failed to perform the task satisfactorily. One commented during debriefing that she thought she was supposed to mark inspector robots rather than victims. After removing these participants a paired t-test shows that in condition-1 (20 meters range scanner) participants explored more regions, $t(16) = 3.097$, $p = 0.007$, as well as found more victims, $t(16) = 3.364$, $p = 0.004$, than under condition-2 (short range scanner). In condition-3 (automated subteam) participants found marginally more victims, $t(16) = 1.944$, $p = 0.07$, than in condition-2 (controlled cooperation) but no difference was found for the extent of regions explored. In the posttest survey, 12 of the 19 (63%) participants reported they were able to control the robots although they had problems in handling some interface components, 6 of the 19 (32%) participants thought they used the interface very well, and only one participant reported it being hard to handle all the components on the user interface but still maintained

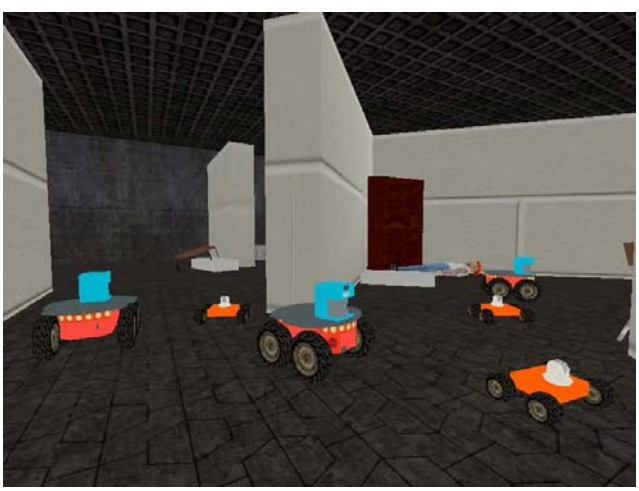


Figure 6 Scout and Explorer robots

she was able to control the robots. Most participants (74%) thought it was easier to coordinate inspectors with explorers with long range scanner. 12 of the 19 (63%) participants rated auto-cooperation between inspector and explorer (the subteam condition) as improving their performance, and 5 (26%) participants though auto-cooperation made no difference. Only 2 (11%) participants judged team autonomy to make things worse.

6.2.2 *Coordination effort*

During the experiment we logged all the control operations with timestamps. From the log file CDs were computed for each type robot according to equation 2 in section 2. Figure 7 shows a typical (IT,FT) distribution under condition 1 (20 meters sensing range) in the experiment with a calculated CD for the explorer of 0.185, a CD for the inspector of 0.06. The low CDs reflect that in trying to control 6 robots the participant ignored some robots while attending to others. The CD for explorers is roughly twice the CD for inspectors. After the participant controlled an explorer, he needed to control an inspector multiple times or multiple inspectors since the explorer has a long detection range and large FOV. In contrast, after controlling an inspector, the participant needed less effort to coordinate explorers.

Figure 8 shows the mean of measured CDs. We predicted that when the explorer has a longer detection range, operators would need to control the inspectors more frequently to cover the mapped area. Therefore a longer detection range should lead to higher CD for explorers. This was confirmed by a two tailed t-test that found higher coordination demand, $t(18) = 2.476$, $p = 0.023$, when participants controlled explorers with large (20 meters) sensing range.

We did not find a corresponding difference, $t(18)=.149$, $p=0.884$, between long and short detection range conditions for the CD for inspectors. This may have occurred because under these two conditions the inspectors have exactly the same capabilities and the difference in explorer detection range was not large enough to impact inspectors' CD for explorers. Under the subteam condition, the automatic cooperation within a subteam decreased or eliminated the coordination requirement when a participant controlled an inspector. Within participant comparisons shows that the measured CD of inspectors under this condition is significantly lower than the CD under condition 2 (independent control with 5 meters detection range), $t(18) = 6.957$, $p < 0.001$. Because the explorer always tries to automatically follow an inspector, we do not report CD of explorers in this condition.

As auxiliary parameters, we evaluated the total attention demand, i.e. the occupation rate of total interaction time in the whole control period, and the action pattern, the ratio of control times between inspector and explorer, as well. Paired t-test shows that under long sensing conditions, participants interacted with robots more times than under short sensing

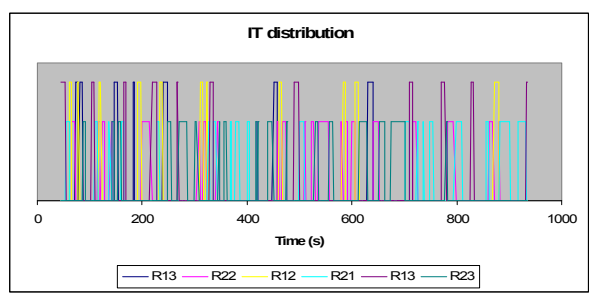


Figure 7 Typical (IT,FT) distribution (higher line indicates the interactions of explorers).

which implies that more robot interactions occurred. The mean action patterns under long and short range scanner conditions are 2.31 and 1.9 respectively. This means that with 20 and 5 meters scanning ranges, participants controlled inspectors 2.31 and 1.9 times respectively after an explorer interaction. Within participant comparisons shows that the ratio is significantly larger under long sensing condition than under short range scanner condition, $t(18) = 2.193$, $p = 0.042$.

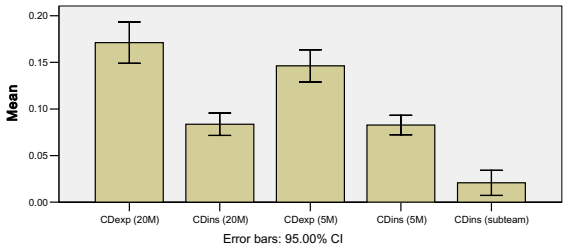


Figure 8 CDs for each robot type

6.2.3 *Analyzing Performance*

As an example of applying CDs to analyze coordination behavior, the performance over explorer CD and total attention demand under the 20 meters sensing range condition reveals three abnormal cases A, B, and C low on both CD and TAD. Associating these cases with recorded map snapshots, we observed that in case A, one robot was entangled by a desk and stuck for a long time, for case B, two robots were controlled in the first 5 minutes and afterwards ignored, and in case C, the participant ignored two inspectors throughout the entire trial.

7. CONCLUSIONS

We proposed an extended Neglect Tolerance model to allow us to evaluate coordination demand in applications where an operator must coordinate multiple robots to perform dependent tasks. Results from the first experiment that required tight coordination conformed closely to our hypotheses with the teleoperation condition producing CD=1 as predicted and heterogeneous teams exerting greater demand than homogenous ones. The CD measure proved useful in identifying abnormal control behavior revealing inefficient control by one participant through irregular time distributions and close CDs for P2ATs under homogeneous and heterogeneous conditions (0.23 and 0.22), a mistake with extended recovery time (41 sec) in another, and a shift to asatisficing strategy between homogeneous and heterogeneous conditions revealed by a drop in CD (0.17 to 0.11) in a third.

As most target applications such as construction or search and rescue require weaker cooperation among heterogeneous platforms the second experiment extended NT methodology to such conditions. Results in this more complex domain were mixed. Our findings of increased CD for long sensor range may seem counter intuitive. Our data show, however, that this effect is not substantial and provide an argument for focused metrics of this sort which measure constituents of the human-robot system directly. Moreover, this experiment also shows how CD can be used to guide us to identify and analyze aberrant control behaviors.

We anticipated a correlation between performance (found victims) and process (measured CDs) measures. However, we did not find the expected relationship in this experiment. From observation of participants during the experiment we believe that high level strategies, such as choosing areas to be searched and path planning, had a significant impact on the overall performance. The participants had few problems in learning to jointly control *explorers* and *inspector*s but needed time to figure out effective strategies for performing the task. Because CD measures control behaviors not strategies these effects were not captured. These experiments have demonstrated the utility of measuring the process of human-robot interaction as well as outcomes to diagnosing operator performance and identifying aspects of the task, particularly for multiple robots, that might benefit from automation. While our approach to measuring CD was supported in the last two experiments the third experiment suggests the need for more sophisticated measures that can take into account strategies and patterns of actions as well as their durations.

8. ACKNOWLEDGEMENTS

This work was supported in part by the Air Force Office of Scientific Research under Grant FA9550-07-1-0039 and Office of Naval Research Grant N00014-03-1-0580.

9. REFERENCES

[1] Balakirsky, S., Carpin, S., Kleiner, A., Lewis, M., Visser, A., Wang, J., and Zipara, V. Toward hetereogeneous robot teams for disaster mitigation: Results and performance metrics from RoboCup Rescue, Journal of Field Robotics, 2007, 24(11-12): p. 943-967.

[2] Carpin, S., Wang, J., Lewis, M., Birk, A., and Jacoff, A. (2005). High fidelity tools for rescue robotics: Results and perspectives, Robocup 2005 Symposium.

[3] S. Carpin, T. Stoyanov, Y. Nevatia, M. Lewis, M.,J. Wang, (2006a). Quantitative assessments of USARSim accuracy". Proceedings of PerMIS 2006

[4] Carpin, S., Lewis, M., Wang, J., Balakirsky, S., Scrapper, C. (2006b). Bridging the gap between simulation and reality in urban search and rescue. Robocup 2006: Robot Soccer World Cup X, Springer, Lecture Notes in Artificial Intelligence

[5] J. W. Crandall, M. A. Goodrich, D. R. Olsen, and C. W. Nielsen. Validating human-robot interaction schemes in multitasking environments. IEEE Transactions on Systems, Man, and Cybernetics, Part A, 35(4):438–449, 2005

[6] B. Gerkey and M. Mataric. A formal framework for the study of task allocation in multi-robot systems. International Journal of Robotics Research, 23(9):939–954, 2004.

[7] A. Jacoff, E. Messina and J. Evans, Experiences in deploying test arenas for autonomous mobile robots. In Proceedings of the 2001 Performance Metrics for Intelligent Systems (PerMIS) Workshop, Mexico City, Mexico, September 2001.

[8] M. Lewis, S. Hughes, J. Wang, M. Koes and S. Carpin, Validating USARsim for use in HRI research, Proceedings of the 49th Annual Meeting of the Human Factors and Ergonomics Society, Orlando, FL, 457-461, 2005.

[9] C. Pepper, S. Balakirsky and C. Scrapper. Robot Simulation Physics Validation, Proceedings of PerMIS'07, 2007.

[10] N. Schurr, J. Marecki, M. Tambe, P. Scerri, N. Kasinadhuni, and J. Lewis, The future of disaster response: Humans working with multiagent teams using DEFACTO, In AAAI Spring Symposium on AI Technologies for Homeland Security, 2005.

[11] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz and M. Goodrich, M. (2006). Common Metrics for Human-Robot Interaction, 2006 Human-Robot Interaction Conference, ACM, March, 2006.

[12] B. Taylor, S. Balakirsky, E. Messina and R. Quinn. Design and Validation of a Whegs Robot in USARSim, Proceedings of PerMIS'07.
[13] Wang, J. Human Control of Cooperating Robots, dissertation, University of Pittsburgh, http://etd.library.pitt.edu/ETD/available/etd -01072008-135804/unrestricted/Wang_EDC_2007-final2.pdf (accessed 7/22/2008), 2007.
[14] J. Wang and M. Lewis (2007) Human control of cooperating robot teams, 2007 Human-Robot Interaction Conference, ACM.
[15] M. Zaratti, M. Fratarcangeli and L. Iocchi. A 3D Simulator of Multiple Legged Robots based on USARSim. Robocup 2006: Robot Soccer World Cup X, Springer, LNAI, 2006.

The Gestural Joystick and the Efficacy of the Path Tortuosity Metric for Human/Robot Interaction

Richard Voyles
University of Denver
2390 S. York Street
Denver, CO USA
1 (303) 871-2481
rvoyles@du.edu

Jaewook Bae
Samsung Corp.
1st 75 W. Plumeria Drive
San Jose, CA USA
baex0021@umn.edu

Roy Godzdanker
University of Denver
2390 S. York Street
Denver, CO USA
roygod@gmail.com

ABSTRACT
This paper describes the design and evaluation of the "gestural joystick," a wearable 2-D pointing controller for mobile robots in hazardous environments that uses hand gestures. Hazardous environments, such as that of a collapsed building search, require operators to wear a significant amount of protective clothing. This protective clothing, which may include hard hats, suits, gloves, goggles, etc., reduces comfort, mobility, dexterity, load capacity, and ability to interact with conventional computer input devices. The gestural joystick, which is embedded in protective clothing, mitigates some of these impacts, but at the cost of lesser familiarity for the user and, therefore, potentially lesser performance. Effective performance metrics are required to evaluate this interface mechanism. Path tortuosity has been proposed as a performance metric for the evaluation of teleoperation of a robot, but has not been proven to be distinct from time-to-complete metrics. By injecting controlled uncertainty between the user and robot, we show, for the first time, that path tortuosity is a useful and distinct metric for the evaluation of robot teleoperation.

Categories and Subject Descriptors
B.4.2 **[Input/Output Devices]**

General Terms
Human Factors

Keywords
Human/robot interaction, path tortuosity, gestures.

1. INTRODUCTION
Urban search and rescue (USAR) has received much interest, recently, from robotics researchers around the world. Robots have a great deal to offer in that they are more expendable than humans, can be made smaller than humans to enter confined spaces, and can tolerate harsher conditions. Robots are routinely deployed during emergency response exercises and are increasingly being deployed during actual emergencies. For instance, after the September 11th attack on the World Trade Center, robot scientists contributed to USAR teams to search for trapped victims [1]. USAR robots are currently controlled exclusively via teleoperation but traditional input devices have proven inappropriate due to the protective clothing required [2]. Traditional human/computer interaction (HCI) devices are hampered by protective clothing including safety glasses, hard hats, respirators, and, most importantly, gloves. Gloves have been found to reduce operator effectiveness in many tasks [3]. Since emergency responders are required to wear heavy gloves to insulate themselves from the hazardous environment, we have been investigating ways to embed wearable human/robot interfaces into the bulky clothing itself -- exploiting it as an asset, rather then a liability.

The prototype wearable glove-based input interface was

Figure 1: Using the Gestural Joystick to operate a commercial Inuktun robot.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

presented in [13] (and shown in Figure 1) for our USAR robot, TerminatorBot [4], which is a miniature robot for core-bored inspection tasks. This interface paradigm grew from the burdens of providing an intuitive interface and controller for the TerminatorBot with conventional interaction devices (Figure 2). Conventional input devices, such as touch screen, mouse, keyboard and joystick, are cumbersome to operators wearing safety gears, even though they are small enough to carry conveniently. Conventional glove-based interfaces, such as the Power Glove, CyberGlove, MIT LED glove and others [5], have shown to be highly accurate for the recognition of various hand gestures [6], [7], [8], [9]. However, these glove interfaces require complex and fragile sensor structures and wires and are not suitable for use in rugged environments. We are developing a new paradigm of gestural joysticking for USAR tasks that reduces encumbrances, yet remains intuitive to non-technical operators. Evaluating novel paradigms is challenging, however, and requires great care.

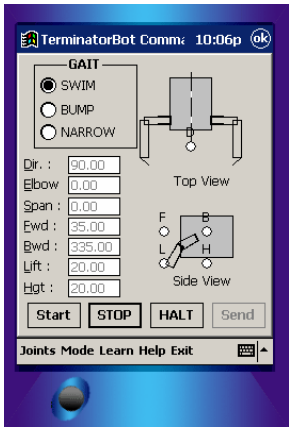


Figure 2: The initial PDA-based interface for the TerminatorBot is too confusing for non-technical users.

1.1 HRI Performance Metrics

In the evaluation of user efficacy, metrics are necessary to measure how well a rescuer can control the robot under specified tasks [10], [11]. The development of metrics for USAR robots is still in its infancy, though the standard "time-to-complete" metric is commonplace. Steinfeld [11] tried to generalize common metrics for task-oriented HRI and built a standardized framework in HRI. However, the framework is difficult to apply to such a specific domain as that represented by USAR and development is still in progress.

The time required to complete a specified, closed-ended task is a common metric for the efficacy of user interfaces. This metric is simple to implement, it results in a single, quantitative value that is easily compared, and it applies to many types of tasks and interfaces. If one interface requires the user to spend more time in completing a standardized, relevant task than another interface, it is reasonable to conclude that the former interface is inferior to the latter.

More recently, researchers have begun to examine other metrics that relate more specifically to the robot teleoperation task. It has been suggested that the degree of smoothness of resulting motion of a human-directed robot control task is a good indicator of the quality of the human/robot interface [10]. Voshell et al termed this metric "path tortuosity" [10] and quantified it through fractal dimension [12]. They used the 2-D projection of a mobile robot's motion on the ground to show that path tortuosity correlated well with user subjective opinions and time-to-complete. However, to our knowledge, nobody has clearly demonstrated that path tortuosity is actually distinct from time-to-complete. Until now, the question has been left unanswered as to whether or not path tortuosity adds information or merely confirms time-to-complete measurements (which are easier to gather). In fact, path tortuosity often provides similar results to the easier to measure time-to-complete metric.

In this paper, we employ two different metrics to evaluate the human/robot interface for a specific task: time-to-complete and path tortuosity. In a test of non-technical users, we introduced varying degrees of uncertainty between the operator console and the robot, unknown to the operator. In effect, we were varying the quality of control they had over the robot in a controlled way, independent of the interface device. Our results provide evidence that path tortuosity is distinct from time-to-complete and provides a complimentary measure of efficacy in human/robot interaction.

2. SYSTEM CONFIGURATION

The gestural input system (see Figure 3), which we call "WRIST" for Wearable Responder Interface for Search Tasks, forgoes the complex gestural capabilities of other glove interfaces, but also eliminates the encumbrances of these other interfaces such as wires, batteries, and active components. By simplifying the interface to only the pointing task, a robust, wearable, wire-free solution results that is suitable for hazardous environments with none of the difficulties of a complex alphabet to memorize. Based on permanent magnets and sensitive Giant Magnetoresistive (GMR) detectors, analog forward/backward and right/left gestures are encoded into commands that can be relayed to a robot, personal computer, or other device requiring pointing information.

Calibration of the sensitive but highly nonlinear GMR detectors is described in [15]. User evaluation tests are presented below for a simulated search and rescue task to compare controllability to a conventional joystick. These

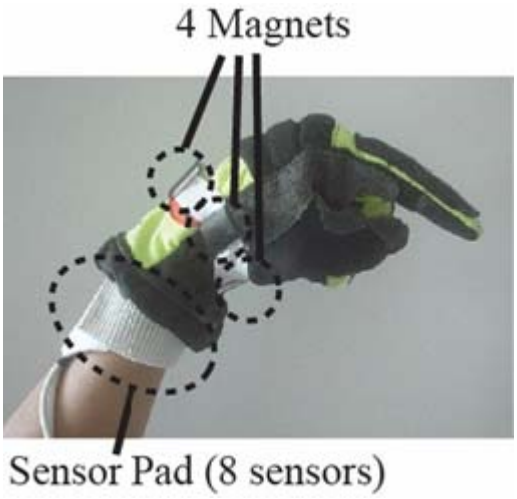


Figure 3: The WRIST system consists of an array of magnets on the glove, an array of sensors on the sleeve, and a data processor.

tests employ not only the standard metric of time-to-complete, but add a metric for path tortuosity based on the fractal dimension of the robot's traced path. Possibly for the first time, we demonstrate the independence of the path tortuosity metric from the time-to-complete metric for pointing controllability.

WRIST provides unencumbered wearability that can be easily equipped on many styles of traditional gloves with minimal modifications. Moreover, the hardware structure of the system should permit intuitive and robust operation.

The embedded rare-earth permanent magnets (Figure 4) have relatively small size and consume no power. The sensor pad is integrated with eight GMR sensors and is worn on the wrist or embedded in the fringe of the sleeve. The magnets and sensors interact with no interconnecting wires allowing the glove to be easily removed.

The signal strength of each magnetic sensor changes with wrist motion. The redundant sensor signals are continually monitored by the CPU where they are translated into the four directional components as described later. The sensor pad contains 8 GMR sensors (AA004, NVE Co., Eden Prairie, Minnesota, USA) and 8 op-amps on a flexible printed circuit board (PCB). The sensor pad can be completely wrapped around the wrist. The PCB is about the thickness of a piece of paper making it easy to embed into a sleeve.

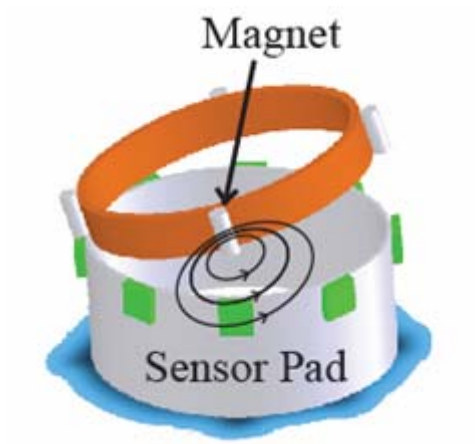


Figure 4: Configuration of the magnets and GMR sensors.

The CPU module is built around the Atmel Corporation's 8-bit AtMega128 RISC microprocessor. Its Analog/Digital converters can process eight analog signals from the eight magnetic sensors. Each Analog/Digital converts a signal into a 10-bit digital output. The glove of WRIST includes nothing but four rare-earth magnets. Therefore, it is washable and virtually unbreakable.

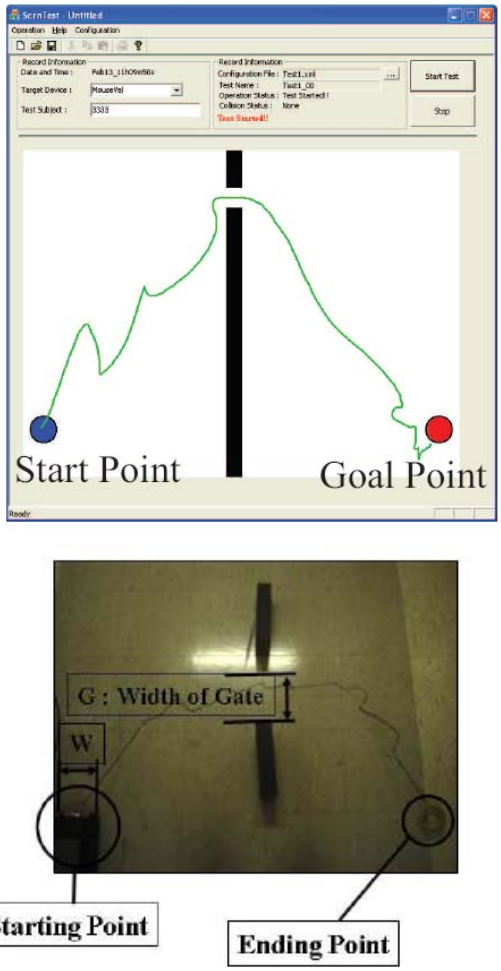


Figure 5: Usability tests based on the NIST gate model. Top is a virtual robot driven through a single gate to a goal position. Bottom is the trace of a physical Inuktun robot driven through a single gate to a goal position.

3. USABILITY EVALUATION

A variety of usability tests were developed to test operator performance with WRIST in comparison to a conventional joystick and a mouse in two different modes (position and velocity mode). Fitz' Law tests were performed as well as physical mobility tests and virtual and physical robot control tests based on the NIST "gate" model. The focus of this paper will be on the virtual and physical robot control tests.

In the NIST gate model, a single robot is teleoperated to navigate through a narrow gate, as shown in Figure 5. While the tests by NIST have used multiple gates, we use only one, for simplicity.

While the path tortuosity metric has been proposed before [10], it has not been proven to provide information that is independent of the time-to-complete metric. Although it is intuitive to use such a metric, many tests show close correspondence between the two. For example, in Figure 6, we show the results of five separate trials for the WRIST system and a conventional joystick. (Time to complete is on the top while fractal dimension is on the bottom.) Both metrics trend downward as gap width increases, meaning that the quality of control increases (task time gets shorter and the path gets smoother) for each interface device as the course gets easier. This is the expected result and shows that path tortuosity is consistent with intuition and time-to-complete, but it does not demonstrate path tortuosity is actually providing new information.

3.1 Verification of the Uniqueness of Path Tortuosity

In an effort to validate the independence of the metric, we deliberately introduced random error into the control software of certain test subjects without the test subject's knowledge. This error manifested itself in the form of a random angular offset in the motion direction of the cursor commanded by the user. In effect, this error controlled the degree of inaccuracy with which the user could control the cursor for the devices tested.

The distribution of the angular error was Gaussian and we made no modification to the commanded magnitude. For our operator pool, we used over one hundred non-engineering undergraduate students in a laboratory class that introduces them to technology. The task was explained to them and they were given one practice trial with each of five different interface devices. The interface devices included the WRIST system, a joystick, a trackball, a mouse in position mode and a mouse in velocity mode. Only a small subset of the data is presented here.

In comparing these "perturbed" runs to normal runs across several users, we found that time-to-complete was not reliably correlated with the actual inaccuracy of the trial, which is assumed to be dictated by the introduced error. The path tortuosity, as computed through Fractal

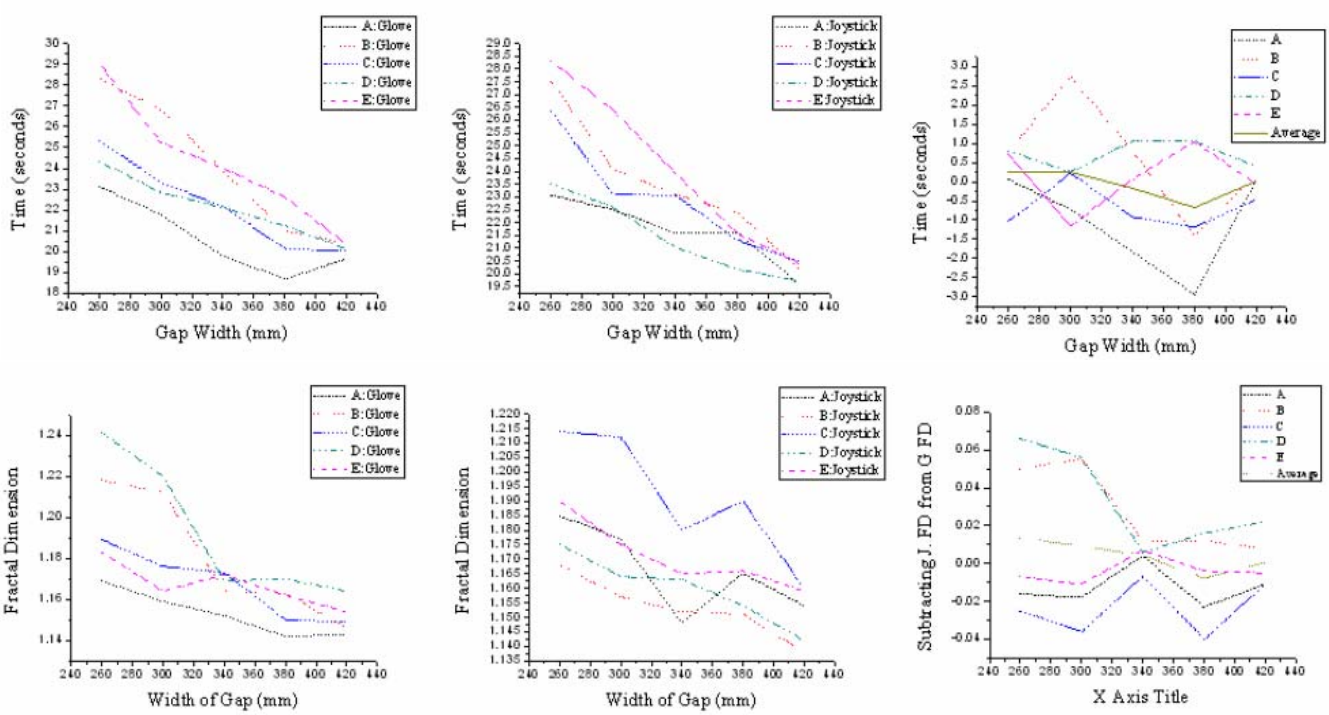
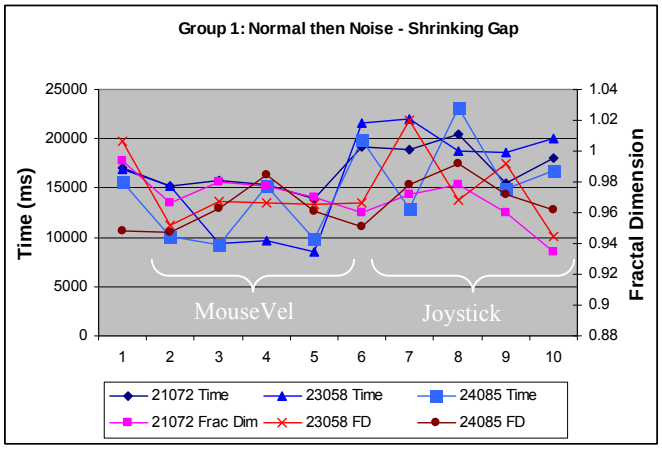


Figure 6: Top set of plots show the time-to-complete metric for five trials each of the WRIST and a conventional joystick as gate width increases. The bottom set of plots show path tortuosity for the same devices and gate tasks. The two metrics trend the

dimension, was a more reliable predictor of the introduced error.

The data are difficult to interpret as a pronounced learning effect is evident in the results (Figure 7). The subjects are not particularly tech savvy so many of the interface devices were new to them and they were only given one practice trial. As they moved through the five trials with each device, negative slopes are apparent as time-to-complete and path tortuosity both decrease with practice. Furthermore, the gate width was varied for many of the test subjects, which varies the difficulty of the task. This introduces a negative or positive slope depending on whether the width is increasing or decreasing.



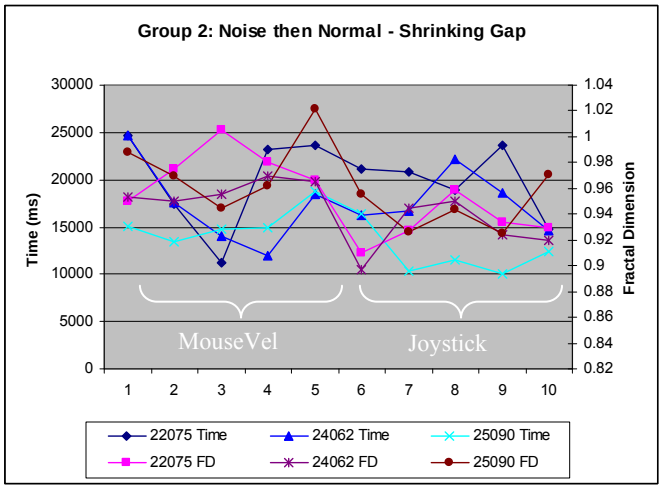


Figure 7: Group 1, top, and Group 2, bottom, indicating the trend over two devices and noise conditions for both time-to-complete metric and path tortuosity. Test subjects are denoted by number.

To counteract these effects, we look at complementary test suites that reverse the effects. For example, we took two groups of test subjects and had them use devices in the same order; in this case, mouse in velocity mode followed by the joystick. (Five trials with each device are completed.) Over the five trials, the gate width is gradually reduced, making the task more difficult and offsetting the learning effect. The complementary tests groups had noise injected at different times, however. By examining the slope and intercept of the trends, we can spot significant artifacts in the metrics.

Group 1 had noise introduced into the mouse tasks but not the joystick tasks while group 2 had noise introduced into the joystick tasks but not the mouse tasks. With the offsetting effects of learning and increasing difficulty, the slopes are near zero.

With the slope near zero, we can focus on the intercept to see which is higher or lower in relation to the device and noise. The intercepts are tabulated below and we see that path tortuosity is a more reliable predictor of the introduced noise than time-to-complete. (Erroneous cases are highlighted in Table 1.)

Table 1: Regression statistics of groups of user data.

Group	Device/Noise	Time Intercept	Fractal Dimension Intercept
Group 1	Mouse/No	17310	0.988
Group 1	Joystick/Yes	20150	0.979
Group 1	Mouse/No	18680	0.992
Group 1	Joystick/Yes	22220	1.000
Group 1	Mouse/No	13960	0.942
Group 1	Joystick/Yes	18790	0.966
Group 2	Mouse/Yes	18950	0.964
Group 2	Joystick/No	22830	0.919
Group 2	Mouse/Yes	22810	0.945
Group 2	Joystick/No	18070	0.919
Group 2	Mouse/Yes	12790	0.959
Group 2	Joystick/No	14620	0.938

To further demonstrate the independence of path tortuosity, we gathered another set consisting of two groups of data: a "Noisy" group of 6 test subjects with random error introduced and a "Normal" group of 22 test subjects. Since it is meaningless to directly compare the performance of individual test subjects, we exhaustively compared the average performance (both time-to-complete and path tortuosity) of every subgroup of four test subjects from each group. This resulted in 15 Noisy subgroups ($^{6}C_4$ – 6-choose-4) and 7315 Normal subgroups ($^{22}C_4$).

Comparing every Noisy subgroup to every other Normal subgroup involved over 100,000 comparisons. Of these comparisons, the fractal dimension of the noisy subgroup was greater than the fractal dimension of the normal

subgroup 99.5% of the time while the time-to-completion of the noisy subgroup was greater than the time-to-completion of the normal subgroup 49.4% of the time (essentially chance).

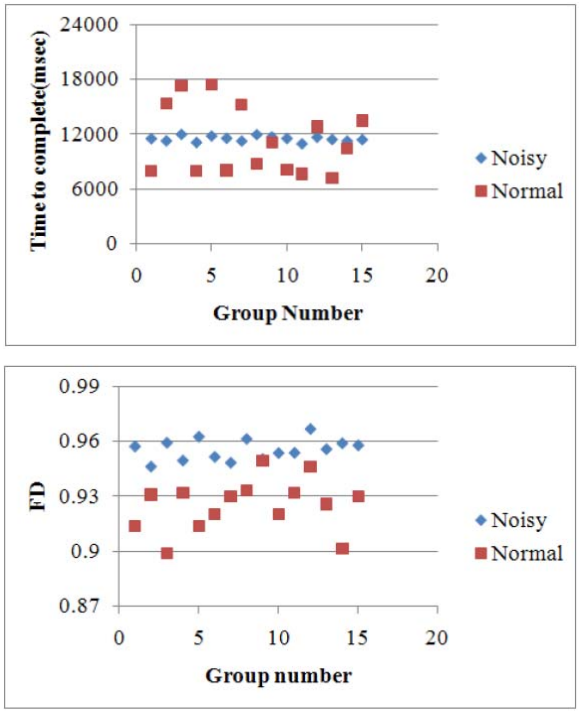


Figure 8: Random sampling of group comparisons with time-to-complete, top, and path tortuosity, bottom, for robot teleoperation tasks with and without noise introduced.

4. CONCLUSION

A large number of tests have been conducted on both technical and non-technical users to determine the usefulness of the WRIST system in human/robot interaction. Over 120 users were tested with various input devices, including the WRIST system, a mouse, a joystick, and a trackball on both simulated robots and physical robots. We gathered both quantitative results and qualitative results. Our quantitative user test results indicate users were not as proficient, in either time-to-complete or path tortuosity, in using the WRIST as compared to the joystick or mouse. This is not surprising as nearly all users had some familiarity with these devices. In fact, learning rates were extrapolated from the data and the WRIST, indeed, shows a much higher improvement rate over the trials. In qualitative evaluations from non-technical users who had little prior exposure to use of a joystick, users preferred the WRIST system for its more intuitive nature. Coupling this result with the portability of the device and other unmeasured benefits of its wire-free nature leads us to believe it will be a successful interface when fully developed.

The discrimination between the time-to-complete and path tortuosity metrics is highly meaningful. It seems intuitive that the two metrics would be distinct, but we believe this is the first result that demonstrates they are, in fact, distinct and separably valuable.

5. ACKNOWLEDGMENTS

Partial funding for this work was provided by the NSF Safety, Security and Rescue Research Center and by NSF through grant 0719306. Thanks to Robin Murphy and the NSF R4 program for the loan of the Inuktun robot.

6. REFERENCES

[1] R. Murphy, “Human-robot interaction in rescue robotics,” IEEE Systems, Man and Cybernetics, vol. 34, no. 2, May 2004.

[2] K. Newman, “The open interface: Beyond keyboards and mice,” e.nz Magazine, May 2002.

[3] G. I. of Technology, “Real-time cooperative behavior for tactical mobile robot teams,” Georgia Tech College of Computing and Georgia tech research Institute: Final Report, vol. A003, 2001.

[4] R. M. Voyles, A. C. Larson, M. Lapoint, and J. Bae, “Core-bored search and rescue applications for an agile limbed robot,” in Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 1, 2004, pp. 58–66.

[5] D. Sturman and D. Zeltzer, “A survey of glove-based input,” in IEEE Computer graphics and Applications, 1994.

[6] G. Kessler, L. Hodges, and N. Walker, “Evaluation of the cyberglove as a whole-hand input device,” in ACM Transactions on Computer-Human Interaction, vol. 2, no. 4, 1995, pp. 263–283.

[7] B. H. Thomas and W. Piekarski, “Glove based user interaction techniques for augmented reality in an outdoor environment,” Virtual Reality, vol. 6, pp. 167–180, 2002.

[8] B. Thomas and W. Piekarski, “Glove based user interaction techniques for augmented reality in an outdoor environment,” Virtual Reality, vol. 6, no. 3, pp. 167–180, 2002.

[9] R. Voyles and P. Khosla, “A multi-agent system for programming robots by human demonstration,” Integrated Computer-Aided Engineering, vol. 8, no. 1, pp. 59–67, 2001.

[10] M. Voshell, D. Woods, and F. Phillips, “Overcoming the keyhole in human-robot coordination: simulation and evaluation,” in Proceedings of the Human Factors and Ergonomics Society, pp. 26–30, 2005.

[11] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich, “Common metrics for human-robot interaction,” in Human-Robot Interaction Conference, March 2006.

[12] Mandelbrot, B., “How Long is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension,” Science, v. 156, n. 3775, pp. 636-638, 1967.

[13] J. Bae and R. Voyles, “Wearable joystick for gloves-on human/computer interaction,” in Proceedings of the SPIE Defense and Security Symposium, Orlando, FL, April 2006.

[14] M. Goldstein, D. Chincholle, and M. Backstrom, “Assessing two new wearable input paradigms: The finger-joint-gesture palm-keypad glove and the invisible phone clock.” Personal and Ubiquitous Computing, vol. 4, no. 2/3, 2000.

[15] J. Bae, A. Larson, R.M. Voyles, R. Godzdanker, J. Pearce, "Development and User Testing of the Gestural Joystick for Gloves-On Hazardous Environments," in Proc. of the IEEE Intl. Symp. on Robot & Human Interactive Communication, Jeju, Korea, Aug. 2007, pp. 1096-1101.

Modeling of Thoughtful Behavior with Dynamic Expert System

Vadim Stefanuk
Institute for Information Transmission Problems,
Russian Academy of Science,
Bolshoy Karetny per. 19
127051 Moscow, Russia
7 495 6504679
stefanuk@iitp.ru

ABSTRACT
The concept of Dynamic Expert System, used in this paper to cope with changing data and knowledge, was introduced at the start of 1990th.

Implemented via a quasi-static approach, the Dynamic Expert System presents a certain step forward in the modeling of intelligent behavior, which reminds that of human beings. There are two reasons for such a claim. First, in general case Dynamic Expert System never interrupts its activity, occasionally interrogating the user if it suspects that some of the previously entered data are obsolete or become unsatisfactory in some other way. This mode reminds the behavior of an alive, as it was discussed in a number of our previous publications.

The present paper considers a second important reason. It shows that Dynamic Expert System in certain sense resolves the Frame Problems of AI, demonstrating the level of reasoning, which is very close to that of human beings. A simple example of a rule-based traffic light control illustrates this property.

Keywords
Dynamic Expert System, Frame Problem, Artificial alive creature, Thoughtful Behavior, traffic lights.

1. INTRODUCTION
Recently there was a new wave of interest to the AI systems working in realistic surroundings as the ideas of data accumulation and the knowledge evolution were recognized as their important characteristics. In this paper it is proposed to treat corresponding problems via the concept of Dynamic Expert System. This term and the concept were introduced at the start of 1990th when a quasi-static approach to the system dynamics has been proposed [5]. The quasi-static approach provides a smooth combination of static and change, when during certain time the knowledge used considered to be stable allowing for an inference to be made. However over larger time intervals some drastic modifications of the knowledge are allowed.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1 $5.00.

The implemented system found its first application in the seismology domain [5-7]. The Dynamic Expert System in this application never stops reacting to any new information on the various seismic predecessors arriving to the system. From time to time the system itself interrogates the user if it suspects that previously entered data are obsolete. In this respect the system behaves itself as an alive. In the present paper we will try to develop further this observation and to show that its behavior may be considered also as a thoughtful one.

The results of this paper supplement the paper [8], where it was shown that the concept of Dynamic Expert System is tightly related to the area of Empirical Bayesian Schemes and to the modern area of Modeling Field Theory Dynamic [4].

2. CLASSIC FRAME PROBLEM
The proposed concept of Dynamic Expert System is not only useful in applications as described in [6]. There are some grounds to expect it may be a new step in the development of the whole area of Artificial Intelligence as it leads to the design of an artifact system, which is alive and has the intellectual behavior in a real sense of the word.

The reason for so ambitious statement lies in the fact that our Dynamic Expert System resolves the Frame Problems first mentioned by J.McCarthy and P.Heyes in [3], which was considered to be the main obstacle on the way of further progress in Artificial Intelligence.

The original formulation of the problem is the following one. Let in the situational calculus [3] it is asserted that

Holds (Situation_0, Black (Block A))

Holds (Situation_0, Separate (Block A))

Holds (Situation_0, Separate (Block B))

Is the following statement correct?

Holds (Result (Put-on (Block A, Block B), Situation_O),

Black (Block A))

In other words if one puts the Block A on the Block B, is it true that the color of the Block A would remain black? It turned out that rather obvious for a human being positive answer is not automatically obtainable in the Situation Calculus Certain additional statements have to be made to arrive to such an answer, like the following one [3]:

Holds (Situation_O, Black(Block A))

à Result (Put_On(Block A, Block B), Situation_0), Black(Block A)) ,

which explicitly shows that the Block A does not change its color.

The Frame Problem became a tough problem for AI due to the fact that the number of additional frame statements may be enormous one to be realistic in applications. There were many attempts to overcome the problem by the proper choice of logic formalisms, which have created actually some other equivalent problems (see the information on WWW).

One practical solution was to use 'a natural law of inertia' or 'do not touch sleeping dog' which roughly admits that things do not change if the change was not mentioned explicitly in the system. The Frame Problem was solved to some extend in the system STRIPS to control a robot, where the objects to be moved were real things shifted around research laboratory, which obviously 'do not change their colors'.

The other important side of the Frame Problem was stressed by Marvin Minsky: it is not clear at all how to find and remove from the memory of a robot all the consequences of an action if the situation changed and this action was not performed (see also [1]).

Such situations are quite natural for a human, who has to reconsider his actions many times a day when he comes across with new situations. He understands clearly, what has preserved and what changed in transition form one situation to another. It looks like the problem does not really exist for a human being.

It must be especially difficult for an intelligent computer controlled robot due to the fact that even if some data and knowledge were fixed initially, the inferences from those may combine with the inferences obtained from the knowledge, which is modifiable, and the total result of the inferences becomes unclear.

3. PHILOSOPHICAL ASPECTS

For the purpose of our paper it is important to note that the Frame Problem obtained also a philosophical interpretation, which demonstrates that this problem distinguishes the behavior of an alive from that of an ordinary computer program. In other words, if one finds an adequate solution for the Frame Problem he probably deals with an artificial creature, which may be referred to as an alive.

Dannett in [1] formulated an Epistemological Frame Problem, how 'a cognitive creature … with many beliefs about the world" can update those beliefs when it performs an act so that they remain "roughly faithful to the world" '?

Fodor puts a similar question [2]: 'How does the machine's program determine which beliefs the robot ought to re-evaluate given that it has embarked upon some or other course of action?'

Almost the same is said in the publication [9], where an importance of limited time and adaptation to changes are stressed: 'A more intricate problem in the notion of adaptiveness, how can we expect to reuse results from earlier proofs when the circumstances change? The author plans to investigate these questions for the particular application domain of intelligent help'.

Concluding the paragraph we have to say that the logic is 'quite correct' at this point: the color of a block may change when the situation changes, for example due to a subtle change in the illumination or something similar. The Frame Problem should not be resolved by changing the logical formalism used. It is the problem that must be recognized and taken into account in the design of an advanced computer system. For us it is important to note that the system taking this problem into consideration should be recognized as a new step in creation of artificial intellectual entity, similar to the intelligent things existing in Nature.

In our Dynamic Expert System some modification of any property is possible only when a certain rule in the Knowledge Base admits such a modification or the rule being brought into the system by the user from outside. Then, it is the responsibility of our system to modify accordingly the believes, which currently exist in the system. Thus, the Dynamic Expert System resolves the Frame Problem in a way it is done by a human.

4. QUASI-STATIC ARCHITECTURE

The quasi-static mode we described in [5,6] and elsewhere. It results from a certain modification of our static ES ZNATOK and a recursive use of this shell during one session of interaction with the user. At each recursive step practically all the data and all involved rules may be modified. Of course, if there are reasons for such modifications in the original KB or they are forced by the interaction with the system user.

Our quasi-static architecture resolves the classic problem – how to remove all the consequences, which follows from a certain fact if the fact is not true any more. At least it is achieved in "a pure ES shell" with only pure rules [6].

When the static shell ZNATOK is used in the dynamic mode, the Knowledge Engineer should take special care for the possible side effects in order to obtain an appropriate behavior of the system. This requires from the Knowledge Engineer a deep understanding both system architecture and the problem domain.

By the way, as it was mentioned in [6], the total number of possible side effects is doubled in a quasi-static mode with respect to the static mode. For each relative side effect, created by the form F1 for form F2 in static case, in the dynamic mode one has to consider the possibility to have a side effect of the form F2 for the Form F1. However, the side effects are the most powerful way to achieve a flexibility and effectiveness of the system performance and we want to keep them.

One of the important tasks for the Knowledge Engineer is to achieve an *operational coherence* when the side effects are used. Partially, operational coherence may be tested if one tries the shell in a number of static environments.

To achieve the operational coherence and the effectiveness the types for the system attributes were introduced in the system such as *dynamic*, *static*, *fuzzy*, and etc. The dynamic type restricts the number of attributes that may be reconsidered. The fuzzy type reduces the required computational efforts by distinguishing ordinary attributes from the fuzzy ones [7].

5. TRAFFIC LIGHTS CONTROL

Our Dynamic Expert System, which is originally "tuned to" modification of the data and rules of the system on the run, naturally solves the Frame Problem in this application.

In this application the data considered are to be either of static or dynamic type. Based on the Dynamic Expert System Shell "Seismo" [5,7] the traffic system is designed in such a way that new dynamic value automatically causes a proper change to all other values involved in the inference. Due to this and other important properties we avoid the difficulty that was referred to as the Frame Problem.

This property will be illustrated here with an example of the traffic light control, which looks a bit more practical in comparison to the Block Word models of AI.

In case of pedestrian controlled traffic light, if the variable *color* is announced to be static one the light would behave itself incorrectly. Either it would not change the color, or the color would change itself accordance with a rule stored in the Knowledge Base of the Dynamic Expert System. If the type of the variable *color* were said to be dynamic one, the pedestrian traffic light would work quite correctly.

Indeed, normally the pedestrian controlled traffic light will be in the state of 'green' towards approaching cars, thus creating no obstacle for the traffic. It changes its color for the yellow and then for the red one, if a person wants to cross the street and pushes the corresponding button.

6. CONCLUSION

The Dynamic Expert System presents a certain step forward in the modeling of intelligent behavior, which reminds that of human beings. There are several reasons for such a claim. In the present paper it is shown that Dynamic Expert System resolves the Frame Problems of AI in a certain sense, thus demonstrating the level of reasoning, which is very close to that of a human. A simple example of a rule-based traffic light control illustrates this property. In this example the Dynamics of our system becomes quite essential.

7. ACKNOWLEDGEMENTS

This research was partially supported by the Russian Fond for Basic Research and with the Fundamental research program by Presidium of Russian Academy of Science.

8. REFERENCES

[1] Dennett, D. Brainstorms. Cambridge, MIT Press, 1978.

[2] Fodor, J.A. The Modularity of Mind. Cambridge, MIT Press, 1983.

[3] McCarthy, J. & Hayes, P.J. "Some Philosophical Problems from the Standpoint of Artificial Intelligence", in Machine Intelligence 4, ed. D.Michie and B.Meltzer, Edinburgh: Edinburgh University Press, pp. 463-502, 1969.

[4] Perlovsky L.I., Neural networks and intellect: Using model based concepts, Oxford University Press, New York (2001).

[5] Stefanuk V.L. The Behavior of Quasi-static Shell in Changing Fuzzy Environment, The Proceedings of the IY National Artificial Intelligence conference KII'94, Rybinsk, V.1, pp.199-203, 1994, (in Russian)

[6] Stefanuk V.L., Dynamic expert systems. KYBERNETES, The International Journal of Systems & Cybernetics. V.29, Issue 5, pp. 702-709, MCB University Press: 2000.

[7] Stefanuk V., Intellectual Performance using Dynamical Expert Knowledge in Seismic Environment. Proceedings of the Workshop "Performance Metrics for Intelligent Systems (PerMIS'06)", http://www.isd.mel.nist.gov/PerMIS_2006/proceedings/PerMIS_Program.pdf

[8] Stefanuk V.. Dynamic Knowledge Processing – Practical Approach. In Proceedings of 2007 International Conference on Integration of Knowledge Intensive Multi-Agent Systems (KIMAS'07), Copyright © IEEE, Prining House, Inc., Stoughton: 2007, pp. 22-27.

[9] Wern, A., Reactive Abduction. Proceedings of ECAI-92, pp.159-163.

Ontological Perspectives for Autonomy Performance

Hui-Min Huang, Elena Messina, Tsai-Hong Hong, Craig Schlenoff
National Institute of Standards and Technology (NIST)
100 Bureau Drive, MS 8230 Gaithersburg, Maryland 20899 USA
+1.301.975. { 3427, 3510, 3444, 3456 }
{ hui-min.huang, elena.messina, tsai-hong.hong, craig.schlenoff }.nist.gov

ABSTRACT

Goals must be assigned for any unmanned system's (UMS) operation before the system's autonomous performance can be measured. This paper reports the early results of construction of an ontology for mission goals that could serve as a template for stating the goal. The Autonomy Levels for Unmanned Systems (ALFUS) Framework is a key element in the ontology. In other words, we design the goal ontology in terms of mission, environment, and operator interaction aspects. We also leverage a collection of related efforts to further evolve the goal ontology, including the Urban Search and Rescue (US&R) robots requirements set, the Perception System for Dynamic Manufacturing, and an extension to the NIST-participated Defense Advanced Research Projects Agency (DARPA) Spoken Language Communication and Translation System for Tactical Use (TRANSTAC) project.

Categories and Subject Descriptors

J.2 [physical sciences and engineering] *ontology, unmanned systems performance*

General Terms

Measurement, Performance, Design, Human Factors, Standardization, Verification

Keywords

ALFUS, communication, environment, goal, manufacturing, metrics, mobility, ontology, sensor, terminology, urban search and rescue (US&R)

1. INTRODUCTION

Unmanned systems (UMSs) have been playing increasingly important roles in many aspects of the society. In a broad sense, UMS includes the unmanned vehicles that aid military operations, the robots that aid bomb disposal tasks, the robots that help the search and rescue operations, and the automation machine systems that perform the part manufacturing and assembly tasks. It is vital for practitioners to be able to model the systems and measure their performances.

We propose to develop an ontology for a generic UMS. The ontology should provide a comprehensive and structural organization for the UMS knowledge, including the hardware, software, interfaces, performance, etc. Figure 1 provides an overview. All the aspects must be defined and formally related.

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
ACM ISBN 978-1-60558-293-1/08/08.

Typically, a UMS is to perform goals that the operator assigns. Many issues need to be addressed, including how to acquire a UMS that fits the needs, how to state the goals to allow precise execution, how to evaluate the performance, etc. We envision the ontology to be able to provide all these features for practitioners.

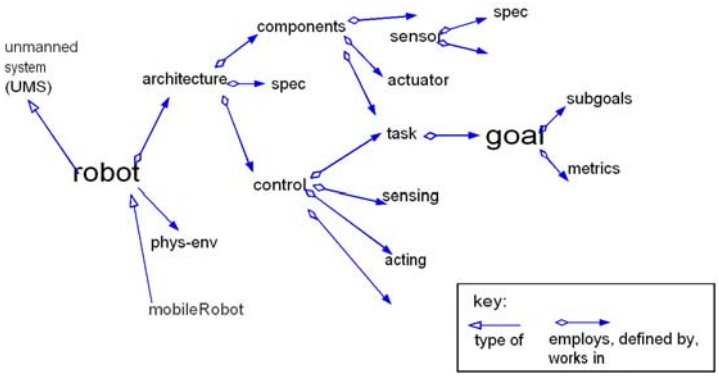


Figure 1: High Level View of the Ontology

The development effort is underway. This paper describes the first increment of results from this effort.

1.1 Scope of the Ontology

This paper describes the development of a particular subset of the ontology, the UMS goal. Given the fact that UMSs may be commanded to perform different types of tasks, it is important to devise a sound approach to knowledge acquisition and organization. Our plan is to develop a structure such that the generic aspects can be instantiated and applied to different applications. We also describe how performance metrics are included in the ontology and are measured by comparing the metrics and the goal.

1.2 Aspects of the Ontology

In the area of information technology, we define ontology as a rigorous or formal model that encompasses a collection of concepts and their relationships for the topic of interest [1, 2, 3, 4, 5, 6, 7]. As such, we propose that ontology should cover the following aspects:

- Terminology and definitions
- Requirements/capability attributes
- Performance metrics

- Engineering specifications
- Standards

In other words, to develop an ontology for a topic, the engineer should:

- identify the key concepts from the terms and definitions used in the domain,
- identify the user, capability, and performance requirements for the program, and
- utilize engineering specifications and applicable standards for the UMS and its subsystems and components.

In this first increment for development of a UMS ontology, we begin with the terminology, metrics, and requirements aspects and leave the others for future development.

1.3 Performance Metrics—ALFUS Framework Overview

The performance of UMS must be measured in terms of the assigned goals. It is desirable to have a generic and standard structure for stating the goal. There are multiple concerns in the goal statement, including operation (or mission), environment, and operator interactions. These lead us to apply the Autonomy Levels for Unmanned Systems (ALFUS) Framework that NIST has been developing [8]. The ALFUS Framework describes that the autonomy of a UMS can be characterized with Contextual Autonomous Capability (CAC). The CAC model is composed of the following three aspects (or axes), Mission Complexity (MC), Environmental Complexity (EC), and Human Independence (HI), as shown in Figure 2. Each axis is further decomposed into a set of performance metrics.

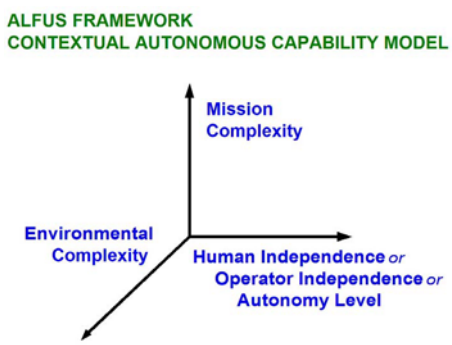


Figure 2: The ALFUS Framework

The ALFUS Framework was developed originally by the ad hoc ALFUS Working Group (WG) that was lead by NIST. The group has later joined the Society for Automotive Engineers (SAE), a standards development organization, as the UMS Performance Measures Subcommittee.

During the first ALFUS workshop, it was determined that the first objective of the group should be terminology [9]. This is because the ALFUS WG decided to take a definition-based approach. The fundamental terms, including autonomy and UMS were defined. The key words and their relationships were further developed into additional terms, sub-relationships, as well as metrics. This fits well with the ontology concept. Figure 3 illustrates this construct.

The MC metrics correspond to, among other aspects, the accuracy and repeatability aspects of a goal. The EC metrics correspond to the spatial and temporal aspects of the goal statement. Human interaction may be critical in assisting the robot to reach a safe state or to reach the goal.

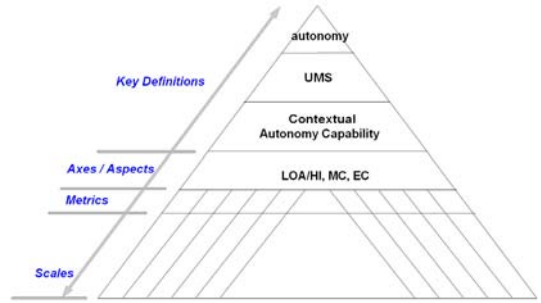


Figure 3: ALFUS Framework Structure

1.4 The Involved Use Cases and Concepts

In the three project activities that we use to further evolve this ontology, the US&R robots project has generated a comprehensive requirements set through intense interactions with the user community. The perception system for dynamic manufacturing project also derived a set of requirements from a workshop attended by the industry. We use these results to iterate the goal ontology. The TRANSTAC project uses a System, Component, and Operationally-Relevant Evaluation (SCORE) approach to evaluate the candidate systems. We began to apply some of the key concepts of SCORE in formulating the ontology [10, 11].

2. PERFORMANCE CHARACTERISTICS AND REQUIREMENTS

The performance characteristics or requirements of various UMS systems were analyzed and pertinent elements were extracted for the development of the ontology, namely, the ontological model of the mission goal for UMS. The following described two use cases.

2.1 Teleoperation Requirements for US&R

Federal Emergency Management Agency (FEMA) US&R Task Force personnel would be among the users who would teleoperate the robots for search and rescue operations. Figure 4 provides an example of the types of terrain which the US&R robots might have to traverse [11, 12].

The users conveyed to the NIST project team how they anticipate the robotic technology could help them. This typifies the interaction between users and technologists who may be implementing the requirements. The entire set of the requirements can be seen in [13]. The existent ontology work in this area includes [14, 15].

Figure 4: Rubble Pile that Robots Might have to Traverse

For this development effort, we examine a subset of the requirements and attempt to restructure them to facilitate the ontology implementation. We also apply engineering disciplines to attempt to explore further details of Requirements. Table 1 illustrates our analysis effort, with the left column stating a subset of the Requirements and the right column listing the robotic functions or subsystems that are involved to fulfill the requirements. The intent is to organize the robotic knowledge through the ontology.

Requirement	**Involved UMS Functions**
To project remote situational awareness (SA), beyond line of sight, into compromised or collapsed structures or to convey other types of information; or be able to operate around corners of buildings; as such, SA for near, far, and field of view are required	Video, command and control signals in tight space
To enable use of video in confined spaces and for short-range object identification, which can wash out from excessive illumination of the scene; therefore, adjustability is required	Variable illumination
To support the SA, the robot should be able to ingress a specified number of meters into the worst case collapse, a reinforced steel structure	Maneuver within tight space
To use this system in sensitive public situations where maintaining control of remote systems is imperative and limiting access to video images and other communications to authorized personnel is prudent; should be shielded from jamming interference and encrypted for security	Security
To project remote situational awareness or to convey other types of information down range within line of sight	Video, command and control over long range

Table 1: Mapping of Requirements to US&R UMS Functions

Beyond this example, we reviewed several other Requirements observing that, overall, the following features are keys to the US&R robotic operations:

- Situation awareness facilitation
- Maneuvering in tight space
- Radio link for video and command and control data
- Usability to the operator
- Minimum training
- System monitoring
- Sufficient power supply
- Compatible with current logistics system

These are to be reflected in the ontology work.

2.2 Requirements for Next Generation Manufacturing

An impediment to advancing manufacturing to its next level is lack of adequate sensors and perception systems, because the involving environments would be dynamic and unstructured. Therefore, dynamic metrology is a key technology. Those perception systems must be comprehensive, pervasive and providing redundancy. In scenarios such as a robot grasping a moving part off an assembly line, a single, narrowly focused sensor will not be sufficient. The sensor may fail, may not be robust enough for the task, may not sense other objects that could become obstacles, or may not be able to adequately sense humans in the workspace to prevent accidents. Perception for such scenarios would require sensor fusion and control logic to facilitate arbitrations among multiple subsystems.

To further discuss the roles and requirements of advanced sensor and perception for the future manufacturing, a workshop was held in October, 2007, which brought together people from Government, industry and academia [16]. A number of the participants indicated that the ability to measure the positions and orientations (6DOF) of components in dynamic environment would result in considerable cost savings in applications such as automobile manufacturing. The installations with more intelligent combinations of sensing and automation will better enable US manufacturers to compete globally. They also called for reference standards, as those would assist the community in establishing clear performance metrics for systems and algorithms.

All these requirements were collected and will be incorporated into a comprehensive robotic ontology. One section of the ontology deals with sensors. A generic sensor includes major attributes of sensing function, sensing target, and sensor specification. Figure 5 provides an illustration of this model using the Protégé tool[1]. Each of the attributes is further elaborated. For example, sensing targets include the properties of spatial-

[1] Disclaimer: Certain trade names and company products are mentioned in the text or identified in certain illustrations to facilitate communication. In no case does such an identification imply recommendation or endorsement by the NIST, nor does it imply that the products are necessarily the best available for the purpose

temporal, chemical, climate, kinematic, etc. This effect is illustrated in Figure 6, using the same tool. Specifications may include speed, drift, resolution, etc. The generic structure, as represented in an ontology, allows the information for specific sensors to be recorded. This ontology will serve as a knowledge base or as a tool for a sensory requirements analysis. It will also be useful in standardizing terminology for robots and sensors and in understanding the application of metrics and standards.

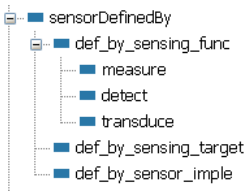


Figure 5: Ontological Model for Sensor Definition

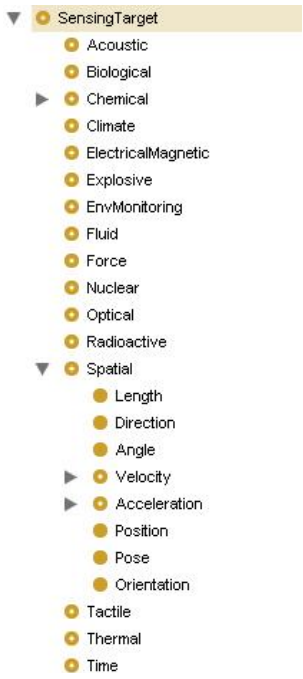


Figure 6: Ontological Model for Sensing Targets

3. ONTOLOGY: ELEMENTARY FEATURES

The previous sections have described the project requirements and the metrics model. Now we will focus on developing the ontology that encompasses those aspects. We first defined sets of fundamental references and relationships that are used to develop the UMS ontology.

3.1 Relationships

Two types of fundamental relationships are defined to associate the ontological entities. These relationships are further subtyped and instantiated to form hierarchical structures.

- Part-whole: *partOf*

  This relationship can be further instantiated to specific needs. The following are two subsets:

  Physical Subtypes: *enclosedBy, boundedBy, composedOf*

  Logical Subtypes: *controlledBy, integratedIn, boundedBy, composedOf*

  These can be further instantiated for particular applications. There are also issues of mapping of numbers (a car is composed of auto parts), complementary active and passive relationships (*composedOf* and *consistOf*).

- Peer: *associatedWith*

  Physical Subtypes: *connectedWith*

  Logical Subtypes: *interfaceWith, integratedWith*

- Generalization/specialization: *typeOf*
- Requirement: *required, optional; and, or*

3.2 Spatial Aspect

3.2.1 References

- *distance, pose, area, tolerance typeOf spatial_feature*
  - *position partOf pose*
  - *orientation partOf pose*
    - *coordinate partOf position*
    - *angle partOf orientation*
  - *coordinate partOf area*
- *range typeOf distance*

3.2.2 Environmental Features

- *object_class typeOf spatial_feature*
  - *ground_ object_class typeOf object_class*
  - *maritime_object_class typeOf object_class*
  - *aerial_ object_class typeOf object_class*

3.3 Temporal Aspect

- *time, tolerance typeOf temporal_feature*
- *duration typeOf temporal_feature*

3.4 Spatial and Temporal

- *speed typeOf temporal_feature and spatial_feature*
- *acceleration typeOf temporal_feature and spatial_feature*

4. GOAL FOR UMS

We began to generate an ontology for the concept of mission goal for the UMS. The identification of the goal attributes is facilitated through the project requirements as described in Section 3. The hierarchical layout of the section headings reflects the structure of the ontology. The mission goal is composed of the subgoals for the subsystems. The subgoals cover operational, environmental, and operator aspects.

Given the fact that UMSs may be commanded to perform vastly different types of tasks, our bottom-up approach is to develop a structure and ontological attributes that might be generic to certain applications. As such, the following subsections describe what is equivalent to an illustrative goal ontology suitable for a particular

set of applications. Different types of applications may need separate sets of the ontology.

4.1 Subgoal for Mobility Subsystem

A UMS includes a mobility system. Therefore, the following is modeled:

mobility subsystem integratedWith UMS

The subsystem's goal can be modeled as:

- *mobility_goal partOf UMS_goal*
  - *pose partOf mobility_goal*
  - *object_class partOf mobility_goal*
  - *time partOf mobility_goal*
  - *duration partOf mobility_goal*
  - *speed partOf mobility_goal*
  - *acceleration partOf mobility_goal*
  - *mobility_goal boundedBy tolerance*

4.2 Subgoal for Sensor Subsystem

A UMS typically includes a sensor system. Therefore, the following is modeled:

sensor subsystem integratedWith UMS

The subsystem's goal can be modeled as:

- *sensory_goal partOf UMS_goal*
  - *range partOf sensory_goal*
  - *frequency partOf sensory_goal*

4.3 Subgoal for Communication Subsystem

A UMS typically includes a communication system. Therefore, the following is modeled:

communication subsystem integratedWith UMS

The subsystem's goal can be modeled as:

- *comm_goal partOf UMS_goal*
  - *cover_range partOf comm_goal*
  - *set_bandwidth partOf comm_goal*
  - *receive_send_line_of_sight partOf comm_goal*

4.4 Subgoal for Power Subsystem

A UMS typically includes a power system. Therefore, the following is modeled:

power subsystem integratedWith UMS

The subsystem's goal can be modeled as:

- *power_subgoal partOf UMS_goal*
  - *set_peak_power_output partOf power_subgoal*
  - *save_min_power partOf power_subgoal*

4.5 Subgoal for Mission Package Subsystem

A robot typically carries additional subsystems beyond the platform. They may be manipulators, tools, special sensors, weapon, etc. Subgoals must be developed for them.

mission package integratedWith UMS

The subsystem's goal can be modeled as:

- *mission_subgoal partOf UMS_goal*
  - *identify_victim partOf mission_subgoal*
  - *assemble_door partOf mission_subgoal*
  - *establish_observation_post partOf mission_subgoal*
  - *translate_text partOf mission_subgoal*

4.6 Subgoal for Chassis Subsystem

A UMS includes a chassis system. Therefore, the following is modeled:

chassis subsystem integratedWith UMS

The subsystem's goal can be modeled as:

- *chassis_goal partOf UMS_goal*
  - *set_illumination_intensity partOf chassis_goal*

4.7 Subgoal for Human-Robot Interaction (HRI) Subsystem

A UMS includes a HRI system. Therefore, the following is modeled:

HRI subsystem integratedWith UMS

The subsystem's goal can be modeled as:

- *HRI_goal partOf UMS_goal*
  - *sound_alarm partOf HRI_goal*
  - *display_health_status partOf HRI_goal*

4.8 Collaboration

A UMS goal may be to collaborate with other systems. The subsystems of a UMS may need to collaborate among themselves.

- *collaboration_subgoal partOf UMS_goal*

The subsystem's goal can be modeled as:

  - *wait_until partOf collaboration_subgoal*
  - *synchronize_with partOf collaboration_subgoal*

Note that the performance metrics for the collaboration will be described later in this paper.

4.9 Prioritization and Management

The UMS entities may be given multiple goals. The entities may receive multiple goals from their collaborators.

- *manage_goals partOf UMS_goal*
  - *multi-tasking partOf manage_goals*
  - *prioritize_goals partOf manage_goals*

- *value partOf prioritize_goals*
  - *costs partOf prioritize_goals*

Note that being able to prioritize the goals would enhance the autonomous performance of the UMS.

4.10 Additional Subgoals

Additional subsystems may be employed and the subgoals should be identified.

5. ALFUS METRICS

As described earlier, the definition of performance metrics is an important aspect of the UMS knowledge and, therefore, it is a part of the ontology. We have developed the autonomous performance of UMS using the ALFUS metrics.

- *Autonomy_performance partOf UMS_performance*
  - *MC partOf Autonomy_performance*
    - *{ subtasks_structure, precision, repeatability, uncertainty, safety_level, risk_level } partOf MC*
    - *{ control_echelon, interoperability, knowledge_shared } partOf MC*
    - *{ situation_analysis, replans } partOf MC*
    - *{ perception } partOf MC*
  - *EC partOf Autonomy_performance*
    - *solution ratios partOf EC*
    - Scale: No impediment, one impediment out of N *possibilities*, (N-1) out of N, n out of N, N out of N
    - *solution_difficulty_levels partOf EC*
    - *{ energy consumption levels, computation load levels, look-ahead ability } partOf solution_difficulty_levels*
  - *HI partOf autonomy_performance*
    - *{ % of plan generated by UMS, % of plan pre-generated, % of plan execution that operator is involved, % of robot vs. operator initiated interactions, required training, workload } partOf HI*

6. GOAL STATE-METRICS ASSOCIATIONS

Various aspects of the ontology should be related. In this section, we illustrate how the requirements, goal, and autonomous performance aspects fit together, focusing on the US&R application example.

- The goal model *is* driven by the Requirements, which state that the robot "should be able to ingress a specified number of meters into the worst case collapse" and "project remote situational awareness or to convey other types of information down range." These are reflected in the mobility and communication subgoals in the ontology.

- The US&R Requirements call for the operator control unit (OCU) displays to be clear and legible in outdoor situations and under *ambient* light conditions. This is to facilitate UMS control and to reduce the stress level of the user, one of the autonomous performance metrics in the ALFUS Framework.

- The US&R Requirements further call for "enable use of video in confined spaces and for short-range *object* identification." This corresponds to the *solution_difficulty* and *situation_analysis* metrics in ALFUS.

7. SUMMARY

An ontological approach is used to model certain aspects of UMS. The objective is to facilitate the investigation of the performance of UMSs. In particular, we represented the goal of a UMS in the ontology, followed by representing the autonomous performance. We further identified several cases illustrating that the two are integrated. Our ultimate goal is to expand on this work for a broad scope ontology that can be helpful to a large audience in the community.

8. REFERENCES

[1] Vujasinovic, M., "An Industrial Validation of a Semantic Mediation Architecture," IEEE Internet Computing, October 2007

[2] Fiorentini, X. et al., "An Ontology for Assembly Representation," NIST Interagency/Internal Report (NISTIR) 7436, Gaithersburg, MD, July 2007

[3] Obrst, L., et al., "The 2006 Upper Ontology Summit Joint Communique," Applied Ontology Journal 2006

[4] Deshayes, L. et al., "An Ontology Architecture for Standards Integration and Conformance in Manufacturing," Proceedings of the IDMME 2006 Grenoble, France, May 17-19, 2006

[5] Schlenoff, C. I., et al., " An Intelligent Ground Vehicle Ontology for Multi-Agent System Integration," Proceedings of the 2005 Knowledge Intensive Multi-Agent Systems (KIMAS) Conference, Waltham, MA, April 18-21, 2005

[6] Schlenoff, C. I., et al., "A Standard Intelligent System Ontology," 2005 SPIE Defense and Security Symposium, March 2005, Florida

[7] Schenk, J.R. and Wade, R.L., "Robotic Systems Technical and Operational Metrics Correlation," PerMIS'08 Workshop, Gaithersburg, Maryland, October 2008

[8] *Autonomy Levels for Unmanned Systems (ALFUS) Framework, Volume II: Framework Models Version 1.0, NIST Special Publication 1011-II-1.0*, Huang, H. et al., Ed., National Institute of Standards and Technology, Gaithersburg, MD, December 2007

[9] *Autonomy Levels for Unmanned Systems Framework, Volume I: Terminology, Version 1.1*, Huang, H. Ed., NIST

Special Publication 1011, National Institute of Standards and Technology, Gaithersburg, MD, September 2004

[10] Weiss, B.A. and Schlenoff, C., "Evolution of the SCORE Framework to Enhance Field-Based Performance Evaluations of Emerging Technologies," PerMIS'08, Gaithersburg, MD, October 2008

[11] Schlenoff C, et al., Applying SCORE to Field-Based Performance Evaluations of Soldier Worn Sensor Technologies. *Journal of Field Robotics* 2007 September;24(8/9):671-98

[12] Messina, E., Jacoff, A., "Performance Standards for Urban Search and Rescue Robots," Proceedings of the SPIE Defense and Security Symposium, Orlando, FL, April 2006

[13] Messina, E. R., et al., Statement of Requirements for Urban Search and Rescue Robot Performance Standards, NIST Draft Report, May 2005

[14] Schlenoff, C. I., et al., " Test Methods and Knowledge Representation for Urban Search and Rescue Robots," Chapter in the Climbing and Walking Robots Book, August 2007

[15] Schlenoff, C. I. and Messina, E. R., "A Robot Ontology for Urban Search and Rescue," 2005 CIKM Conference: Workshop on Research in Knowledge Representation for Autonomous Systems, Bremen, Germany, October 2005

[16] Draft Proceedings of the Dynamic Measurement and Control for Autonomous Manufacturing Workshop, To Be Published as a NIST Internal Report, Gaithersburg, MD, October 2008

Robotic Systems Technical and Operational Metrics Correlation

Jason R. Schenk, Ph.D.
DeVivo AST
2225 Drake Ave Suite 2
Huntsville Al 35805
256-489-4614
jasonschenk@devivoast.com

Robert L. Wade
US Army Aviation and Missile Research,
Development and Engineering Center
Software Engineering Directorate (SED)
Redstone Arsenal, AL 35898
256-842-6174
robert.l.wade@us.army.mil

ABSTRACT

As unmanned systems become more prevalent, so does the emphasis on quantifying their performance using standard measures. Several research efforts have attempted to elicit important attributes associated with unmanned systems performance, with limited applicability and success. We identify several reasons for this, including the lack of adherence to a strict ontology and taxonomy, the use of means-based rather than goal-based tests, and the lack of a clear correlation between tests and predicted performance. This paper provides a conceptual framework for the development of correlations between technical characteristics and operational performance. These correlations provide a foundation for generating predictive performance test standards.

Categories and Subject Descriptors

B.8.2 [**Hardware**]: Performance and Reliability – *performance analysis and design aids*

General Terms

Algorithms, Measurement, Performance, , Human Factors, Standardization

Key Words

Analytic Hierarchy Process (AHP), Explosive Ordinance Disposal (EOD), unmanned systems, correlation, performance metrics

1. Introduction

A multiplicity of technical performance measures have been developed for requirements documents, system performance specifications, and research analysis of current Department of Defense (DoD) robotic systems. None are standardized or widely accepted. Furthermore, their correlation to operational performance is not known or well-understood within the robotics community. In fact, operational performance and technical characteristics are often ill-defined, mixed, or applied inconsistently. The establishment of a set of consistent normative technical and operational metrics based on a strict and logical ontology is critical to informing the design, comparison, and selection of unmanned systems [1].

The Robotic Systems Technical and Operational Metrics Correlation (RSTOMC) project aims to develop a methodology for establishing testable quantitative measures that correlate with unmanned systems performance. This project is sponsored by the United States Army and the Joint Robotics Ground Enterprise (JGRE), Office the Undersecretary of Defense (OUSD). Explosive ordinance disposal (EOD) is a current pressing need for the Army, so the focus of the current RSTOMC project is on tele-operated EOD platform domain analysis. This focus adds the benefit of working systems and operators with a significant historical experience. The RSTOMC project will develop a preliminary model for technical to operational correlation using quantitative and qualitative information from this domain. Based on these correlations, the project allows for the development of a method for selecting a set of technical performance tests and metrics for evaluating systems.

Standardized robotic system metrics will enable the DoD to define, specify, distinguish, evaluate, and test robotic systems in a more systematic manner. This project will attempt to derive more reliable correlations between the technical and operational characteristics of robotic systems. This correlation has the potential to simplify and reduce the cost of analysis. Successful execution of this effort will provide the foundation for a project manager or user to objectively predict the impact of technical changes on operational performance, or the operational performance of a proposed system, to an accuracy of 80% by performing a set of low cost technical tests. While the creation of specific tests may be left to future work, the RSTOMC effort will produce a model for predicting operational performance based on a set of technical measures. We will refine the model based on the data available from within the EOD robotics community and develop a software based system that allows for the continued maturation of the model as well as the incorporation of

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

additional domains. Finally, we propose a method for test creation and near-optimal set selection.

2. RSTOMC Plan

The overall goal for this project is to improve the ability to objectively assess robotic systems by developing a tool that, through simple tests and measures, can predict the mission-specific operational performance to at least 80% accuracy. To reach this goal, a methodology for robotic systems performance measurement will be developed to identify and assess both technical measures and the operational performance of the systems in application specific usage scenarios. *Technical measures* are defined here as a quantitative assessment of a characteristic in a controlled environment. A *technical characteristic* is a physical ability of a system. This is different from *operational performance measures*, which are defined relative to mission goals or tasks. Correlations between the technical and performance measures will be determined. The resulting correlations will be captured in a software tool to be distributed to appropriate agencies. The RSTOMC project consists of the following tasks:

Task 1) Collect and Analyze Operational Scenario.

- **Subtask 1)** Select the method of analysis.
- **Subtask 2)** Select the data source and domain.

Task 2) Define Technical Performance Characteristics.

- **Subtask 3)** Identify relevant technical characteristics (TC) of EOD systems.

Task 3) Define Operational Score (O_s). This task may include defining mission-specific criteria which we call operational measures (O_m).

- **Subtask 4)** Identify operational requirements and their relative importance for successful EOD system performance.

Task 4) Establish Correlations between Operational Performance and TC. Operational performance may be defined as the O_s or O_m.

- **Subtask 5)** Identify the correlation between TC and operational performance.
- **Subtask 6)** (Future work) Design technical performance metrics. and measures (TPM), validating results where possible.
- **Subtask 7)** (Future work) Create a method for selecting a near-optimal set of tests. Incorporate and document the results within a software product.

2.1 Subtask 1: Select Method

Broadly speaking, efforts to measure robotic systems quality fall into two main categories. The first is based on task decomposition, such as [2], where each mission-specific task is broken down into its most fundamental subtasks and the physical requirements or resources required to accomplish them. The overall performance is assumed to be the sum of a systems ability to complete each subtask. These efforts generally start with subject matter experts listing tasks or functions that a system must perform, and their frequency or importance. Such decomposition is often limited by experts' ability to quantify the relative relationships of functions to performance. This becomes difficult or impossible in high complexity systems with multiple interaction effects and high order dependencies [3]. Further, such analyses risk focusing on means rather than ends, limiting its relevance back to overall system performance.

One effort based on task decomposition to create a set of prioritized and measureable requirements for urban search and rescue robot performance comes from the National Institute of Standards and Technology (NIST) [4]. The study used subject matter experts to identify 103 requirements across 13 robotic platform categories from aerial to land to aquatic. There were 21 requirements that were most widely applicable across the 13 robot categories. Tests and measures were developed for each of these requirements, such as "communication range – beyond line of sight".

The National Institute of Justice has also used task decomposition to determine metrics for assessing unmanned system in law enforcement [5]. In their study, they used a focus group to obtain nine key attributes critical to system selection. These attributes were: purchase cost, platform speed, ability to complete its mission, minimal weight, minimal down time, minimal maintenance cost and maintenance requirements, manipulator lift capacity, and operating range. A survey was then conducted to determine acceptable values for each.

An effort by the Air Force Research Laboratory (AFRL) attempts to link technical measures to operational requirements for explosive ordinance disposal (EOD) robotic vehicles [6][7]. This project used Quality Function Deployment (QFD) to elicit requirements from subject matter experts, as well as one-on-one interviews to gain detailed insight into requirements. The results of their most recent project are used in the current research, and are discussed in greater detail in the following sections.

Task decomposition allows for a complete enumeration of expected task requirements. However, decomposition is platform and strategy dependent, meaning different designs or solution strategies cannot be tested as effectively or without bias. Secondly, decomposition of tasks does not test interaction effects, unless explicitly enumerated during the decomposition. Unmanned systems are complex and as such have complex interactions between component parts or capabilities. These interactions may form the basis for important emergent behaviors or qualities that allow the system to function or fail in surprising ways [8].

A second class of research efforts takes a gestalt approach, starting with the systems-level requirements of a system. The assumption is that complex systems have complex, inseparable interactions that produce emergent behavior. These emergent properties cannot be captured except by looking at the system as a whole. For example, Steinfeld et al. list five features that are required of a task-oriented human-robot system [9]. These are: navigation, manipulation, perception, management, and social, which are further broken down into more specific measurable

tasks. These tasks remain relatively generic and mission independent. It is again up to the subject matter expert to determine the relative importance of each of these tasks to the specific mission profile. Cummings et al. generated five sets of metrics classes using a separate ontological framework, based on human robot interaction (HRI) theory [10]. These classes are human behavior efficiency, human behavior precursors, Unmanned Vehicle (UV) behavior efficiency, collaboration, and mission effectiveness.

Systems-level approaches are useful in that they include the complex and emergent properties of a system in its analysis. However, it is often difficult to create a set of technical tests to measure the top-level requirements a system. Additionally, such tests do not provide much specific information about a system. On the other hand, they are much more goal oriented, allowing for fair comparison of different platform designs or solution strategies. Such high-level tests are often very expensive, approaching the cost and complexity of full-blown field trials. Finally, the authors are not aware of any top-down approach that has generated specific weights for the importance of each of the proposed high-level requirements to overall performance.

Another distinction one must consider is the ontology used in the approach. Two common ontologies are illustrated by the Architecture Framework for Unmanned Systems (AFUS), which standardizes a set of terms, definitions, and attributes common across all domains of unmanned systems [11]. The two ontologies are the Conceptual View (means-oriented) and the Capabilities View (goal-oriented). The Conceptual View describes unmanned systems in terms of component structure, knowledge store, and gross actions. The Capabilities describes capabilities and behaviors a system is able to perform. For example, a laser range-finder is a *concept* used to describe the composition of a system, while object-detection and tracking (which may use a laser range-finder) is a *capability* of the platform. The ontology chosen drives the development of the criteria, tests, and measures that are chosen to predict system performance.

2.1.1 Analytic Hierarchy Process

Ideally, technical as well as operational performance data would be available from rigorous lab and field experimentation, and real-world observation across a wide range of platforms. Due to cost and logistics limitations, it is often more practical to obtain data through the elicitation of expert opinions. There are several ways in which to accomplish this, including Quality Function Deployment (QFD) and Analytic Hierarchy Process (AHP). The fundamental mathematics of the RSTOMC project comes from the AHP, which provides a systematic approach to correlating low level features with high level system requirements [12]. The raw data comes from a QFD and interviews performed by AFRL. The following is a brief description of the proposed mathematical relationships. These equations represent an idealized, hierarchical, and theoretical model of the relationships. The actual manifest structure of our analysis will depend on the specifics of the data.

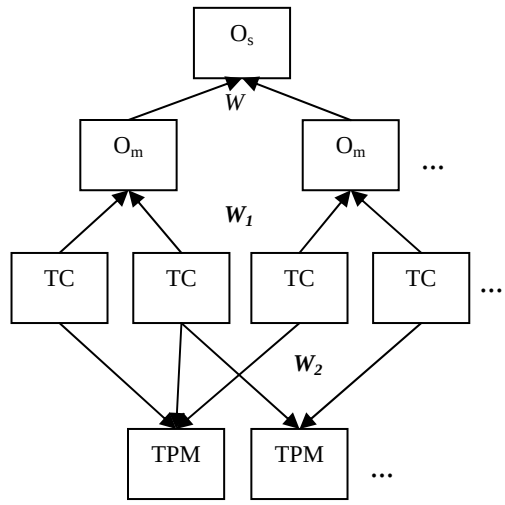


Figure 2 Conceptual hierarchy of EOD unmanned systems analysis

Operational requirements, O_m, include all the important criteria that may arise during an EOD mission. Examples might include probability of successful disarming or average time to disarm. In addition, their frequency or relative importance to the completion of the mission provides the basis for weighting. The weighted sum of these requirements yields the operational performance score of the system, O_s.

$$\boldsymbol{O_s} = [\boldsymbol{W}][\boldsymbol{O_m}] \tag{1}$$

Technical characteristics refer to the physical systems and their interactions, such as end-effector lift capacity or screen resolution. These characteristics each have a unique impact on the set of operational requirements. The nature of this impact provides the weighting factors:

$$[O_m] = [\boldsymbol{W_1}][TC] \tag{2}$$

To further complicate matters, available metrics and derived tests may not correspond 1:1 to technical characteristics, which would require another set of weights, $\boldsymbol{W_2}$. However, if a 1:1 correspondence can be established (that is, each test measures only one technical characteristic), then this weighting matrix is equal to the identity, I. These tests and metrics, which we call technical performance metrics, or TPM, may be related back to the operational performance score, in the following manner:

$$O_s = [W][\boldsymbol{W_1}][\boldsymbol{W_2}]^{-1}[TPM] \tag{3}$$

As an alternative, TPM may be directly related to operational requirements, or even operational performance scores, using weights derived from a separate survey. In this way, only one set of weights are needed, indifferent to the relationship of operational requirements and technical characteristics:

$$O_s = [\tilde{W}][TPM] \tag{4}$$

This holds true only if the TPM are a minimum necessary and sufficient measure of the operational score (no zero rows or columns in the weighting matrix, $\boldsymbol{W_2}$). In either case, the relationship is linear. Therefore, there are no interaction terms between TPM. This assumption is unlikely for complex systems, especially if each TPM corresponds to about one TC. For example, it would be difficult to design a TPM measuring manipulator reach and a TPM measuring lift capacity that are independent of one another, since lift capacity is relative to the extension of the arm. We are motivated to develop TPM based on some other criteria besides decomposition to the most fundamental tasks.

2.2 Subtask 2: Select Data Source and Domain

We chose the AFRL analysis to provide the raw data for the project, because it is one of the few analyses that quantify the relationships between characteristics and performance. The analysis uses Quality Function Deployment and significant domain expert input to formulate operational requirements, technical characteristics, and their relationship to one another. The inclusion of this relational data makes it ideal for our analysis.

This analysis draws from interviews and focus groups with 270 DOD unit-level EOD technicians from 45 different organizations across the four United States armed forces services. The goal of the analysis was to generate requirements for the combat variant of the Next Generation EOD Remote Control Vehicle (NGEODRCV). The report identified and ranked, using QFD, 67 "technical characteristics" based on 79 "customer needs." The expert interviews provided details to the technical characteristics, breaking them down further into approximately 271 stated needs, preferences, tasks, design elements, features, etc.

The set of roughly 271 failure modes, design preferences, tasks, features, specifications, and customer requirements that make up the responses do not represent a clear and consistent ontology for use in our analysis. In order to be applicable to the RSTOMC project, they must be organized in such a way that a small set of valid technical metrics can be derived to be predictive of operational performance. By creating a set of technical tests and metrics, and using a set of weights derived from the domain expert interviews, we can use the AHP methodology to generate a first order linear model of operational performance.

2.3 Subtask 3: Identify TC

Technical characteristics were established by the TSWG NGEODRCV Common Systems Architecture dated 14 January 2003. These characteristics were placed in a three-tiered hierarchy. For example, Navigation is broken into "path planning" and "path execution". Each of these is further broken down. For example, path execution is broken down into "obstacle detection" and "obstacle avoidance".

Table 2 Sample of Technical Characteristics from the QFD

Navigation					Payload				
Path Planning			Path Execution		Sensor Integration			Tool Integration	
Position capability (D-1-a)	Mapping capability (D-1-b)	Waypoint navigation (D-1-c)	Obstacle avoidance (D-2-a)	Obstacle Detection (D-2-b)	Logical (E-1-a)	Electrical (E-1-b)	Physical (E-1-c)	Electrical (E-2-b)	Physical (E-2-c)

These characteristics were further broken down and defined in surveys. For example, "Tether Durability" was described as the ability to resist being run over by the platform (51%), resist tensile forces (25%), resist rough terrain (9%), and resist both rough terrain and being run over (15%). In a QFD, these characteristics are of the same level of scope, type, and fidelity. In other words, the ontology should be consistent. Typically, these are measurable characteristics with a possible range, such as "battery life: 0-24 hours". However, in the present QFD, characteristics are a mixture of resistance to failure modes (e.g. tether resistance against being run over, prevention of dirt clogging treads), design modality preferences (e.g. joystick vs. keyboard), tasks (e.g. ability to place counter-measures on a vehicle, ability to reach into a 55-gal drum), typical customer requirements (e.g. fine/smooth motor control, "observability" of platform orientation), specifications (e.g. 12-hour battery life, maximum arm lift of 25-50 lbs.), and features (e.g. rangefinder). The experts' reasoning generally appears within the interview responses. Therefore, this feature of the analysis may be addressed by (and provides further motivation for) goal-based tests.

As a result of trying to fit all of the needs into the pre-determined hierarchy, several of the 271 needs are the same under multiple headings (e.g. depth perception appears under Visual (B-2-a) as well as Displays (C-2-d)). Therefore, the ranked list may be skewed, since some important needs may be diffused among multiple technical characteristics, while other characteristics represent aggregates of several important needs that could be broken-up further.

In an effort to tighten the taxonomy, we re-sort and re-classify the needs into new technical characteristics categories. Each listed need maintains its constituent weight (the weight of the characteristic it belonged to times the percent of respondents citing it for that characteristic). The result is a list of approximately twenty-five new technical characteristics categories. Under each of these categories, the same 271 TC remain (condensed to 96 by combining duplicate and redundant TC), only in a more useful way for the RSTOMC project.

2.4 Subtask 4: Identify Operational Requirements

In a QFD, user requirements in the form of qualities are listed along the left side of the table [13][14][15]. These requirements might include general qualities such as "ease of deployment" or "mobility". Note that the emphasis is on general goals, rather than specific means or tactics. In the present case, the requirements represented possible tasks that a system would need to perform during operations, as determined by a joint service focus group of combat experienced EOD technicians. These included tasks such as "offload robot" and "negotiate terrain", along with the subtasks required to accomplish these. The subtasks were weighted via survey as to their importance to the overall mission. As discussed previously, tests based on task decomposition methods tend to dictate the solution strategy, limiting its ability to compare innovative design or operational strategies [16]. More general goal-oriented requirements allow comparison of disparate designs over ordinal measures. At the least, the sum total of these requirements provides Os. The derivation of a more goal-focused set of operational measures, O_m, may result from a separate survey. The formulation of tests based on these results will be discussed further in section 2.6.

Table 1 Sample of User Requirements from the QFD

Requirement
Off-load Robot
Unsecure robot
Power-up robot
Remove robot from vehicle
Pick up robot
Drive robot off vehicle
Off-load OCU
Remove OCU from vehicle
Set up OCU
Attach tether to OCU

2.5 Subtask 5: Identify the correlation

The body of the QFD matrix represents the correlation between customer requirements and technical characteristics. Since we are reorganizing the data, the correlations must be carried along with the lowest level TC as they move to our new taxonometric structure. Since the operational analysis interview results cite the rationale and the relative number of respondents citing each rationale, we are able to carry that response over to the new parent category without losing that response's relative contribution to overall system performance.

2.6 Subtask 6: Generate TPM Tests

This section details the important considerations associated with test selection. Although this and the following section are identified as future works, it is important to mention here. This is because the present work is intended to ultimately facilitate the creation of standard tests, so preliminary work must be cognizant of this goal.

The level of testing selected is based on cost, resources, the desired number and complexity of tests, and the required level of specificity and "predictiveness" of tests. Since each TC has its own weighting, tests may be devised that measure the each independently, collectively, across categories, or some variation of the three. The type of tests selected depends on 4 factors:

1. The natural grouping of the technical requirements. Some sets of requirements are coupled or difficult to segregate. Interactions and emergent properties may also exist that cannot be measured if TC are measured in isolation. A solid taxonomy and ontology will facilitate this.
2. Cost or difficulty of the test. A set of tests that measure individual components separately may be more or less difficult or expensive than a single aggregate test. Cost is also impacted by interaction effects, as it may be more difficult to isolate TC that are naturally coupled.
3. The "informativeness" of the test. More atomic tests provide more specific information, while aggregate tests may be more predictive of actual operational performance, due to interaction effects and their goal rather than means focus.
4. The number of tests. A smaller set of tests are preferable to a large battery of tests.

Overall, the goal is to provide a small set of informative tests that together predict performance to the highest accuracy possible. Below are three levels of tests along the spectrum of aggregation. The weights included in the tables are estimated from the results of the operational analysis and our effort to re-sort the results for our purpose.

2.6.1 Atomic tests

A large number of very specific "atomic" tests designed to measure each TC separately provide very specific information about a given platform. It provides clear evidence of what specific features or abilities are strong or weak in a platform, but the correlation to real world operational performance may not be as high. This is because individual tests would only provide for a first order linear model where interactions between components are ignored.

Also, specific tests of "means" dictate a solution strategy and limit the possibility for innovation. For example, if system performance for a hypothetical mission is defined as the ability to cross a bridge, then tests of waypoint sensors, the drive train, stability, distance sensors and obstacle avoidance might be devised to see if the robot could cross the bridge. However, a system designed to bypass the bridge altogether and fjord the river might accomplish the

mission objective equally well, but score poorly on these atomic, "means" tests. Further, due to unmeasured interactions, the system that scores well on such tests might still be unable to accomplish the mission objectives, for example if its extra features make it too heavy or wide to cross the bridge.

Table 5 Examples of "atomic" level tests

Test	Details	$\hat{w}$
Light board	Tests the manual dexterity of the robot, the controllability of the system, and the visual feedback of the device	~4%
Lift strength	Tests the strength of arm, effector, and the stability of the platform	~4%
Reach	Tests the range of motion (solid angle or volume) of the arm	~3%
Climb	Tests the maximum grade of stairs, curbs, and slopes scalable	~3%

2.6.2 Aggregate tests

Alternatively, higher level "aggregate" tests are designed to measure sets of technical characteristics that are naturally coupled. These types of tests tend to be fewer in number, but more complex. As a result, interactions are captured and the subsequent model can be more predictive of operational performance. The tradeoff is that the information provided by the results of tests is less specific. The determination of specific deficiencies in the platform requires follow-up analysis. Using our river-crossing example, the test would measure how well (quickly, reliably, etc.) the robot could cross a bridge, rather than the individual systems that provide that affordance. As with the atomic tests, this level still risks measuring means rather than goals.

Table 6 Examples of "aggregate" level tests

Test	Details	$\hat{w}$
Advanced Visual Acuity	Tests the ability to detect various types and levels of signals	~15%
Signal Strength	Tests the ability to maintain a strong signal across distance and with physical and electronic interference	~23%
Manual dexterity	Tests the ability to access and manipulate objects of various types, sizes despite possible obstructions	~15%

2.6.3 Task level tests

A third design philosophy might be to create tests that are "task" or "goal" based. These tests are also very aggregate in that they test many components and the interactions in concert. The overall performance on a task represents a system-level emergent capability, thereby allowing for multiple robot design types or strategies to be compared. Task outcomes (such as the ability to get to the other side of a river) are measured rather than means (bridge crossing, swimming, etc.). This method has the potential for being the most predictive of operational performance and best able to compare disparate robotic systems, because its tests more closely approximate the real world. As with all more aggregate tests, however, insight into specifics require post hoc analysis. In addition, the results of these tests will most likely not be independent, so the sum weightings may not be equal to 1. Finally, the resource and time costs associated with these tests approach those of full-blown field trials, the current method with which this effort is intended to improve upon.

Table 7 Examples of "task" level tests

Test	Details	$\hat{w}$
Retrieve	Enter a room and retrieve an object. There may be other objects in room, doors, stairs/slopes, and other obstacles, variable lighting, target object may be obstructed, buried, or inside a container, object may be large, small, fragile, slippery, or heavy.	~25-75%
Arrange	Move objects to a predetermined arrangement. Specific tasks might include stacking, clearing an area, manipulating small objects (Chess, Perfection, Toy Ball), etc.	~25-75%

Regardless of the level of testing, tests must be correlated back to overall operational performance. The AFRL study provides a basis for this, as the previous section detailed. Verification may be conducted through additional surveys or interviews, using AHP or another method.

2.7 Subtask 7: TPM Test Set Selection

Given the sheer number of alternative performance test combinations, an empirical method of test selection should be derived. This method is fundamentally a combination of two classical problems. First, it is a cost benefit-analysis. Each proposed test has a unique cost associated with it, in terms of resource requirements, time requirements, and costs associated with the difficulty of testing and measurement. The goal is to find a set of tests that can predict with the requisite level of accuracy at the minimum cost. To achieve this accuracy, the set of tests must span the relevant system characteristics that correlate with overall performance. This is a set covering problem. The solution must cover characteristics that account for at least 80% of the variability. In other words the overall test selection problem becomes: minimize the cost of the selected tests, such that the sum of $\hat{w}$ of the TC covered by the selected tests $\geq$ 80%. This problem is NP-hard, and therefore we propose future work use a solution derived

from a heuristic, such as a greedy algorithm or evolutionary computation.

3. Preliminary Findings

The present work and review of similar efforts has produced the following findings, representing common themes that appear throughout the literature and research. These findings represent principles that must be addressed in order to adequately represent the problem and provide a useful solution:

1. **Taxonomy must be hierarchical** – taxonomy, or the hierarchical categorization of a set of concepts, must be well-formed. This means that at each level of abstraction, categories must be distinct (exclusive), of the same level (scope), and of the same disciminant (type).
2. **Ontology is critical** – without a solid ontology, analyses become confused, where customer requirements are confused with subtasks and strategies, which are in turn confused with technical characteristics. A good ontology to develop tests will identify requirements based on the goals of the deployment, and tests based on the ability to achieve goals, regardless of method or strategy of the platform.
3. **Characteristics are coupled** – it is difficult to separate some characteristics from one another, because they are interrelated. For example, usability, of either the input or output sub-system, is difficult to exclude from the sensory or motor sub-systems of a platform, respectively.
4. **Weighting is variable** – each platform design may employ a different strategy to accomplish goals, meaning that there is an interaction between platform and the importance (weights) of test scores, if those are based on technical characteristics rather than goals. For example, one platform might rely more heavily on its strength to clear obstacles, while a second platform uses its robust drive system to scale most obstacles, and a third uses its small size and superior environmental awareness to circumnavigate obstacles. The weight of associated tests would be different in each case.
5. **Ends over means** – tests based on technical characteristics specify the solution strategy. By focusing on technical characteristics rather than goals, tests are biased towards a preconceived solution strategy. This biases tests against innovation, and potentially forces systems to maintain "vestigial" features in order to satisfy test requirements.
6. **Performance over characteristics** – the goal of metrics is to inform acquisition decisions based on performance under intended use. Goal-based assessments are preferable to characteristics that, although interesting, are not directly predictive of performance.
7. **Roles change goals *and* means** – designs are optimized towards their roles. Platforms designed for one role cannot be compared to those designed for a different role, except relative to its *in-class* performance. For example, a tank cannot be compared to a fighter aircraft, except through comparing how good the tank is at its role versus how good the fighter is at its role.
8. **Expertise variability valuable to measuring learning curve** – by using pairs of operators matched based on observed covariates, expertise levels can be controlled for (neutralized). Also, main effects plots on expertise will show the learning curve for each platform [17].

4. Summary

The AFRL study provides the basis for a set of operational requirements, O_m, and their relative importance to the overall operational performance O_s. A second set of O_m that are more mission and goal targeted may be derived from a separate survey. Subject matter experts interviewed in the study related the technical characteristics, TC, to the operational requirements. By testing these characteristics, one can determine technical performance metrics, TPM, that use these relationships to measure the operational performance objectively. Each test may measure technical characteristics either individually or in logical aggregate groups.

This process will provide a methodology for generating and weighting technical performance tests and metrics that inform design and selection decisions for EOD robotic systems as well as other unmanned systems in a standardized way. The successful completion of the project will see the following deliverables: a set of TC and weights that can predict 80% of the operational performance of a tele-operated EOD robotic system, insight into the critical design characteristics of a tele-operated EOD robotic system, and a methodology for the construction of an objective, quantifiable, reliable, and empirically valid testing standard for predicting unmanned systems performance.

5. References

[1] van Rees, R. 2003. Clarity in the Usage of the Terms Ontology, Taxonomy and Classification. Construction Infomatics Digital Library. http://itc.scix.net/paperw78-2003-432.content (accessed 7/1/08).

[2] Mitchell, D.K. 2005. Soldier Workload Analysis of the Mounted Combat System (MCS) Platoon's Use of Unmanned Assets. Technical Report, ARL-TR-3476, U.S. Army Research Laboratory, Aberdeen Proving Ground, MD.

[3] Woods, D.D. and Hollnagel, E. 2006. Joint Cognitive Systems: Patterns in Cognitive Systems Engineering. CRC Press, Boca Raton.

[4] Messina, E., Jacoff, A., Scholtz, J., Schlenoff, C., Huang, H., Lytle, A., & Blitch, J. 2005. Statement of Requirements for Urban Search and Rescue Robot Performance Standards (Preliminary Report). Department of Homeland Security Science and Technology Directorate and National Institute of Standards and Technology.

http://www.isd.mel.nist.gov/US&R_Robot_Standards/Requirements%20Report%20(prelim).pdf (accessed 7/1/08).

[5] National Institute of Justice. 2000. Final Report on Law Enforcement Robot Technology Assessment. TSWG Task T-150B2. http://www.nlectc.org/jpsg/robotassessment/robotassessment.html (accessed 7/1/08).

[6] AFRL. 2006. Next Generation EOD Remote Controlled Vehicle: Military Combat Operational Analysis (DRAFT). Combating Terrorism Technology Support Office Technical Support Working Group.

[7] AFRL. 2003. Next Generation EOD Remote Controlled Vehicle: Operational Analysis. Combating Terrorism Technology Support Office Technical Support Working Group.

[8] Hollnagel, E., Woods, D.D., & Leveson, N. 2006. Resilience Engineering: Concepts and Precepts. Ashgate; Burlington, VT.

[9] Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A., and Goodrich, M. 2006. Common metrics for human-robot interaction. In *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-Robot interaction* (Salt Lake City, Utah, USA, March 02 - 03, 2006). HRI '06. ACM, New York, NY, 33-40.

[10] Donmez, B., Pina, P., & Cummings, M. 2008. Evaluation Criteria for Human-Automation Performance Metrics. PerMIS conferece proceedings; August 19-21, 2008, Gaithersburg.

[11] Society of Automotive Engineers AS-4A. 2008. Architecture Framework for Unmanned Systems. Architecture Framework Subcommittee, Society of Automotive Engineers.

[12] Saaty, T.L. 1980. The Analytic Hierarchy Process. McGraw-Hill, New York, NY.

[13] Kolarik, W.J. 1999. Creating Quality. McGraw-Hill, Boston.

[14] Kusiak, A. 1999. Engineering Design – Products, Processes, and Systems. Academic Press, San Diego.

[15] Lowe, A.J. 2000. QFD-HOQ tutorial. http://www.gsm.mq.edu.au/cmit/qfd-hoq-tutorial.swf (accessed 5/5/08).

[16] Saaty, T.L. 1999. Fundamentals of the Analytic Network Process. ISAHP; August 12-14, Kobe, Japan.

[17] Allen, T.T. 2006. Introduction to Engineering Statistics and Six Sigma: Statistical Quality Control and Design of Experiments and Systems. Springer-Verlag, London.

Survey of Domain-Specific Performance Measures in Assistive Robotic Technology

Katherine M. Tsui and Holly A. Yanco
University of Massachusetts Lowell
Department of Computer Science
1 University Avenue
Lowell, MA, USA
{ktsui, holly}@cs.uml.edu

David J. Feil-Seifer and Maja J. Matarić
University of Southern California
Department of Computer Science
941 West 37th Place
Los Angeles, CA, USA
{dfseifer, mataric}@usc.edu

ABSTRACT

Assistive robotics have been developed for several domains, including autism, eldercare, intelligent wheelchairs, assistive robotic arms, external limb prostheses, and stroke rehabilitation. Work in assistive robotics can be divided into two larger research areas: *technology development*, where new devices, software, and interfaces are created; and *clinical application*, where assistive technology is applied to a given end-user population. Moving from technology development towards clinical applications is a significant challenge. Developing performance metrics for assistive robots can unveil a larger set of challenges. For example, what well established performance measures should be used for evaluation to lend credence to a particular assistive robotic technology from a clinician's perspective? In this paper, we survey several areas of assistive robotic technology in order to demonstrate domain-specific means for evaluating the performance of an assistive robot system.

Categories and Subject Descriptors

A.1 [**Introductory and Survey**]

General Terms

Performance measures

Keywords

Assistive technology, human-robot interaction, robotics, end-user evaluation

1. INTRODUCTION

Assistive robotics may have therapeutic benefits in domains ranging from autism to post-stroke rehabilitation to eldercare. However, it can be challenging to transition an assistive device developed in the lab to the target domain. This problem can occur even when the device was designed with a specific end user in mind. Römer et al. provided guidelines for compiling a technical file for an assistive device for transfer from academic development to manufacturing [52]. Their guidelines state that documentation of an assistive device must include its "intended use, design specifications, design considerations, design methods, design calculations, risk analysis, verification of the specifications, validation information of performance of its intended use, and compliance to application standards" [52]. Academic and industrial research labs are the piloting grounds for new concepts. However, due to the institutional separation between the research environment and end-users, special care must be taken so that a finished project properly addresses the needs of end-users. As such, it is imperative for the development of assistive robotic technologies to involve the end-user in the design and evaluations [28]. These end-user evaluations, with the proper performance measures, can provide the basis for performance validation needed to begin the transition from research pilot to end product.

Does there exist a ubiquitous set of performance measures for the evaluation of assistive robotic technologies? Time to task completion or time on task are common measures. Römer et al. propose an absolute measure for time to task completion, where the time is normalized with an able-bodied person's performance [52]. Task completion time fits many robotic applications, such as retrieving an object with a robotic manipulator. However, it may not suit other applications, such as a range of motion exercise in the rehabilitation of an upper limb. Römer et al. also acknowledge other factors in determining performance measures, namely "user friendliness, ease of operation, [and] effectiveness of input device" [52].

Aside from the very general metrics described above, should there even be a ubiquitous set of performance metrics? This lack of a ubiquitous set has occurred in part because each domain has very specific needs in terms of performance. Most metrics do not translate well between domains or even subdomains. The field of assistive robotics technology has used a wide variety of performance measures specific to domains for end-user evaluations. However, there are observable similarities between various employed metrics and how they are devised. In order to evaluate an assistive robotic technology within a particular domain, clinical performance measures are needed to lend validity to the device.

Clinical evaluation is the mechanism used to determine the clinical, biological, or psychological effects of an evaluated intervention. Clinical evaluations use The Good Clin-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08 August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

ical Practice Protocol, which requires clearly stated objectives, checkpoints, and types and frequency of measurement [68]. Well established domains can have well established performance measures. For example, the Fugl-Meyer motor assessment, created in 1975 [25], is commonly used when evaluating upper limb rehabilitation for patients post-stroke. FIM [42] is popular when measuring the function independence of a person with respect to activities of daily living (ADLs). The two evaluations have little correlation, if any, to each other because they are domain-specific. However, they are both used for studying potential end-users that do not use assistive technology, and can serve as an effective method for assessing performance relative to the established baseline.

In this paper, we explore contemporary end-user evaluations and the performance measures used in evaluating assistive robotic technology. We detail the performance measures and discuss for which evaluations and contexts they would be appropriate.

2. ASSISTIVE ROBOTIC TECHNOLOGIES

Haigh and Yanco surveyed assistive robotics in 2002 [30]. A historical survey of rehabilitation robotics through 2003 can be found in Hillman [32]. Simpson surveyed intelligent wheelchairs through 2004 [57]. We present a contemporary survey of assistive technologies that have been evaluated by end-users. We believe that the primary focus of end-user evaluations should be on the *human* performance measurements, and secondarily on the performance of the robot. This section highlights six areas of assistive technology development: autism; eldercare; intelligent wheelchairs, assistive robotic arms; prosthetic limbs; and post-stroke rehabilitation. For each area, we describe a few examples of performance metrics and how they are employed/applied.

2.1 Autism Spectrum Disorder

An increasing number of research institutions are investigating the use of robots as a means of interaction with children with autism spectrum disorder (ASD), including the National Institute of Information and Communications Technology [37], University of Hertfordshire [49, 50, 48], Université de Sherbrooke [43, 53], University of Southern California [21], University of Washington [60], and Yale University [55, 56]. The goal of these systems is to use robots as a means of affecting the social and communicative behavior of children with autism for either assessment or therapeutic purposes.

2.1.1 End-user Evaluations

The University of Hertfordshire has conducted several observation studies with children with ASD [49]. In one study, four children interacted with Robota, a robot doll, over a period of several months. Post-hoc analysis of video footage of interaction sessions yielded eye gaze, touch, imitation, and proximity categories. Performance measures included frequency of the occurrence of the categories. Another study used the hesitation and duration of a drumming session as a task-specific measure of engagement with a drumming robot [50]. In addition, measures for observing social behavior were taken from existing work from the autism research community regarding methods for using video coding for observing social behavior [64] to determine if a robot was an isolator or mediator for children with autism [48].

The Université of Sherbrooke conducted an observation study of four children with autism spectrum disorder over seven weeks [43]. The children interacted with Tito, a human-character robot, three times per week for five minutes. Video was collected during the interactions. In post-hoc analysis, the interactions were categorized into shared attention, shared conventions, and absence of shared attention or conventions; all video data were coded using twelve-second windows. Performance measures included frequency of the occurrence of categories. Other work involved the use of automated interaction logs in order to model a user's play behavior with the robot [53]. Performance measures included correlation of recognized play with observed behavior.

The National Institute of Information and Communications Technology (NICT) conducted a longitudinal observation study in a day-care setting [37]. Groups of children interacted with a simple character robot, Keepon, in twenty five three-hour sessions over five months. Each session was a free-play scenario that was part of the regular day-care schedule. Children were given the opportunity to interact with the robot, or not, and children were allowed to interact with the robot in groups. Video of these interactions was recorded and analyzed in a qualitative fashion. In particular, they observed changes in dyadic interaction between the child, the robot, and peers.

The University of Southern California (USC) conducted a study with children with autism interacting with a bubble-blowing robot [20]. This research uses a repeated-measures study to compare two types of robot behavior, contingent (the robot responds to the child's actions) and random (the robot executes an action after a random amount of time has passed). The scenario involved the child, the robot, and a parent observed for forty-five minutes. Post-hoc analysis of video data was used to identify joint-attention, vocalizations, social orienting, and other forms of social interaction, identified by target (parent, robot, or none). These behaviors were taken from a diagnostic exam, the Autism Diagnostic Observation Schedule (ADOS) [40], which uses a similar scenario to the one used in the experiment, providing a key for identifying relevant evaluative behavior. Results from this work supported the hypothesis that a robot behaving contingently provoked more social behavior than a robot behaving randomly. Performance measures included frequency and richness of the interaction observed between sessions.

The University of Washington developed a study that compared a robot dog, AIBO, to a simple mechanical stuffed dog [60]. After a brief introductory period, the participants, parent and child, interacted with the one of the artifacts for a period of thirty minutes. The sessions were videotaped, and coded for behavior. The behavior coding included verbal engagement, affection, animating artifact, reciprocal interaction, and authentic interaction. The data were compared between sessions with each dog. The performance measure used was the amount of coded social behavior observed.

Yale University has been developing robots for diagnostic and therapeutic applications for children with autism. Specifically, they are developing passive sensing techniques along with robots designed to exhibit social "presses" in order to provoke and observe the behavior of children with autism [55]. One example of this approach was the use of observing gaze behavior as a means for providing diagnostic information [56]. In one study, children were outfitted with eye-tracking equipment and their gaze was tracked

with various visual and auditory stimuli. This experiment tested both children with autism and typically developing children. The performance measure for this study was to determine if the gaze tracker could identify significant differences between the gaze patterns of children with autism and typically developing children. Another study compared affective prosody given from either a human or robot speech therapist [36].

2.1.2 Analysis

One common technique for measuring performance in the ASD domain is coding, followed by a post-hoc analysis to create keywords, phrases, or categories from video data [51]. Categories and definitions are defined from these units. The data, such as open ended responses to questions or recorded, can be annotated with the categories. To ensure reliability, multiple coders are trained on the units and definitions. When multiple coders are used, inter-rater reliability needs to be established, usually assessed using Cohen's kappa [12]. However, in each case, the basic unit of time for behavior data could be vastly different, ranging from tenths of a second [49], to twelve seconds [43], to assessments of the entire session [37]. The resulting performance measures use the number of occurrences within the categories.

While these assessments are in most cases driven by existing tools used in developmental or autism-specific settings, there is little evidence shown so far that the measures used translate well to real-world improvements in learning, social skill development, and psychosocial behavior. It is important to note that autism is considered a spectrum disorder and that there is a great deal of heterogeneity to the population [24]. While studies can show effects for a small subgroup of children, it is important to analyze how generalizable the results are. One strategy for ensuring that the observed data are somewhat grounded in the field of autism research is to draw the analysis metrics from existing communities [51, 20].

2.2 Eldercare

Studies show that the elderly population is growing worldwide [6]. Roboticists from research institutions, such as NICT [70], USC [63], and University of Missouri [72] are investigating robots for use as minders, guides, and companions.

2.2.1 End-user Evaluations

The University of Missouri in conjunction with TigerPlace, an eldercare facility, studied assistive technology for aging in place [72], where residents who would otherwise be required to have full-time nursing-home care are able to live in their current residence and have health services brought to them instead. As part of this effort, they developed a fuzzy-logic augmentation of an existing day-to-day evaluation, the Short Physical Performance Battery (SPPB) [29]. This test measures the performance for balance, gait, strength, and endurance.

NICT conducted a five-week study of twenty three elderly women interacting with Paro, the therapeutic care robot seal, in an eldercare facility. Interaction occurred one to threes times per week [70]. Performance measures included self assessment of the participant's mood (pictorial Likert scale [39] of 1 (happy) to 20 (sad)) before and after the interaction with Paro; questions from the Profile of Mood States questionnaire [41] to evaluate anxiety, depression, and vigor (Likert scale of 0 (none) to 4 (extremely)); and urinary specimens to measure stress.

Researchers at the USC are currently developing a robot for exercise therapy in adults suffering from dementia [63]. Exercise therapy was part of the regular care regiment provided by the staff at the nursing home location of the experiment, but keeping the elders engaged in the task was a challenge for the staff. The experiment scenario involves using a robot to demonstrate, coach, and monitor exercises. The real-world performance measure for success is compliance to the exercise regimen, measured by time on task (from recorded video data post-hoc), or overall health of the residents. Initial studies involved using a focus group to assess resident's reactions to the robot. For the focus group interaction, performance was measured by the number of residents showing willingness to interact with the robot.

2.2.2 Analysis

Most of the above systems are currently at the feasibility stage of implementation, an important stage of evaluation for determining if the technology is ready for deployment in a real-world environment. User and behavior studies of eldercare systems, such as with Paro, serve to describe the effects that such systems have on users and their environment. By emphasizing social interaction and fitness, these performance measures implicitly measure changes in quality of life (QoL).

Current evaluations of eldercare systems occur over a period of days or weeks. As these systems become more permanent fixtures in eldercare environments, the assessment of QoL becomes more important. There exist standardized questionnaires for observing QoL at multiple points of time. Therefore, QoL can be a good method of observing the long-term effectiveness of a change in the eldercare environment [76]. For example, the SF-36 survey [1] is used to assess health-related QoL, while the 15-D [59] survey is used to measure QoL along several elements of a subject's lifestyle.

2.3 Intelligent Wheelchairs

Intelligent wheelchairs can potentially improve the quality of life for people with disabilities. Research has focused on autonomous and semi-autonomous collision-free navigation and human-robot interaction (i.e., novel input devices and intention recognition) and has been conducted by both research institutions and companies.

2.3.1 End-user Evaluations

In 2005, MobileRobots (formerly ActivMedia) and the University of Massachusetts Lowell evaluated the Independence – Enhancing Wheelchair (IEW) [45, 46] with several end-users at a rehabilitation center. The original testing design planned to use of a maze-like obstacle course made of cardboard boxes. However, this scenario did not work well with the participants. They were frustrated by a maze that was not like their regular driving environments and viewed boxes as moveable objects.

Instead, the participants operated the IEW as they would typically use a wheelchair in their everyday lives (e.g., going to class which entailed moving through corridors with other people and passing through doorways). The performance measure, number of hits/near misses and time on task, was not modified. The results have not yet been published.

End-user trials have also been completed by intelligent wheelchair companies, such as DEKA [16] and CALL Centre [8], seeking government approval to prove the safety of these systems. The University of Pittsburgh has conducted an evaluation of DEKA's iBOT with end-users [13].

2.3.2 Analysis

In the domain of intelligent wheelchairs, the majority of user testing has been in the form of feasibility studies with able-bodied participants. As noted in Yanco [77], able-bodied participants are more easily able to vocalize any discomforts and stop a trial quickly. These pilot experiments pave the way for end-user trials.

One barrier to end-user trials of robotic wheelchair systems is the need for the use of a participant's seating on the prototype system. While seating can be moved from the participant's wheelchair to the prototype system (if compatible) and back, this seating switch can take thirty to sixty minutes in each direction, making multiple testing sessions prohibitive.

We discuss performance measures commonly used thus far in feasibility studies. One of the most common tests of an autonomous intelligent wheelchair is passing through a doorway [58]. Passing through a doorway without collision is one of seven "environmental negotiations" that a person must perform in order to be prescribed a power wheelchair for mobility [67]. Other tasks include changing speed to accommodate the environment (e.g., cluttered = slow), stopping at closed doors and drop offs (e.g., stairs and curbs), and navigating a hallway with dynamic and stationary objects (e.g., people and furniture).

In the case of these power mobility skills, the user is rated based on his/her ability to *safely* complete the task. In contrast, robotic performance measures are not binary. Performance measures include time to completion (i.e., time to pass through the doorway), number of interactions, and number of collisions. Recent performance measures include accuracy, legibility, and gracefulness of the motion used to pass through the doorway [9, 62].

2.4 Assistive Robotic Arms

Robotic arms can improve the quality of life by aiding in activities of daily living (ADLs), such as self-care and pick-and-place tasks. Robotic arms can be used in fixed workstations, placed on mobile platforms, or mounted to wheelchairs. Research focuses both on building robot arms and the design of human-robot interaction. One topic of interest is retrieving an object from a shelf or floor (i.e., pick-and-place task), one of the most common ADLs [61]. Institutions investigating assistive robotic arms include Clarkson University [26], Delft University [65], Stanford University [71], University of Massachusetts Lowell [66], University of Pittsburgh [11], and TNO Science & Industry [65].

2.4.1 End-user Evaluations

Stanford University conducted an experiment with twelve spinal cord injury patients on two user interfaces for ProVAR, a vocational workstation [71]. After using each interface, each participant answered an evaluation questionnaire. Performance measures included open-ended responses to positive and negative questions on the robot's appearance, navigation, ease of use, error messages, complexity, usefulness, and functionality, and also on the participant's satisfaction.

The University of Pittsburgh evaluated the effects of a Raptor arm, a commercially available wheelchair-mounted robotic arm, on the independence of eleven spinal cord injury patients [11]. Participants first completed sixteen ADLs without the Raptor arm, then again after initial training, and once more after thirteen hours of use. At each session, the participants were timed to task completion and classified as *dependent*, *needs assistance*, or *independent*.

Clarkson University evaluated eight multiple sclerosis patients over five ADLs with and without the Raptor arm [26]. The participants in this study all required assistance with self-care ADLs. Participants were evaluated before and after training on the Raptor arm. At each session, the participants were timed to task completion and interviewed. They also rated the level of difficulty of task performance and the Psychosocial Impact of Assistive Devices Scale (PIADS) [15].

University of Massachusetts Lowell conducted an experiment of a new visual human-robot interface for the Manus Assistive Robotic Manipulator (ARM), a commercially available European robot arm. Eight individuals who used wheelchairs and had cognitive impairments participated in an eight week controlled experiment to control the robot arm in a pick-and-place task. Performance measures included time to task completion (i.e., object selection time), level of attention, level of prompting (based on measurement of functional independence [42]), and survey responses (i.e., preference of interface, improvements).

TNO Science & Industry and Delft University conducted a four person case study [65]. The end-users were people who use power wheelchairs and have weak upper limb strength and intact cognition. TNO Science & Industry evaluated their alternative graphical user interface for the Manus ARM. The performance measures included number of mode switches, task time, Rating Scale of Mental Effort (RSME) [78], and survey responses.

2.4.2 Analysis

As demonstrated by Tsui et al. [66], Tijsma et al. [65], and Fulk et al. [26], it is also important to account for the user's experience with respect to cognitive workload and mental and emotional state. The basis for the user's experience performance measure must be derived or adapted from an existing clinical measure.

In Tsui et al. [66] and Tijsma et al. [65], the participants were rated or rated themselves with respect to cognitive workload. In Tsui et al. [66], the level of prompting during a trial was a cognitive measure based on FIM, which is a scale that measures functional independence [42]. A person is rated on a Likert scale (1 = needs total assistance to 7 = has complete independence) on a variety of ADLs. FIM may also be applied as a cognitive measure to activities such as "comprehension, expression, social interaction, problem solving, and memory" [42]. In Tijsma et al. [65], RSME was used as a cognitive performance measure. RSME is a 150 point scale measuring the mental effort needed to complete a task, where 0 = no effort and 150 = extreme effort. The Standardized Mini-Mental State Examination [47] is another cognitive performance measures used in older adults.

In Fulk et al. [26], participants explicitly ranked the perceived difficulty of the task and their mental and emotional state were recorded using PIADS. PIADS is a twenty six item questionnaire in which a person rates their perceived

experience after completing a task with an assistive technology device [14]. It measures the person's feelings of competence, willingness to try new things, and emotional state. PIADS is well established and significantly used in the US and Canada [14]. An alternative emotional performance measure is the Profile of Mood States [41] used in Wada et al. [70].

2.5 External Limb Prostheses

Robotic prostheses can serve as limb replacements. Research institutions, such as Hong Kong Polytechnic University [38], Massachusetts Institute of Technology [3], Northwestern University [44], and the Rehabilitation Institute of Chicago [44], have investigated creating new robotic prosthetics and control strategies.

2.5.1 End-user Evaluations

The Rehabilitation Institute of Chicago (RIC) and Northwestern University conducted a clinical evaluation of six individuals who underwent targeted muscle reinnervation surgery [44]. After the upper limb prosthetic device was optimally configured for each patient's electromyography signals (EMG), functional testing occurred after the first month, third month, and sixth month. The functional testing was comprised of a series of standard tests: box and blocks, clothespin relocation, Assessment of Motor and Process Skills (AMPS) [23], and the University of New Brunswick prosthetic function [54]. Performance measures included time to complete task, accuracy, and AMPS score.

Researchers at the Massachusetts Institute of Technology conducted a clinical evaluation with three unilateral, transtibial amputees [3]. Data collection included oxygen consumption, carbon dioxide generation, joint torque, and joint angle. Kinematic and kinetic data were collected using a motion capture system for the ankle-foot prosthesis and unaffected leg. The resulting performance measures were metabolic cost of transport (using oxygen consumption as a parameter), gait symmetry between the legs, vertical ground reaction forces, and external work done at the center of mass of each leg.

Hong Kong Polytechnic University conducted a clinical evaluation with four transtibial amputees over the course of three consecutive days [38]. Data collected included motion capture and open-ended responses about the participant's comfort and the prosthesis' stability, ease of use, perceived flexibility, and weight. Stance time, swing time, step length, vertical trunk motion, and average velocity were derived from the motion capture data. Performance measures included ranking of the prostheses used (with respect to comfort, stability, ease of use, perceived flexibility, and weight), gait symmetry, and ground force reactions.

2.5.2 Analysis

Performance measures involving ADLs can be used in evaluating prostheses because ADLs include functions such a locomotion and self-care activities. Locomotion includes walking and climbing stairs, and self-care activities involve a high level of dexterity. Heinemann et al. [31] proposed the Orthotics and Prosthetics Users' Survey (OPUS). Burger et al. [7] in turn evaluated the Upper Extremity Functional Status of OPUS with sixty one users with unilateral, upper limb amputations and found that the scale was suitable for the measuring functionality of the population. The Upper Extremity Function Status is comprised of twenty three ADLs, rated in a Likert scale fashion (0 = unable to complete, 3 = very easy to complete. Similarly, AMPS is also comprised of ADLs but in a more flexible fashion; there are eighteen categories of ADLs with up to eleven choices within a category [2]. Another measure of quality of life is FIM, which is comprised of eighteen ADLs.

2.6 Stroke Rehabilitation

Robots are being investigated for gait training at Arizona State University [73], upper-limb recovery at RIC and Northwestern University [33], and wrist rehabilitation at Hong Kong Polytechnic University [34]. It is well documented that stroke patients regain most of their mobility through repetitions of task training [35]. Many researchers are investigating the use of robots as a way to augment current rehabilitation strategies for post-stroke patients.

2.6.1 End-user Evaluations

An example of a typical rehabilitation robot study using stroke patients was conducted by RIC and Northwestern University of the Therapy Wilmington Robotic Exoskeleton (T-WREX). The team conducted a clinical evaluation of twenty three stroke survivors over sixteen weeks comparing robot-assisted therapy to a traditional rehabilitation therapy regimen [33]. The researchers observed functional arm movement, quality of affected arm use, range of motion, grip strength, a survey of patient satisfaction of therapy, and the use of the affected arm in the home when not undergoing therapy. Performance assessments with or without the robot included Fugl-Meyer [25] and Rancho Functional Test for Upper Extremity [74] to measure ability to use the arm. In addition, they measured use of the arm outside of the experimental setting by using the Motor Activity Log [69], a self-report, to determine how the arm was used in the home. Finally, to assess the costs of using the robot, they measured the amount of time that the user needed assistance in order to use the T-WREX.

The early stages of rehabilitation robot development involves evaluations of the performance of the robot in a pilot setting. Some evaluations are users studies, where the robot is used with small number of users to determine what needs to be altered [73]. Performance measures used involve satisfaction surveys, measures of robustness, and analyses of the quantifiability of sensor data for clinical purposes. These measures are specific to the robot being evaluated, and in general cannot be used in the field in general.

The primary assessment of post-stroke rehabilitative robotics involves the use of clinical assessments of patient function. Discussed above was the Fugl-Meyer and Rancho Functional Test. However, there are many others used. At at Northwestern University and RIC, Ellis et al. [18] supplemented the Fugl-Meyer with several other measures, including the Chedokee McMaster Stroke Assessment, the Reaching Performance Scale, and the Stroke Impact Scale. At Hong Kong Polytechnic University, Hu et al. [34] used four other measures: the Motor Status Score (MSS, used to assess shoulder function) [22], the Modified Ashworth Scale (MAS, used to measure of increase of muscle tone) [4], the Action Research Arm Test (ARAT, used to assess grasp, grip, pinch, and gross movement) [17], and FIM (used to asses functionality in ADLs) [42]. These performance measures provide the picture of the clinical definition of effectiveness.

2.6.2 Analysis

Stroke rehabilitation is an established medical domain. Thus, the evaluations of these experiments use relevant clinical evaluations to determine the effectiveness of the robot-augmented therapy. The scope of rehabilitative robotics for patients post-stroke is quite large, ranging from upper-limb recovery to gait training and wrist rehabilitation. Even within a domain, the specific performance measures differ depending on the therapy and may not translate well to another sub-domain. For example, the MSS is applicable to the T-WREX [33] upper-arm rehabilitative aid, but not evaluating gait rehabilitation.

Functional evaluations, such as the Fugl-Meyer [27] and Wolf Motor Function [75], are crucial to comparing the effectiveness of robot-augmented therapies to one another in addition to comparing them with non-robot augmentations for current therapies. It is through these comparisons that robots can truly be evaluated as a rehabilitative device.

3. CONCLUSIONS

We believe that performance measures should be specific to the domain and relevant to the task. Domains with clear, well-established medical or therapeutic analogs can leverage existing clinical performance measures. For example, the Fugl-Meyer motor assessment, founded in 1975 [25], is popular when evaluating upper limb rehabilitation of post-stroke patients. Domains without strong therapeutic analogs can appropriately borrow clinical performance measures. Alternatively, they may draw inspiration from a clinical performance measure to create a new one or augment an existing one if none of the existing measures are appropriate [29].

Further, we believe that evaluations conducted with end-users should focus at least as highly on *human* performance measures as they do on system performance measures. By placing the emphasis on human performance, it becomes possible to correlate system performance with human performance. Celik et al. has taken the important first steps for stroke-rehabilitation by examining trajectory error and smoothness of motion with respect to Fugl-Meyer [10]. Similarly, Brewer et al. has used machine learning techniques on sensor data to predict the score of a person with Parkinson's disease on the Unified Parkinson Disease Rating Scale (UPDRS) [5, 19].

Existing performance measures for most of assistive robotic technologies do not provide sufficient detail for experimental and clinical evaluations. We provide a summary of performance measures used (see Table 1) and offer guidelines as to choosing appropriate and meaningful performance measures:

- Consult a clinician who specializes in the particular domain, if possible.
- Choose an appropriate clinical measure for the domain. A domain's "gold standard" will provide the best validity to clinicians.
- Choose an appropriate method to capture a participant's emotional and mental state.
- Consider an appropriate quality of life measurement.
- Administer the human performance measures at least before and after the experiment.
- Consider coding open ended responses, comments, and/or video.
- Concretely define each enumeration in a Likert scale.

By choosing meaningful performance measures, robotics researchers provide a common ground for interpretation and acceptance by the clinical community. In addition, the researchers of a given system are also given clear guidelines for how to observe and define performance of a given system.

Through this survey, we seek other well-established performance measures to apply to assistive robotic technologies. Common performance measurements will allow researchers to both compare the state of the art approaches within specific domains and also to compare against the state of the practice within the field outside of the robotics community.

4. ACKNOWLEDGMENTS

This work is funded in part by the National Science Foundation (IIS-0534364, CNS-0709296) and the Nancy Laurie Marks Family Foundation.

5. REFERENCES

[1] N. K. Aaronson, C. Acquadro, J. Alonso, G. Apolone, D. Bucquet, M. Bullinger, K. Bungay, S. Fukuhara, B. Gandek, S. Keller, D. Razavi, R. Sanson-Fisher, M. Sullivan, S. Wood-Dauphinee, A. Wagner, and J. E. Ware Jr. Intl. Quality of Life Assessment (IQOLA) Project. *Quality of Life Research*, 1(5):349–351, 2004.

[2] AMPS.com. Assess of Motor and Process Skills. In *http://www.ampsintl.com/tasks.htm*, 2008.

[3] S. Au. *Powered Ankle-Foot Prosthesis for the Improvement of Amputee Walking Economy*. PhD thesis, MIT, 2007.

[4] R. Bohannon and M. Smith. Interrater reliability of a modified Ashworth scale of muscle spasticity. *Physical Therapy*, 67(2):206–7, 1987.

[5] B. R. Brewer, S. Pradhan, G. Carvell, P. Sparto, D. Josbeno, and A. Delitto. Application of Machine Learning to the Development of a Quantitative Clinical Biomarker for the Progression of Parkinson's Disease. In *Rehab. Eng. Society of North America Conf.*, 2008.

[6] J. Brody. Prospects for an ageing population. *Nature*, 315(6019):463–466, 1985.

[7] H. Burger, F. Franchignoni, A. Heinemann, S. Kotnik, and A. Giordano. Validation of the orthotics and prosthetics user survey upper extremity functional status module in people with unilateral upper limb amputation. *J. of Rehab. Medicine*, 40(5):393–399, 2008.

[8] CALL Centre. Smart wheelchair. In *http://callcentre.education.ed.ac.uk/Smart_WheelCh/smart_wheelch.html*, 2008.

[9] T. Carlson and Y. Demiris. Human-wheelchair collaboration through prediction of intention and adaptive assistance. In *IEEE Intl. Conf. on Robotics and Automation*, 2008.

[10] O. Celik, M. K. O'Malley, C. Boake, H. Levin, S. Fischer, and T. Reistetter. Comparison of robotic and clinical motor function improvement measures for sub-acute stroke patients. In *IEEE Intl. Conf. on Robotics and Automation*, 2008.

[11] E. Chaves, A. Koontz, S. Garber, R. Cooper, and A. Williams. Clinical evaluation of a wheelchair mounted robotic arm. Technical report, Univ. of Pittsburgh, 2003.

[12] J. A. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46, 1960.

[13] R. Cooper, M. Boninger, R. Cooper, A. Dobson, J. Kessler, M. Schmeler, and S. Fitzgerald. Use of the Independence 3000 IBOT Transporter at home and in the community. *J. of Spinal Cord Medicine*, 26(1):79–85, 2003.

Table 1: Summary of Performance Measures Used in Assistive Robotic Technology Domains

Domain	Performance Measures Used
Autism	Behavior coding, correlate sensor modeling of behavior to human-rated behavior
Eldercare	Activities of daily living (SBBP, etc.), mood, stress (Standardized Mini-Mental State), quality of life (SF-36, 15-D, etc.)
Intelligent Wheelchairs	Number of hits/near misses, time on task, accuracy, gracefulness
Assistive Robotic Arms	Activities of daily living, time to task completion, mental state (RSME, Profile of Mood States), attention, level of prompting
Prostheses	Functional tests (AMPS, OPUS, FIM, etc.), measures of effort (oxygen consumption, etc.), accuracy, time to complete task, comfort, ease of use
Post-Stroke Rehabilitation	Functional measures (Fugl-Meyer, MSS, ARAT, FIM, Chedokee McMaster, Reaching Performance Scale, MAS, Wolf Motor, etc.), use of affected limb in home (Motor Activity Log, etc.)

[14] H. Day and J. Jutai. Piads in the world. In *http://www.piads.ca/worldmapshtm/worldmap.asp*, 2008.

[15] H. Day, J. Jutai, and K. Campbell. Development of a scale to measure the psychosocial impact of assistive devices: lessons learned and the road ahead. *Disability and Rehab.*, 24(1–3):31–37, 2002.

[16] DEKA Research and Development Corporation. DEKA Evolved Thinking. In *http://www.dekaresearch.com*, 2008.

[17] W. DeWeerdt and M. Harrison. Measuring recovery of arm-hand function in stroke patients: a comparison of the Brunnstrom-Fugl-Meyer test and the Action Research Arm Test. *Physiother Can*, 37(2):65–70, 1985.

[18] M. D. Ellis, T. Sukal, T. DeMott, and J. P. A. Dewald. ACT^{3D} exercise targets gravity-induced discoordination and improves reaching work area in individuals with stroke. In *IEEE Intl. Conf. on Rehab. Robotics*, 2007.

[19] S. Fahn, R. Elton, et al. Unified Parkinson's Disease Rating Scale. *Recent developments in Parkinson's disease*, 2:153–163, 1987.

[20] D. J. Feil-Seifer and M. J. Matarić. Robot-assisted therapy for children with autism spectrum disorders. In *Conf. on Interaction Design for Children: Children with Special Needs*, 2008.

[21] D. J. Feil-Seifer and M. J. Matarić. Toward socially assistive robotics for augmenting interventions for children with autism spectrum disorders. In *Intl. Symposium on Experimental Robotics*, 2008.

[22] M. Ferraro, J. Demaio, J. Krol, C. Trudell, K. Rannekleiv, L. Edelstein, P. Christos, M. Aisen, J. England, and S. Fasoli. Assessing the Motor Status Score: A Scale for the Evaluation of Upper Limb Motor Outcomes in Patients after Stroke. *Neurorehabilitation and Neural Repair*, 16(3):283, 2002.

[23] A. Fisher. AMPS: Assessment of Motor and Process Skills Volume 1: Development, Standardisation, and Administration Manual. *Ft Collins, CO: Three Star Press Inc*, 2003.

[24] B. Freeman. Guidelines for Evaluating Intervention Programs for Children with Autism. *J. of Autism and Developmental Disorders*, 27(6):641–651, 1997.

[25] A. Fugl-Meyer, L. Jaasko, I. Leyman, S. Olsson, and S. Steglind. The post-stroke hemiplegic patient. 1. a method for evaluation of physical performance. *Scandinavian J. of Rehab. Medicine*, 7(1):13–31, 1975.

[26] G. Fulk, M. Frick, A. Behal, and M. Ludwig. A wheelchair mounted robotic arm for individuals with multiple sclerosis. Technical report, Clarkson Univ., 2005.

[27] D. Gladstone, C. Danells, and S. Black. The Fugl-Meyer Assessment of Motor Recovery after Stroke: A Critical Review of Its Measurement Properties. *Neurorehabilitation and Neural Repair*, 16(3):232, 2002.

[28] D. Greenwood, W. Whyte, and I. Harkavy. Participatory Action Research as a Process and as a Goal. *Human Relations*, 46(2):175–192, 1993.

[29] J. Guralnik, E. Simonsick, L. Ferrucci, R. Glynn, L. Berkman, D. Blazer, P. Scherr, and R. Wallace. A short physical performance battery assessing lower extremity function: association with self-reported disability and prediction of mortality and nursing home admission. *J. of Gerontology*, 49(2):M85–94, 1994.

[30] K. Haigh and H. A. Yanco. Automation as caregiver: A survey of issues and technologies. In *AAAI-2002 Workshop on Automation as Caregiver: The Role of Intelligent Technology in Elder Care*, 2002.

[31] A. W. Heinemann, R. K. Bode, and C. O'Reilly. Development and measurement properties of the orthotics and prosthetics users' survey (opus): a comprehensive set of clinical outcome instruments. *Prosthetics and Orthotics Intl.*, 27(3):191–206, 2003.

[32] M. Hillman. Rehabilitation robotics from past to present - a historical perspective. In *IEEE 8th Intl. Conf. on Rehab. Robotics*, 2003.

[33] S. J. Housman, V. Le, T. Rahman, R. J. Sanchez, and D. J. Reinkensmeyer. Arm-training with t-wrex after chronic stroke: Preliminary results of a randomized controlled trial. In *IEEE Intl. Conf. on Rehab. Robotics*, 2007.

[34] X. L. Hu, K. Y. Tong, R. Song, X. j. Zheng, I. F. Lo, and K. H. Lui. Myoelectrically controlled robotic systems that provide voluntary mechanical help for persons after stroke. In *IEEE Intl. Conf. on Rehab. Robotics*, 2007.

[35] W. Jenkins and M. Merzenich. Reorganization of neocortical representations after brain injury: a neurophysiological model of the bases of recovery from stroke. *Progress in Brain Research*, 71:249–66, 1987.

[36] E. S. Kim, E. Newland, R. Paul, and B. Scassellati. A Robotic Therapist For Positive, Affective Prosody in High-Functioning Autistic Children. In *Poster Pres. at the Intl. Meeting for Autism Research*, 2008.

[37] H. Kozima and C. Nakagawa. Longitudinal child-robot interaction at preschool. In *AAAI Spring Symposium on Multidisciplinary Collaboration for Socially Assistive Robotics*, pages 27–32, 2007.

[38] W. Lee, M. Zhang, P. Chan, and D. Boone. Gait Analysis of Low-Cost Flexible-Shank Trans-Tibial Prostheses. *IEEE*

Trans. on Neural Systems and Rehab. Eng., 14(3):370–377, 2006.

[39] R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 140(5):1–55, 1932.

[40] C. Lord, S. Risi, L. Lambrecht, E. H. C. Jr., B. L. Leventhal, P. C. DiLavore, A. Pickles, and M. Rutter. The autism diagnostic observation schedule-generic: A standard measure of social and communication deficits associated with the spectrum of autism. *J. of Autism and Developmental Disorders*, 30(3):205–223, 2000.

[41] D. M. McNair, M. Lorr, and L. F. Droppleman. Profile of mood states. In *Educational and Industrial Testing Service*, 1992.

[42] MedFriendly.com. Functional independence measure. In *http://www.medfriendly.com/functionalindependencemeasure.html*, 2007.

[43] F. Michaud, T. Salter, A. Duquette, H. Mercier, M. Lauria, H. Larouche, and F. Larose. Assistive technologies and child-robot interaction. In *AAAI Spring Symposium on Multidisciplinary Collaboration for Socially Assistive Robotics*, 2007.

[44] L. Miller, K. Stubblefield, R. Lipschutz, B. Lock, and T. Kuiken. Improved Myoelectric Prosthesis Control Using Targeted Reinnervation Surgery: A Case Series. *IEEE Trans. on Neural Systems and Rehab. Eng.*, 16(1):46–50, 2008.

[45] MobileRobots Inc. Independence-enhaning wheelchair. In *http://www.activrobots.com/RESEARCH/wheelchair.html*, 2008.

[46] MobileRobots Inc. Robotic chariot. In *http://activrobots.com/robots/robochariot.html*, 2008.

[47] D. Molloy and T. Standish. A Guide to the Standardized Mini-Mental State Examination. *Intl. Psychogeriatrics*, 9(S1):87–94, 2005.

[48] B. Robins, K. Dautenhahn, and J. Dubowsky. Robots as Isolators or Mediators for Children with Autism? A Cautionary Tale. In *AISB05: Social Intelligence and Interaction in Animals, Robots and Agents*, 2005.

[49] B. Robins, K. Dautenhahn, R. te Boekhorst, and A. Billard. Robots as assistive technology – does appearance matter? In *IEEE Int. Workshop on Robot and Human Interactive Communication*, 2004.

[50] B. Robins, K. Dautenhahn, R. te Boekhorst, and C. Nehaniv. Behaviour Delay and Robot Expressiveness in Child-Robot Interactions: A User Study on Interaction Kinesics. In *Intl. Conf. on Human-Robot Interaction*, 2008.

[51] R. Robins, C. Fraley, and R. Krueger. *Handbook of Research Methods in Personality Psychology*. Guilford Press, 2007.

[52] G. Römer and H. Stuyt. Compiling a Medical Device File and a Proposal for an Intl. Standard for Rehabilitation Robots. *IEEE Intl. Conf. on Rehab. Robotics*, pages 489–496, 2007.

[53] T. Salter, F. Michaud, D. Létourneau, D. Lee, and I. Werry. Using proprioceptive sensors for categorizing interactions. In *Human-Robot Interaction*, 2007.

[54] E. Sanderson and R. Scott. UNB test of prosthetic function: a test for unilateral amputees [test manual]. *Fredericton, New Brunswick, Canada, Univ. of New Brunswick*, 1985.

[55] B. Scassellati. How Social Robots Will Help Us to Diagnose, Treat, and Understand Autism. *Robotics Research: Results of the 12th Intl. Symposium ISRR*, 28:552–563, 2007.

[56] F. Shic, B. Scassellati, D. Lin, and K. Chawarska. Measuring context: The gaze patterns of children with autism evaluated from the bottom-up. *Intl. Conf. on Development and Learning*, pages 70–75, 2007.

[57] R. Simpson. Smart wheelchairs: A literature review. *J. of Rehab. Research Development*, 42(4):423–36, 2005.

[58] R. C. Simpson. *Improved automatic adaption through the combination of multiple information sources*. PhD thesis, Univ. of Michigan, Ann Arbor, 1997.

[59] H. Sintonen. The 15-d measure of health related quality of life: Reliability, validity and sensitivity of its health state descriptive system. Working Paper 41, Center for Health Program Evaluation, 1994.

[60] C. Stanton, P. Kahn, R. Severson, J. Ruckert, and B. Gill. Robotic animals might aid in the social development of children with autism. In *Intl. Conf. on Human Robot Interaction*, pages 271–278, 2008.

[61] C. Stranger, C. Anglin, W. S. Harwin, and D. Romilly. Devices for assisting manipulation: A summary of user task priorities. *IEEE Trans. on Rehab. Eng.*, 4(2):256–265, 1994.

[62] T. Taha, J. V. Miró, and G. Dissanayake. Pompdp-based long-term user intention prediction for wheelchair navigation. In *IEEE Intl. Conf. on Robotics and Automation*, 2008.

[63] A. Tapus, J. Fasola, and M. J. Matarić. Socially assistive robots for individuals suffering from dementia. In *Human-Robot Interaction Intl. Conf., Workshop on Robotic Helpers: User Interaction, Interfaces and Companions in Assistive and Therapy Robotics*, 2008.

[64] C. Tardif, M. Plumet, J. Beaudichon, D. Waller, M. Bouvard, and M. Leboyer. Micro-analysis of social interactions between autistic children and normal adults in semi-structured play situations. *Intl. J. of Behavioral Development*, 18(4):727–747, 1995.

[65] H. Tijsma, F. Liefhebber, and J. Herder. Evaluation of new user interface features for the manus robot arm. In *IEEE Intl. Conf. on Rehab. Robotics*, pages 258–263, 2005.

[66] K. Tsui, H. Yanco, D. Kontak, and L. Beliveau. Development and evaluation of a flexible interface for a wheelchair mounted robotic arm. In *Intl. Conf. on Human Robot Interaction*, 2008.

[67] Univ. of Illinois Chicago. Power mobility skills checklist. In *http://internet.dscc.uic.edu/forms/0534.pdf*, 2008.

[68] US Food and Drug Administration. Guidance for industry, E6 good clinical practice: consolidated guidance. *Federal Register*, 10:691–709, 1997.

[69] G. Uswatte, E. Taub, D. Morris, K. Light, and P. Thompson. The Motor Activity Log-28: Assessing daily use of the hemiparetic arm after stroke. *Neurology*, 67(7):1189, 2006.

[70] K. Wada, T. Shibata, T. Saito, and K. Tanie. Effects of robot-assisted activity for elderly people and nurses at a day service center. *IEEE*, 92(11):1780–1788, 2004.

[71] J. Wagner and H. Van der Loos. Training strategies for the user interface of vocational assistive robots. *Intl. Conf. on Eng. in Medicine and Biology Society, EMBC*, 2, 2004.

[72] S. Wang, J. Keller, K. Burks, M. Skubic, and H. Tyrer. Assessing Physical Performance of Elders Using Fuzzy Logic. *IEEE Intl. Conf. on Fuzzy Systems*, pages 2998–3003, 2006.

[73] J. A. Ward, S. Balasubramanian, T. Sugar, and J. He. Robotic gait trainer reliability and stroke patient case study. In *IEEE Intl. Conf. on Rehab. Robotics*, 2007.

[74] D. Wilson, L. Baker, and J. Craddock. Functional test for the hemiparetic upper extremity. *American J. of Occupational Therapy*, 38(3):159–64, 1984.

[75] S. Wolf, P. Thompson, D. Morris, D. Rose, C. Winstein, E. Taub, C. Giuliani, and S. Pearson. The EXCITE Trial: Attributes of the Wolf Motor Function Test in Patients with Subacute Stroke. *Neurorehabilitation and Neural Repair*, 19(3):194, 2005.

[76] S. Wood-Dauphinee. Assessing quality of life in clinical research: From where have we come and where are we going? *J. of Clinical Epidemiology*, 52(4):355–363, 1999.

[77] H. Yanco. Evaluating the performance of assistive robotic systems. *Performance Metrics for Intelligent Systems (PerMIS) Workshop*, pages 21–25, 2002.

[78] F. Zijlstra. *Efficiency in work behaviour: A design approach for modern tools*. PhD thesis, Delft Univ., 1993.

Refining the Cognitive Decathlon

Robert L. Simpson, Jr.
Applied Systems Intelligence, Inc.
3650 Brookside Parkway, Suite 500
Alpharetta, Georgia, USA

bsimpson@asinc.com

Charles R. Twardy
OnLine Star, Inc.
2515 Red Cedar Dr.
Bowie, Maryland, USA

ctwardy@onlinestarinc.com

ABSTRACT
We argue that cognitive tests of intelligent agents should use modern intelligence theory to help ensure the test battery covers key aspects of cognition and decomposes them as diagnostically as possible. To this end we assess the recent BICA cognitive decathlon proposal [15] on the Cattel-Horn-Carrol (CHC) factor model of human intelligence [11], and suggest tests to fill the gaps. Some of those tests come from cognitive performance software developed by NTI [17 & 18]. Appealing again to CHC theory, we note remaining gaps and suggest known tests which can fill them.

Categories and Subject Descriptors
I.2.0 [**Artificial Intelligence**]: General – *cognitive simulation*

General Terms
Measurement, Performance, Experimentation, Human Factors

Keywords
Cognitive decathlon, Integrated cognitive agents, Intelligence theory, Cattel-Horn-Carrol Model (CHC), BICA

1. INTRODUCTION
The idea of a cognitive decathlon dates back at least to the early 1990s when: "Vere proposed creating a "Cognitive Decathlon" to create a sociological environment in which work on integrated cognitive systems can prosper. Systems entering the Cognitive Decathlon are judged, perhaps figuratively, based on a cumulative score of their performance in each cognitive "event." The contestants do not have to beat all of the narrower systems in their one specialty event, but compete against other well-rounded cognitive systems." [23, p. 460]. In Newell [16] as well as Anderson and Lebiere [1], the goal is to resist specialization, and return AI to a broad vision of integrated intelligence. Anderson and Lebiere said their article could be viewed as a proposal for events in the decathlon, with initial scores provided by ACT-R and classical connectionism. Recognizing that goal, DARPA's Information Processing Technology Office (IPTO) has been looking for a "Cognitive Grand Challenge" to rival the highly-successful vehicle Grand Challenge. In January 2005, IPTO held a Grand Challenge workshop. They commissioned MITRE to prepare a report [2] detailing why previous Grand Challenge proposals had failed. Participants at the workshop were given copies of the report. It concluded that a Grand Challenge must meet these criteria:

- Clear and compelling demonstration of cognition
- Clear and simple measurement
- Decomposable and diagnostic
- Ambitious and visionary, but not unrealistic
- Compelling to the general public
- Motivating for researchers

These in turn were explained in some detail. For example, to be "clear and compelling":

a. The test should be a proxy for a range of problems requiring cognitive capabilities.

b. The test should not be "game-able" or solvable by "cheap tricks"

c. It should not be solvable by brute force computation, alone, and it should not lend itself to idiot savant solutions.

d. It should require integration of multiple cognitive capabilities.

The best general categories were "Physical Activity", like RoboCup, and "Take a Test". The MITRE review placed a Cognitive Decathlon into the "Take a Test" camp. But what sort of test?

2. RIGHT IDEA, WRONG TEST
RPI's Selmer Bringsjord [2, 4 &5] proposed that AI agents simply be given the Wechsler Adult Intelligence Scale (WAIS), a popular IQ test. Others proposed the New York Regent's exams, or the California STAR tests, which are performance tests, not aptitude tests. Bringsjord calls the general approach "Psychometric AI". Unlike the all-or-none Turing Test, failure on a *broad* test like WAIS *is* diagnostic – the pattern of successes and failures on the questions will tell us what the agent does well and poorly.

Furthermore, intelligence tests have been used for clinical diagnosis, opening up intriguing possibilities for "diagnosing" agents – which we expect will show a great many deficits when compared with humans, perhaps in characteristic patterns. For example, Paul Harrison [12] argues that statistical methods based on the Gaussian distribution react "autistically" to outliers.

However, the choice of test matters. WAIS and other standard tests are deficient because they cover mainly memory and attention, things which computers are very good at [9]. Indeed,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

Sanghi and Dowe [21] claim to have written a simple 960-line Perl program that gets average human scores on various IQ tests[1], which is clearly a *reductio* for those tests, since the program earns its scores on arithmetic, logical, and pattern questions, not language or semantic ones.

We argue that this is a problem with the *particular* tests, not the general idea. As we discuss in the next section, modern CHC theory holds that human intelligence factors into at least 10 broad aptitudes, only 2 or 3 of which are exercised by standard IQ tests. Flanagan et al. [10, p.54] claim, "*The Wechsler verbal/nonverbal model does not represent a theoretically or empirically supported model of the structure of intelligence*." Our proposal, in a nutshell, is to make sure that all are covered. We want the individual tasks to map fairly cleanly onto cognitive "modules" – cohesive units of cognitive function. So we need to know what those units are.

3. CHC: MODERN PSYCHOMETRIC THEORY

The underlying model of intelligence has changed in the hundred years since psychometric testing began.

Flanagan et al. [10 & 11] describe the progression of intelligence theories & tests from single-factor theories to modern theories. They argue that the most well-supported theory of cognitive factors is "modern G*f*-G*c*" theory. Their version is CHC theory, so called because it merges Carroll's 8-factor model based on an exhaustive review of the factor analysis literature[2] with the Horn-Cattell 10-factor model. CHC theory has 10 broad cognitive abilities, each of which subsumes between 2 and 14 more narrow abilities.

The most common intelligence tests (Stanford-Binet, and the Wechsler tests, including WAIS) *do not* match up with modern CHC theory. They were originally designed for single-factor or dichotomous theories of intelligence, and later revisions – the SB:IV and WISC-III or WAIS-III – have only been slightly updated: they do not correspond to the current consensus on the most likely cognitive factor/ability boundaries. Indeed, according to a new study, the recently-updated WISC-IV "measures [only] crystallized ability (G*c*), visual processing (G*v*), fluid reasoning (G*f*), short-term memory (G*sm*), and processing speed (G*s*); some abilities are well-measured, others are not" [13].

3.1 The CHC Factors

The factors are [17], pp.30-31, 42-45):

Gf – fluid intelligence: what we use when faced with a novel task; inductive and deductive reasoning.

Gc – crystallized intelligence: acquired knowledge; "the sage"

Gq – quantitative knowledge, esp. arithmetical

Grw – reading/writing ability; basic written comprehension & expression

Gsm – short-term memory; storage for a few seconds; working memory

Gv – visual processing: including spatial orientation

Ga – auditory processing: "the ability to perceive, analyze, and synthesize patterns among auditory stimuli, and discriminate subtle nuances in patterns of sound" (p.42)

Glr – long-term storage and retrieval: long-term memory performance (not content)

Gs – processing speed: "attentive speediness"; on the order of 2-3 minutes (p.44)

Gt – decision/reaction time or speed: on the order of seconds or parts thereof

To get a good measure of human cognitive abilities, CHC theory suggests at least two independent tests for each of these 10 broad abilities, preferably using relatively unrelated "narrow" abilities from within the broad ability. For example, a measure of Fluid Intelligence, G*f*, might include a test on "General Sequential Reasoning" and on "Induction".

Mueller et al. [15] present a cognitive decathlon they designed for DARPA's Biologically Inspired Cognitive Architecture program, BICA. BICA sought to "develop comprehensive biological embodied cognitive agents that could learn and be taught like a human." Mueller et al. developed three complex Challenge Scenarios, 23 Cognitive Decathlon[3] tasks, and a Biovalidity Assessment. We are concerned with the Decathlon tasks.

In Table 1, we have labeled the BICA decathlon tests according to the CHC abilities *we* think they measure. Unsurprisingly, the Visual tests measure Visual Processing, G*v*. Some of the more challenging ones may also measure long-term memory, G*lr*, given that they involve remembering and recognizing places previously visited. Likewise, the advanced Search tasks involve Processing Speed (G*s*), Memory (G*lr* & G*sm*), and possibly some Fluid Intelligence (G*f*) when the agent must learn hiding patterns.

Language and Knowledge areas test long-term memory (G*lr*) and Crystallized Intelligence (G*c*) and possibly Short-term Memory (G*sm*).

Part II of Flanagan et al. [10] is basically a how-to guide for constructing a minimal but sound cross-battery test to measure CHC abilities. Since we will see the cross-battery approach again with the NTI "Armory", we should consider Flanagan et al.'s core design ideas (pp.210-213)

• Use good theory (e.g. CHC) so you have good factors. That way we are more likely to cut at the joints, getting scores for each separate cognitive faculty, which is especially important in clinical settings, such as diagnosing learning difficulties.

• Use relatively pure indicators. Ideally, each task should measure a single factor, otherwise our indicator (the task score) contains reliable variance that is associated with another CHC construct, leading to confusion and misdiagnosis.

• Conversely, Use at least two distinct, qualitatively different narrow abilities to measure a broad ability. Otherwise you're not

[1] They did not attempt the Wechsler tests or the Stanford-Binet tests, presumably because they are not publicly available. Nor should we expect their program to do well on them. As we see later, those tests have heavy language and semantic components. Sanghi and Dowe used the ACE, Eysenck tests 1—8, I.Q. Test Labs, test Testedich.de, and an I.Q. test from Norway. They scored poorly on the last 3 (59, 84, and 60) respectively. See their Table 1 (p.4) and references.

[2] Carroll reviewed 1500 studies covering 461 data sets.

[3] No one interprets decathlon literally to mean 10 tasks.

measuring G*c*, but just VL or LD or LS, etc. (The Wechsler Verbal Comprehension Index is actually quite good in this regard.)

CHC theory describes a relative complete taxonomy of *cognitive* functions, but does not directly test abilities like attention, kinesthetic ability, causal understanding, tracking & timing, etc.

Table 1: Our Assessment of CHC Abilities Measured by the BICA Decathlon

Task	Level	CHC*
Vision	Invariant Object Identification	G*v*
	Object ID: Size discrimination	G*v*
	Object ID with rotation	G*v*
	Visual Action/Event Recognition	G*v*, G*lr*
Search & Navigation	Visual Search	G*v*
	Simple Navigation	G*v*, G*s*
	Travelling Salesman Problem	G*v*, G*s*, G*lr*
	Embodied Search	G*v*, G*s*, G*lr*
	Reinforcement Learning	G*v*, G*s*, G*lr*, G*f*, G*sm*
Manual Control & Learning	Motor Mimicry	--, G*sm*, G*v*
	Simple (1-hand) Manipulation	--, G*sm*, G*v*
	Two-hand manipulation	--, G*sm*, G*v*
	Device Mimicry	--, G*sm*, G*v*
	Intention Mimicry	--, G*sm*, G*v*
Knowledge Learning	Episodic Recognition Memory	G*lr*, G*sm*?
	Semantic Memory/Categorization	G*lr, Gf,* G*sm*?
Language & Concept Learning	Object-Noun Mapping	G*c*, G*lr*
	Property-Adjective	G*c*, G*lr*
	Relation-Preposition	G*c*, G*lr*
	Action-Verb	G*c*, G*lr*
	Relational Verb-Coordinated Action	G*c*, G*lr*
Simple Motor Control	Eye Movements	--
	Aimed manual Movements	--

* If presented verbally, all tasks also involve some auditory processing G*a*, and language, G*c*.

Manual Control and Simple Motor Control tasks test abilities outside the scope of CHC theory. However, some of the manual control tasks involve integrating a series of visual actions (part of G*v*) and remembering short sequences (G*sm*).

The BICA materials suggest that task instructions are presented verbally, in which case they also test auditory processing (G*a*) extensively. Nevertheless, if we are looking for more complete tests of cognitive ability in artificial agents, then CHC theory suggests we may want to supplement these decathlon entries with some that exercise other abilities:

G*f* – fluid intelligence

G*rw* – reading/writing ability

G*t* – decision/reaction time

G*s* – processing speed

G*q* – quantitative knowledge

Of these, perhaps the hardest to measure is G*f*. But we can make some progress with the others.

We might also be less interested in innate aptitudes than in cognitive *performance.* After all, cognitive systems are supposed to learn, so we might assess their capabilities at a specific time. For such repeated testing, it would be helpful to have a large "armory" of tests which can be composed on the fly. That armory idea comes from O'Donnell et al [17 & 18] at NTI.

4. THE NTI ARMORY

In this section, we look at a cognitive *performance* evaluation "armory" developed and computerized by NTI, Inc. of Fairborn, OH; see O'Donnell et al. [17 & 18]. The original goal of this effort was to permit researchers to generate unique test batteries from the armory that would be tailored to the performance demands of specific jobs for people. NTI reviewed existing taxonomies including the CHC, and created a list of 18 broad "performance attributes" or cognitive functions such as Sustained Attention, Working Memory, Decision Making, Spatial Visualization, and Time/Velocity estimation. The creation of an armory of tests that have been described in terms of a single defined set of performance and cognitive skills is noteworthy for our Cognitive Decathlon purposes. The NTI report summarizes a vast literature, and took a big step towards *applying* that literature to cognitive metrics.

Their idea was to rate each potential test/task against all of the 18 cognitive functions, creating a characteristic signature vector. The NTI software then creates an "optimal" test battery on the fly to match the skills needed by a particular job.

According to O'Donnell[4], "Since the armory was developed as a cognitive performance assessment tool, it has been used as a state measure, and has never been validated or compared to trait measures [such as CHC]. Some of the tests in the armory may have some history in the area of intelligence testing, but this was not our focus." However, we can use CHC categories even to guide performance assessments.

We envision a future Cognitive Decathlon web site where researchers could test the capabilities of their integrated cognitive agents by having their agents examined via administration of all or a subset of these and perhaps other tests. The advantage of this type of Decathlon is fairly straightforward. First, like CHC-based tests, these performance tests have a built-in comparison to human performance. Second, the tests are well understood within the

[4] Personal communication.

psychological testing community. Third, the average "man on the street" can understand the intuition of administering the same test to natural and artificial intelligent systems.

The NTI Armory is not sufficient for BICA's goals, particularly because someone could use a collection of subroutines each of which was optimized and specialized for a particular test, rather than an integrated cognitive agent. But our goal is to refine the BICA Decathlon, not replace it. CHC theory has led us to look for diagnostic tests that fill the gaps, and decompose relatively cleanly.

The specific tests in the NTI Armory are listed in Table 2. We provide a short description of some of these tests below.

Some tests – Dichotic Listening, Stroop Visual, and Visual Vigilance – depend on fine details of human cognition that we do not expect to see duplicated in machine cognition. For example, the famous Stroop test presents a color word like "red" in another color (as we did here, for those viewing this in color). The participant must try to name the color of the word, but humans find that difficult, and are prone to mistakenly say the conflicting color name from the word itself. (There is no difficulty with non-color words like "car".)

Table 2: The NTI Armory Tests

Continuous Memory	Reaction Time - Choice
Dichotic Listening	Reaction Time - Simple
Digit Span	Relative Motion (Join-Up)
Manikin (Low/High)	Sternberg - Letters
Match to Sample	Sternberg - Symbols
Math Processing	Stroop - Visual
Motion Inference	Tower of Hanoi (Low /High)
Novascan C (1 + 7)	Tracking - Pursuit
Precision Timing	Tracking - Unstable
Peripheral Information-Processing	Visual Vigilance
Rapid Decision Making	Wisconsin Card Sorting

In fact, a CMU-led team [8] showed that a simple neural net would generate human-like Stroop results so long as you trained it with more word-naming than color-naming tasks, so that word-naming was relatively automatic. That matched MacLeod and Dunbar's [14] showing that color naming itself was relatively automatic when paired with the even less well-trained task of shape naming, and that sufficient training on shape naming reversed that effect. So tests like the Stroop task are very good candidates for systems that learn like humans do.

Understanding the instructions may well be harder than taking the test itself. We do not want special-purpose agents that already know the task, so we must be able to describe the task to a general-purpose agent. The BICA proposal presumes a fair bit of verbal natural language processing (NLP). At minimum, agents would need to parse a formal language which can say, "You will get a task like this, and must remember x. Then you will get a distracter task where you have to do y, after which you will be asked to compare u and v to x."

Many of the NTI tasks put a lot of effort into directing human attention. As BICA imitation tasks like "do this" acknowledge, the ability to manage attention and indexical reference so the agent can have its attention directed is itself already a major achievement. The first round is likely to present tasks as the full set of percepts.

Let us now consider a few of the NTI tests. As our goal is not necessarily to exactly duplicate human performance, we should be prepared to use "staircase" techniques (e.g. [22]) to quickly find the system's limits, and then explore them. We recommend adding such features to almost all of the tests.

5. SOME NTI TESTS

5.1 Test 1: Continuous Memory

The continuous memory test consists of a random series of visual presentations of numbers which the operator must encode in a sequential fashion. As each number in the series is presented for encoding, a probe number is presented simultaneously. The operator must compare this probe number to a previously presented item at a pre-specified number of positions back in the series. Once the operator has made the appropriate recall, he or she must decide if that item is the same as, or different from, the probe number. Thus, the task exercises working memory functions by requiring operators to accurately maintain, update, and access a store of information on a continuous basis. Task difficulty is manipulated by varying the length of the series which must be maintained in memory in order to respond to recall probes.

Potential for Cog Decathlon: Obviously this is easy for a special-purpose program. However, it may still be a challenge for cognitive architectures like ACT-R which deliberately have a very limited working memory. For example, a recurrent neural net model will have a Markov horizon because computational constraints limit the chain depth. Also, any system operating "in the loop" with rich perceptions will be forced to limit attention and recall. We may have to bar some systems based on architecture, unless we can rely on a system gaming this test to fail another. Presenting the input "visually" (as images or feature vectors) is harder for "honest" systems, but still simple for special-purpose programs: just pipe the output of a trained digit classifier to a simple list processor, for example.

5.2 Test 4 and 5: Manikin

The Manikin Test as described here is a derivative of a task originally developed by [3] and popularized by the UTC-PAB [20]. The test is designed to index ability to mentally manipulate objects and determine orientation of a given stimulus. In this version, the test shows a vehicle such as an aircraft or a car. To one side is a male figure, and to the other side is a female figure. These figures and the object are lined up horizontally. Below the object and figures is a single query figure (male or female). The agent must determine whether the figure matching the query figure is to the right or the left of the vehicle, *in the vehicle's frame of reference* . The figures may appear either upright or upside down and facing either toward or away from the subject. The 16 combinations of orientation, stimuli and side are pseudo-randomly ordered. The number of trials selected for a given training or test session is under experimenter control. The NTI software uses stock images of the front or back of a sports car, and schematic man or woman figures (as you might see on restrooms, but in uniform). Unpracticed humans find most of the trials to be easy, but not when the vehicle is presented upside down and backwards. This test is considered to have two states in

which somewhat different cognitive skills are measured. In the LOW TRAINING condition, the subject is familiar with the task, but has not reached a level of "automaticity." In the HIGH TRAINING condition, the subject is so practiced that a different group of cognitive skills, such as procedural and working memory, are used to process the task.

Potential for Cog Decathlon: This requires visual presentation and an understanding of handedness. It is likely a test that machines would find difficult to do as fast as humans, since it involves a lot of image rotation, an understanding of "front/back/side", spatial awareness, and typical shapes of people and vehicles. It is still possible for a special-purpose program, but less so if the objects are chosen from a very large (and possibly unknown) set, and if we can apply obscurations to the image. Presuming a time limit, we can adapt this easily to test machines by reducing the time limit using a binary search. The metric for humans and machines could be the time-limit where they get 50% wrong.

5.3 Test 8: Motion Inference (Time/Velocity Estimation)

During the task, the subject sees a moving stimulus traversing a curved path. Approximately half way to a hash mark, the stimulus disappears. The subject's task is to determine when the stimulus, moving at a constant speed, would have reached a hash mark located in a random position along the curved path. The hash mark range of positions is set in the test's configuration program and can be anywhere between the beginning and end of the curve, but typically located in the last third of the path. The subject must infer how long the stimulus would take to reach the hash mark. The response required is a button press when the subject believes the stimulus would have reached the mark. The distracter is a simple "semantic" task. When the stimulus disappears, a series of four letters of the alphabet appear on the screen. The subject must immediately decide whether any of the letters are vowels. This decision is indicated with a response using a designated button on the response device (e.g., mouse). In effect, this interpolated task acts as a distracter to the subject in estimating the inferred motion. In this way, the subject is precluded from using methods such as counting, tapping, or singing to infer the motion. Once the response to the letters is made, the subject is required to estimate when the stimulus would have reached the stop point, and is to indicate this by pressing the designated button. This task really seems to require some practice.

Potential for Cog Decathlon: In addition to an interesting tracking task in itself, the distraction task forces us to consider how we present the directions to the cognitive agent. This is a good test, because it requires division and direction of attention between different tasks demanding different capacities. The distraction task could easily be gamed (if x in vowels: …), but once again, we seek other ways to prevent gaming. The visual tracking task should provide a challenge, and any agent capable of doing the visual tracking (given, say, a series of pixel planes) should be able to do visual inference of letters, which will make the task somewhat less trivial.

5.4 Test 9: NovaScan C

This test represents a special adaptation of the "multi-tasking" approach. Generally, in multi-tasking efforts the subject is free to adopt any strategy he or she wishes in order to achieve a final composite performance. This introduces some degree of difficulty in analyzing the task, particularly in diagnosing the nature of any decrements observed. NovaScan attempts to eliminate this ambiguity by using what has been called a "directed attention" rather than a "divided attention" paradigm. In the directed attention approach, the subject is still required to multiplex between two or more skill requirements. However, instead of being free to attend to each one whenever he/she wishes, the test directs the person to the test that must be attended to at any given time. This is done by having only one test appear on the screen at a time. In effect, the person has to keep one test's requirements in memory, while actively performing another test. In this way, the subject's strategy is highly constrained, and it is easier to determine where a cognitive decrement or improvement has occurred. Of course, it is still possible to introduce more than one task requirement at a time, as long as the demands of the tasks can be controlled.

NovaScan is a generic paradigm. There are many tests that can be introduced into it, just as there are many tests that can be used in the traditional divided attention approach. The present application of NovaScan, (C) uses two of the individual tests described elsewhere in the armory (Manikin and Continuous Memory). In each, a task appears on the screen (e.g., Manikin) and the subject must perform it for some period of time. At irregular intervals, this task is replaced by another task (e.g., Continuous Memory), and the subject must process this for some period of time. When that task is again replaced with the first (Manikin) task, the subject must remember the demands of the second task (Continuous Memory) while again performing the first. This alternation continues for some defined period of time or number of presentations. In addition to these demands, the subject typically must monitor a dial in which the pointer is moving at a constant rate, but in an inconsistent manner (the Dial Task). The subject must detect when the dial has gone into a "danger" zone. To do this, the subject must establish a scan rate for the dial that optimizes the opportunity to detect a danger indication, while allowing time to optimally process the other tests. This paradigm therefore approximates complex real-world tasks where two or more basic cognitive or psychomotor requirements must be attended to, and an optimal multiplexing strategy must be adopted based on current experience.

Potential for Cog Decathlon: The NovaScan paradigm offers a very flexible way to help prevent spoofing, since the agent must not only be able to do single tests, but switch between them. We should expect agents to have to learn the new combination, and then improve. Consider, for example, that learning to drive involves this kind of sequential directed attention, where subtasks are gradually automated. In fact, such considerations drove some of the early rule-generating systems. The instructions may still be the hardest part.

5.5 Test 12: Rapid Decision Making

The basic concept of this test is to present the subject with a display containing three "areas" that represent three levels of unspecified "danger". These areas are clearly marked with respect to the level of danger. At various times, symbols appear on the display indicating that a "vehicle" has entered into one of the areas. The vehicle appears as one of three types of symbol. One type clearly indicates that the vehicle poses minimal threat;

another indicates that the vehicle is a clear threat, and third type indicates that it is uncertain whether the vehicle is friend or foe. The subject's task is to decide on the level of threat, based on the type of vehicle and the area of the display in which it is located, and to make a differential response based on that decision. This is to be done as rapidly as possible. The test is paced so that only a short period of time is available to make the decision before the next stimulus appears, and this interval may be adjusted by the experimenter. In essence, this test is a complex choice reaction time test where higher level cognitive processes must be used to determine what the stimulus means, and where there is a complex response selection.

Potential for Cog Decathlon: This is obviously useful for a missile defense scenario. It naturally suggests a game where score is determined by a payoff matrix, where the values can be chosen at the start of the test. The main control variable is pacing. Other possibilities include number of locations and/or vehicle types. The uncertain vehicle is a nice touch, because optimal reward will require some utility calculations based on degree of certainty. The degree of uncertainty could be made to vary in a clear way, such as merging shape, or fading the image, or even just tagging it as uncertain and at what level.

5.6 Test 24: Wisconsin Card Sorting

In the armory's computerized version of this test, four groups of figures (called "key cards") are shown to the subject on the screen. Each card shows different shapes, and a different number of shapes. Also, the shapes on each card are a different color. They are typically arranged as shown in Figure 1, and this pattern of colors, shapes, and number is the default option.

Figure 1: Example shapes for the four "key" cards

The participant is then presented a series of "test cards" containing various combinations of the shapes, colors, and number of objects shown in the key cards. The task is to decide which key card "matches" the presented test card. Since there are three different ways a test card can match a key card (by color, shape, or number) the subject must decide which sorting criterion to use. No rule is given to the subject for matching cards. However, feedback is given for each attempted match on whether it was "right" or "wrong". This is based on a pre-established sorting criterion. Once the subject discovers the correct criterion and answers "correctly" six consecutive times, the criterion is switched to one of the other two. If the subject appears to be answering correctly for any number lower than six, and then makes an error, the count starts over (i.e., the subject must answer correctly six consecutive times). Normally, the types of shift in criterion are specified in the default condition. Among many dependent measures that may be collected, the number of matching categories completed and the number of "perseverative" errors (i.e., the number of matches attempted in which the same incorrect matching criterion was used) are perhaps most common. Perseverative errors indicate difficulty in changing approaches to problem solving, or inhibiting previously learned approaches. The task measures first the ability of the subject to conclude that there are 3 possible criteria by which to match the cards, and then assesses cognitive flexibility by requiring the subject to switch criteria to continue being successful at the task. The test is a good measure of adaptability and avoidance of perseveration.

Potential for Cog Decathlon: This card sorting task is a good test of a cognitive agent's ability to perform rule induction. It has the added twist that the rule has to be revised under some executive control when the rule is changed. An important control for comparison to human performance is the degree that the human has had prior experience with rule learning. This can be controlled somewhat with the prior knowledge that the cognitive agent has about the card representations. Like some of the other tests, e.g., the "join up" and "pursuit" tasks, if the agent has the appropriate learning mechanism this task should be easy and the advantage should be with the cognitive agent. Noise in the representation or other distracters could be added but then the difficulty goes up for the human subject perhaps beyond performance.

6. GAPS IN THE NTI ARMORY

The NTI Armory offers only partial coverage of all the potential dimensions of cognition. NTI's expert panel rated each test across all 18 of their defined cognitive functions. At least four dimensions of cognition are not well represented: Problem Sensitivity, Math Functioning, Language/Semantics, and Declarative Memory.

Problem sensitivity is the ability to recognize that a problem exists, not necessarily the ability to solve it. It is valued among emergency responders. Math functioning and language/semantics are self-explanatory. Declarative memory is memory of things from more than 20 minutes ago, hence a form of LTM but distinct from procedural LTM. So it would include both episodic (time-based) memory and other declarative (fact-based) memory.

We have identified several potential supplementary tests. To cover fact-based declarative memory, we could include the California Verbal Learning Test (CVLT). Also, using Table 5.1 from Flanagan et al. [10], the following items from common intelligence batteries are strong tests of LTM (G*lr*):

Tests of Associative Memory

- WJ-R Memory for Names
- WJ-R Delayed Recall Memory of Names
- WJ-R/III Visual-Auditory Learning
- WJ-R/III Delayed Recall Visual-Auditory Learning
- KAIT Rebus Learning
- KAIT Delayed Recall Rebus Learning

Tests of Ideational Fluency, Naming, or Declarative Memory

- WJ-III Retrieval Fluency (Ideational Fluency)
- WJ-III Rapid Picture Naming (Naming Facility)
- Visual paired-comparison (Declarative Memory)

Specific tests of language/semantics include the California Verbal Learning Test (CVLT) or any number of tests of crystallized intelligence from the WAIS and other intelligence batteries[5]:

- DAS Similarities

[5] List based on Flanagan et al. 2000a, Table 5.1.

- SB:IV Verbal relations
- SB:IV Comprehension
- SB:IV Absurdities
- WJ-III Verbal Comprehension
- WAIS Verbal Comprehension

To remedy the shortcomings in the Math Functioning dimension, we could include either the Wechsler-Arithmetic which contains 14 mental arithmetic brief story problems, the WJ-R/III Calculation and Applied Problems tests, or any number of similar tests of basic arithmetic. Story problems will require some language ability of course, while straight calculation tests could be trivial, if the agent can encode them directly into parseable code. We have been unable to identify a suitable test for problem sensitivity. An incident commander we spoke with suspects that this ability is usually assumed for emergency responders: training exercises will often require the responder to say, "Scene survey" and wait for the instructor to say "The scene is secure" or else fail, but no actual survey is performed.[6]

7. EPISODIC MEMORY

Episodic memory is declarative long-term memory (Glr) specifically associated with times or events – episodes – in an agent's history. It is a form of associative memory. Our ability to organize memories by events, such as yesterday's meeting or our last summer vacation, depends on (or exemplifies) episodic memory. The NTI test armory is weak here, especially because tests of long-term memory require, on their interpretation, 20-minute intervals. One of these tests would have to be paired with other tests that ran in the interval. However, the BICA tests are relatively strong. For example, one task requires the agent to remember which objects they have already encountered in which rooms.

There are some dedicated episodic memory tests. One is the University of Southern California Repeatable Episodic Memory Test [19]. It consists of:

...seven different lists, each co mposed of 15 semantically unrelated, high-frequency nouns. The words are presented in a different order on three study-test trials. After each study trial the subject recalls the words in any order. The test takes about 10 min to administer and score. The recall protocol can be scored for (a) global mnemonic efficiency, (b) primary and secondary memory, (c) subjective organizati on, (d) recall consistency and (e) recall as a function of serial position.

It has been applied in several clinical papers (for example, to Alzheimer's patients) to determine the pattern of memory deficiencies. Although it does not specifically require a 20-minute delay, it could. It is designed to be repeatable, and could be made even more so by using WordNet to generate lists of the required type on demand.

However, it requires a fair bit of semantic knowledge. Participants are expected to recall things by category, for example. Eventually, we want agents to be able to do this. In the meantime, however, we need a non-linguistic test of episodic memory. Several researchers in animal behavior (ethology) have been working on the problem.

[6] Bob Koester, personal communication.

Figure 2: Western Scrub-Jay caching or retrieving food. Scenery provides context for "episodic" memory. From N.S. Clayton, http://www.psychol.cam.ac.uk/cplcl/. Used with permission.

Researchers at Cambridge have been investigating food-cache storing in corvids, especially scrub-jays [6 & 7]. The design, as shown in the photo (Figure 2), involves a set of cache locations cued by features of the environment. The experimenters then compare the performance of caching birds, observing birds, and naïve birds on retrieval, attempting to control for various efficient search strategies.

A similar experiment could be set up as a software task (like those in the NTI armory), using successive still images or video. A simpler version could use very "cartoon" locations and stimuli. The agent being tested can then be asked where agent Green placed the items, or agent Blue. This would allow us to test episodic memory for agents that do not yet meet all of the BICA presumptions. (Of course, with cognitive agents it need not be food caching!)

8. CONCLUSION

In this paper we have reviewed the history of the idea of a cognitive decathlon as a methodology for testing the capabilities of an intelligent agent. We argued that the CHC criteria summarized by Flanagan et al.'s [10] presentation of modern intelligence theory – the Cattel-Horn-Carrol model lead nicely to specific cognitive categories for a decathlon. We also looked at a specific set of tests, the "NTI Armory," as candidates for a potential battery of tests. Admittedly, the NTI Armory offers only partial coverage of all the potential dimensions of cognition. We still need to complete the battery of tests for missing dimensions of cognition and describe how the tests would be administered to agent subjects.

9. ACKNOWLEDGMENTS

This paper is partially based upon work funded by DARPA and any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the U.S. Government.

10. REFERENCES

[1] Anderson & Lebiere 2003. The Newell Test for a theory of cognition. BBS 26:6, p.5.

[2] Bayer, S., Damianos, L., Hirschman, L. & Strong, G. A Summary of Previous Grand Challenge Proposals for Cognitive Systems. The MITRE Corporation. Prepared for DARPA IPTO, September 2004 Version 1.4 http://www.mitre.org/work/tech_papers/tech_papers_05/05_0947/index.html

[3] Benson, A. J., Gedye, J.L. and Jones, G. M. Disorientation in flight due to a covert vestibular disorder, with associated generalised, muscular tension. *Aerosp Med.* 1963 Jul;34:649–654.

[4] Bringsjord, S. & Schimanski. B. 2003. "What is Artificial Intelligence? Psychometric AI as an Answer," Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03) (San Francisco, CA: Morgan Kaufmann), pp. 887-893.

[5] Bringsjord, S. & B. Schimanski 2004. " 'Pulling It All Together' via Psychometric AI," Achieving Human-Level Intelligence Through Integrated Systems and Research Technical Report FS- 04-01 (Menlo Park, CA: AAAI Press), pp. 9-16.

[6] Clayton, N. S. and A. Dickinson. 1998. Episodic-like memory during cache recovery by scrub jays. *Nature* 295, 272—278.

[7] Clayton, N. S. and J. M. Dally and J. D. Gilbert and A. Dickinson. 2005. Food caching by Western scrub-jays (Aphelocoma Californica): a case of prospective cognition. *J. Exp. Psychol: Anim. Behav. Proc.* 31, 115—124.

[8] Cohen, J.D., Servan-Schreiber, D. and McClelland, J. 1992. A parallel distributed processing approach to automaticity. *American Journal of Psychology*, 105(2), 239—269.

[9] Cohen, P. R. 2005. If Not Turing's Test, Then What? AI Magazine 26(4): Winter 2005, 61–67

[10] Flanagan, D. P., McGrew, K. S. & Ortiz, S.O. 2000a. The Wechsler Intelligence Scales & Gf-Gc Theory: A Contemporary Approach to Interpretation. Allyn & Bacon, Boston.

[11] Flanagan, D. P., Ortiz, S.O., & McGrew, K. S. 2000b. Contemporary issues in intellectual assessment. Powerpoint presentation.

[12] Harrison, P. F. 2005. Is Autism a sensitivity to outliers? http://www.logarithmic.net/pfh/autism

[13] Keith, T., Fine, J., Taub, G., Reynolds, M. & Kranzler, J. 2006. Higher-Order, Multi-Sample, Confirmatory Factor Analysis of the Wechsler Intelligence Scale for Children—Fourth Edition: What Does it Measure? *School Psychology Review, 35 (1)*, 108-127. (From a review posted on McGrew's website, http://intelligencetesting.blogspot.com/2006/04/what-does-wisc-iv-measure-chc.html).

[14] MacLeod, C. M. & Dunbar, K. 1988. Training and Stroop-like interference: Evidence for a continuum of automaticity. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 14, 126—135.

[15] Mueller, M. S., Jones, M., Minnery, B. S. and Hiland, J. M. H. 2007. The BICA Cognitive Decathlon A Test Suite for Biologically-Inspired Cognitive Agents. BRIMS 2007, Norfolk, VA. http://obereed.net/docs/MuellerBRIMS2007.pdf

[16] Newell, A. (1990). Unified Theories of Cognition. Cambridge, MA: Harvard University Press.

[17] O'Donnell, R. D., Moise, S. and Schmidt, R. 2004. Comprehensive computerized cognitive assessment battery. Arlington, VA: Office of Naval Research; 2004. Contract No. N00014–01-C-0430. NTI, Inc.

[18] O'Donnell, R. D., Moise, S. and Schmidt, R. Generating performance test batteries relevant to specific operational tasks. *Aviation, Space, and Environmental Medicine* 2005; 76(7, Suppl.):C24–30. http://www.ingentaconnect.com/content/asma/asem/2005/00000076/A00107s1/art00007

[19] Parker, E. S., Eaton, E. M., Whipple, S.C., Heseltine and Bridge. 1995. *J Clin Exp Neuropsychol*. Dec;17(6):926-36. University of Southern California Repeatable Episodic Memory Test.

[20] Perez, W. A., Masline, P. J., Ramsey, E. G. and Urban, K. E. Unified Tri-Services Cognitive Performance Assessment Battery: Review and Methodology, Technical Report Apr 1984-Feb 1987. Systems Research Labs Inc., Dayton Ohio.

[21] Sanghi, P. and. Dowe, D. L. 2003. A Computer Program Capable of Passing I.Q. Tests, in P P Slezak (ed), Proceedings of the Joint International Conference on Cognitive Science, 4th ICCS International Conference on Cognitive Science & 7th ASCS Australasian Society for Cognitive Science (ICCS/ASCS-2003), 13-17 July 2003, Sydney, NSW, Australia, pp 570-575. http://www.csse.monash.edu.au/~sanghi/iq.html

[22] Watson, A. B. & Pelli, D. G. (1983) QUEST: a Bayesian adaptive psychometric method. *Percept Psychophys, 33 (2),* 113-20. http://www.psych.nyu.edu/pelli/pubs/watson1983quest.pdf

[23] Vere S. 1992. Planning, Encyclopedia of AI, Shapiro (ed.), Second Ed.

Using Metrics to Optimize a High Performance Intelligent Image Processing Code

Scott Spetka
SUNY Institute of Technology
and ITT Corp.
Utica and Rome, NY, USA
scott@cs.sunyit.edu

Susan Emeny
ITT Corp.
775 Daedalian Drive
Rome, NY, USA
Susan.Emeny@rl.af.mil

George O. Ramseyer
Richard Linderman
Air Force Research Laboratory
Rome, NY, USA
George.Ramseyer@rl.af.mil

***Abstract*—** Optimizing the execution of intelligent codes on high performance computer's (HPC's) has become more challenging as the numbers of processors increases. Single processors in many HPC's have been replaced with dual processors, and more recently multiprocessors. This, combined with the inherent complexities of multi-core processors, has made the processing of intelligent codes even more complex on the latest HPC's. The coming availability of thousands of processors in more affordable medium sized HPC's offers the potential for improved performance for codes that can scale sufficiently to take advantage of hundreds of teraflops. Additionally, techniques for harnessing the performance potential of multi-code processors require the appropriate location of data in shared memories, or even shared level-2 caches, and can afford additional orders of magnitude performance increases. The key to designing code that uses the available teraflops wisely is an understanding of the application's behavior. For intelligent systems, whose behavior may depend on heuristics evaluated at runtime, measurements and profiling runs provide the basis for system design decisions, regarding distribution of data and processing. This paper focuses on the metrics needed to optimize intelligent codes, and how a specific image processing code was instrumented to produce the required metrics.

***Keywords*:** *HPC, Latency, Performance, PCID, Optimization, FFT*

I. Introduction

Because of recent developments in high performance computing, many codes that were optimized for older HPC systems need to be optimized again to take advantage of current HPC technologies. In addition, intelligent codes require techniques for dynamically adapting to changes in system behavior to optimize performance. This leads to variations in processing approaches, depending on the inputs to the executing code. In these cases, the selection of a processing technique can depend not only on the approach taken, but also on how well it would scale across available processing resources. It would also depend on the underlying system architecture and configuration. Distribution of processing on an HPC Linux cluster, built from SONY Playstation IIIs, would be different from a cluster of dual quad-core Xeon systems.

At a lower level, improvements in networking and multi-core processors have introduced the possibility of remote DMA for networks [1] and shared level-2 cache, in addition to shared memory. Codes that were optimized for 100Mb Ethernet or even gigabit Ethernet can benefit from rethinking algorithms for Infiniband, where network latency can be as little as 10 usec. Shared cacheing on dual-core and quad-core processors are being implemented to allow limited sharing of level-2 cache [2].

To study tradeoffs and optimization techniques, we started with a multiple frame blind deconvolution algorithm, called PCID (Physically Constrained Image Deconvolution) [3]. We profiled and measured the code to develop optimization techniques that could be used for improving its performance. We used a suite of test cases, representative of a broad range of inputs that we expect to find for real situations. In section 2, we introduce the test systems. In section 3, higher level measurements help to determine the percentage of peak performance that is currently being achieved by the code and the parts of the code that offer the best potential rewards for optimization. Section 4 focuses on lower level code behavior of networking and cache performance. Section 5 presents our conclusions.

II. Test Hardware and software

This project is focused on two test HPC systems and the PCID software. The test systems both run the Linux operating system. One of the test systems is Jaws, a 5,120 processor

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

Dell PowerEdge 1955. Each node contains two dual-core Xeon 3.0 GHz 64-bit Woodcrest CPUs, 8GB of RAM, and 72GB of local disk space. The nodes are connected via Cisco Infiniband, running at 10Gbits/sec (peak).

Figure 1. Jaws at MHPCC.HPC.MIL.

The other HPC system has 12 head nodes, each with two quad-core Xeon 3.0 GHz 64-bit processors and 32GB of RAM. The head nodes are interconnected with Infiniband. Each head node has 24 SONY Playstation III backnodes connected through their gigabit Ethernet ports. Each of the 288 Playstations has a Cell processor [4] with a PowerPC and 6 Synergistic Processing Elements (SPEs).

The PCID software is used for image deblurring. It uses a sequence of frames to compute a deblurring function that can be applied to input images to create a corrected image. The code is processor intensive, and offers choices for processing that allow users to select processing approaches to adapt the algorithm to their inputs. Optimizing PCID for out test environments requires determining appropriate test cases to exercise the PCID code, especially for cases that currently require longer executions. These cases may show the biggest improvement if our optimization succeeds.

III. High Level Measurements

We measure the high level behavior of the system, running the PCID software, to determine the amount of time that is spent in key, processor intensive computation. The time is measured, both as a percentage of total execution time and as a measured elapsed time spent in each of the computationally complex functions. We expect that Fast Fourier Transforms (FFTs) will account for a high percentage of the processing time for an image processing algorithm that does a lot of matrix multiplication.

The *gprof* profiling tool [5] was used to generate call graphs and percentages of time spent executing each of the PCID functions. In addition, we used the Multi-Processing Environment (MPE) [6] to create graphical displays for communications between processors for the parallel execution of the PCID code.

We also instrumented the PCID code with time measurement software to measure the elapsed time for each of the computationally intensive parts of the software. In particular, we measured the time spent in each of the FFT functions and also broke the times down according to the calling function for the FFT.

Measuring the number of FFTs processed and the total processing time for FFTs allowed us to create a baseline performance metric for gigaflops/Sec. The number of floating point operations (flops) required to process FFTs in the code, based on the formula for the number of flops needed to perform a 1-D complex FFT of length n: 5n*log(n).

High level measurements helped to determine where there was the most potential for improvement in the code and to identify communication problems, for example, cases where processors were idle, while processes waited for communication from other processes. It also helped to identify metrics to evaluate our success in optimizing the code, by establishing initial baseline metrics.

IV. Low Level Measurements

After identifying the functions that accounted for a large percentage of processing time, we proceeded to explore possible approaches to optimization at the lower level. Two basic approaches were used at this level. We implemented tests to verify that processor cores could share level-2 cache on quad-core processors. Some of the new processors keep track of cache hit rates and allow software to query the values. Also, performance measurements can reflect a relative lack of cache if performance degrades.

We also tried alternative libraries for FFT implementation, including Intel's Integrated Performance Primitives (IPP) [7] and our own custom FFT implementation. The PCID code initially used the FFTW library [8]. Understanding the advantages and disadvantages of each FFT implementation would be difficult analytically. Developing metrics that suggest which code is best suited for each situation leads to improved overall optimization.

A combination of fine-tuning a particular FFT code and having alternative implementations to select for specific situations seems to be a successful approach to optimize this important class of HPC image processing codes.

IV. Conclusion

We are currently performing ongoing system measurements and optimizations of PCID. An advantage of using metrics for continual optimization of an HPC code is that, as new challenging uses of the system emerge, they can be addressed in the context of the previous optimizations. We do not have to wait for a complete analysis of the system, because we expect it to be changing with increased use.

Another advantage of using metrics for performance is that bugs in the system can be uncovered when unexpected results are produced. This is particularly useful when the code is

under active development, as many intelligent systems are, since the system is always adapting to new knowledge.

Our approach works well, since it determines at an early stage in the optimization process, whether there is reason to expect a significant improvement in performance. It also helps find appropriate points to attack inefficiency for improved performance. New techniques will be developed and generalized for application to all HPC implementations, based on an improved understanding of the behavior of multi-core processors networked with Infiniband or the next generation of lower latency gigabit Ethernet.

REFERENCES

[1] Understanding Infiniband – Cisco White Paper, http://www.cisco.com/en/US/prod/collateral/ps6418/ps6419-/ps6421/prod_white_paper0900aecd8043ba1d.html.

[2] Quad-Core Intel Xeon Processor 5400 Series ftp://download.intel.com/products/processor/xeon/dc54kprodbrief.pdf

[3] Parallelization and Automation of a Blind Deconvolution Algorithm http://ieeexplore.ieee.org/iel5/4134019/4134020/04134075.pdf

[4] The Cell project at IBM Research - http://www.research.ibm.com/cell/

[5] GNU Profiler - http://sourceware.org/binutils/docs-2.16/gprof/

[6] Multi-Processing Environment (MPE) http://www-unix.mcs.anl.gov/mpi/www/www4/MPE.html

[7] IPP http://www.intel.com/cd/software/products/asmo-na/eng/302910.htm

[8] FFTW – Fastest Fourier Transform in the West – www.fftw.org

Measurement Techniques for Multiagent Systems

Robert N. Lass, Evan A. Sultanik, William C. Regli
Drexel University
Department of Computer Science
College of Engineering
3141 Chestnut Street
Philadelphia, PA 19104
{urlass, eas28, regli}@cs.drexel.edu

ABSTRACT

Multiagent Systems (MAS) are a software paradigm for building large scale intelligent distributed systems. Increasingly, these systems are being deployed on handheld computing devices, or on non-traditional networks such as mobile ad-hoc networks and satellite links. These systems present new challenges for computer scientists in describing the performance of a system and analyzing competing systems. This paper surveys existing metrics that can be used to describe MASes and related components, and provides a framework for analyzing MASes with a case study using DCOPolis, a distributed constraint reasoning system.

1. INTRODUCTION

An agent is a situated computational process with one or more of the following properties: autonomy, pro-activity and interactivity. A multiagent system (MAS) is a system with one or more agents. MASes are a software paradigm for building large scale intelligent distributed systems. Increasingly, these systems are being deployed on handheld computing devices, or on non-traditional networks such as mobile ad-hoc networks and satellite links. These systems present new challenges for computer scientists in describing the performance of a system and analyzing competing systems.

Much of the research in this area is entirely theoretical, in the sense that no examples of large-scale systems of this type exist. As a result, most work utilizes simulators or metrics that have not been validated against real-world results. Furthermore, there is a lack of standard terminology or even an agreed upon set of functions that a MAS must provide. Hopefully the recently published Agent Systems Reference Model [7] will provide this in the same way that the OSI reference model has for the field of computer networking.

This is not to say that the simulators or metrics have no value. So many variables exist that comparing a fielded system of this type against another fielded system is not a straightforward task. In some cases the researchers may not even have access to hardware or enough experience to successfully run experiments with real systems [43].

Problems with current methods of evaluating decentralized systems are discussed at length in [23]. Specifically, the authors claim that current practices have a tendency to be inappropriately generalized, to use technically inappropriate but "standard" evaluation techniques, and to focus too heavily on feasible systems. Generalization is caused by only evaluating the performance of the system in a small portion of the environmental and workload space. Standard evaluation techniques bias research towards systems that perform well with regard to those techniques. The difficulty of establishing new methods may cause systems to be evaluated at points that are not commensurate with their intended use. As a result of these three points, research may become ossified: increasing the difficulty of making new discoveries. Lastly, the authors discuss robustness: focusing on a few evaluation points may not uncover behavior that may occur in a more dynamic environment.

The main contributions of this paper are a procedure for developing and testing frameworks and procedures for MASes to avoid these problems, and to present results from an example application of this procedure to a framework for distributed constraint optimization. As advocated in [23], the framework separates the implementation of the algorithms being studied from the platform (simulator, real 802.11 network, *etc.*) to allow code to be written once and then tested in the simulator or run as part of a real system. The latter also allows the simulation data, as well as algorithm metrics to be verified.

2. MULTIAGENT SYSTEM MODEL

Models describe relationships between components of a system to facilitate reasoning about the system. Many abstract models of MASes have been written. In this paper, we use the model derived in the Agent Systems Reference Model (ASRM), and classify metrics based on the layer of this model to which they are applied. We believe that this model is more relevant than others for applied researchers because the various components of the model were informed by reverse engineering of existing multiagent systems, rather than theories about how multiagent systems ought to operate. In addition, the ASRM was not written by a single research group, but a collection of people from industry, government and academia.

2.1 Abstract Model of a MAS

An abstract representation of a MAS is shown in Fig. 1,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS '08 August 19–21, 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

and is taken from the Agent Systems Reference Model (ASRM) [7]. At the top of the diagram are *agents*, represented as triangles. Conceptually, an agent is a process with a sensor interface that determines the state of the world. It gives information about the world to a controller, performs some computation, and may result in the effector taking some action to modify the world. A thermostat could be taken as simple example: the sensor consists of a thermometer, the controller decides whether or not to turn the air-conditioner or heater on, and the effector is the air conditioner or heater interface.

The agent is supported by an *agent framework*. An agent framework is the software components that support agent execution. In some agent systems, the agent framework may be trivial, if the agents run natively on the platform (as opposed to in a virtual machine or some other local execution environment). Most agent systems, however, are based on a framework that supports key functionality agents commonly use, such as services for migration, agent messaging, and matchmaking. Examples of such systems are JADE [27], Cougaar [14], and A-Globe [22].

Under the framework is the *platform*. The platform consists of all non-agent software present, such as the operating system, databases, networking software or window managers. As depicted in Figure 1, each platform may have multiple frameworks on top.

The platform executes on a computing device, or a *host*. This is the physical computing platform on which the software is executing. A host may have multiple platforms executing on it. These hosts are distributed in the physical world, which is the bottom layer in the figure.

To summarize, measurement can take place at four layers in the model: agent, framework, platform, and environment/host. In addition, *system* measurements that cover the whole system (*i.e.* all of the components functioning together) can be taken. Within each of these layers, there are different levels and classifications of components to be measured, such as:

- **Framework:** The OSI [52] layer 7 application protocol could be analyzed, the memory footprint, cpu usage, and other framework related metrics.

- **Platform:** Except for in the trivial case where the agents run directly on the platform, the OSI [52] layers 2–6 occur within the platform. Measurement could occur at any of these levels. This means the performance of 802.11, Internet Protocol (IP), Transmission Control Protocol (TCP), Session Initiation Protocol (SIP), and Secure Sockets Layer (SSL) may all be measured, each of which is at a different OSI layer.

- **Environment:** This layer is primarily composed of the OSI layer one.

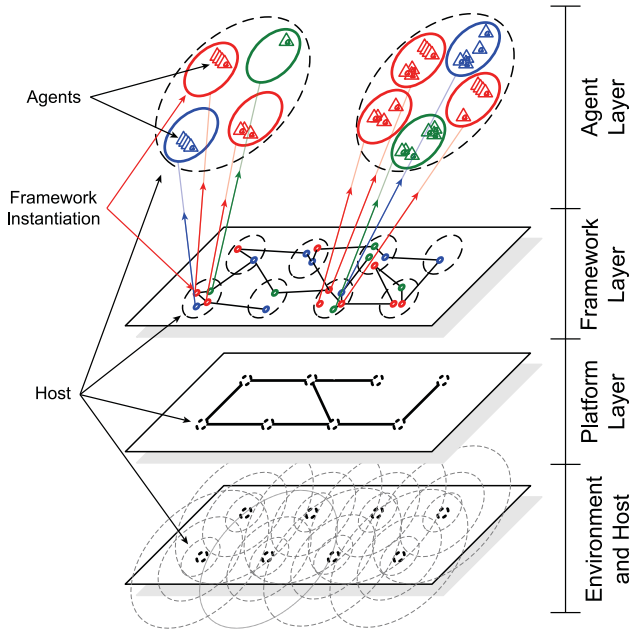


Figure 1: An abstract representation of a multiagent system. This diagram shows all of the different agents, frameworks, platforms (along with their non-agent software), and hosts needed to deliver the functionality of the system. Different colored circles represent different types of frameworks. The framework layer links show the logical connections that exist between hosts, the platform layer shows the communications connections that exist between hosts, and the environment layer shows the state of the physical medium (in this case, radio signals) used for communications between hosts.

3. METRICS SURVEY

3.1 Meta-Metrics

There are a number of *types* of metrics that can be applied at each layer of the model. They are generally classified based on their purpose, or based on the domain of the values they may take on.

3.1.1 Effectiveness vs Performance

Measures of Effectiveness (MoE) quantify the system's ability to complete its task in a given environment. In some cases this could be a binary value (the system succeeds or it fails) while in other cases it could be a range of values (the system saved $x\%$ of the hostages).

Measures of Performance (MoP) are quantitative measures of some system characteristic, such as bandwidth required, power consumed, communications range or time to perform some task. They do not describe the quality of the solution, but the quality of obtaining the solution.

Often, these two types of measures will be combined to say something about the performance required to achieve a level of effectiveness with a system. For example, one might produce a graph showing the trade off between time and solution quality for a certain system.

3.1.2 Data Classification

The most widely adopted method of data classification divides data into one of four different categories [47]. Nominal measurements are labels that are assigned to data. Ordinal measurements are rankings; greater than and less than can be applied to the measurements, but meaningful arithmetic transformations are not possible. Interval measurements are also numbers, but the difference between them has meaning. Ratio measurements are the same as interval measurements, except that there is a known zero point.

3.2 Agents and Frameworks

An agent is a situated computational process, and for our purposes agents are the components that achieve the desired functionality of the system at the highest level. The framework is a part of the system that provides functionality to the agents. Depending on the system, due to the way in which agents and frameworks are differentiated, metrics that are relevant for the agents in one system may be relevant to the framework in another system, and *vice versa*. Therefore, such metrics are categorized together herein.

A method for comparing ontology matching algorithms that is compatible with the accepted criteria of recall and precision is proposed in [17]. The author states that this is more accurate as it takes into account semantics, not just syntax.

Quantifiable measures for autonomy are described in [5]. The autonomy metric is a number between 0 and 1, and is always defined with respect to a goal and an agent. The number is calculated by determining the percentage of decisions that are made by the agent to reach the goal. For example, consider an agent a_1 working to achieve goal g_1. If one out of every four decisions used to reach g_1 were made by a_1, the autonomy metric with respect to a_1 and g_1 is 0.25. If the agents vote on decisions, then the metric is the weight that this agents vote has on the final decision. For example, if five agents each cast votes of equal weight for each decision, each agent's autonomy metric is 0.2.

For conflict resolution, metrics such as those used by the distributed constraint reasoning community may be used. Cycle-based runtime (CBR) [16], NCCC [36] and ENCCC [42] are three popular metrics based on logical clocks that may be extended to measure virtually any asynchronous decision process.

3.3 Platform

3.3.1 Distributed Systems

Many MAS are distributed systems, and it follows that techniques for analyzing distributed systems can be applied to MAS analysis. Hollingsworth summarized metrics [25] and techniques for evaluating the performance of distributed systems [24]. Lynch gives an overview of many widely used distributed algorithms, along with their analysis in [34]. The reader is referred to these three publications.

3.3.2 Networking

Networking is major component of most realistic MASes, and very diverse and active research in networking metrics is ongoing. First is a brief overview of two types of networking metrics, connectivity and capacity, followed by a discussion of MANETs.

Connectivity.

Connectivity refers to the state of the communications links between computers in a network. Often, this is represented as a graph with the nodes representing the computers and the edges representing communications links. If a computer can communicate with another computer, there is a communications link between them.

The volatility of these links depends on the type of network. On traditional wired networks, the communications links between computers are relatively static. On a MANET, the links between nodes change as the nodes move spatially.

The performance of the network at OSI layer three, specifically the routing protocol, is also critical. In [28], three different ad-hoc routing protocols are each tested under three different scenarios. In these experiments, the scenarios differed in terms of the network load.

Capacity.

There are several ways of measuring cross-layer network performance. First, by using a program such as `netperf` [29], throughput can be measured for a given network topology and configuration.

MANET.

Various metrics specifically for evaluating MANETs are described by [15], mainly at the four lowest layers of the OSI model. End-to-end throughput and delay, route acquisition time, percentage out-of-order delivery and efficiency. The latter is a general term describing the overhead involved in sending data. Three example efficiency ratios are given: bits transmitted to bits delivered, control bits transmitted to data bits delivered and control and data packets transmitted to data packets delivered. It also describes different contexts under which a MANET may operate. These contexts include network size, connectivity, topological rate of change, link capacity, fraction of unidirectional links, traffic patterns, mobility and the fraction and frequency of sleeping nodes. When evaluating the performance of a MANET system, it is important to note the context under which the researcher performs the evaluation.

3.4 Environment / Host

These metrics describe some aspect of the environment in which the system was tested are environmental measures. In the case of a robot, this might be the physical world. In the case of a software agent, this is the services, users, and other agents with which it interacts.

In [1], the quantification of the complexity of a test environment is proposed. The intent is to allow one to measure the task performance of an agent with respect to the tests' complexities. A more specific example of the complexity of a test environment is given in [38], which describes three metrics for describing the "traverseability" of terrain by a robot. Two are for roughness and one is for "crossability."

Since humans can be part of the environment in which agents operate, it may be useful to describe the level of interaction agent(s) have with them. A classification system for autonomous systems is proposed in [26], based on the complexity of the mission, independence from humans and difficulty of the operational environment. Terms and other metrics for autonomous systems are also defined in this work.

3.5 System

System metrics are overall metrics that measure something about the system as a whole. When comparing different systems for a task, often the evaluator wants a brief summary they can present to others on the overall performance or effectiveness of the system, making these metrics some of the most relevant.

An approach to evaluating performance in surveillance systems is presented in [20], along with domain specific metrics are also proposed.

There is much research in the literature on evaluating the effectiveness of robots, which have many similarities with MAS. A brief overview of metrics related to human-robot interaction is provided by [46]. The robots' performance usually cannot be measured in terms of optimality, as they have to deal with a messy environment, making it difficult to objectively assess them. Often this means that the robots are given a number of tasks that are taken to be representative and evaluated based on how well they are able to complete these tasks. For example, [4] deals with autonomous robots in a disaster scenario. The authors propose metrics that award points based on how well the robots are able to map their environment and find disaster victims. In a hybrid human-robot system, one method of analysis is to measure the effect of the robot on the human user's effectiveness at a task [9]. Other works on evaluating human-robot interaction include [21], which presents a framework for evaluating performance and testing procedures for human-robot teams, and [39] which evaluates a number of metrics for how well robots help humans complete tasks.

Another human-machine hybrid system is the integrated automobile crash warning system, presented in [18]. Three metrics describing effectiveness were given based on the warnings the system gave: percent true, percent false and percent missed. A measure of performance (see next section) was also used describing how far before an area of danger a driver will be able to stop.

A discussion of endurance testing for robots and a recommendation for all safety, security and rescue (SSR) robots to undergo endurance testing is given in [31]. Using WEKA[50], statistical analysis was performed on the failure data collected to determine the causes of fault.

The results of a long term experimental use of robots in a joint effort between Swedish academic and military organizations was described in [33]. It included a qualitative analysis of the users attitude towards the robots before, during and after the study.

When the optimal or actual solution is known, one way to evaluate effectiveness is to compare the optimal or actual solution to the solution produced by the MAS. In [19], the authors present a new approach to analysis of road recognition algorithms. In this approach, the feature extraction results are compared to actual features from a National Institute of Standards and Technology (NIST) database. The feature search trees were also used to describe the computational complexity of the search.

When it's not clear what the optimal solution is, or when there are many ways to describe the effectiveness of the system's solution, several metrics may be needed. In [45], the experimenters set up a slalom course, and wanted to measure the performance of a hybrid human-robot system at navigating the course. There is no single metric that describes how well the system performed, so a number of performance measures were recorded such as time to navigate the course, gates passed through, symbols seen, and the results of a human user's survey. The latter covered efficiency, effectiveness and user satisfaction.

Some of this work is more general. For example, in [30], the authors proposes a general effectiveness metric $P = A - B$, where A is the success metric, B is the failure metric and P is the combined performance metric. This type of metric works with a wide variety of systems that have some notion of success and failure. Similarly, in [11], the authors proposes an information theoretic metric for evaluating the quality of the amount of information processed by an intelligent system.

If a single performance metric is desired, a number of metrics can be combined. In [49], the authors propose a number of performance metrics for the Mars Rover and a formula for generating a composite performance score. The scores are combined using a technique inspired by information theory, described in [41].

A definition of performance, scalability and stability in terms of multiagent systems, and an example of analyzing a MAS for these factors is presented in [32]. In general, performance is computational cost and throughput (computational complexity and message complexity), scalability is the rate at which the overhead increases as the agent population increases, and stability is whether or not there is an equilibrium point that the system will return to after perturbations. An example of analyzing these factors is given for a MAS that solves a standard contract net [44] problem.

In many cases, a number of different metrics are needed to get a sense of the performance of the system. An example of this is [13], which investigates benchmarks for UGVs in terms of reconfiguration, communications and adaptation and learning. Another is [8], in which four metrics (2 MOE, 2 MOP) are used to evaluate an algorithm for transforming disparate data sets to a common coordinate system: convergence (MOP), speed (MOP), translation error (MOE) and rotation error (MOE).

4. ANALYSIS FRAMEWORK FOR MULTI-AGENT SYSTEMS

This section presents a framework for applying these metrics to a decision making task. There are three main components: selection, collection, and application. First the evaluator decides which metrics to use, which must be grounded in some overall goal for the system. Next, the metrics are

collected by performing experiments. Finally, the metrics are applied to the original goals to determine if the system meets the goal or perhaps if one system performs better than another.

4.1 Selection

There are an infinite number of metrics that could be applied to a system, and an infinite number of ways to apply them. How does a researcher go about deciding which metrics need to be measured for his or her system?

The Goal, Question, Metric (GQM) [6] approach for evaluation was developed during a series of projects at the NASA Software Engineering Lab. This technique is intended to provide a focus to investigation and evaluation of systems.

In this approach the evaluator first chooses goals for different products, processes, and / or resources. There are four parts to a goal: the purpose, the issue, the object and the viewpoint. The example given in [6] is "Improve (purpose) the timeliness of (issue) change request processing (object) from the managers viewpoint (viewpoint)."

Next, the evaluator selects questions, usually with quantifiable answers, that must to be answered to understand if the system meets the goal. Each goal may need multiple questions.

Finally, the metric is a set of data associated with the questions that can be subjective (depends on the point of view, such as ease of use of a UI) or objective (independent of the point of view, such as program size). This data is used to answer the questions, which in turn informs the evaluator about the goals.

As an example scenario to illustrate how the GQM approach works, consider evaluating two systems for solving a Distributed Constraint Reasoning (DCR) problem [51] on a MANET. First, we must decide on a goal, such as "Select (purpose) the system (object) providing the lowest average runtime in a bandwidth constrained environment (issue) from the point of view of the last agent to converge on a solution (viewpoint)." There is still some fuzziness to the statement, but the scope is narrower. For example, this goal is not concerned with the networking cost of a system, the amount of information an algorithm leaks to other agents, or memory utilization. There are usually multiple goals in a real evaluation, but for the rest of this example we will only look at this single goal.

The next step is to select questions that allow to characterize the object with respect to the goal, such as "How long does the system take to converge with test data A?" "How long does the system take to converge with test data B?"

Alternatively, there may be metrics or tests that are commonly used in the domain in which the system operates. For example, [31], describes a specific type of test that the authors recommend performing on a certain class of robots.

4.2 Collection

From the questions chosen in the previous section we need to select a set of metrics to collect that will allow us to answer them. In the example questions we selected we were only concerned with time to completion. So, we need to collect runtime information for the system. This could be a sophisticated solution, such as instrumenting the code to record timing information, or it could be something more informal such as having the user time it with a stopwatch.

Papers describing practices for conducting research, methods for analyzing data are classified here as "empirical methods." Some of them are general, such as [3] which provides some "rules of thumb" to keep in mind when comparing algorithms, and an example of the application of each of those rules. One widely cited resource for empirical methods for MASes is [12].

4.3 Application

In our example scenario, we asked the questions "How long does the system take to converge with test data A?" and "How long does the system take to converge with the test data B?" to help us meet the goal "Select the system providing the lowest average runtime from the point of view of the last agent to converge on a solution." Al, all that is left is to compare the runtime of each system to determine which is the lowest. If the runtime using both sets of test data is lower for one system, clearly that is the system to select. If one system has a lower runtime for test data A and another has a lower runtime for test data B, then we have to either decide which data set is most similar to the use the system will see once deployed, or we need to create new goals and re-assess the system.

5. CASE STUDY: DCOPOLIS

A large class of multiagent coordination and distributed resource allocation problems can be modeled as distributed constraint reasoning (DCR) problems. DCR has generated a lot of interest in the constraint programming community and a number of algorithms have been developed to solve DCR problems [37, 35, 40, 10]. A formal treatment of DCOP is outside of the scope of this paper, and the reader is referred to [51] for an introduction to the topic.

Informally, DCR is a method for agents to collaboratively solve constraint reasoning problems distributedly with only local information. The four main components of a DCR problem are variables, domains, agents and constraints. Each *agent* has a set of *variables*, to which it must assign *values*. Each variable has an associated *domain*, which is the set of all possible value assignments to the variable. *Constraints* are a set of functions that specify the cost of any set of partial variable assignments. Finally, each agent is assigned one or more variables for which it is responsible for value assignment. DCOP algorithms work by exchanging messages between agents, who give each other just enough information to allow each agent to make a globally optimal variable assignment.

DCOPolis [48] is a framework for comparing *and deploying* DCR software in heterogeneous environments. DCOPolis has three key points:

1. The communications platform, DCR algorithm, and problems instances are all modular and may be swapped for a truly comprehensive analysis of algorithmic performance;

2. DCOPolis contributes to comparative analysis of DCR algorithms by allowing different state-of-the-art algorithms to run in the same simulator under the same conditions or to be deployed on "real" hardware in "real" scenarios; and

3. DCOPolis introduces a new form of distributed algorithm simulation that shows promise of accurate prediction of real-world runtime.

DCOPolis has three primary abstract components: *problems*, *algorithms*, and *platforms*. The main function of DCOPolis is to provide an interface through which the three components can interact. By writing a new instance of any of these components that properly adheres to DCOPolis' API, any algorithm should be able to solve any instance of any problem while running on any platform—even without prior knowledge of such. This makes implementation and testing of new algorithms and platforms trivial.

In keeping with the example given in Section 4 ("Select (purpose) the system (object) providing the lowest average runtime in a bandwidth constrained environment (issue) from the point of view of the last agent to converge on a solution (viewpoint)."), let us apply the framework to DCOPolis running two different algorithms.

- **Agent:** DCOPolis agents are instantiated with a local view of the problem and then assign values to their variables, send messages to other agents, and change the assigned values based on the messages they received from other agents. Here, we need to record the time each agent takes to converge upon a solution.

- **Framework:** In this case, the framework is what the ASRM refers to as a "NULL framework." The functionality is contained within the agents, which interact directly with the platform through the Java Virtual Machine. There is nothing to be measured here.

- **Platform:** Any of the metrics in Section 3.3.2 can measure the performance of the network at the platform layer. Then an estimation of the system's performance were the bandwidth to drop below our test environment's could be made. Also at this level, the metric in [2] could also be used to determine which bottlenecks could be optimized, if we were interested in improving as well as comparing the systems.

- **Environment:** Our goal stated that we must be concerned with bandwidth constrained environment. The main thing to measure here is available bandwidth.

6. CONCLUSION

MAS are complicated systems made up a number of interconnected components. Measuring these systems presents new challenges, especially when these systems are deployed in dynamic environments such as mobile ad-hoc networks. This paper surveyed a number of metrics that can be used to measure these types of systems, as well as a general framework for analyzing MASes and its application to an example MAS, DCOPolis.

7. REFERENCES

[1] Michael L. Anderson. A flexible approach to quantifying various dimensions of environmental complexity. In *2004 Performance Metrics for Intelligent Systems Workshop*, 2004.

[2] Thomas E. Anderson and Edward D. Lazowska. Quartz: a tool for tuning parallel program performance. In *Proceedings of the 1990 ACM SIGMETRICS conference on measurement and modeling of computer systems*, pages 115 – 125, New York, NY, USA, 1990. ACM Press.

[3] S. Balakirsky and T. R. Kramer. Comparing algorithms: Rules of thumb and an example. In *2004 Performance Metrics for Intelligent Systems Workshop*, 2004.

[4] S. Balakirsky, C. Scrapper, S. Carpin, and M. Lewis. Usarsim: Providing a framework for multi-robot performance evaluation. In *2006 Performance Metrics for Intelligent Systems Workshop*, 2006.

[5] KS Barber and CE Martin. Agent autonomy: Specification, measurement, and dynamic adjustment. *Proceedings of the Autonomy Control Software Workshop at Autonomous Agents (Agents2019 99)*, pages 8–15, 1999.

[6] V. Basili, G. Caldiera, , and H.D. Rombach. *The Goal Question Metric Approach*, pages 528–532. John Wiley and Sons, Inc., 1994.

[7] Brandon Bloom, Christopher J. Dugan, Tedd Gimber, Bernard Goren, Andrew Hight, Moshe Kam, Joseph B. Kopena, Robert N. Lass, Israel Mayk, Spiros Mancoridis, Pragnesh Jay Modi, William M. Mongan, William C. Regli, Randy Reitmeyer, Jeff K. Salvage, Evan A. Sultanik, and Todd Urness. *Agent Systems Reference Model*. Drexel University, Philadelphia, PA, 2006. http://gicl.cs.drexel.edu/people/regli/reference_model-v1a.pdf.

[8] Bruce Brendle. 3d data registrnation based on human perception. In *2006 Performance Metrics for Intelligent Systems Workshop*, 2006.

[9] Jennifer L. Burke, Robin R. Murphy, Dawn R. Riddle, and Thomas Fincannon. Task performance metrcis in human-robot interaction: Taking a systems approach. In *2004 Performance Metrics for Intelligent Systems Workshop*, 2004.

[10] Anton Chechetka and Katia Sycara. No-commitment branch and bound search for distributed constraint optimization. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 1427–1429, New York, NY, USA, 2006. ACM Press.

[11] Rama Chellappa and Amit K. Roy Chowdhury. An information theoretic evaluation criterion for 3d reconstruction algorithms. In *2004 Performance Metrics for Intelligent Systems Workshop*, 2004.

[12] Paul R. Cohen. *Empirical methods for artificial intelligence*. MIT Press Cambridge, MA, USA, 1995.

[13] Sesh Commuri, Yushan Li, Dean Hougen, and Rafael Fierro. Evaluating intelligence in unmanned ground vehicle teams. In *2004 Performance Metrics for Intelligent Systems Workshop*, 2004.

[14] BBN Corporation. Cognitive Agent Architecture (Cougaar). http://www.cougaar.org/.

[15] S. Corson and J. Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. RFC 2501, January 1999.

[16] John Davin and Pragnesh Jay Modi. Impact of problem centralization in distributed constraint optimization algorithms. In *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems*, pages 1057–1063, New York, NY, USA, 2005. ACM Press.

[17] Jérôme Euzenat. emantic precision and recall for

ontology alignment evaluation. In *IJCAI '07: Proceedings of the 21st International Joint Conference on Artificial Intelligence*, 2007.

[18] Jack J. Ference, Sandor Szabo, and Wassim G. Najm. Performance evaluation of integrated vehicle-based safety systems. In *Performance Metrics for Intelligent Systems Workshop*, pages 85 – 89. NIST Special Publication, 2006.

[19] M. Foedisch, C. Schlenoff, and R. Madhavan. Performance analysis of symbolic road recognition for on-road driving. In *2006 Performance Metrics for Intelligent Systems Workshop*, 2006.

[20] Michael Freed, Robert Harris, and Michael Shafto. Measuring autonomous uav surveillence. In *2004 Performance Metrics for Intelligent Systems Workshop*, 2004.

[21] A. Freedy, J. McDonough, R. Jacobs, E. Freedy, S. Thayer, and G. Weltman. A mixed initiative human-robots team performance assessment system for use in operational and training environments. In *2004 Performance Metrics for Intelligent Systems Workshop*, 2004.

[22] Agent Technology Group. A-globe. http://agents.felk.cvut.cz/aglobe/.

[23] Andreas Haeberlen, Alan Mislove, Ansley Post, and Peter Druschel. Fallacies in evaluating decentralized systems.

[24] Jeffrey K. Hollingsworth, James Lumpp, and Barton P. Miller. Techniques for performance measurement of parellel programs. In *Parallel Computers: Theory and Practice*. IEEE Press, 1995.

[25] Jeffrey K. Hollingsworth and Barton P. Miller. Parallel program performance metrics: A comparison and validation. In *Proceedings of the 1992 ACM/IEEE conference on Supercomputing*, pages 4 – 13, Los Alamitos, CA, USA, 1992. IEEE Computer Society Press.

[26] Hui-Min Huang. The autonomy levels for unmanned systems alfus framework. In *Performance Metrics for Intelligent Systems Workshop*, pages 47 – 51. NIST Special Publication, 2006.

[27] Telocom Italia. Java Agent DEvelopment Framework (JADE). http://jade.tilab.com/.

[28] Per Johansson, Tony Larsson, Nicklas Hedman, Bartosz Mielczarek, and Mikael Degermark. Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. In *MobiCom '99: Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pages 195–206, New York, NY, USA, 1999. ACM Press.

[29] Rick Jones. Netperf. http://www.netperf.org/netperf/NetperfPage.html.

[30] Balajee Kannan and Lynne E. Parker. Fault tolerance based metrics for evaluating system performance in multi-robot teams. In *2006 Performance Metrics for Intelligent Systems Workshop*, 2006.

[31] Jeffrey A. Kramer and Robin R. Murphy. Endurance testing for safety, security and rescue robots. In *Performance Metrics for Intelligent Systems Workshop*, pages 247 – 254. NIST Special Publication, 2006.

[32] Lyndon Lee, Hyacinth Nwana, Divine Ndumu, and Phillipe De Wilde. The Stability, Scalability and Performance of Multi-agent Systems. *BT Technology Journal*, 16(3):94–103, 1998.

[33] C. Lundberg, H.I. Christensen, and R. Reinhold. Intellectual performance using dynamical expert knowledge in seismic environments. In *2006 Performance Metrics for Intelligent Systems Workshop*, 2006.

[34] Nancy A. Lynch. *Distributed Algorithms*. Morgan Kaufmann, 1996.

[35] Roger Mailler and Victor Lesser. Solving distributed constraint optimization problems using cooperative mediation. In *AAMAS '04: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 438–445, Washington, DC, USA, 2004. IEEE Computer Society.

[36] A. Meisels, E. Kaplansky, I. Razgon, and R. Zivan. Comparing performance of distributed constraints processing algorithms, 2002.

[37] Pragnesh Jay Modi, Wei-Min Shen, Milind Tambe, and Makoto Yokoo. An asynchronous complete method for distributed constraint optimization. In *AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, pages 161–168, New York, NY, USA, 2003. ACM Press.

[38] V. Molino, R. Madhavan, E. Messina, T. Downs, A. Jacoff, and S. Balakirsky. Treversability metrics for urban search and rescue robots on rough terrain. In *2006 Performance Metrics for Intelligent Systems Workshop*, 2006.

[39] Dan R. Olsen. and Michael A. Goodrich. Metrics for evaluating human-robot interactions. *Proceedings of PERMIS*, 2003, 2003.

[40] Adrian Petcu and Boi Faltings. A distributed, complete method for multi-agent constraint optimization. In *CP 2004 - Fifth International Workshop on Distributed Constraint Reasoning (DCR2004)*, Toronto, Canada, September 2004.

[41] Guillermo Rodriguez and Charles R. Weisbin. A New Method to Evaluate Human-Robot System Performance. *Autonomous Robots*, 14(2):165–178, 2003.

[42] Marius Silaghi, Robert N. Lass, Evan Sultanik, William C. Regli, Toshihiro Matsui, and Makoto Yokoo. Constant Cost of the Computation-Unit in Efficiency Graphs. In *The Tenth Annual Workshop on Distributed Constraint Reasoning*, May 2008.

[43] Emin Gun Sirer. Sextant deployment (accessed 10/31/2007). http://www.cs.cornell.edu/People/egs/sextant/deployment.php.

[44] Reid G. Smith. The contract net protocol: High level communication and control in a distributed problem solver. In *IEEE Transactions on Computers*, volume C-29 12, pages 1004 – 1113, december 1980.

[45] Brian Stanton, Brian Antonishek, and Jean Scholtz. Development of an evaluation method for acceptable usability. In *Performance Metrics for Intelligent Systems Workshop*, pages 263 – 267. NIST Special Publication, 2006.

[46] Aaron Steinfeld, Terrance Fong, David Kaber, Michael

Lewis, Jean Scholtz, Alan Schultz, and Michael Goodrich. Common metrics for human-robot interaction. *ACM SIGCHI/SIGART Human-Robot Interaction*, pages 33–40, 2006.

[47] S. S. Stevens. On the theory of scales of measurement. *Science*, 1946.

[48] Evan A. Sultanik, Robert N. Lass, and William C. Regli. DCOPolis: A framework for simulating and deploying distributed constraint optimization algorithms. In *The Ninth Annual Workshop on Distributed Constraint Reasoning*, September 2007.

[49] Edward Tunstel. Performance metrics for operational mars rovers. In *Performance Metrics for Intelligent Systems Workshop*, pages 69 – 76. NIST Special Publication, 2006.

[50] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2005.

[51] M. Yokoo, E.H. Durfee, T. Ishida, and K. Kuwabara. The distributed constraint satisfaction problem: formalization and algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 10(5):673–685, 1998.

[52] Herbert Zimmerman. OSI reference model—the ISO model of architecture for open system interconnection. *IEEE Transactions on Communications*, 28(4):425–432, April 1980.

Performance of Super-Resolution Enhancement for LADAR Camera Data

Shuowen Hu
Army Research Laboratory
2800 Powder Mill Rd
Adelphi, MD 20783
1-301-394-1678
shuowen.hu@us.army.mil

S. Susan Young
Army Research Laboratory
2800 Powder Mill Rd
Adelphi, MD 20783
1-301-394-0230
ssyoung@arl.army.mil

Tsai Hong
NIST
100 Bureau Dr
Gaithersburg, MD 20899
1-301-975-3444
hongt@cme.nist.gov

ABSTRACT

Laser detection and ranging (LADAR) camera systems are increasingly used in robotics applications for autonomous navigation and obstacle avoidance. Their compact size, high frame rate, wide field-of-view, and low cost are key advantages over traditional scanning LADAR devices. However, these benefits are achieved at the cost of spatial resolution. Therefore super-resolution image reconstruction technology can be applied to improve the resolution of LADAR camera data. Previous work by Rosenbush et al. applied the super-resolution algorithm of Vandewalle et al. to LADAR camera data, and observed quantitative improvement in image quality in terms of number of edges detected. This study uses the super-resolution algorithm of Young et al. to enhance the resolution of range data acquired from a commercial available LADAR camera. This work applies a preprocessing stage that increases the accuracy of sub-pixel shift estimation for improved registration of multiple LADAR camera frames and uses the triangle orientation discrimination methodology for a subjective evaluation. The objective is to measure the improvement in probabilities of target discrimination at various ranges achieved by super-resolution enhancement of LADAR camera data.

Categories and Subject Descriptors

D.3.3 [**Image Processing and Computer Vision**]: Enhancement – *Filtering, Registration.*

General Terms

Algorithms, Performance, Human Factors

Keywords

Super-resolution, LADAR camera, image registration, triangle orientation discrimination methodology

This paper is authored by employees of the United States Government and is in the public domain.

PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

1. INTRODUCTION

Laser detection and ranging (LADAR) is a crucial component for navigation in autonomous or semi-autonomous robots. Current small robots generally employ a 2D scanning LADAR that scans along a single line and therefore cannot detect objects above or below the detection line [1, 2]. In indoor urban environments where the setting is highly cluttered with overhanging objects such as tabletops, the 2D scanning LADAR systems may not be sufficient for navigation and obstacle avoidance [1]. A new generation of small and compact 3D LADAR devices, named LADAR camera, offers a promising solution to small robot navigation in urban environments where modern warfare is often conducted.

LADAR camera devices are compact and lightweight sensors that acquire a 3D range image of the surrounding environment. The SR-3000[1] LADAR camera device

Figure 1. SR3000 Error! Bookmark not defined.
LADAR Camera

[1] Certain commercial equipment, instruments, or materials are identified in this paper in order to adequately specify the experimental procedure. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the materials or equipment identified are necessarily best for the purpose.

(Figure 1) used in this study weighs 162 g and measures (5.0 x 6.7 x 4.23) cm [3]. LADAR camera devices emit diffuse modulated near-infrared light and measure the subsequent phase shift between the original emitted light and the reflected light. The phase measurements are combined to calculate the range data based on the time-of-flight principle [3]. The detector utilized by LADAR camera devices is a focal plane array (FPA), which is typically limited to a maximum size of (256 x 256) detectors. Consequently, these devices cannot achieve the resolution of scanning LADAR systems. This disadvantage of LADAR camera systems may be rectified by the application of super-resolution image reconstruction.

Super-resolution algorithms utilize a series of low-resolution frames containing sub-pixel shifts to generate a higher resolution image. These algorithms are typically composed of two major stages: registration stage and reconstruction stage. During the registration stage, the shift with respect to a reference frame (usually the first frame of the series) is computed to sub-pixel (i.e. decimal pixel) accuracy. The second stage utilizes this sub-pixel information to interpolate the low-resolution frames onto a higher resolution grid. A necessary condition for successful super-resolution enhancement is the presence of differing shifts between the frames in the series. The differing shifts of each frame provide additional information from which to reconstruct the super-resolved imagery. Previous work by Rosenbush et al. [4] applied a super-resolution algorithm [5] to LADAR camera data, and observed improvement in image quality in terms of number of edges detected. In this work, the super-resolution algorithm of Young et al. [6] is applied to LADAR camera imagery. This algorithm separates the registration stage into a gross shift (i.e. integer pixel shift) estimation stage and a sub-pixel shift (i.e. decimal pixel shift) estimation stage for improved registration accuracy. Both sub-stages use the correlation method in the frequency domain to estimate shifts between the frame series and the reference image. The reconstruction stage of Young et al.'s algorithm applies the error-energy reduction method with constraints in both spatial and frequency domains to generate a high-resolution image [6]. Because LADAR camera imagery is inherently smoother than visible light imagery (LADAR camera data does not capture the texture or color of the scene), this work develops a preprocessing stage for improved image registration. Specifically, a wavelet edge filtering method [7] and a Canny edge detection method [4] are investigated and compared against the accuracy achieved with no preprocessing. The wavelet edge filtering method provided more accurate shift estimation for LADAR camera data.

To assess the improvement in super-resolution enhanced LADAR camera data, the authors conducted perception experiments to obtain a human subjective measurement of quality. The triangle orientation discrimination (TOD) methodology [8,9] was used to measure the improvement achieved with super-resolution. The TOD task is a four-alternative forced-choice perception experiment where the subject is asked to identify the orientation of a triangle (apex up, down, right, or left) [9]. Results show that the probability of target discrimination as well as the response time improves with super-resolution enhancement of the LADAR camera data.

2. METHODOLOGY

2.1 Preprocessing Stage

The purpose of the preprocessing stage is to emphasize LADAR camera image edges for improved frame registration. One investigated method was the use of multi-scale edge-wavelet transforms [10] to calculate the horizontal and vertical partial derivatives of the input image at the second wavelet scale for each frame of the series. The two derivatives were then combined using sum of squares to produce a wavelet edge enhanced frame series. Another investigated method was the use of Canny edge detection algorithm to generate binary edge frame series.

To assess the benefit of preprocessing, the following procedure was followed. A synthetic frame series was generated with known sub-pixel shifts. First, an oversampled non-aliased scanning LADAR reference image (204 x 204) pixels was interpolated by an upsampling factor of eight (1632 x 1632) pixels using a Fourier windowing method [10]. Then, the simulated high-resolution image was sub-sampled at different factors to produce several low-resolution frame series, each with a different degree of aliasing. Figure 2 shows the un-aliased spectrum of a discrete space signal (e.g. scanning LADAR image) produced by oversampling a continuous space signal at a sampling frequency greater than the Nyquist frequency. If the sampling frequency is below Nyquist (simulated by sub-sampling the oversampled image), the spectrum of the sampled signal is aliased with distorted higher frequency components as depicted in red in Figure 2.

Synthetic frame series was generated by sub-sampling every m pixels in both dimensions of the simulated high-resolution image, where m = 4, 8, 12, 16, 20, 28, 36, 48, 56. Therefore the undersampling factors are $m/8$ (i.e., 0.5, 1, 1.5, 2, 2.5, 3.5, 4.5, 6, 7), simulating different degrees of aliasing. For each undersampling factor, the sub-pixel shifts for each frame of the synthetic series were generated by varying the starting pixel position of sub-sampling according to a uniform random distribution (30 frames for each series). Then preprocessing using either the wavelet or Canny method was performed. Sub-pixel shift estimates from the preprocessed and no preprocessing series were compared to the known shifts. Let $\varepsilon_i = (\varepsilon_{xi}, \varepsilon_{yi})$ denote

the registration error vector of the ith frame where ε_{xi} and ε_{yi} are the registration errors in the x and y directions. A mean absolute error (MAE) can be calculated for the frames of each synthetic series using the following equation:

$$E = \frac{1}{n}\sum_{i=1}^{n}\|\varepsilon_i\| = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\varepsilon_{xi}^2 + \varepsilon_{yi}^2}$$

where $n = 30$. The registration errors of the wavelet preprocessing method was compared to that of Canny and no preprocessing methods to assess the accuracy at each undersampling factor.

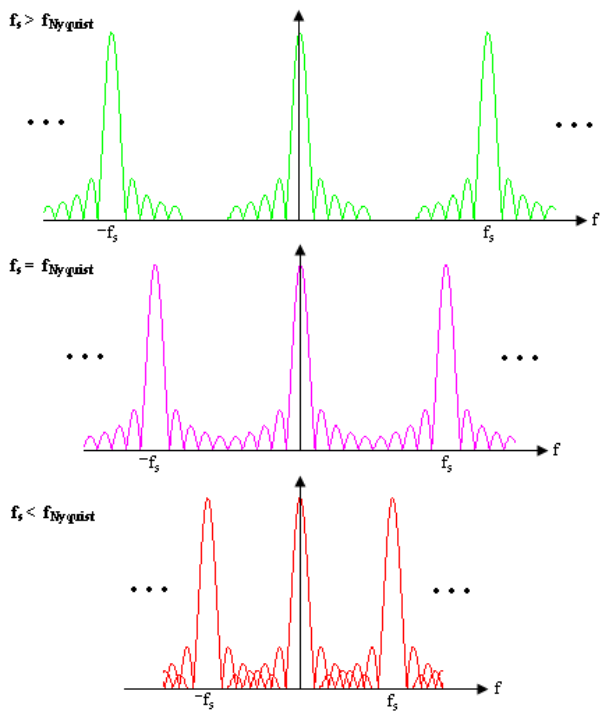


Figure 2. (Top) Un-aliased spectrum of signal sampled above Nyquist frequency, (mid) at Nyquist, and (bottom) aliased at below Nyquist frequency.

2.2 Triangle Orientation Discrimination (TOD) Methodology

The TOD methodology, developed by Netherlands TNO Physics and Electronics Laboratory (TNO-FEL), is a perception study that allows human subjects to provide a measure of image quality at various target ranges [9]. The test pattern is an equilateral triangle in one of four possible orientations (apex up, down, left, or right), and the measurement process is a four-alternative forced-choice psychological procedure that requires the observer to indicate the orientation. Variation of triangle contrast/size by changing target ranges results in a correct discrimination percentage between 25 % (pure guess) and 100 %. Probabilities of target discrimination at different ranges can then be calculated to measure the quality of both the original and super-resolved data.

The TOD method is suitable for electro-optical and optical imaging systems, and has been widely used in thermal and visual domain imagers. This methodology provides a simple task that has a close relationship to real target acquisition, and the results are free from observer bias [8,9]. The TOD methodology was adapted to LADAR camera data by using a target consisting of a square white foam board target (50 x 50) cm with an equilateral triangular hole (7.5 cm per side) cut into the board as shown in Figure 3. The triangular hole provides the necessary depth contrast against the board.

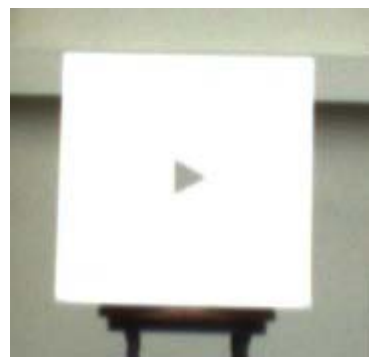


Figure 3. TOD setup

2.3 LADAR Camera

The device utilized in this study is the SwissRanger SR-3000**Error! Bookmark not defined.** LADAR camera. The camera emits diffuse 850 nm near-infrared light modulated at a default frequency of 20 MHz from a bank of 55 light emitting diodes. The non-ambiguity range achieved at this modulation frequency is 7.5 m. The SR-3000**Error! Bookmark not defined.** has a pixel array of 176 x 144 with a field-of-view of 47.5° x 39.6°, and can capture images at a maximum rate of 50 frames/s (variable with respect to the integration time setting).

2.4 Data Collection

Data collection for the experiment was conducted at a laboratory in the National Institute of Standards and Technology. The SR-3000**Error! Bookmark not defined.** LADAR camera was placed 6.5 m from a beige wall as depicted in Figure 3. The target was positioned at (3, 3.5, 4, 4.5, 5, 5.5, and 6) m from the camera. The investigated ranges were limited to between (3 and 6) m because inaccurate behavior of LADAR cameras was observed at very close and very far target distances [11]. At each range, the triangle was positioned at one of four possible orientations (apex up, down, left, right) with the center approximately 1 m high. For each orientation at each

range, four trials were conducted. Each trial consisted of a sequence of 32 frames acquired by holding the camera. The natural motion of the hand while holding the camera provided the shifts required for the super-resolution algorithm. Motion is assumed to be limited to translations in the x (horizontal) and y (vertical) planes. Though slight rotation and translation in the z-plane (depth) might have occurred from holding the camera, these parameters were not considered in the current study.

2.5 Data Processing

For each series of 32 frames, the first 25 frames are utilized for super-resolution image reconstruction. The first frame was used as the reference frame from which pixel shifts were calculated for successive frames. The use of 25 frames resulted in a resolution improvement factor of five in each direction for the super-resolved image. To ensure that the monitor modulation transfer function (MTF) was not a limiting factor in the experiment, the super-resolved images (250 x 250 pixels) were bilinearly interpolated by a factor of two to 500 x 500 pixels. The original imagery (50 x 50 pixels) was bilinearly interpolated to 500 x 500 pixels for consistency between the baseline and super-resolved imagery.

2.6 Perception Experiment

The perception experiment was a four-alternative forced-choice procedure (up, down, left, right). The imagery in this experiment was organized in the image cells and their naming convention is shown in Table 1. As shown in the row of original images in Table 1, the grayscale baseline range imagery was grouped into seven cells corresponding to the seven different target ranges. Each cell consisted of 16 original low-resolution LADAR camera images (4 orientations x 4 trials). Similarly, the grayscale super-resolved range imagery was grouped into seven cells consisting of 16 images each as shown in the row of super-resolved images in Table 1. The experiment therefore consisted of 14 cells with a total of 224 images.

Range m	A 3.0	B 3.5	C 4.0	D 4.5	E 5.0	F 5.5	G 6.0
Original Image	AA	BA	CA	DA	EA	FA	GA
Super-resolved image	AB	BB	CB	DB	EB	FB	GB

Table 1. Image cell format and naming convention

Nine subjects participated in the experiment in July 2008. The subjects were shown one image at a time with randomized presentation of cells and randomized presentation of images within each cell. The display was a 19 inch flat panel (Dell 1908FP**Error! Bookmark not defined.**) with a resolution of 1280 x 1024 pixels.

3. RESULTS AND DISCUSSION

3.1 Assessment of Registration Accuracy

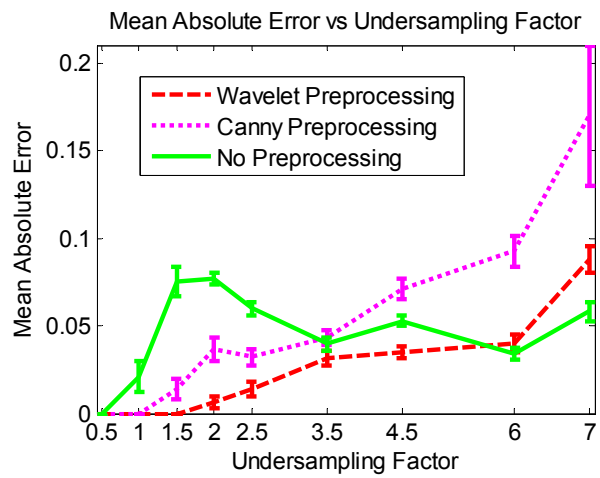


Figure 4. Mean absolute registration error with standard deviation for each undersampling factor.

Figure 4 shows the mean absolute error of registration at each undersampling factor for the generated synthetic experiments. The unit of error is fraction of a pixel. Wavelet preprocessing outperformed both the Canny method and no preprocessing method for undersampling factors of less than 6. Wavelet preprocessing was especially effective at low and moderate degrees of aliasing (undersampling factor of less than 3.5). If the imagery contained severe aliasing (undersampling factor greater than 6), then no preprocessing resulted in higher registration accuracy. The observed trend is expected. LADAR camera data is characteristically smooth due to the lack of texture information, so edge filtering with the wavelet method will improve registration. But if the data is severely undersampled that its mid to high frequency components are corrupted by aliasing, then wavelet edge filtering (which uses these severely corrupted frequency components) will result in poorer registration. The degree of aliasing in the imagery acquired with the SR-3000**Error! Bookmark not defined.** is expected to be in the moderate range as super-resolved imagery using wavelet preprocessing yields fewer artifacts than imagery produced without preprocessing.

3.2 Triangle Orientation Discrimination Perception Experiment

Figure 5 shows grayscale and color images (color-coded to distance) of the TOD target oriented up at a distance of 5 m from the camera. The orientation of the equilateral triangular hole is difficult to discern in the original images at this distance as the triangular hole appears like a blurred circle. By contrast, the orientation is clear in the super-resolution enhanced imagery. For imagery with target distances greater than 5 m, the orientation, as expected, was still more difficult to discern using the original LADAR camera imagery. But super-resolution at these greater distances proved to be effective.

(a) original **(b) super-resolved**

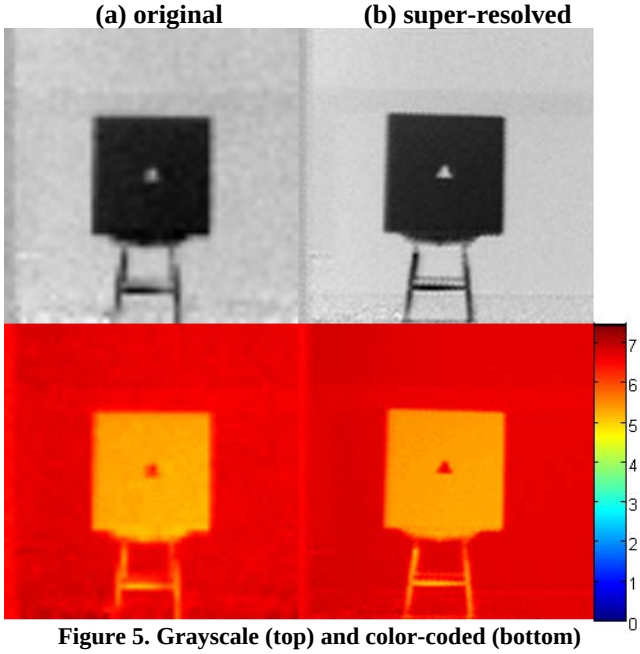


Figure 5. Grayscale (top) and color-coded (bottom) LADAR camera imagery for (a) original image and (b) super-resolved image of TOD target at 5 m.

Figure 6 shows grayscale and color images of the TOD target oriented left at a distance of 4 m from the camera. As the target distance decreases, the orientation of the triangular hole becomes more visible in the original imagery, though the triangular hole still appears distorted. In the super-resolved image, the triangular hole does not appear distorted, and is shaped more like a triangle.

Figure 7 shows the group-averaged chance-corrected probability of target discrimination at each target range. At all ranges, super-resolved imagery produced a higher probability of target discrimination with smaller inter-subject variability. At a target distance of 3 m, the original imagery had a 77 % of the probability of target discrimination, while the super-resolved imagery reached 100 %. The target discrimination performance is increased by 30 % using the super-resolution algorithm. As the target distance increased, subjects had more difficulty to discriminate the target orientation. At a target distance of 6 m, the original imagery had a 20 % of the probability of target discrimination, while the super-resolved imagery reached 90 %. That is a 350 % improvement in target discrimination performance. In summary, the probability of target discrimination is increased by 30 % to 350 % for the target ranges from 3 m to 6 m using the super-resolution algorithm.

(a) original **(b) super-resolved**

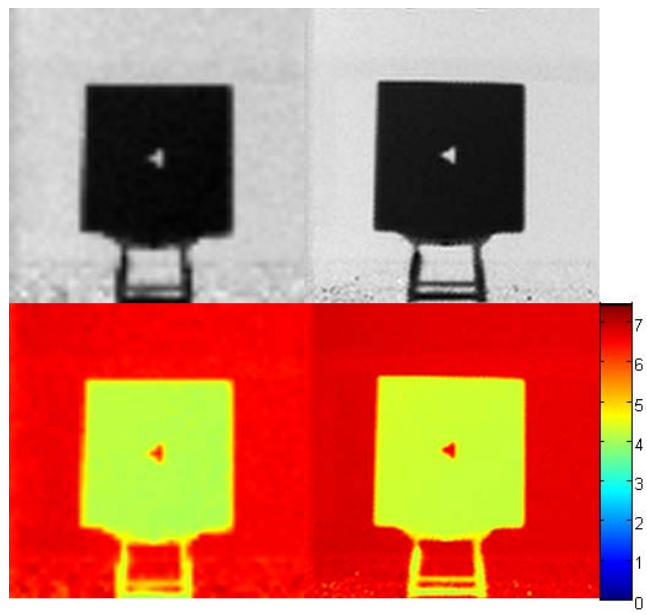


Figure 6. Grayscale (top) and color-coded (bottom) LADAR camera imagery for (a) original image and (b) super-resolved image of TOD target at 4 m.

.

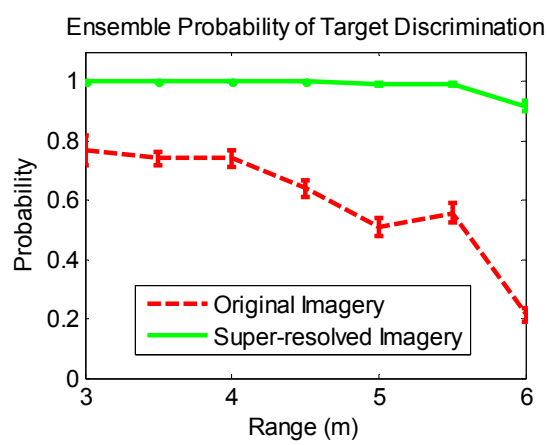


Figure 7. Chance-corrected probability of target discrimination at each target range.

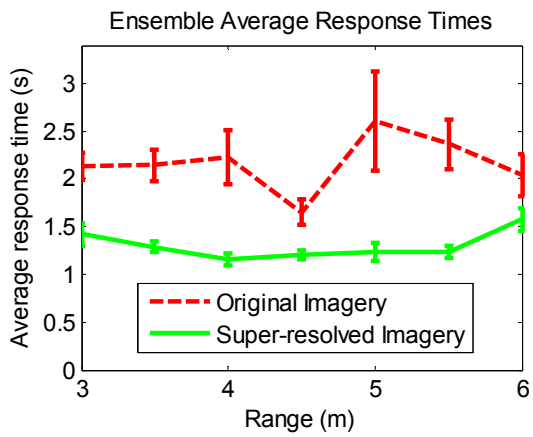


Figure 8. Response times at each target range with standard error bars showing inter-subject variability.

Not only were subjects able to achieve higher accuracy at all ranges with super-resolved imagery, but the response times were also faster with less variability for super-resolved imagery at all ranges. Figure 8 shows the group-averaged response times at each range with standard error bars representing inter-subject variability. At a range of 6 m, subjects responded in an average time of 1.58 s using the super-resolved imagery, 23 % faster than the response time using original imagery.

4. CONCLUSION

Super-resolution image reconstruction is complemented by a wavelet preprocessing stage for improved image registration, yields significant benefits for LADAR camera imagery. In the triangle orientation discrimination experiment, subjects achieved higher accuracy at all investigated target ranges with faster response times and reduced inter-subject variability for super-resolved imagery. Complemented by super-resolution image reconstruction, the high frame rate, small size, and light-weight LADAR camera sensors will be ideal for autonomous or semi-autonomous robot navigation in urban indoor environments. In semi-autonomous robot navigation, super-resolution enhancement will provide human operators with increased target discrimination. In fully autonomous mode, super-resolved imagery will allow guidance software to improve obstacle avoidance. The incorporation of super-resolution into the US Army's robotic applications will improve small robot performance, contributing to the soldier's survivability and lethality.

5. ACKNOWLEDGMENTS

The authors would like to thank Dr. Joe Reynolds and Dr. Keith Krapels at US Army's Night Vision and Electronic Sensors Directorate for their insights and helpful discussions. We thank Tommy Chang at NIST for helping the data in the NIST vision lab.

6. REFERENCES

[1] Ng, T. C. 2005. SIMTech technical reports, Vol. 6, No. 1, 13-18, 2005.

[2] Committee on Army Unmanned Ground Vehicle Technology 2002. "Technology development for Army unmanned ground vehicles", Sandia Report.

[3] MESA Imaging, 2006. SwissRanger SR-3000 Manual. http://www.mesa-imaging.ch/.

[4] Rosenbush, G., Hong, T. H., Eastman, R. D. 2007. Super-resolution enhancement of flash LADAR range data. In Proceedings of the SPIE, Vol. 6736, 67314.

[5] Vandewalle P., Susstrunk S., Vetterli, M. 2005. A frequency domain approach to registration of aliased images with application to super-resolution. EURASIP Journal on Applied Signal Processing, 71459

[6] Young, S. S., Driggers, R. G. 2006. Super-resolution image reconstruction from a sequence of aliased imagery. Applied Optics, Vol. 45, 5073-5085.

[7] Devitt, N., Moyer, S., Young, S. S. 2006. Effect of image enhancement on the search and detection task in the urban terrain. In Proceedings of SPIE, Vol. 6207, 62070D 1-13.

[8] Driggers, R. G., Krapels, K., Murrill, S., Young, S., Thielke, M., Schuler, J. 2005. Super-resolution performance for undersampled imagers. Optical Engineering, Vol. 44(1), 14002.

[9] Bijl, P., Valeton, J. M. 1998. Triangle orientation discrimination: the alternative to MRTD and MRC. Optical Engineering, Vol. 37(7), 1976-1983.

[10] Young, S. S., Driggers, R. G., Jacobs, E. L. 2008. Signal Processing and Performance Analysis for Imaging Systems. Artech House, Norwood, MA.

[11] Anderson, D., Herman H., Kelly A. 2005. Experimental characterization of commercial flash LADAR devices. In Proceedings of the International Conference of Sensing and Technology.

Performance Evaluation of Laser Trackers

B. Muralikrishnan, D. Sawyer, C. Blackburn, S. Phillips, B. Borchardt, W. T. Estler
Precision Engineering Division
National Institute of Standards and Technology
Gaithersburg, MD 20899

ABSTRACT
The American Society for Mechanical Engineers (ASME) recently released the ASME B89.4.19 Standard [1] on performance evaluation of spherical coordinate instruments such as laser trackers. At the National Institute of Standards and Technology (NIST), we can perform the complete set of tests described in the Standard, and have done so for a variety of laser trackers. We outline the tests described in the Standard, discuss our capabilities at the large-scale coordinate metrology group, and present results from B89.4.19 tests conducted on a few trackers. We also outline an analysis approach that may be used to evaluate the sensitivity of any measurement, including the tests described in the B89.4.19 Standard, to different geometric misalignments in trackers. We discuss how this approach may be useful in determining optimal placement of reference lengths to be most sensitive to different geometric misalignments.

Keywords
ASME B89.4.19, large-scale metrology, laser tracker, performance evaluation, sensitivity analysis

1. INTRODUCTION
Spherical coordinate measurement systems such as laser trackers, scanners and other devices are increasingly used in manufacturing shop floors for inspection, assembly, automation *etc*. These instruments are also sometimes used in the calibration of other lower-accuracy instruments such as industrial robots and even certain Cartesian coordinate measuring machines (CMMs). The spherical coordinate systems provide high accuracy at relatively low cost (in comparison to more conventional Cartesian CMMs), and are portable and convenient to use. Because of the proliferation of such devices in recent years, there has been an increasing need for a uniform set of performance tests so that users and manufacturers share a common understanding of the capabilities of the instrument.

In 2007, the American Society for Mechanical Engineers (ASME) published the ASME B89.4.19 Standard titled "Performance Evaluation of Laser-Based Spherical Coordinate Measurement Systems". This Standard, for the first time, defined a common set of tests that can be performed by both the user and the manufacture to establish if an instrument meets the manufacturer's performance specifications (MPE). This Standard, although limited to instruments that use a cooperative target such as retro-reflector, represents a significant step forward. It is the first and to date, the only performance evaluation Standard for spherical CMMs and establishes a framework for testing and evaluation of laser trackers and related devices.

In this paper, we present an overview of the B89.4.19 Standard and highlight the different tests described in it. We discuss capabilities of the large-scale coordinate metrology group at NIST where a complete set of B89.4.19 tests may be performed. We show examples of laser trackers tested at NIST that meet the manufacture's performance specifications and others that do not. Systematic errors due to geometrical and optical misalignments within a tracker are a major source of uncertainty in tracker measurements. An ideal performance evaluation test has high sensitivity to all misalignment parameters in a tracker's error model. Given the error model, it is possible to numerically determine the sensitivity of each of the B89.4.19 tests to different misalignment parameters. We have performed such analysis and briefly discuss our method and results.

2. THE ASME B89.4.19 STANDARD
The ASME B89.4.19 Standard describes three types of tests to be performed on trackers to evaluate their performance. These are the ranging tests, the length measurement system tests and two-face system tests.

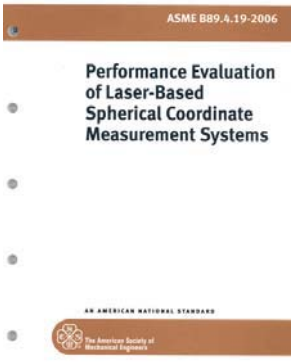


Figure 1. The ASME B89.4.19 Standard.

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

2.1 Ranging Tests

Ranging tests assess the distance (or displacement) measurement capability of the instrument. The ranging system (an interferometer or an absolute distance measurement (ADM) system) establishes the unit of length and is therefore a critical component of the system. The tests as described in the Standard require the tracker to measure several calibrated lengths aligned along the line-of-sight of the tracker. The reference lengths employed may be calibrated artifacts, realized by free standing targets, or a laser-rail system.

2.2 Length Measurement System Tests

The length measurement system tests are similar to volumetric length tests on Cartesian CMMs. A calibrated reference length is placed at different positions and orientations in the measurement volume and is measured by the tracker. The error in the measured length is compared against the MPE to determine conformance to specification. There are several sources of mechanical and optical misalignments within the construction of a tracker that produce systematic errors in the measured angle and range readings and therefore in measured lengths. The length measurement system tests are designed to be sensitive to these misalignments. Again, the reference lengths employed may be calibrated artifacts, realized by free standing targets, or a laser-rail system.

2.3 Two-face System Tests

Some geometric misalignments are such that the errors in the measured angles of a fixed target change in sign when the same target is measured in the backsight of the instrument. Such frontsight-backsight measurements of a single target are called two-face tests. These tests are extremely useful because they capture a large number of geometric misalignments and they do not require a calibrated reference length. The Standard requires two-face tests to be performed at different positions within the work volume of the instrument. More details on the test positions may be found in [1,2].

3. LARGE-SCALE METROLOGY AT NIST

The large-scale coordinate metrology group at NIST has the capability of performing the complete set of B89.4.19 tests.

The ranging tests are performed in the 60 m long-length test facility where a laser-rail and carriage system is operational (see Figure 2). The carriage has two opposing retro-reflectors. One retro-reflector is used for the tracker under test while the other is used for the reference interferometer on the other end of the rail. The expanded uncertainty (k = 2) of reference length L is $U(L)$ = 5 µm + 0.3 × $10^{-6}L$.

Figure 2. The NIST 60 m laser rail facility viewed from the tracker under test end; note the movable carriage with retroreflector.

The length measurement and two-face system tests are performed in the large-scale laboratory. Currently, the reference length for the length measurement tests is realized using the laser-rail and carriage system (LARCS) [3] (see Figure 3). The LARCS (different from the 60 m laser-rail facility used for range calibration) employs a reference interferometer mounted on a rail (about 3 m long) that can be oriented in different ways (horizontal, vertical, inclined) to meet the B89.4.19 requirements. A carriage with two retro-reflectors rides on the rail. The tracker uses one retro-reflector while the reference interferometer utilizes the other. The expanded uncertainty (k = 2) of a nominal reference length L is $U(L)$ = 3.4 µm + 0.5 × 10^{-6} L for the LARCS system. We are now evaluating different artifacts that may be used as the reference length instead of the LARCS system [4].

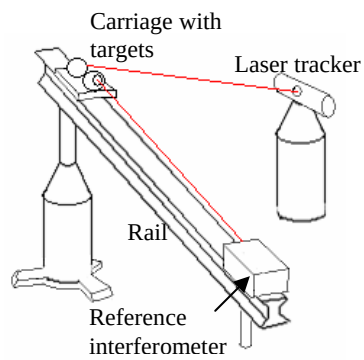


Figure 3. The LARCS laser rail.

4. TRACKER CALIBRATION EXAMPLES

We have performed the B89.4.19 tests on different trackers at our facility at NIST. Some trackers that were tested met the manufacture's specifications while others did not. We show results of ranging test performed on Tracker A in Figure 4. The data were recorded in the ADM mode of the tracker and substantially more points were recorded than required by the Standard. The errors in the measured lengths were within the manufacturer's specifications and therefore acceptable.

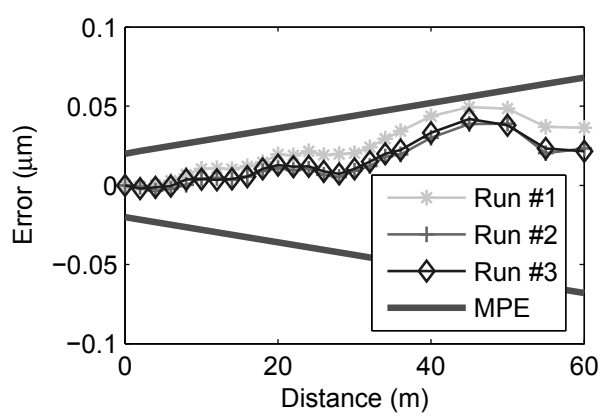


Figure 4. Ranging test results for Tracker A.

We present the results from the length measurement and two-face system tests for three trackers in Figures 5, 6, 7, 8, 9, and 10.

The length measurement system test charts (Figures 5, 7, and 9) may be interpreted as follows. The 35 length tests are in the order in which they appear in the Standard. Test 1 is the horizontal length measurement at the near position (1 m away, azimuthal angle of 0°). Tests 2 through 5 are the horizontal length measurements at four orientations of the tracker (0°, 90°, 180° and 270°) at the 3 m distance.

Tests 6 through 9 are the horizontal length measurements at four orientations of the tracker at the 6 m distance. Tests 10 through 17 are the vertical length measurements. Tests 18 through 25 are the right diagonal length measurements and tests 26 through 33 are the left diagonal length measurements. Tests 34 and 35 are the user-defined positions.

The two-face charts (Figures 6, 8, and 10) may be interpreted as follows. The 36 two-face tests are in the order in which they appear in the Standard. Therefore, tests 1 through 4 are the two-face measurements at the near position (1 m) with the target on the floor for four orientations of the tracker (0°, 90°, 180° and 270°).

Tests 5 through 8 are the two-face measurements at the near position (1 m) with the target at tracker height for four orientations of the tracker. Tests 9 through 12 are the two-face measurements at the near position (1 m) with the target at twice the tracker height for four orientations of the tracker (0°, 90°, 180° and 270°).

Tests 13 through 24 are a repetition of tests 1 through 12 but with the tracker 3 m away from the target. Tests 25 through 36 are a repetition of tests 1 through 12 but with the tracker 6 m away from the target. We discuss the tracker performance evaluation results next.

4.1 Tracker A

Tracker A (see Figures 5 and 6) clearly meets the manufacture's specification for length measurement and two-face system tests. The error in the measurement of a calibrated 2.3 m length placed 6 m away from the tracker was less than 25 μm, well under the manufacturer's specification of 100 μm. Small two-face errors, under 50 μm, for this tracker indicate that most of its geometric errors have been properly compensated.

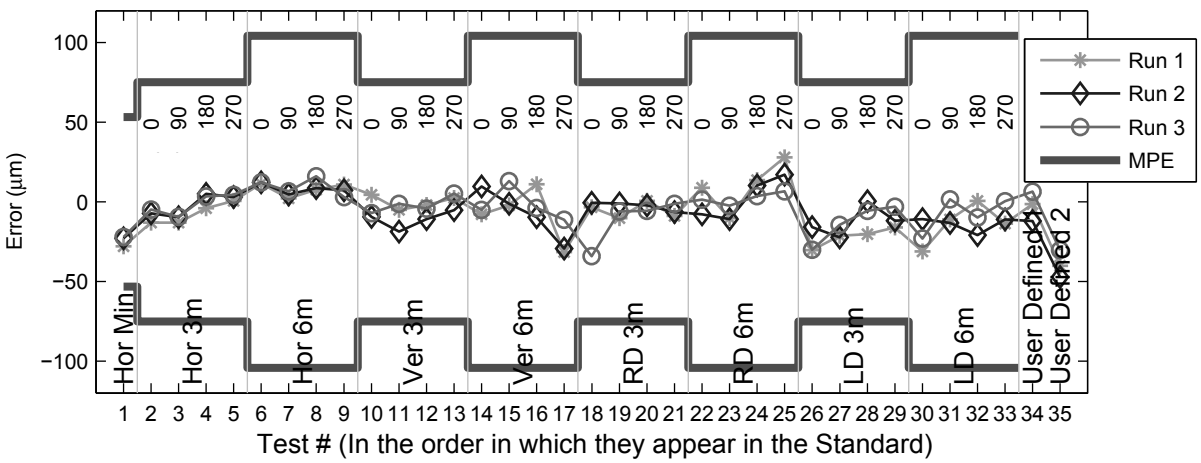


Figure 5. Length measurement system test results for Tracker A.

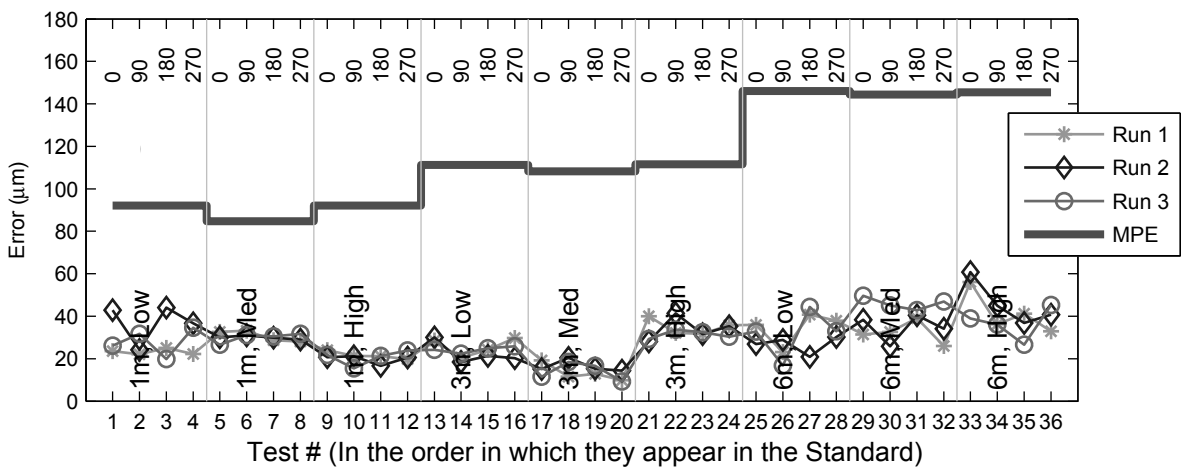


Figure 6. Two-face system test results for Tracker A.

4.2 Tracker B

Figures 7 and 8 show the results of length measurement and two-face system tests for Tracker B. This tracker appears to have satisfactory length measurement performance, but demonstrates large two-face errors.

Notice that the two-face error (Figure 8) demonstrates periodicity that is a function of the azimuth. In addition, the average two-face error (approximately 1.2 mm in Figure 8) does not change with distance of the target from the tracker. The average two-face error with increasing distance may arise from an offset in the beam from its ideal position (Tracker B does not have a beam steering mirror. Rather, the source is located directly in the head). Such an offset will result in decreasing error in the measured angle farther away from the tracker; consequently, the two-face error will be independent of range.

An explanation for the periodicity in the measured two-face data involves some subtlety. An eccentricity in the horizontal angle encoder will result in two-face errors showing periodicity that is a function of the azimuth. A least-squares fit of the data will therefore provide an estimate of the eccentricity. However, it turns out that when the two-face error, which is a convolved distance from horizontal and vertical angle error, is isolated into its constituent angle errors, the periodicity is in the vertical angle error. The vertical angle encoder has no functional relationship with the horizontal angle and therefore, the periodicity does not appear to be due to a geometric misalignment. The observed periodicity may be the result of stressing and relaxation of tension in cables within the tracker, or other such causes that are not considered in a geometric error model.

Although Tracker B did have large two-face errors, the B89.4.19 length measurement system test results (Figure 7) do not seem to be adversely affected by the large beam offsets. A careful consideration of the beam offset misalignment reveals that this term does not impact symmetrically placed lengths with respect to the tracker because the error at ends of the reference length cancel each other. Almost every B89.4.19 position is symmetrically placed with respect to the tracker. Asymmetrically positioned lengths will demonstrate large errors and may be used as the user-defined positions during the test.

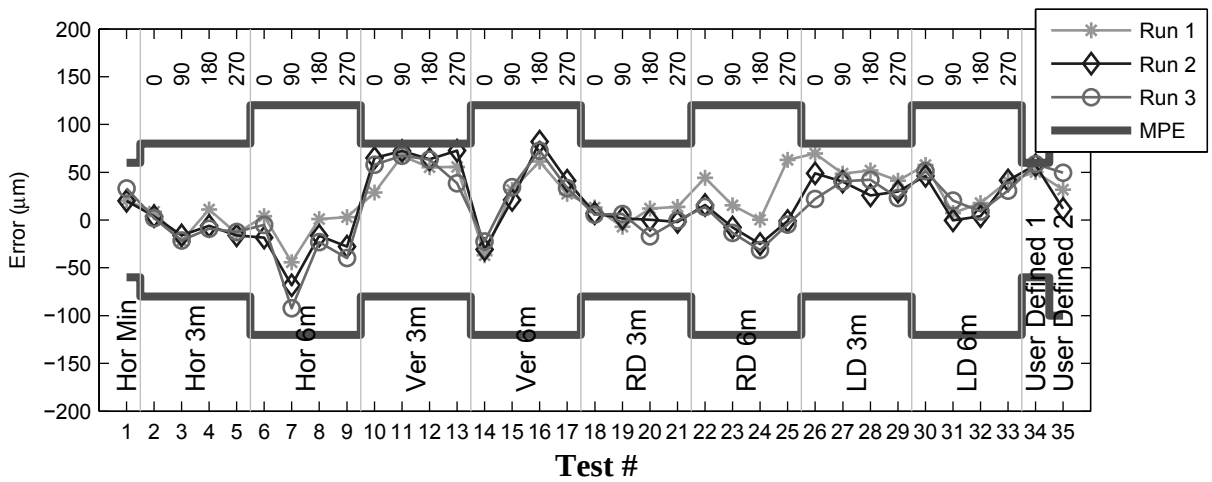


Figure 7. Length measurement system test results for Tracker B.

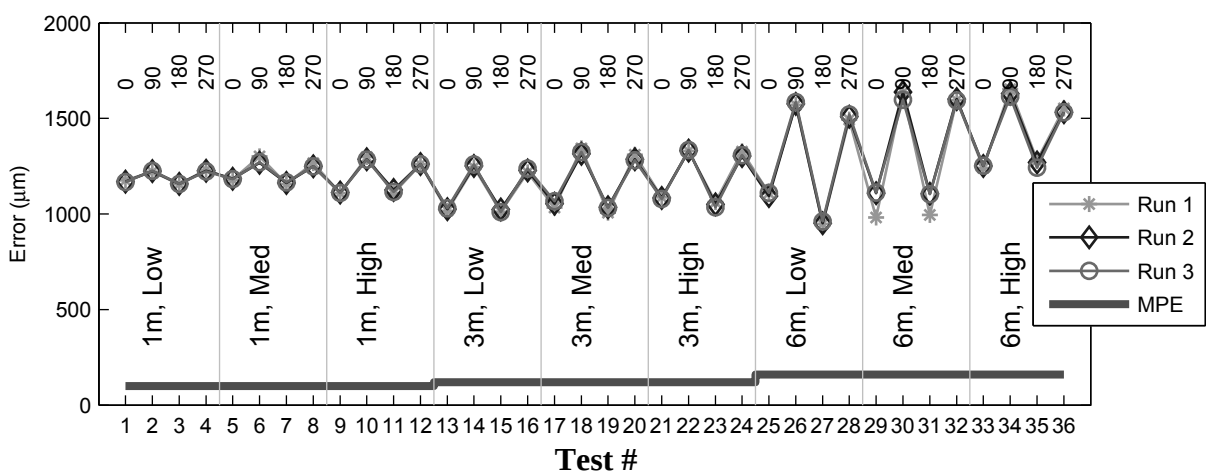


Figure 8. Two-face system test results for Tracker B.

Several interesting points raised in this section are worth summarizing:

- Large length measurement or two-face system test errors typically suggest that geometric misalignments have not been properly compensated.
- Two-face errors as reported in the B89.4.19 Standard are the convolved errors in both the horizontal and vertical angles, and scaled by the range. Raw horizontal and vertical angle errors from a two-face test contain more diagnostic information.
- The purpose of extracting the magnitude of physical misalignments from B89.4.19 tests is to estimate the error in other length measurements made within the work volume of the tracker.

4.3 Tracker C

Tracker C shows large errors in the length measurement system tests. The periodicity of the errors for Tracker C (Figure 9) may be the result of an eccentricity in the horizontal angle encoder, as well as a tilt in the beam as it emerges from its source. This tracker has the source located in the base and a beam steering mirror directs the beam to the target. Therefore, any tilt in the beam will also be a function of the azimuth. A least-squares best-fit may be performed to determine the magnitude of the eccentricity and tilt.

The two-face error (Figure 10) shows increasing error farther away from the tracker. A tilt in the beam as it emerges from the source will produce a constant angle error that when amplified by the range results in larger two-face error farther away from the tracker. We do not have the manufacturer's specification for two-face error for this tracker. The errors in Figure 10 are comparable to those of Tracker A and may therefore be acceptable, but nevertheless, there is evidence of improper compensation for geometric errors.

In this section, we have suggested that the B89.4.19 test results may be employed in a diagnostic mode where physical misalignments are determined. A geometric error model is required for this purpose. Error models are also useful in performing sensitivity analysis where the sensitivity of any given test to any geometric misalignment is determined through numerical simulation. We describe sensitivity analysis next.

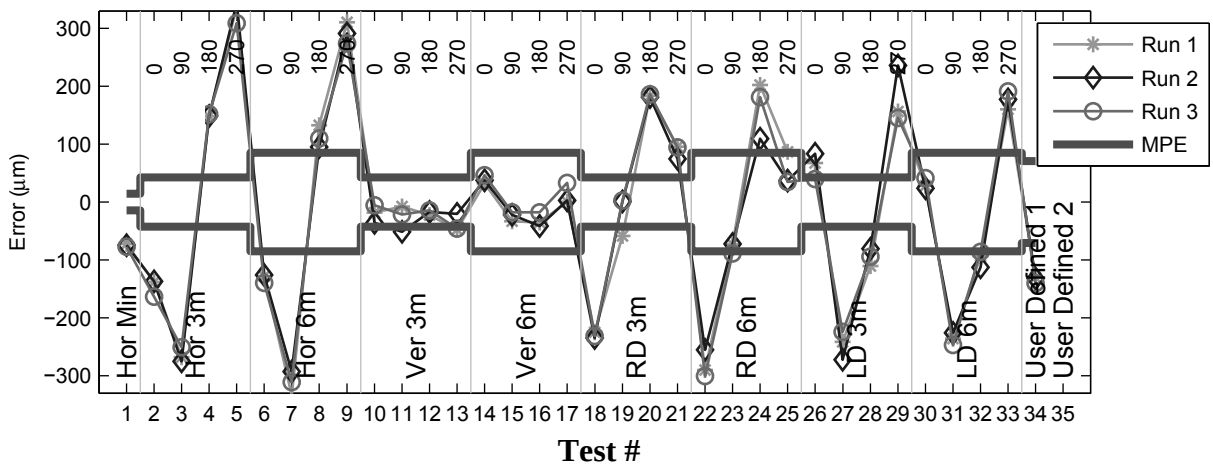


Figure 9. Length measurement system test results for Tracker C.

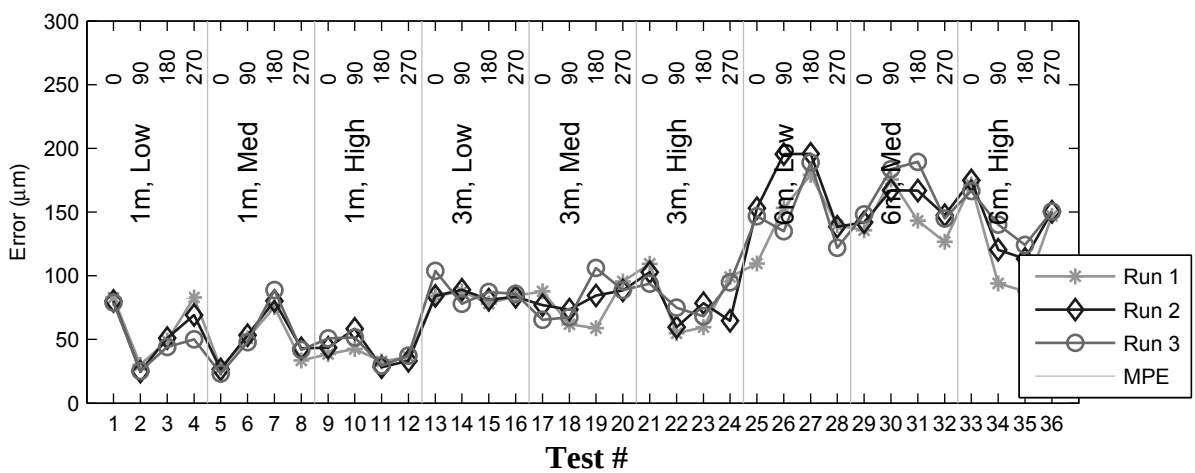


Figure 10. Two-face system test results for Tracker C.

5. SENSITIVITY ANALYSIS

A spherical coordinate instrument such as a laser tracker is a mechanical assembly of different components and therefore subject to misalignments such as offsets (offset in the beam from ideal position, offset between the standing and transit axes *etc*), tilt (tilt in the beam, tilt in the transit axis *etc*) and eccentricity (encoder eccentricity with respect to axis) during construction and assembly. It is general practice to correct for these misalignments by software compensation. A geometric error model [5] is required for this purpose that relates the corrected (true) range and angles to measured range and angles, and geometric misalignments within the tracker.

The corrected range (Rc) and angles (horizontal angle: Hc, vertical angle: Vc) of any coordinate in space are functions of several misalignment parameters within the construction of the tracker and also of the measured coordinate values at that location (Rm, Hm, Vm). The corrections ΔRm, ΔHm and ΔVm in Rm, Hm and Vm respectively may be expressed as (linear models may be sufficient as a first approximation),

$$
\begin{aligned}
Rc - Rm &= \Delta Rm = \sum_{i=1}^{n} x_i u_i(Rm, Hm, Vm) \\
Hc - Hm &= \Delta Hm = \sum_{i=1}^{n} x_i v_i(Rm, Hm, Vm) \\
Vc - Vm &= \Delta Vm = \sum_{i=1}^{n} x_i w_i(Rm, Hm, Vm)
\end{aligned}
\tag{2}
$$

where x is any misalignment parameter (eccentricity in encoder, beam offset, transit axis offset from standing axis, beam tilt, *etc*), and u, v and w are functions of measured range and angles.

Because different commercially available laser trackers have different mechanical constructions, an error model applicable to one tracker may not necessarily be applicable to another. At NIST, we have modeled three broad classes of trackers: a) tracker with a beam steering mirror for which the Loser and Kyle [5] model is applicable, b) tracker with the laser source in the rotating head and c) scanner with source mounted on the transit axis with a rotating prism mirror that steers the beam to the target.

As an example, an error model for a tracker with the source located in the head is given by

$$
\begin{aligned}
Rc &= Rm + x_2.\sin(Vm) + x_8 \\
Hc &= Hm + \frac{x_{1t}}{Rm.\sin(Vm)} + \frac{x_{4t}}{\sin(Vm)} + \frac{x_5}{\tan(Vm)} + x_{6x}\cos(Hm) \\
&\quad - x_{6y}\sin(Hm) + x_{9a}\sin(m.Hm) + x_{9b}\cos(m.Hm) \\
Vc &= Vm - \frac{x_{1m}}{Rm} + \frac{x_2\cos(Vm)}{Rm} + x_3 + x_{7n}\cos(Vm) \\
&\quad - x_{7z}\sin(Vm) + x_{9c}\sin(m.Vm) + x_{9d}\cos(m.Vm)
\end{aligned}
\tag{3}
$$

where x_{1t} and x_{1m} are beam offsets along the transit axis and its normal, x_2 is the transit offset, x_3 is the vertical index offset, x_{4t} is the beam tilt, x_5 is the transit tilt, x_{6x} and x_{6y} are the horizontal angle encoder eccentricities, x_{7x} and x_{7y} are the vertical angle encoder eccentricities, x_8 is the bird-bath error, and x_{9a}, x_{9b}, x_{9c} and x_{9d} are the components of the mth order scale error in the encoder (1st order is not distinguishable from encoder eccentricity. Higher orders beyond 2nd order may be neglected).

This error model may now be used to numerically estimate the sensitivity of any given test to geometric misalignments that are included in the model. As an example, consider the beam offset terms. We describe how the coefficients for the parameters in the error model were obtained, and then discuss the sensitivity of different B89.4.19 tests to this misalignment.

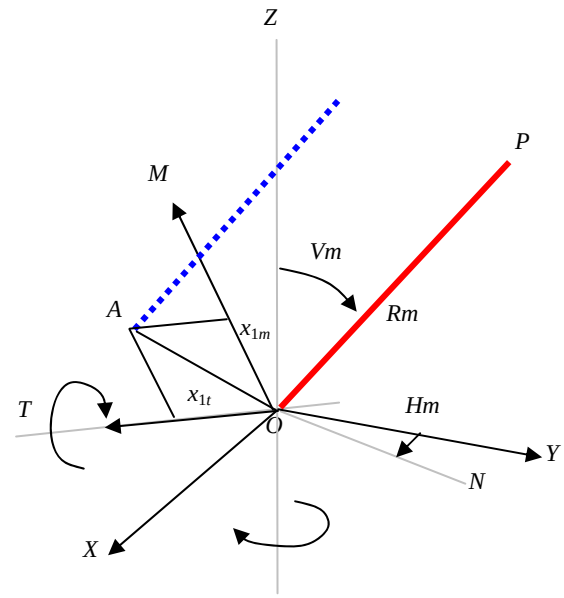


Figure 11. Schematic of beam offset in a tracker where the beam originates from the head (there is no beam steering mirror). Axes *OT*, *ON* and *OM* are fixed to the tracker's head and therefore rotate with the head about the *Z* axis.

5.1 Error Model Coefficients

The beam originating from the source (at O) may be displaced from its ideal position by a constant offset (OA in Figure 11) to emerge from A, a misalignment parameter referred to as beam offset. The offset can be resolved into components along two orthogonal axes, M and T (x_{1m} and x_{1t}). In Figure 11, OT is the transit axis, P is the target, ON is the projection of the beam to the target on the XY plane, and OM is the normal to both the transit axis and beam to the target. XYZ is a fixed Cartesian coordinate system with origin at O. TNZ is Cartesian coordinate system, also with origin at O, but attached to the tracker head so that it can rotate about the Z axis. TPM is also a Cartesian system with origin at O and attached to the tracker so that it can rotate about the transit axis OT. The offset component along the transit (x_{1t}) produces an error in the measured horizontal angle. The correction for the beam offset is given by $\Delta Hm = \dfrac{x_{1t}}{Rm.\sin(Vm)}$.

The component along its normal (x_{1m}) produces an error in the measured vertical angle, and its correction is given by $\Delta Vm = \dfrac{-x_{1m}}{Rm}$.

5.2 Sensitivity to Two-face System Tests

The effect of beam offset on two-face tests described in the B89.4.19 Standard can be determined as follows. The corrections for the measured horizontal and vertical angles of a target placed distance *Rm* away from the front face of the tracker are given above. These corrections reverse in sign when the tracker is in the backsight mode. The apparent distance *E* in a two-face test is therefore given by $E = 2\Delta Hm.Rm.\sin(Vm) = 2x_{1t}$ for offset along the transit axis and $E = 2\Delta Vm.Rm = 2x_{1m}$ for offset along *OM*. Both beam offset parameters are therefore sensitive to every two-face test described in the Standard by the same sensitivity factor of 2.

5.3 Sensitivity to Length Measurement System Tests

Systematic errors in measured range and angles lead to an error in the determination of the coordinates of each end of the reference length. This however does not necessarily imply an error in the calculated length between the two ends because the error vectors (vector between true coordinate and measured coordinate) at the two ends may simply result in translation and/or rotation of the length, but not a change in its magnitude. Sensitivity to length tests is achieved primarily if the error vectors at the two ends produce components along the length with non-zero sum. Components perpendicular to the length are generally not sensitive.

Any symmetrically placed reference length (such as the horizontal, vertical or diagonal length tests in the Standard) is not sensitive to beam offset because it only serves to translate and rotate the length. The default position for the first user-defined test (asymmetrical vertical length test) is sensitive to beam offset along the *M* axis because the asymmetrical positioning of the reference length creates unequal error components at the two ends of the reference length which do not completely cancel each other.

6. SUMMARY

The complete set of B89.4.19 tests may be numerically simulated and the sensitivity of each test to every parameter in the error model can be determined. The results, tabulated as a two-dimensional matrix with B89.4.19 tests in one axis and misalignment parameters in the other, is a "sensitivity matrix." We have created such sensitivity matrices for the three classes of trackers for which we have developed error models. Such matrices are useful in assessing the capabilities and limitations of any set of performance evaluation tests.

Analysis of such sensitivity matrices further leads to the identification of optimal length positions that are sensitive to specific geometric misalignments. For example, we mentioned that the beam offset misalignment parameter was not sensitive to horizontal, vertical, or diagonal length tests because of the symmetrical nature of the positioning of the reference length with respect to the tracker. Sensitivity analysis indicated that asymmetrical positioning is advantageous, and therefore may be considered as a user-defined test.

7. CONCLUSIONS

The recently released ASME B89.4.19 Standard provides a common set of performance evaluation tests that may be performed by both the manufacturer and the user to evaluate if the instrument meets the manufacture's specifications.

The Standard contains three types of tests. The ranging tests assess the instrument's distance (or displacement) measuring capability. The length measurement and two-face system tests identify any systematic errors within the instrument's construction, such as mechanical and optical misalignments. The length measurement system tests require a calibrated reference length (typically 2.3 m long) realized either as an artifact or using laser-rails, or between free standing targets calibrated by other means. The ranging tests require a reference interferometer and a laser-rail and carriage system where long lengths may be calibrated or some other means to independently measure long lengths reliably. The two-face tests require no reference lengths. They are simple and easy to perform, and capture a large number of geometric misalignments.

The B89.4.19 test results provide valuable diagnostic information as well. Using geometric error models of the tracker, it may be possible to estimate magnitudes of misalignments in the construction of the tracker. Such information may then be used in determining errors in other length measurements made within the work volume of the tracker.

Geometric error models also serve a more general role. They may be used to determine the sensitivity of any given test to any geometric misalignment within the tracker. Such sensitivity analysis is useful in determining if a given set of performance evaluation tests effectively captures the misalignments, or if any modifications in the placement of reference lengths are necessary.

8. REFERENCES

[1] ASME B89.4.19-2006 Standard – Performance Evaluation of Laser-Based Spherical Coordinate Measurement Systems, www.asme.org.

[2] W.T.Estler, D.S.Sawyer, B.Borchardt, and S.D.Phillips, Laser-scale metrology instrument performance evaluations at NIST, The Journal of the CMSC, Vol. 1, No. 2, pp 27-32, 2006.

[3] D. Sawyer, B. Borchardt, S.D.Phillips, C. Fronczek, W.T.Estler, W. Woo, and R.W. Nickey, A laser tracker calibration system, Proceedings of the Measurement Science Conference, Anaheim, CA, 2002.

[4] D. Sawyer, NIST progress in the development of a deployable high-accuracy artifact for laser tracker performance evaluation per ASME B89.4.19, Proceedings of the CMSC conference, Charlotte, NC, July 21-25, 2008.

[5] Raimund Loser and Stephen Kyle, Alignment and field check procedures for the Leica Laser Tracker LTD 500, Boeing Large Scale Optical Metrology Seminar, 1999.

Preliminary Analysis of Conveyor Dynamic Motion for Automation Applications

Jane Shi, Ph.D.
GM R&D Center
30500 Mound Road
Warren, MI, USA
Jane.Sh@gm.com

ABSTRACT
In order to eliminate the automation stop stations in automotive general assembly (GA) and enable robotic automation on continuous moving conveyors, it is essential that the conveyors' dynamic motion behaviors at the assembly plant floor are characterized and quantified. A six degree of freedom (DOF) accelerometer with embedded data logging capability has been utilized to collect the conveyor dynamic motion data from several assembly plants. This paper summarizes conveyor dynamic motion data collected from several assembly plants, analyzes these dynamic motion data using simple statistical analysis to quantify the conveyor motion's stability in the main direction of travel, and identify any frequency pattern of the conveyor dynamic motion data with the Fast Fourier Transformation (FFT) frequency analysis.

Categories and Subject Descriptors
C.4 [**Performance of Systems**]: Performance attributes.
C.3 [**Special-Purpose and Application-Based Systems**]: Process control systems; Real-time and embedded systems.

General Terms
Measurement, Verification.

Keywords
Motion quantification, Statistical analysis, FFT

1. INTRODUCTION

In automotive general assembly (GA), vehicle bodies are being carried on a continuous moving assembly line. The vehicle bodies are moving through hundreds of workstations where a variety of parts are being assembled together, typically by human operators as shown in Figure 1.

Current automation of assembly tasks are implemented in stop stations, i.e. the vehicle body stops and remains stationary for the entire time when the robotic assembly is being performed. A stop station in GA requires a long section of high speed conveyors before the stop station for job accumulation, then again a long section of high speed conveyors after the stop station for synchronizing with a slow moving conveyor and manual workstations. The GA automation stop stations are expensive: one stop station takes up an equivalent floor space of 7 manual workstations.

Figure 1 Automotive General Assembly Conveyors Transport Vehicle Bodies for Assembly Operations

One of the solutions to eliminate the automation stop stations in general assembly is to automate assembly operations on a moving line. Line tracking is the fundamental enabler for robotic automation on a continuous moving line. In order to develop adequate line tracking capabilities, the conveyor's motion need to be characterized. This paper presents the method to collect the conveyor motion data, summarizes the collected data, and presents the motion frequency analysis results and basic statistical analysis results for both linear acceleration and computed linear speed.

2. CONVEYOR MOTION DATA COLLECTION METHOD

Line tracking of a moving vehicle body for assembly is mainly concerned with the vehicle body moving speed in the 3D space with respect to a fixed reference frame as shown in Figure 2. Given that OwXwYwZw is the fixed frame and OXYZ is the moving frame attached to the moving vehicle body, Xw is the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

main direction of conveyor motion. The goal of conveyor motion data collection is to record the frame OXYZ position or motion speed (Vx,Vy,Vz,Rx,Ry,Rz) or its acceleration with respect to the stationary frame for a coverage of an entire workstation.

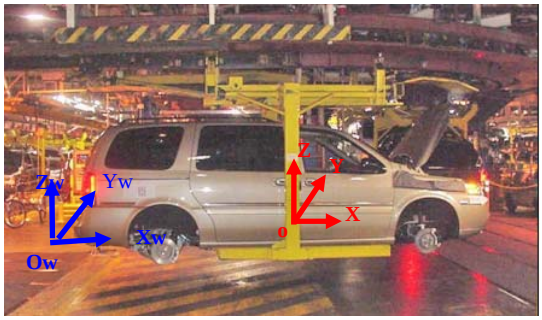


Figure 2 Line Tracking Reference Frames

An ideal method and its associated instrument to collect the vehicle body motion data should be simple to set up, easy to initiate for data recording, and non-intrusive to operators on the production line since the instrument has to be riding with the vehicle body on the conveyors for a number of repeated data sampling. In addition, minimum secondary processing of recorded raw data is desired. Many of high speed camera or laser tracking systems are complex to set up, require line of sight, and are intrusive to the natural production environment on the assembly plant floor. After investigating several available motion sensors, the six degree of freedom (DOF) accelerometer, EDR3D-6DOF [1], as shown in Figure 3, is chosen based on its capabilities of embedded simultaneous data logging of all 6 axes data: three linear accelerations and three angular velocities. EDR3D-6DOF allows about 30 minute data logging of linear acceleration and angular velocity at 500 Hz rate of 10 bit to fill up 4 megabytes of embedded flash memory. An external toggle switch simply turns on or off the data logging. In addition magnets provide convenient way of mounting the instrument to any steel surface.

Figure 3 An Accelerometer with Embedded Data Logging Capabilities

Table 1 below summarizes the measurement range and the resolution for all six axes.

Table 1 Accelerometer Measurement Range and Resolution

Parameter		Measurement Range	Resolution	
Linear Acceleration	X Acceleration	2.23g	0.005g	1.93 in/sec^2
	Y Acceleration	2.28g	0.005g	1.93 in/sec^2
	Z Acceleration	2.30g	0.005g	1.93 in/sec^2
Angular Velocity	X Rotation	78.5 deg/sec	0.0766 deg/sec	
	Y Rotation	77.3 deg/sec	0.075488 deg/sec	
	Z Rotation	79.6 deg/sec	0.077734 deg/sec	

When placed with X direction aligned with conveyor main travel direction, the EDR3D-6DOF is used to collect three linear accelerations and three angular velocities of conveyors from the assembly plants.

3. RAW MOTION DATA COLLECTED

Raw conveyor motion data have been collected from a total of five different types of conveyor at four different assembly plants from late June through early September of 2007. For each conveyor type, the accelerometer was placed on the conveyor to ride through a workstation or a section of multiple workstations. Conveyor motion data are repeatedly sampled five times so that the randomness of conveyor motion can be studied. During the active data recording period, any external disturbances to the natural conveyor movement, such as an operator stepping on the conveyor carriers or a heavy machine loading onto the vehicle body, were noted so that any exceptional signal in the raw data can be detected. In this paper, only representative samples of data are illustrated for two types of conveyors: a platform type of conveyor with stable and smooth motion, a chain type of conveyor with rough and jerky motion.

Figure 4 shows three linear accelerations for a platform type of conveyor with stable and smooth motion.

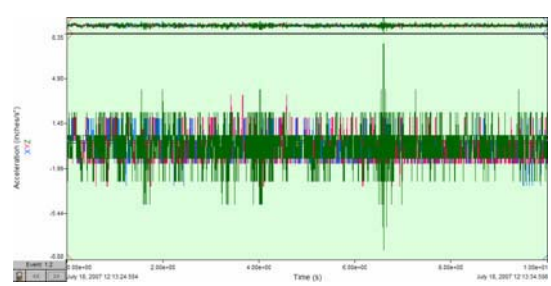


Figure 4 Raw Linear Accelerations of a Stable Conveyor

For a conveyor that exhibits rough and jerky motion, Figure 5 is an example graph of three linear accelerations.

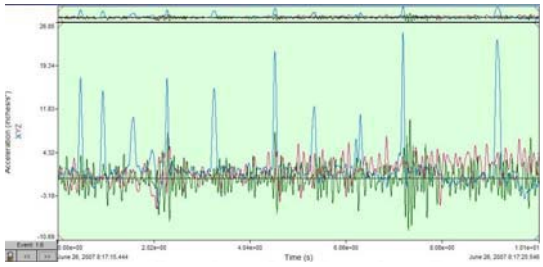


Figure 5 Raw Linear Accelerations for a Jerky Conveyor

Please note that the scale of Figure 5 is significantly larger than the scale of Figure 4.

The linear acceleration in X direction (main conveyor motion direction) exhibits significant jerky motion (blue) that is of short duration with big changes in X acceleration. Figure 6 through 9 are four samples of the same workstation with jerky motion.

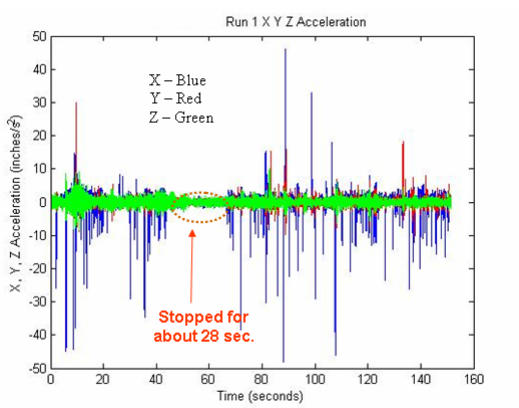


Figure 6 First Run of a Jerky Conveyor

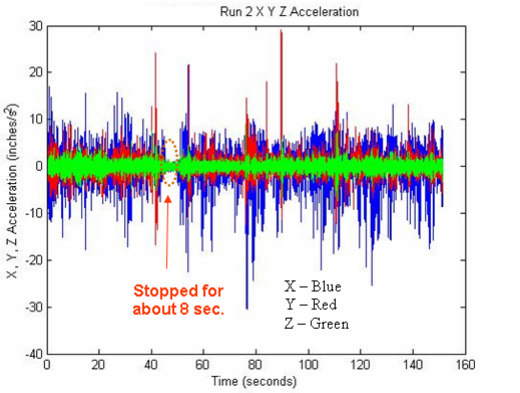


Figure 7 Second Run of a Jerky Conveyor

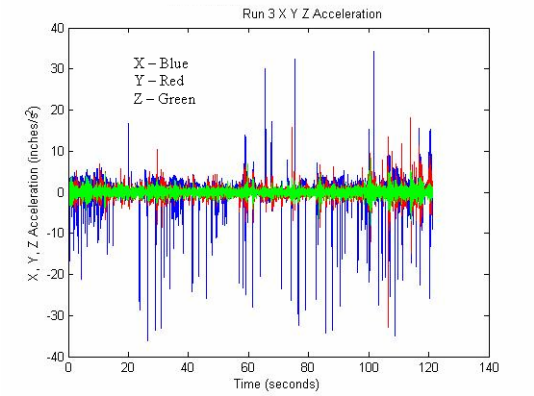


Figure 8 Third Run of a Jerky Conveyor

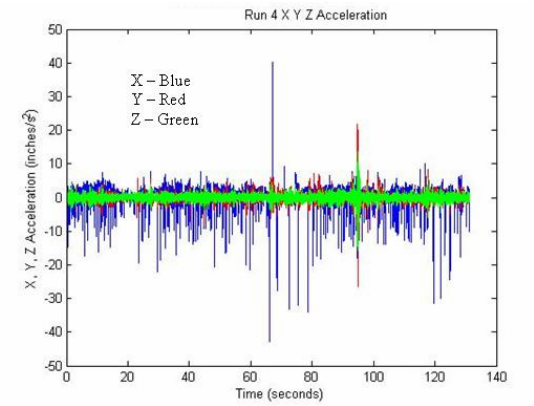


Figure 9 Fourth Run of a Jerky Conveyor

During the data collection period, the vehicle body stops at a random time for a time period and then starts again. Depends on the specific production activity, the stop and re-start occurs randomly. Figure 10 and 11 are two examples of captured stop and/or start of a conveyor.

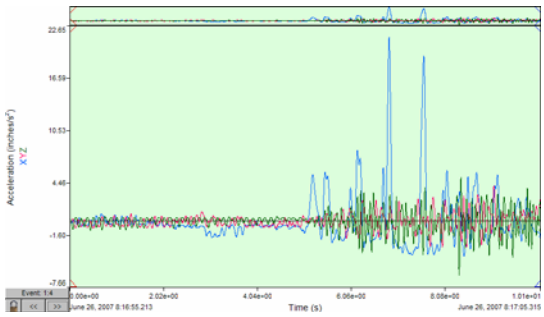


Figure 10 Normal Start of A Jerky Conveyor

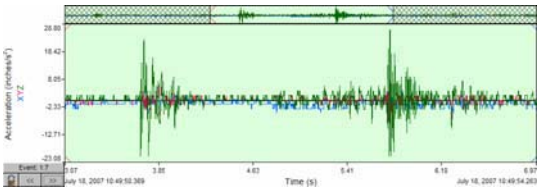


Figure 11 Normal Stop and Start of A Smooth Conveyor

This section illustrates several representative data samples only. Given the multiple data sets for each of five types of conveyors, statistical analysis can identify the stability of each conveyor and allow the comparison of different types of conveyor. Signature motion of a conveyor can be identified so that the emulation of the representative motion can be rendered by other motion devices, such as a robot [2].

4. STATISTICAL ANALYSIS OF CONVEYOR LINEAR ACCELERATIONS

Basic statistical analysis of three linear accelerations, the mean, the standard deviation, maximum and minimum value, has been performed and the results are summarized in Table 2.

Table 2 Three Linear Acceleration Statistics

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Linear Acceleration	Mean (X) in/sec^2	Mean (Y) in/sec^2	Mean (Z) in/sec^2	StdDev(X) in/sec^2	StdDev(Y) in/sec^2	StdDev(Z) in/sec^2	Max(X) in/sec^2	Max(Y) in/sec^2	Max(Z) in/sec^2	Min(X) in/sec^2	Min(Y) in/sec^2	Min(Z) in/sec^2
2	Conveyor 1 Sample 1	0.00	0.00	0.00	0.45	0.31	0.13	1.36	3.07	0.42	-1.42	-5.33	-0.91
3	Conveyor 1 Sample 2	0.00	0.00	0.00	0.53	0.29	0.15	1.83	3.63	1.42	-2.02	-4.63	-2.39
4	Conveyor 1 Sample 3	0.00	0.00	0.00	0.53	0.29	0.15	1.83	3.63	1.42	-2.02	-4.63	-2.39
5	Conveyor 2 Sample 1	0.01	0.00	0.00	0.53	0.28	0.28	1.89	1.09	1.04	-3.66	-1.04	-1.68
6	Conveyor 2 Sample 2	0.00	0.00	0.00	0.62	0.26	0.27	1.62	1.80	1.80	-4.28	-1.20	-1.11
7	Conveyor 2 Sample 3	0.00	0.00	0.00	0.66	0.31	0.32	1.78	1.27	1.44	-3.73	-1.51	-1.12
8	Conveyor 2 Sample 4	0.00	0.01	0.00	0.74	0.59	0.67	3.83	2.85	3.86	-5.71	-3.06	-2.99
9	Conveyor 2 Sample 5	0.00	0.00	0.00	0.58	0.57	0.70	3.77	5.50	5.86	-6.14	-5.68	-4.86
10	Conveyor 3 Sample 1	0.00	0.00	0.00	4.34	1.19	1.66	30.71	10.30	13.26	-26.73	-12.11	-11.14
11	Conveyor 3 Sample 2	0.00	0.00	0.00	5.17	1.45	2.16	44.75	24.47	19.46	-29.76	-12.17	-18.92
12	Conveyor 3 Sample 3	0.00	0.00	0.00	4.26	1.35	1.50	21.95	12.63	9.24	-25.43	-14.99	-11.94
13	Conveyor 3 Sample 4	0.00	0.00	0.00	4.22	2.08	1.76	41.01	17.69	12.62	-25.00	-22.02	-11.83
14	Conveyor 3 Sample 5	0.00	0.00	0.00	5.56	1.74	2.54	50.12	16.68	23.28	-31.13	-12.81	-20.34
15	Conveyor 4 Sample 1	-0.01	0.00	0.00	3.45	1.18	0.65	6.84	9.38	3.92	-38.54	-6.71	-3.20
16	Conveyor 4 Sample 2	0.01	0.00	0.00	3.04	0.82	0.37	20.66	5.30	2.21	-27.13	-5.65	-2.51
17	Conveyor 4 Sample 3	-0.01	0.00	0.00	3.18	0.57	0.48	28.51	2.90	2.85	-36.79	-2.49	-2.93
18	Conveyor 4 Sample 4	0.00	0.00	0.00	4.08	0.99	0.76	39.77	5.05	3.60	-33.76	-5.17	-4.84
19	Conveyor 4 Sample 5	-0.01	0.00	0.00	3.90	0.84	0.67	20.33	7.05	3.70	-37.93	-8.83	-3.14
20	Conveyor 4 Sample 6	0.00	0.00	0.00	3.00	0.90	0.59	5.18	5.97	3.68	-35.70	-12.83	-3.26
21	Conveyor 5 Sample 1	0.00	0.01	0.00	3.94	1.21	0.56	38.57	22.72	3.95	-42.00	-7.67	-5.02
22	Conveyor 5 Sample 2	-0.01	0.00	0.00	3.97	1.79	0.42	17.56	21.22	3.39	-24.71	-11.21	-4.61
23	Conveyor 5 Sample 3	0.00	0.00	0.00	4.46	1.53	0.38	29.12	14.29	3.54	-31.14	-26.22	-6.13
24	Conveyor 5 Sample 4	-0.01	0.00	0.00	4.18	1.23	0.48	37.28	17.27	5.45	-31.56	-15.57	-7.81
25	Conveyor 5 Sample 5	-0.01	0.00	0.00	3.56	1.23	0.36	29.04	16.92	4.07	-34.77	-10.95	-2.84
26	Conveyor 5 Sample 6	0.00	0.00	0.00	4.63	1.35	0.38	30.04	23.30	5.27	-39.49	-13.05	-5.60
27	Conveyor 6 Sample 1	0.00	0.00	0.00	2.22	1.36	1.47	22.62	15.45	15.53	-15.12	-10.17	-12.00
28	Conveyor 6 Sample 2	0.00	0.00	0.00	1.96	0.80	0.81	9.77	5.36	6.59	-5.87	-6.90	-5.01
29	Conveyor 6 Sample 3	0.01	0.00	0.00	2.69	0.83	1.42	12.71	4.71	15.04	-11.83	-8.89	-8.87
30	Conveyor 6 Sample 4	0.00	0.00	0.00	1.73	0.82	0.98	8.37	5.42	5.52	-8.99	-5.27	-5.65
31	Conveyor 6 Sample 5	0.00	0.00	0.00	1.88	0.42	0.75	8.72	2.41	5.40	-6.80	-1.85	-4.84

As it is expected, the mean, as listed in column B,C,D, for three linear accelerations is zero. This means that there is no net acceleration for the linear motion even though the instant value of the acceleration can be quite large in either positive or negative direction as shown by the maximum and minimum values.

The range of acceleration distribution is measured by one standard deviation, as listed in column E,F,G.. So the conveyor motion's stability can be evaluated simply by the value of standard derivation. The smaller the standard deviation value, the more stable the conveyor motion. Although Conveyor 5 moves at lower average speed, it is most unstable conveyor with the widest range of speed and highest magnitude of jerk (change rate of acceleration).The next unstable conveyor is Conveyor 4. Stability of Conveyor 3 is similar to Conveyor 4. Conveyor 6 has better stability than previous two types of conveyor. Both Conveyor 1 and 2 are significantly stable as shown by their standard derivation as well as maximum and minimum values.

More sophisticated statistical analysis can be performed, possibly in future, to further analyze the dynamic motion data for detection of any relationships among the sample sets.

5. FAST FOURIER TRANSFORMATION ANALYSIS OF CONVEYOR LINEAR ACCELERATION

In order to detect any frequency patterns of the collected acceleration data, Fast Fourier Transformation (FFT) analysis has performed for all linear acceleration data. Sample graph of FFT analysis results are illustrated for a smooth conveyor, Figure 12 and 13, and for a jerky conveyor by Figure 14 and Figure 15. The horizontal X axis is the frequency in Hz and the vertical Y axis is the power spectrum distribution at a particular frequency. The peaks are the frequency concentration of the acceleration.

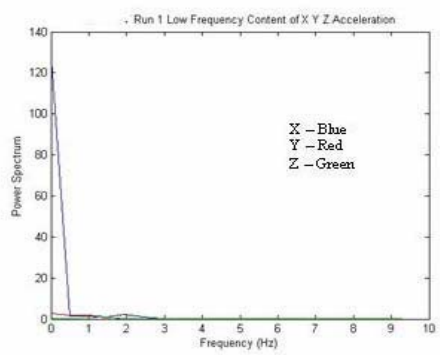


Figure 12 FFT Results of First Run of a Smooth Conveyor

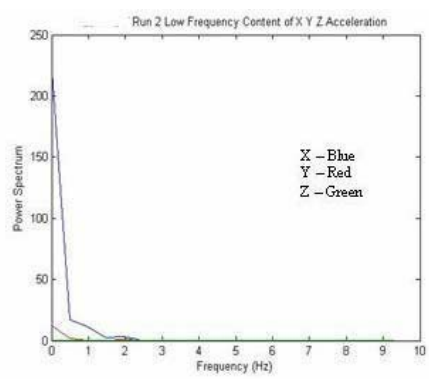


Figure 13 FFT Results of Second Run of a Smooth Conveyor

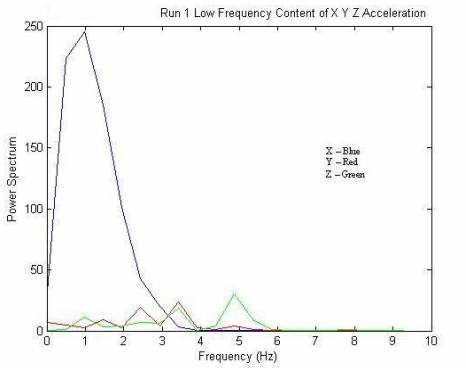


Figure 14 FFT Results of First Run of a Jerky Conveyor

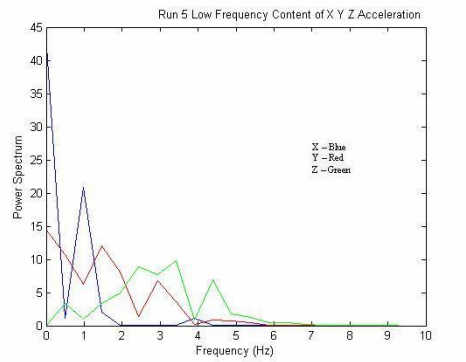


Figure 15 FFT Results of Fifth Run of a Jerky Conveyor

Table 3 is the summary of the frequencies for the first three highest peaks of FFT analysis results for a total of six conveyors.

Table 3 FFT Frequency of Three Linear Accelerations

	A	B	C	D	E	F	G	H	I	J
1	FFT Results of Linear Acceleration	X Frequency (Hz)	X Frequency (Hz)	X Frequency (Hz)	Y Frequency (Hz)	Y Frequency (Hz)	Y Frequency (Hz)	Z Frequency (Hz)	Z Frequency (Hz)	Z Frequency (Hz)
2	Conveyor 1 Sample 1	0.00	1.95	-	0.00	1.95	-	1.46	0.98	-
3	Conveyor 1 Sample 2	0.00	0.49	0.98	0.00	0.49	0.98	0.98	0.00	-
4	Conveyor 1 Sample 3	0.00	0.49	0.98	0.00	1.95	0.98	0.98	0.00	0.34
5	Conveyor 2 Sample 1	1.46	0.00	0.49	0.49	1.46	0.00	1.46	0.00	0.49
6	Conveyor 2 Sample 2	0.49	2.44	0.98	1.46	0.00	0.98	1.46	0.00	2.93
7	Conveyor 2 Sample 3	1.95	0.00	1.46	0.49	1.46	1.95	1.95	0.00	2.44
8	Conveyor 2 Sample 4	1.95	0.49	2.44	1.46	0.98	0.00	1.46	1.95	2.44
9	Conveyor 2 Sample 5	0.00	0.49	2.44	1.46	0.00	1.95	1.46	2.93	0.00
10	Conveyor 3 Sample 1	1.95	0.00	1.46	1.95	0.49	2.93	1.95	2.93	3.91
11	Conveyor 3 Sample 2	1.95	1.46	0.98	0.49	1.95	1.46	2.44	3.42	2.93
12	Conveyor 3 Sample 3	1.95	1.46	0.98	0.49	5.37	1.95	1.95	2.44	2.93
13	Conveyor 3 Sample 4	1.95	1.46	0.98	1.95	1.46	0.49	1.95	3.91	2.93
14	Conveyor 3 Sample 5	1.95	0.00	1.46	1.95	0.00	1.46	1.95	2.93	2.44
15	Conveyor 4 Sample 1	0.98	0.49	1.46	3.42	2.44	1.46	4.88	3.42	0.98
16	Conveyor 4 Sample 2	0.00	0.49	0.98	2.44	1.46	0.00	3.91	1.95	2.93
17	Conveyor 4 Sample 3	0.00	1.46	0.49	0.00	1.95	2.44	4.88	5.37	2.44
18	Conveyor 4 Sample 4	0.98	0.49	2.93	1.46	0.98	1.95	3.91	3.42	1.46
19	Conveyor 4 Sample 5	0.00	0.98	1.46	0.00	0.49	1.46	3.42	2.44	2.93
20	Conveyor 4 Sample 6	0.00	0.49	1.46	0.98	0.49	2.44	3.91	3.42	1.95
21	Conveyor 5 Sample 1	0.49	0.00	0.98	0.00	2.44	1.95	3.42	2.93	2.44
22	Conveyor 5 Sample 2	0.98	1.95	1.46	0.98	3.42	4.39	2.44	3.91	0.49
23	Conveyor 5 Sample 3	0.98	1.95	0.49	2.93	1.46	3.42	3.42	2.44	0.98
24	Conveyor 5 Sample 4	0.49	0.98	1.46	1.46	1.95	2.44	2.93	0.98	1.46
25	Conveyor 5 Sample 5	2.44	0.98	0.49	2.44	1.95	1.46	2.93	2.44	0.49
26	Conveyor 5 Sample 6	1.46	0.98	0.49	1.46	0.00	1.95	3.91	3.42	2.44
27	Conveyor 6 Sample 1	0.00	1.46	0.98	0.98	1.46	0.49	0.00	0.98	0.49
28	Conveyor 6 Sample 2	0.49	0.98	0.00	0.49	0.00	1.95	0.49	0.98	1.46
29	Conveyor 6 Sample 3	0.00	0.49	2.93	0.00	0.49	1.46	1.46	1.95	3.42
30	Conveyor 6 Sample 4	0.49	0.98	1.46	0.00	0.49	0.98	0.98	0.49	0.00
31	Conveyor 6 Sample 5	0.49	0.00	0.98	0.00	0.49	1.95	1.95	0.00	2.44

When the highest dominant peak occurs at zero frequency, it means that zero acceleration is dominant as it is in Conveyor 1 case, and the conveyor motion speed is constant. When multiple and equally strong peaks occur at different frequencies, it means that the conveyor acceleration varies at combined multiple frequencies as it is in Conveyor 5 case. Based on the FFT analysis results, it can be concluded that all jerky motion along the conveyor main travel direction is of random nature, i.e. they do not occur at a fixed frequency or fixed multiple frequencies with a regular pattern.

6. COMPUTED SPEED AND POSITION DATA

Given conveyors' three linear acceleration data sampled at 500Hz, conveyors linear speed can be computed with a numerical integration method. It has turned out that the computed speed and position can not reliably reflect the accurate stop and start scenarios. Any small linear trend or nonzero mean in acceleration data will cumulate over time and it will lead to significant error in computed speed and position. After several attempts and experiments, it is discovered that the following pre-processing and necessary filtering can yield good speed and position data.

- First, the linear trend of the linear acceleration within every 10 second period is removed
- Second, the acceleration data is filtered at 5 Hz since high frequency vibration is of no interest for speed computation.
- Third, the mean is set to zero since it is assumed no net acceleration of the conveyor motion.
- Finally, the speed and position are computed using the first order linear integration with 0.002 second interval.

Sample graphs of computed conveyor speed and the 3D position are illustrated for a smooth conveyor, Figure 16 and 17, and for a jerky conveyor by Figure 18 and Figure 19. The horizontal X axis is the time in seconds and the vertical Y axis is the speed (in

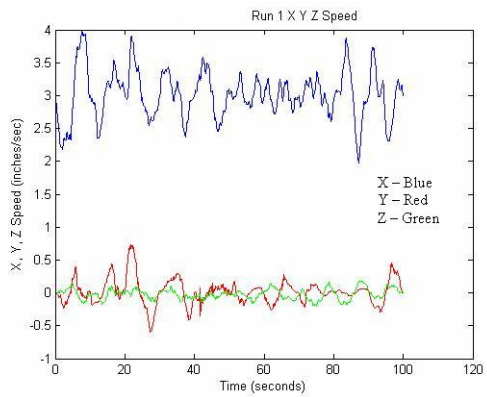


Figure 16 Computed Speed of First Run of a Smooth Conveyor

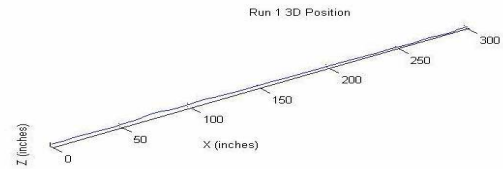


Figure 17 Computed 3D Position of First Run of a Smooth Conveyor

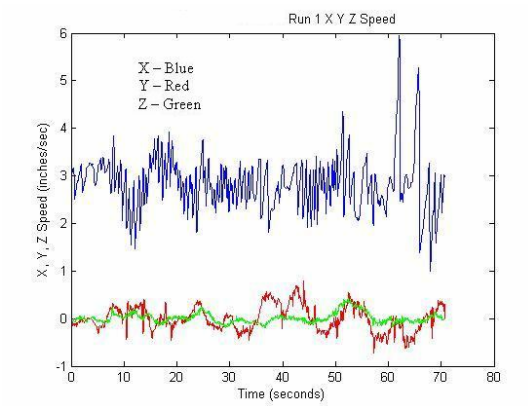


Figure 18 Computed Speed of First Run of a Jerky Conveyor

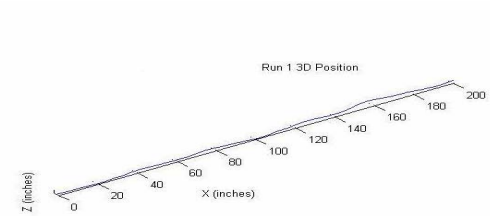


Figure 19 Computed 3D Position of First Run of a Jerky Conveyor

Basic statistical analysis of three linear speeds, the mean, the standard deviation, maximum and minimum value, has been performed and the results are summarized in Table 4.

Table 4 Three Linear Speed Statistics

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Linear Speed	Mean (X) in/sec	Mean (Y) in/sec	Mean (Z) in/sec	StdDev(X) in/sec	StdDev(Y) in/sec	StdDev(Z) in/sec	Max(X) in/sec	Max(Y) in/sec	Max(Z) in/sec	Min(X) in/sec	Min(Y) in/sec	Min(Z) in/sec
2	Conveyor 1 Sample 1	3.01	0.02	-0.02	0.36	0.19	0.08	3.97	0.74	0.18	1.98	-0.60	-0.20
3	Conveyor 1 Sample 2	2.92	0.04	-0.03	0.35	0.16	0.07	3.84	0.44	0.09	2.01	-0.43	-0.30
4	Conveyor 1 Sample 3	2.92	0.04	-0.03	0.35	0.16	0.07	3.84	0.44	0.09	2.01	-0.43	-0.30
5	Conveyor 2 Sample 1	2.78	0.01	0.03	0.37	0.12	0.10	4.44	0.39	0.24	1.78	-0.33	-0.28
6	Conveyor 2 Sample 2	3.00	0.01	0.02	0.49	0.09	0.07	4.62	0.24	0.19	1.14	-0.29	-0.14
7	Conveyor 2 Sample 3	3.01	0.04	0.01	0.43	0.10	0.07	4.67	0.29	0.19	1.03	-0.23	-0.19
8	Conveyor 2 Sample 4	2.94	-0.03	-0.05	0.72	0.12	0.09	5.48	0.30	0.22	0.45	-0.34	-0.36
9	Conveyor 2 Sample 5	2.98	0.00	-0.10	0.32	0.12	0.10	4.72	0.45	0.30	2.01	-0.38	-0.49
10	Conveyor 3 Sample 1	3.00	0.00	0.00	0.63	0.16	0.09	6.00	0.45	0.32	-1.40	-0.54	-0.38
11	Conveyor 3 Sample 2	3.01	-0.01	-0.01	0.71	0.19	0.29	5.94	0.67	2.11	-2.31	-1.32	-1.57
12	Conveyor 3 Sample 3	3.00	0.00	0.00	0.53	0.17	0.08	4.47	0.55	0.31	-1.81	-0.64	-0.40
13	Conveyor 3 Sample 4	3.00	0.00	0.00	0.56	0.21	0.09	4.95	0.73	0.39	0.37	-0.59	-0.29
14	Conveyor 3 Sample 5	3.00	0.00	0.00	0.66	0.16	0.09	6.24	0.91	0.71	-1.48	-0.97	-0.56
15	Conveyor 4 Sample 1	2.83	0.00	0.01	0.53	0.26	0.11	5.94	0.80	0.42	0.99	-0.72	-0.19
16	Conveyor 4 Sample 2	2.88	0.00	0.01	0.63	0.29	0.07	5.65	1.16	0.24	1.11	-0.90	-0.17
17	Conveyor 4 Sample 3	2.92	0.04	0.01	0.91	0.15	0.07	6.22	0.51	0.24	-0.44	-0.44	-0.18
18	Conveyor 4 Sample 4	3.19	-0.10	0.01	0.82	0.32	0.12	5.78	0.67	0.29	0.95	-1.08	-0.45
19	Conveyor 4 Sample 5	2.93	-0.02	-0.03	0.88	0.33	0.10	5.80	0.85	0.28	0.67	-1.05	-0.32
20	Conveyor 4 Sample 6	2.94	0.02	0.06	0.51	0.26	0.11	5.70	0.66	0.42	1.30	-0.77	-0.23
21	Conveyor 5 Sample 1	1.80	0.21	-0.10	0.79	0.32	0.13	4.64	1.76	0.18	-2.33	-2.09	-0.46
22	Conveyor 5 Sample 2	2.51	-0.03	0.06	0.56	0.31	0.08	4.79	2.66	0.26	0.57	-1.17	-0.19
23	Conveyor 5 Sample 3	1.96	0.27	-0.02	0.84	0.32	0.08	4.90	2.43	0.26	-0.83	-1.15	-0.32
24	Conveyor 5 Sample 4	1.74	-0.03	0.04	0.71	0.23	0.08	4.48	1.62	0.33	-1.34	-1.21	-0.19
25	Conveyor 5 Sample 5	2.03	-0.03	-0.01	0.76	0.24	0.09	5.57	1.06	0.23	-1.01	-1.29	-0.33
26	Conveyor 5 Sample 6	2.37	-0.04	-0.08	0.76	0.22	0.09	5.59	0.79	0.15	0.05	-1.27	-0.38
27	Conveyor 6 Sample 1	2.45	0.00	0.04	0.66	0.62	0.24	4.50	3.39	1.36	-0.26	-3.71	-1.45
28	Conveyor 6 Sample 2	2.18	-0.01	-0.05	0.65	0.29	0.14	4.22	1.40	0.71	0.00	-1.76	-1.00
29	Conveyor 6 Sample 3	2.02	-0.01	0.03	1.10	0.55	0.14	5.13	2.09	0.67	-1.34	-1.83	-0.79
30	Conveyor 6 Sample 4	2.35	0.01	0.02	0.70	0.31	0.14	4.56	1.19	0.77	0.26	-0.94	-0.54
31	Conveyor 6 Sample 5	2.33	-0.06	0.07	0.57	0.20	0.09	4.21	0.57	0.40	0.56	-0.54	-0.27

The range of speed distribution is measured by one standard deviation, as listed in column E,F,G.. So the conveyor motion's stability can be similarly evaluated by the value of standard derivation. The smaller the standard deviation value, the more stable the conveyor motion. Although Conveyor 5 moves at lower average speed, it is most unstable conveyor with the widest range of speed and highest magnitude of acceleration (change rate of speed).The next unstable conveyor is Conveyor 4. Stability of Conveyor 3 is similar to Conveyor 4. Conveyor 6 has better stability than previous two types of conveyor. Both Conveyor 1 and 2 are significantly stable as shown by their standard derivation as well as maximum and minimum values. The same conclusion of the conveyor motion stability has been determined using the acceleration standard deviation in Section 4.

7. CONCLUSION AND SUMMARY

This paper presented a method of sampling conveyor motion data, illustrated raw linear acceleration data for two representative conveyors. Standard statistical analysis has been performed for both linear acceleration and computed linear speed to evaluate the conveyor motion stability. The Fast Fourier Transformation (FFT) has been performed on linear acceleration data to identify frequency pattern of the conveyer acceleration. The FFT analysis has shown that all jerky motion along the conveyor main travel direction is of random nature, i.e. they do not occur at a fixed frequency or multiple frequencies and do not have magnitude of regular pattern.

Based on the actual conveyor motion data collected from the assembly plants, the dynamic motion profile library has been built for each of conveyor type that is commonly used in the general assembly. These motion profiles can be used in laboratory experiments for developing new line tracking solutions and validating vendor supplied black box line tracking solutions that are targeted for plant floor deployment.

8. ACKNOWLEDGMENTS

The author would like to acknowledge Bob Scheuerman, Rick F. Rourke Dave Groll Jim W. Wells Thomas McGraw, Charles W. Wampler, Neil McKay, and Roland Menassa. Their contributions of data collection, inputs, and consultations have made this project a success compfrom its initial concept to the final results.

9. REFERENCES

[1] "EDR3D 6DOF Motionmaster User's Manual", IST, Inc. 2005

[2] J. Shi, R. Rourke, D. Groll, P. Tavora,"Qunatification of Line Tracking Solutions for Automotive Applications", PreMIS'08, 2008

3D Part Identification Based on Local Shape Descriptors

Xiaolan Li[1,2] Afzal Godil[1] Asim Wagan[1]
[1] National Institute of Standards and Technology, Gaithersburg, MD 20899, U.S.A.
[2] Zhejiang Gongshang University, Hangzhou, Zhejiang 310018, China
Contact: lixlan@nist.gov

ABSTRACT
This paper explores 3D object recognition based on local shape descriptor. 3D object recognition is becoming an increasingly important task in modern applications such as computer vision, CAD/CAM, multimedia, molecular biology, robotics, and so on. Compared with general objects, CAD models contain more complicated structures and subtle local features. It is especially challenging to recognize the CAD model from the point clouds which only contain partial data of the model.
We adopt the Bag of Words framework to do the partial-to-global 3D CAD retrieval. In this paper the visual words dictionary is constructed based on the spin image local feature descriptor. The method is tested on the Purdue Engineering Benchmark. Furthermore, several experiments are performed to show how the size of query data and the dissimilarity measurement affect the retrieval results.

Categories and Subject Descriptors
I.2.10 [**Vision and Scene Understanding**]: Shape, Representations, data structures, and transforms.

General Terms
Algorithms, Performance, Reliability.

Keywords
CAD model retrieval, bag of words, spin image.

1. INTRODUCTION
Large number of 3D models are created everyday and stored in databases. In order for these 3D databases to be useful, we should be able to search on them. Therefore, identification, retrieval and classification of 3D objects are becoming an increasingly important task in modern applications such as computer vision, CAD/CAM, multimedia, molecular biology, robotics, and so on.

With recent developments in 3D range scanners it is possible to capture 3D shapes in real time. However, because of the limitation of the point of view, the occlusion in the scene, and the real time requirement, only parts of the object can be captured during scanning. This proposes a challenging research problem: given an incomplete point cloud of an object, how to retrieve the corresponding complete model from a database. Solving this problem will also benefit several other applications, such as data registration [Mitra06], model fixing [Founkhouser04], and so on.

Nevertheless, most of the 3D shape retrieval methods are based on global shape descriptors, which require the complete geometry of a 3D object, such as Light Field descriptors [Chen03], spherical harmonics descriptor [Kazhdan03], D2 shape distribution [Osada02]. That these methods are not suitable for solving the problem provides an impetus to create methods for partial-to-global 3D shape identification and matching.

Besides the benefits of partial-to-global retrieval, local descriptors can capture more local details than can the global ones. Compared with general objects, CAD models have more complicated structure with holes and other local features. Using global information, these subtle details can be neglected. From this aspect, local descriptors are better.

In this paper, we present a complete framework for performing 3D partial shape identification on 3D CAD parts. Several experiments are performed to show how the size of query data and the dissimilarity measurement affect the retrieval results.

The organization of the paper is as follows. Several related works are summarized in Section 2. Section 3 outlines the whole framework of our method, and introduces two crucial terms: bag-of-words and spin image. Then, the procedures of feature extraction and similarity computation are described in Section 4. In Section 5, we provide the 3D shape retrieval results on the Purdue Engineering Benchmark.

2. RELATED WORK
In order to perform 3D partial-to-global shape retrieval, the following methods have been proposed. [Podolak06] exploits the symmetry of the shape. [Mitra06] [Frome04] develop local shape signatures. Because of its simplicity and generality, the bag-of-words method, which is insensitive to deformation, articulation and partial missing data, has attracted lots of interest in 2D [Li05] and 3D [Shan06] [Liu06] [Ohbuchi08] fields. In [Li05], the method is applied to images by using a visual analogue of a word, formed by vector quantizing two regional descriptors: normalized 11*11 pixel gray values and SIFT descriptors. In [Shan06] and [Liu06], visual feature dictionary is constituted by clustering spin images in small regions. In order to procure partial-to-whole retrieval, Kullback-Leibler divergence is proposed as similarity measurement in [Liu06], while a probabilistic framework is introduced in [Shan06]. For the sake of collecting visual words, Ohbuchi et. al. [Ohbuchi08] apply SIFT algorithm to depth buffer images of the model captured from uniformly sampled locations on a view sphere. After vector quantization, Kullbak-Leibler divergence measures the similari-

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

ties of the models. But these methods focus on the retrieval of general objects.

Compared with general objects, CAD models have a more complicated topology with holes and other local features. In [Ip07], partial CAD retrieval is achieved based on segmentation, which directly affects the retrieval results. This paper aims to develop a new method for 3D CAD parts identification in similar circumstance as in [Ip07]. That is, given an unknown partial 3D point cloud of a part, we are trying to identify the part based on the known CAD model in a database. Moreover, our framework is closely related to that of [Liu06] and [Shan06], which does not require segmentation at all.

3. OUR FRAMEWORK

We first describe the whole framework of our method, and then introduce the concept of the spin image [Johnson99] and then give several examples.

3.1 Our framework

Our method is divided into two stages as shown in figure 1. The first stage is completed off-line, aims to construct a visual word dictionary based on a 3D database. First, local features are extracted from each model in the database. Second, a clustering or classification method is applied to the feature collection to construct the visual word dictionary. The second stage is on-line comparison. For the query data, we extract local features and search the dictionary for the nearest visual word. We then represent the query data with a feature vector, in which each element corresponds to one visual word in the dictionary, and the value denotes the frequency of the word appearing in the query data. Finally, a certain dissimilarity metric is chosen to compare the difference between the query data and the models in the database. A retrieval rank list is the output of the framework.

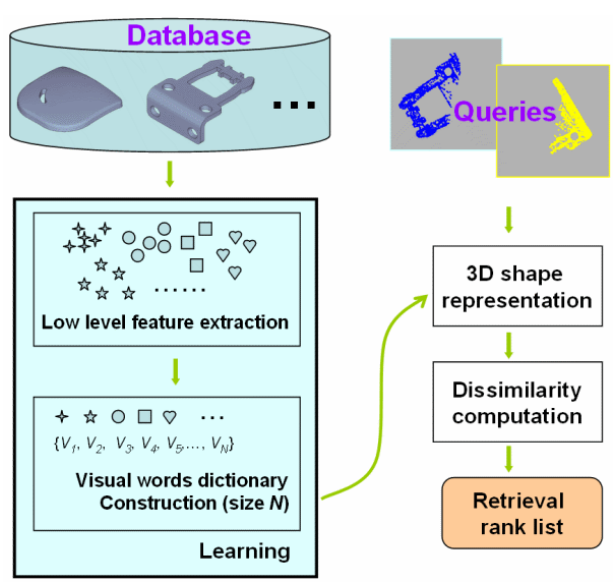


Fig. 1. Our framework

3.2 Spin image

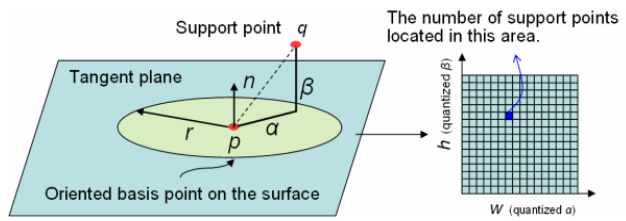


Fig. 2. Extracting low level features with spin images

As shown in Figure 2, the spin image, which is invariant to the rotation and translation transform, characterizes the local appearance properties around its basis point p within the support range r. It is a two-dimensional histogram accumulating the number of points located at the coordinate (α, β), where α and β are the lengths of the two orthogonal edges of the triangle formed by the oriented basis point p, whose orientation is defined by the normal n, and support point q. The final size of the spin images is defined by the width and the height of the spin plane. We choose it as the low level feature descriptor in this paper. Figure 3 demonstrates several spin images extracted from different positions from the bunny.

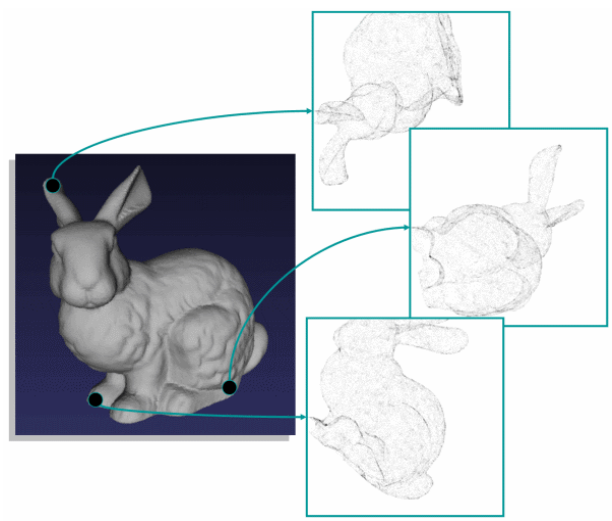


Fig. 3. Demonstration of spin images. The support range r is defined as the radius of the model. The width and height of the spin image are all 256

4. IMPLEMENTATION

In this section we will elaborate the details of the method proposed in previous section.

4.1 Low level feature extraction

Because the 3D meshes may be composed of large and tiny triangles, instead of calculating spin images based on the mesh vertices [Johnson99], a two passes sampling procedure is performed here. Using Monte-Carlo strategy [Osada02], for each 3D mesh, N_b oriented basis points p with normal n and N_s support points q are sampled uniformly on the surface in two passes

respectively, where N_b=800, N_s=50000. Other parameters of spin image are defined as: 1) r =0.4R, where R is the radius of the mesh. 2) the width and height of spin images is set as w=h=10.

Now a large number of spin images are collected from the 3D shape database. Each mesh is represented with N_b spin images.

4.2 Visual words dictionary construction

With N_b*N_m spin images, where N_b is defined previously and N_m is the number of 3D meshes we used for building the visual words dictionary, k-means algorithm is applied to agglomerate N clusters. Here N equals to 1500, which defines the size of the dictionary. Therefore, each spin image is assigned with the index of its nearest cluster. Actually, other clustering algorithms [Moosmann08] can be adopted to do the work. Further research needs to be done to analyze the effects of different clustering algorithms and the size of the dictionary.

4.3 3D shape representation

For a new shape data, no matter if it is a complete model or just a partial point cloud of an object; we represent it using the visual words in the dictionary. The representation can be derived via three steps as follows:

1. Extract the low level features using spin images.
2. Calculate the distances between the spin images and the visual words. The shortest distance indicates that we can use the corresponding visual word to record this spin image.
3. Count the number of times each visual word appears on this shape.

Therefore, each shape is represented by a vector fv=(x_1, x_2, …, x_N). This is explained visually in figure 4.

Fig.4. Shape representation

4.4 Dissimilarity computation

The requirements for dissimilarity measure for the partial-to-global retrieval task are quite different than the global-to-global retrieval problem. As described in [Liu06], suppose there are query data composed of a head and a torso, it is highly probable that a human model is a candidate shape for this query. However, the human model is not a part of this query data. That means the distance between the query data and the model does not equal to the distance between the model and the query data. The dissimilarity metric should reflect this asymmetric property.

To satisfy this requirement, an ordinary symmetric distance measurement, such as L1, L2, is not a suitable choice. KL divergence is one of the metrics which satisfies the asymmetric property. We will demonstrate the different retrieval results using L1 and KL distance metric in the next section.

5. EXPERIMENTAL RESULTS

The Purdue Engineering Benchmark (PEB) [Jaynti06], which contains 801 3D CAD models, is chosen as the 3D shape database. It is classified into 42 classes such as, "Discs", "T-shaped parts" and "Bracket-like parts".

Figure 5 shows the Precision Recall curves [Shilane04] with KL divergence measurement when using different partial sizes of the object as query data. G-G means it is the PR curve for the global-to-global retrieval, P2-G means half of the original model is used as the query data, P3-G means one third of the original model is used as the query data, and so on. It verifies the intuitive feeling that less information will lead to worse retrieval results. However, even with reduced information reasonable performance is observed, suggesting robustness of the method.

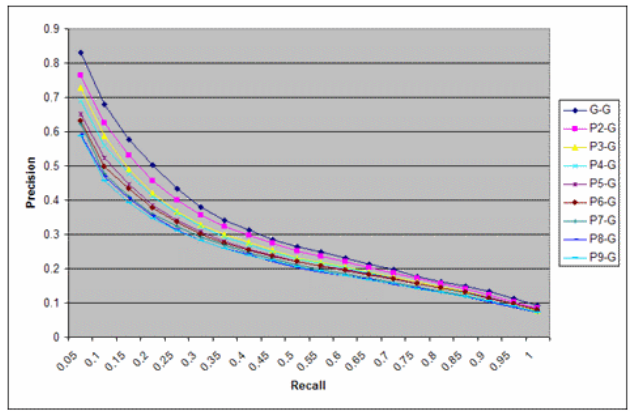


Fig. 5. The precision recall curves regarding with different size of the query data

In order to show the effects of using different distance metric, we draw two PR curves corresponding to these two metrics (see figure 6). Only one sixth of the model is used as query data.

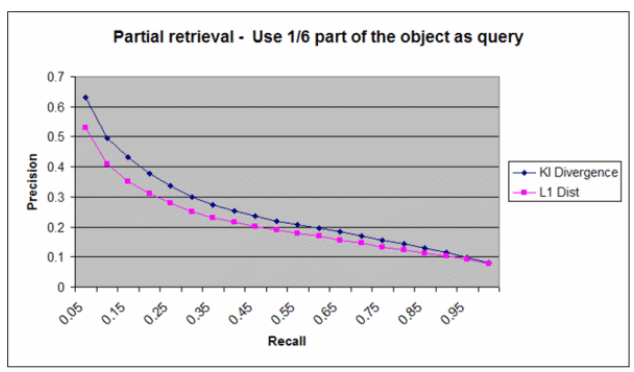
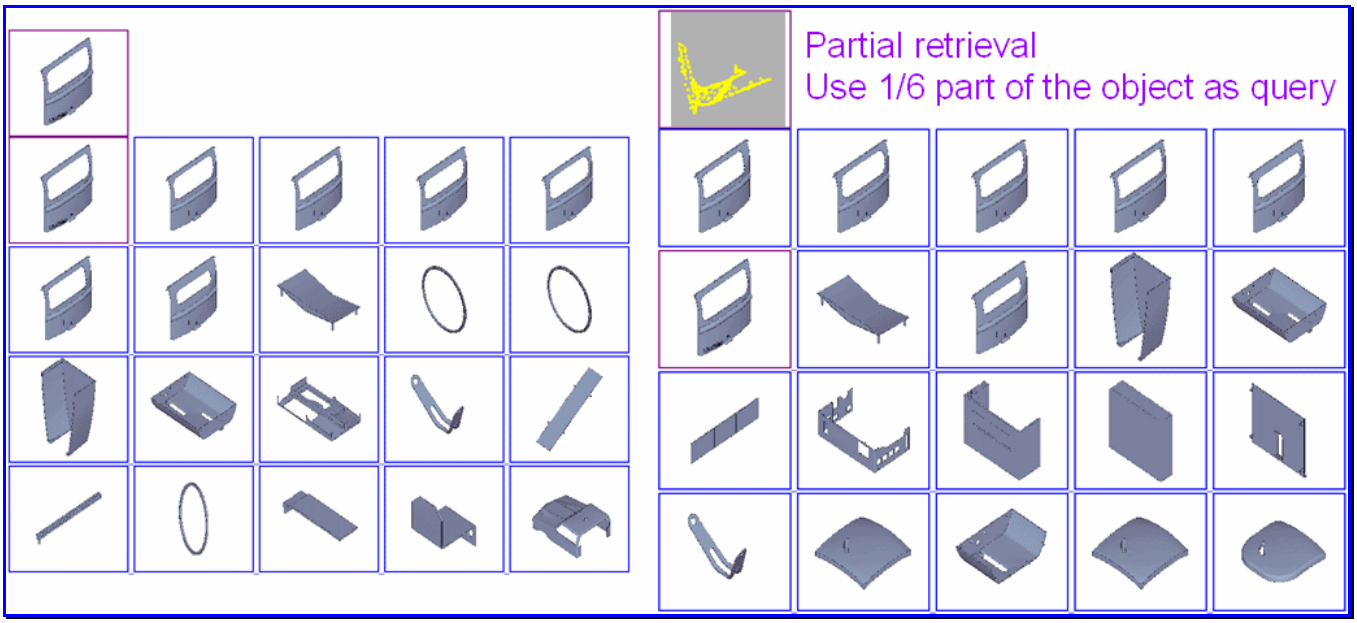


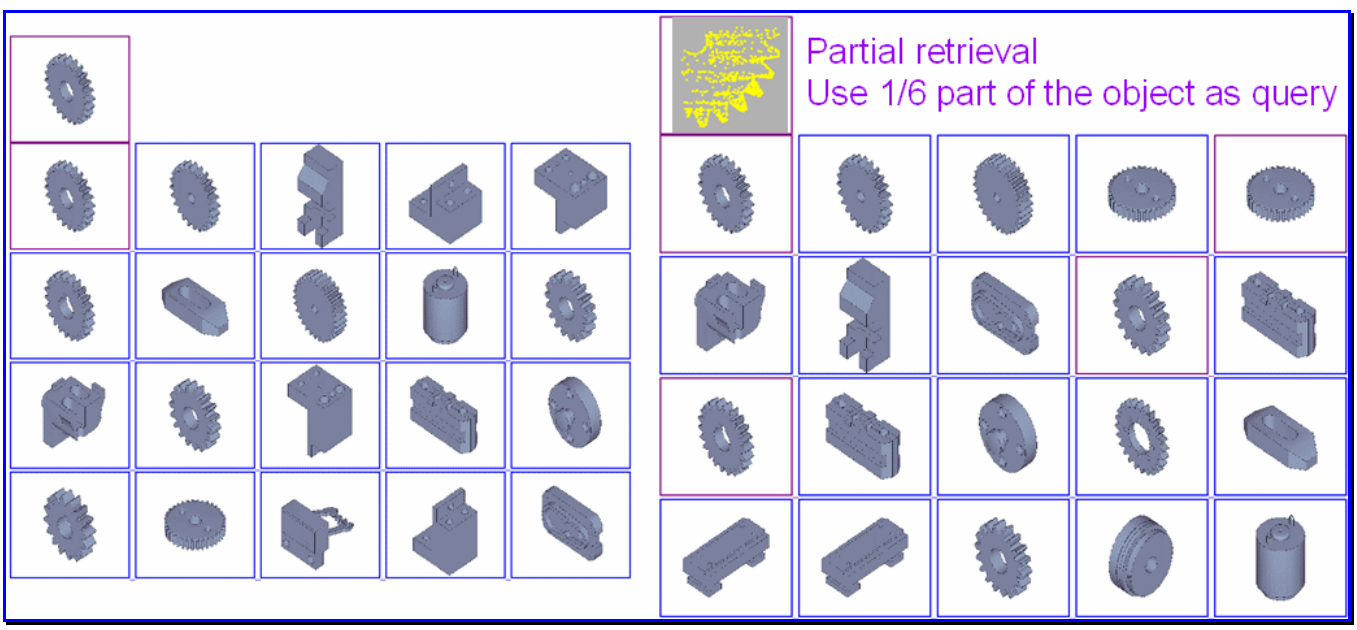
Fig. 6. The precision recall curves regarding with different dissimilarly metrics

Figure 7 provides two examples comparing the retrieval results of global-to-global retrieval and partial-to-global retrieval. The top figure shows the results when using a door as query shape. For Partial-to-Global retrieval, the left top part of the door is used as query data. In fact PEB contains only 7 door models; both G-G and P6-G retrieval rank all the 7 door models on the top of the retrieval list. The bottom figure shows results when using a gear as query shape. It shows that the P6-G re-

trieval is better than the G-G one, since P6-G find out more gears than G-G. Why does the partial-to-global retrieval perform better? It seems impossible. However, recalling the definition of the feature vector will provide some clues to the answer. The feature vector describes the frequency of the visual words appearing in the shape. When using the entire gear model to be the query data, the plane-kind of visual word overwhelm the other features. However, using partial of the object to be the query data, the gear teeth shape dominates the whole shape. So more gears are picked out, and listed on the top of the list.



(a) First example to show the difference between Global-to-Global (G-G) and Partial-to-Global (P-G) retrieval. The left figures show the Global-to-G-G retrieval result using a complete model (the first image listed in the first line) as the query. The right figures show the P-G retrieval result using 1/6 part of the complete model (the second image listed in the first line) as the query. The top 20 models are listed orderly according to the similarity metric.



(b) The second example to show the difference between G-G and P-G retrieval. The layout of the images is the same as that of (a).

Fig. 7. Two examples of retrieval results

6. CONCLUSIONS

In this paper, we propose to use the bag-of-words model for 3D CAD parts retrieval. The spin image is chosen as the local feature detector. We perform experiments to study the effectiveness of the method to solve the problem of partial-to-global 3D shape recognition. The results demonstrate the effectiveness of the method.

ACKNOWLEDGEMENT

We would like to thank the SIMA program and the IDUS program for supporting this work.

REFERENCES

[Chen03] D.Y. Chen, M. Ouyoung, X.P. Tian, Y.T. Shen. On visual similarity. based 3D model retrieval. Computer Graphics Forum (Eurographics '03), pp. 223-232.

[Founkhouser04] T. Funkhouser, Kazhdan, P. Shilane, P. Min, W. Kiefer, A. Tal, S. Rusinkiewicz, and D. Dobkin, Modeling by example, SIGGRAPH 2004, pp. 652-663.

[Frome04] A. Frome, D. Huber, R. Kolluri, T. Bulow, J. Malik, Recognizing objects in range data using regional point descriptors, ECCV 2004, pp.

[Ip07] C.Y. Ip, S.K. Gupta, Retrieving matching CAD models by using partial 3D point clouds, Computer-Aided Design & Applications, Vol.4, No. 5, 2007, pp. 629-638.

[Jaynti06] Jayanti, S. , Kalyanaraman, Y. , Iyer, N. , and Ramani, K. Developing An Engineering Shape Benchmark For CAD Models, Computer-Aided Design, Volume 38, Issue 9, Shape Similarity Detection and Search for CAD/CAE Applications, September 2006, Pages 939-953.

[Johson99] A.E. Johnson, M. Hebert. Using spin images for efficient multiple model recognition in cluttered 3D scenes. IEEE Trans. On PAMI, 21(5), pp. 433-449, 1999.

[Kazhdan03] Michael Kazhdan, Thomas Funkhouser, and Szymon Rusinkiewicz. Rotation Invariant Spherical Harmonic Representation of 3D Shape Descriptors. Symposium on Geometry Processing. June 2003.

[Li05] L. Fei-Fei, P. Perona. A Bayesian Hierarchical model for learning natural scene categories. CVPR 2005, 524-531.

[Liu06] Y. Liu, H. Zha, H. Qin, Shape topics : a compact representation and new algorithms for 3D partial shape retrieval, IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'06), 2006, pp

[Mitra06] N.J. Mitra, L. Guibas, J. Giesen, M. Pauly, Probabilistic Fingerprints for Shapes, Eurographics Symposium on Geometry Processing 2006.

[Moosmann08] F. Moosmann, E. Nowak, F. Jurie. Randomized clustering forests for image classification. PAMI 2008, Vol.30, No. 9. 1632-1646.

[Ohbuchi08] R. Ohbuchi, K. Osada, T. Furuya, T. Banno, Salient local visual features for shape-base 3D model retrieval, IEEE Inter. Conf. on Shape Modeling and Applications (SMI'08),2008, pp. 93-102.

[Osada02] Robert Osada, Thomas Funkhouser, Bernard Chazelle, and David Dobkin. Shape Distributions. ACM Transactions on Graphics, 21(4):807-832, October 2002.

[Podolak06] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, T. Funkhouser, A planar-reflective symmetry transform for 3D shapes, SIGGRAPH 2006, pp.

[Shilane04] P. Shilane, P. Min, M.Kazhdan, T. Funkhouser. The Princeton shape benchmark. *Proc. of the Shape Modeling International 2004 (SMI'04)*, 04(00): 167-178, 2004.

Calibration of a System of a Gray-Value Camera and an MDSI Range Camera

Tobias Hanning
University of Passau
Innstr. 43
D-94034 Passau, Germany
tobias.hanning@uni-passau.de

Aless Lasaruk
University of Passau
Innstr. 43
D-94034 Passau, Germany
aless.lasaruk@uni-passau.de

ABSTRACT

We present a novel calibration approach for a system of a range camera based on the Multiple Double Short-time Integration (MDSI) principle and a regular gray-value camera. We use a white plate with a regular grid of black circles as a calibration target. We expose the target to the camera system in different distances and angles. Our procedure returns the intrinsic camera parameters of both cameras and their relation to each other. We demonstrate the applicability of our calibration procedure by discussing the calibration results for a system of a 64×8-pixel MDSI range camera developed within the European project MIDIAS and a common gray-value camera.

Keywords

Range camera, gray-value camera, calibration

1. INTRODUCTION

Camera calibration is a procedure of obtaining camera model parameters by observing objects of known nature. We call a calibration procedure for a system of cameras a *cross-calibration procedure*, if the calibration parameters of each of the cameras in the system are obtained iteratively from the calibration parameters of the others and from the mutual relation between the cameras in the system. In this paper we present a cross-calibration procedure for a system of a range camera based on the Multiple Double Short-Time Integration (MDSI) approach and a regular gray-value camera.

A *range camera* is a grid of range pixels. Each *range pixel* provides the distance between the observed object and the pixel surface measured along a viewing ray. A feasible way to realize a range camera based on the time-of-flight principle has been presented by Mengel et al. [1, 2, 3]. Their MDSI approach is even suitable for outdoor measurements and thus for automotive applications without making the laser illumination dangerous for the eye. According to the MDSI approach each range pixel integrates the radiation emitted by a near infrared laser and reflected by the observed objects during two time periods of different length. The two time periods are given by repeated application of the following procedure: Two intensity acquisitions, a short and a long one, are made, starting consecutively at time t_i^l and t_i^s respectively. For each of these intensity measurements a separate laser pulse is emitted. Let $\varphi(a,t)$ be the radiation intensity measured in the point a of the active sensor area A at time t and let $\psi(a)$ denote the sensor sensitivity in a. Then the (voltage) signals s_i and l_i corresponding to the two integration measurements are given by the following expressions:

$$\begin{aligned} s_i &= \int_{t_i^s}^{t_i^s+\Delta_s} \int_A \varphi(a,t)\psi(a)\mathrm{d}a\mathrm{d}t, \\ l_i &= \int_{t_i^l}^{t_i^l+\Delta_l} \int_A \varphi(a,t)\psi(a)\mathrm{d}a\mathrm{d}t. \end{aligned}$$

Both acquisition periods start before the backscattered light from the associated laser pulse reaches the sensor surface. It is assumed that the inequality $t_i^s + \Delta_s < t_i^l$ holds, although the difference is very small. The short shutter period Δ_s is chosen in such a way that s_i contains a fraction of the laser radiation reflected from the scene. This fraction depends on the round-trip run-time of the laser pulse. The second shutter period Δ_l is significantly longer than Δ_s. Hence, within the period Δ_l all laser pulse radiation reflected from the solid angle of the scene is collected.

According to the MDSI principle single measurements in (1) are repeated several times. This yields two sequences $s_1, \ldots, s_n$ and $l_1, \ldots, l_n$ of intensity values corresponding to the short and long integration periods respectively. By integrating these intensities we obtain the values

$$s = \sum_{i=1}^{n} s_i \quad \text{and} \quad l = \sum_{i=1}^{n} l_i. \tag{1}$$

We call the number n the *number of analog integrations*. According to [1] the distance to the observed object is linear in the quotient s/l. In other words, for each range pixel and for each integration index n there exist coefficients $\alpha', \beta' \in \mathbb{R}$, such that for each pair of respective values s and l obtained in Equation (1) for the distance to the corresponding observed scene $d(s,l)$ is given by

$$d(s,l) = \alpha' s/l + \beta'. \tag{2}$$

The number of analog integrations can be adjusted for each range pixel distinctly at runtime. Hence, in each measurement cycle an MDSI range camera provides for each

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS' 08 August 19-21, 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

pixel the values s and l and the corresponding number of analog integrations n. For simplicity we assume that n is constant for each measurement and equal for each range pixel. Our approach can be generalized straightforwardly to work with adaptive number of analog integrations. Since the range pixels are located in a rectangular grid on the sensor surface, the matrix of the long and the short intensity values respectively can be interpreted as a gray-value image of the observed surface. We call these images the *long* and the *short intensity image* respectively.

A calibration algorithm for an MDSI range camera is a procedure, which obtains all necessary parameters to reconstruct the surface points incident to the viewing ray of each range pixel. More precisely, a calibration procedure for our MDSI range camera model according to [1] is an algorithm, which takes a series of intensity values $s, l \in \mathbb{R}$ obtained by observing objects of known nature for each range pixel as input. The algorithm returns for each pixel and for each integration index the *surface reconstruction function* $w : \mathbb{R}^2 \to \mathbb{R}^3$. This function takes the intensities s and l as input and returns the coordinates of the observed surface point incident to the viewing ray with respect to some suitable reference coordinate system. Hence, a calibrated MDSI range camera can be utilized as a coordinate measuring machine. Notice that depending on the choice of the reference coordinate system we obtain different surface reconstruction functions. The two obvious reference coordinate systems are: the gray-value and the range camera coordinate system. In this paper we give the surface reconstruction functions w_g and w_r for these two coordinate systems.

In [4] we have shown that the Z-coordinates of reconstructed surface points can be obtained flexibly by observing a planar calibration pattern. In this approach the Z-coordinates are obtained in the coordinate system of the gray-value camera, which observes the same scene, as the range camera. The Z-coordinates are linear in the intensity quotient as well: For each range pixel and for each integration index n there exist coefficients $\alpha, \beta \in \mathbb{R}$, such that for each pair of the values s and l the Z-coordinate of the observed surface point incident to the viewing ray is given by

$$z(s, l) = \alpha s/l + \beta. \tag{3}$$

The price to pay for the simplicity and flexibility of the procedure in [4] was the requirement to keep the spatial relation between the gray-value camera and the range camera fixed after the calibration parameters have been computed. In this article we use and significantly extend the techniques presented in [4]. We present a simple and flexible range camera calibration method, which utilizes a calibrated gray-value camera. We apply the cross-calibration approach by using data-fusion techniques to calibrate the whole system first. From the calibrated system a gray-value camera independent range camera calibration can be easily obtained. This independency from the distance reference device of our calibration procedure is the major original contribution of the present paper.

The outline of the paper is as follows: In Section 2 we introduce our camera models. Section 3 describes the hardware setup and the data acquisition procedure from the viewpoint of the user. In Section 4 we present the cross-calibration procedure for a general camera system combined of an MDSI range camera and a regular gray-value camera. In Section 5 we apply the procedure to the range camera system combined of a 64×8-pixel range camera developed within the European project UseRCams and a common gray-value camera. Section 6 summarizes our work.

2. CAMERA MODELS

We assume the gray-value camera to satisfy the pinhole camera model with radial distortions. Therefore, the intrinsic parameters of the gray-value camera are represented by the distortion function $\sigma_g : \mathbb{R}^2 \to \mathbb{R}^2$ and the mapping $I_g : \mathbb{R}^2 \to \mathbb{R}^2$ with

$$I_g(x, y) = \begin{pmatrix} \alpha_g & 0 \\ 0 & \beta_g \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} x_g \\ y_g \end{pmatrix}. \tag{4}$$

Each space point $p \in \mathbb{R}^3$ with respect to the coordinate system of the gray-value camera is related to its image $p' \in \mathbb{R}^2$ by

$$p' = (I_g \circ \sigma_g \circ \pi)(p), \tag{5}$$

where the projection $\pi : \mathbb{R}^3 \to \mathbb{R}^2$ is defined by $\pi(x, y, z) = (x/z, y/z)$. The mapping I_g describes the transformation from the projection to the image coordinate system. The parameters α_g and β_g are equal to the focal length of the corresponding pinhole camera divided by the pixel width and height respectively. The parameters x_g and y_g define the projection of the optical center on the image plane of the gray-value camera in pixel coordinates. We use the distortion function

$$\sigma_g(x, y) = \begin{pmatrix} x + x \sum_{i=1}^{d} k_i (x^2 + y^2)^i \\ y + y \sum_{i=1}^{d} k_i (x^2 + y^2)^i \end{pmatrix}, \tag{6}$$

where $k_1, \ldots, k_d$ are the distortion parameters and $d = 2$ or $d = 4$. We call the parameters α_g, β_g, x_g, y_g, and $k_1, \ldots, k_d$ the *intrinsic parameters of the gray-value camera*. Notice that for the most gray-value cameras $\alpha_g = \beta_g$ is a good approximation, since the pixels on common gray-value chips are nearly squared. We call the mapping

$$K_g = I_g \circ \sigma_g \circ \pi : \mathbb{R}^3 \setminus \{(x, y, z) \in \mathbb{R}^3 \mid z = 0\} \to \mathbb{R}^2 \tag{7}$$

the *camera mapping of the gray-value camera*.

The range camera can be modeled as a finite pinhole camera. A finite pinhole camera is defined by the mapping $K_r(p) = \pi(P_r R(p - t))$ for a rotation matrix $R \in \mathbb{R}^{3\times3}$, an upper triangular matrix $P_r \in \mathbb{R}^{3\times3}$ and a translation vector $t \in \mathbb{R}^3$ (see [5] for details). The vector t together with R forms the isometric transformation

$$T(p) = R(p - t) \tag{8}$$

from the range camera coordinate system to the coordinate system of the gray-value camera. Thus, for the range camera there exists a regular matrix $A = P_r R \in \mathbb{R}^{3\times3}$ and a vector $b \in \mathbb{R}^3$, such that for each space point $p \in \mathbb{R}^3$ and its corresponding range pixel coordinate $p' \in \mathbb{R}^2$ there exists a $\lambda \in \mathbb{R}$ with

$$\lambda \begin{pmatrix} p' \\ 1 \end{pmatrix} = Ap + b. \tag{9}$$

Our calibration procedure returns the values A and b, from which P_r, R, and t can easily be obtained: Since A is regular, we can determine P_r from A from an RQ-decomposition

$A = P_r R$ into an upper triangular matrix P_r with positive eigenvalues and an orthogonal matrix R. Such a decomposition can be obtained by a slightly modified QR-decomposition algorithm, which guarantees the positivity of the diagonal entries of P_r (see e. g. [6]). Observing that $b = -P_r R t = -At$, we obtain the translation $t = -A^{-1}b$. We denote the matrix A^{-1} by M.

For a range pixel p' and for our fixed number of number of analog integrations consider the range function $z : \mathbb{R}^2 \to \mathbb{R}$ which, given the short and long intensities s and l, returns the Z-coordinate of the observed object point incident to the pixels viewing ray. According to Equation (3) there exist $\alpha, \beta \in \mathbb{R}$ with

$$z(s,l) = \alpha \frac{s}{l} + \beta. \tag{10}$$

Then from Equation (9) we obtain the following relation between the space point p and its image point p':

$$p = \lambda A^{-1} \begin{pmatrix} p' \\ 1 \end{pmatrix} - A^{-1} b = \lambda M \begin{pmatrix} p' \\ 1 \end{pmatrix} + t. \tag{11}$$

Let us have a closer look at the third component of p, which is equal $z(s,l)$. Denoting the third row of M by $r^\top$ we obtain the following equation from which λ can be easily obtained:

$$z(s,l) = \lambda r^\top \begin{pmatrix} p' \\ 1 \end{pmatrix} - r^\top b. \tag{12}$$

Consequently, the surface reconstruction function w_g with respect to the coordinate system of the gray-value camera for the pixel coordinates p' and the chosen fixed number of analog integrations is given by

$$w_g(s,l) = \frac{z(s,l) + r^\top b}{r^\top \binom{p'}{1}} A^{-1} \begin{pmatrix} p' \\ 1 \end{pmatrix} - A^{-1} b. \tag{13}$$

Using our abbreviations $M = A^{-1}$ and $t = -A^{-1}b$ we obtain

$$w_g(s,l) = \frac{\alpha \frac{s}{l} + \beta + r^\top b}{r^\top \binom{p'}{1}} M \begin{pmatrix} p' \\ 1 \end{pmatrix} + t. \tag{14}$$

Our range camera model has two types of parameters: The parameters A and b do not depend on a particular range pixel. On the other hand, the parameters α and β in (14) are pixel dependent. These parameters must be estimated for each range pixel separately. Notice that Equation (14) can also be represented as $w_g(s,l) = u \frac{s}{l} + v$ for $u, v \in \mathbb{R}^3$. The values u and v then define the viewing ray associated with the particular range pixel.

The remainder of this section deals with the problem of obtaining the range camera parameters, which do not depend on the gray-value camera. Let for a certain range pixel the distance reconstruction function d satisfy Equation (2). From Equation (14) we obtain

$$d(s,l) = \alpha' \frac{s}{l} + \beta' = ||w_g(s,l) - t||, \tag{15}$$

which yields

$$d(s,l) = \frac{\alpha \frac{s}{l} + \beta + r^\top b}{r^\top \binom{p'}{1}} ||M \begin{pmatrix} p' \\ 1 \end{pmatrix}||. \tag{16}$$

Notice that $M\binom{p'}{1}$ is independent of s and l. Hence, we have

$$\alpha' = \frac{||M\binom{p'}{1}||\alpha}{r^\top \binom{p'}{1}} \quad \text{and} \quad \beta' = \frac{||M\binom{p'}{1}||(\beta + r^\top b)}{r^\top \binom{p'}{1}}. \tag{17}$$

Finally, we obtain the reconstruction function $w_r : \mathbb{R}^2 \to \mathbb{R}^3$ with respect to the coordinate system of the range camera by

$$w_r(s,l) = d(s,l) \frac{P_r^{-1}\binom{p'}{1}}{||P_r^{-1}\binom{p'}{1}||}. \tag{18}$$

3. HARDWARE SETUP AND DATA ACQUISITION

The hardware setup consists of an off-the-shelf gray-value camera and an MDSI range camera in rigid spatial relation. For a successful calibration the field of view of the range camera must be contained in the field of view of the gray-value camera.

The calibration procedure consists of the following steps:

(i) Acquire calibration data by continuously exposing a calibration pattern to the camera system.

(ii) Determine the intrinsic parameters of the gray-value camera.

(iii) Determine the range camera parameters A and b

(iv) Determine for each range camera pixel (and optionally for each integration index n) the Z-coordinate reconstruction parameters α and β in Equation (3).

(v) Optimize all parameters by nonlinear optimization using a suitable error function.

(vi) Optionally compute the range reconstruction function for each range pixel using Equation (17) and (18).

It is possible to use one calibration target to perform all the above steps. A suitable calibration target is for example a white planar pattern with a regular grid of black circles painted on it (see Figure 1). The size of the circles should be chosen in a way that correspondent circles can be recognized as well in the gray-value images as in at least one of the intensity images provided by the range camera. If the resolution of the range camera is to low to recognize the circles needed for precise plane reconstruction in step (iv) one can choose a different calibration target (for example with bigger circles) to determine the range camera parameters A and b in the step (iii). This is the only critical step, where the target features must be reconstructed within the intensity images of the range camera.

4. CROSS-CALIBRATION PROCEDURE

We now turn to the details of our cross-calibration procedure. The estimation of the intrinsic parameters of a gray-value camera is a well studied problem (see e. g. [7, 8, 9]) Since our calibration pattern is suitable for a calibration of the intrinsic parameters of the gray-value camera, we assume the gray-value camera to be roughly calibrated. In fact, we use the algorithm presented in [9] to perform this calibration. During our cross-calibration procedure the parameters of the gray-value camera will be further improved.

We now turn to the step (iii) of our calibration procedure. In step (i) we have obtained a sequence of synchronously acquired gray-value and intensity images. Let j be the image pair index within this sequence. Suppose, we have obtained the centers of the black circles $p_{1j}, \ldots, p_{mj} \in \mathbb{R}^2$ in the gray-value image and the corresponding centers of the same circles

$p'_{1j}, \ldots, p'_{mj} \in \mathbb{R}^2$ in the short or long intensity image of the range camera.

Given a calibrated gray-value camera K_g and the centers of the circles $c_1, \ldots, c_m \in \mathbb{R}^3$ with respect to the calibration pattern, we obtain the positions of the centers observed by the gray-value camera by determining

$$(R_j, t_j) = \underset{(R,t)\in \mathrm{SO}_3 \times \mathbb{R}^3}{\operatorname{argmin}} \sum_{i=1}^{m} \|K_g(Rc_i + t) - p_{ij}\|^2 \tag{19}$$

(see e. g. [10]). We set $c_{ij} := R_j c_i + t_j$. Then c_{ij} is the position of the center of the i-th circle observed in the j-th image pair. With this we obtain for each circle the relation

$$\lambda \begin{pmatrix} p'_{ij} \\ 1 \end{pmatrix} = Ac_{ij} + b, \tag{20}$$

for $\lambda \in \mathbb{R}$. Equation (20) is linear in A and b and, hence, admits a DLT algorithm for the estimation of these camera parameters. The equations for all i and for different positions of the calibration pattern form an overdetermined system of linear equations. We obtain a least-squares solution for A and b by Housholder-Transformations [11]. For stability we additionally apply the RANSAC algorithm to remove outliers. Notice that, in contrast to the calibration procedure in [9] we do not need to impose additional constraints on A and b, like for example orthogonality constraints.

The estimate of A and b obtained by the DLT-algorithm turns to be very rough for back-projection purposes. More precisely, let

$$w_{ij} = \frac{(c_{ij})_z + r^\top b}{r^\top \binom{p'_{ij}}{1}} M \begin{pmatrix} p'_{ij} \\ 1 \end{pmatrix} + t \tag{21}$$

be obtained by replacing the z-coordinate $z(s,l)$ in of the reconstructed space point from Equation (13) by the Z-coordinate $(c_{ij})_z$ of c_{ij}. Then the Euclidean distance $||K_g(w_{ij}) - p_{ij}||$ between the projections of the reconstructed surface points w_{ij} and their corresponding original image points p_{ij} is high with respect to the gray-value image size. To determine the parameters A and b precisely we refine them by minimizing the back-projection error

$$e(A,b) = \sum_{i=1}^{j} \sum_{i=1}^{m} (K_g(w_{ij}) - p_{ij})^2. \tag{22}$$

We solve the problem induced by the error function in Equation (22) by non-linear optimization using for the BFGS algorithm [12]. Notice that we do not need any range information of the range camera in this step.

After estimates of A and b have been obtained, the range parameters α and β from Equation (3) can be estimated for each range pixel. We extend the procedure already given in [4]. Suppose, for the range pixel p' we have the intensity values s and l while observing the plane with the equation $n^\top p = d$, where $n \in \mathbb{R}^3$ with $||n|| = 1$. Then, according to Equation (14), we have

$$n^\top w(s,l) = d. \tag{23}$$

Substitution of $w(s,l)$ from Equation (14) yields

$$\alpha \frac{s}{l} + \beta = r^\top \begin{pmatrix} p' \\ 1 \end{pmatrix} \frac{d - n^\top t}{n^\top M \binom{p'}{1}} - r^\top b. \tag{24}$$

Equations (24) (for a fixed range pixel) for each pair of synchronously obtained images of the calibration plate form a system of linear equations in the parameters α and β. We obtain a least-squares solution for these parameters by Housholder-Transformations [11]. Again, the RANSAC algorithm is additionally applied for stability (see Figure 4). Notice the significant extention of the procedure described in [4], where only plate positions orthogonal to the optical axis of the gray-value camera were concerned. Indeed, Equation (24) simplifies for $n = (0,0,1)^\top$ to

$$z(s,l) = \alpha \frac{s}{l} + \beta = d. \tag{25}$$

The final optimization step involves all collected data to refine the parameters A and b and the parameters of the gray-value camera. There are several error functions, which can be considered in this step. According to our application, we need besides optimal surface reconstruction properties also optimal back-projection properties into the image of the gray-value camera. Therefore a suitable error function is

$$e(A,b,K_g) = \sum_{i=1}^{j} \sum_{i=1}^{m} (K_g(w_g(s_{ij}, l_{ij})) - p_{ij})^2. \tag{26}$$

Here K_g is a short notation for all gray-value camera parameters. In particular K_g contains the distortion parameters of σ_g. We solve the optimization problem induced by the error function in Equation (26) by the BFGS algorithm [12]. The error function in Equation (26) involves also the optimization of the gray-value camera parameters K_g, which are improved by this final optimization step. This completes the cross-calibration procedure.

5. EXPERIMENTAL RESULTS

We have applied the calibration procedure described in Section 3 and 4 to a system of a gray-value camera and a 64× 8-pixel MDSI range camera developed within the European project UseRCams and a common gray-value camera.

We use a white plate with a regular grid of black circles painted on it as a calibration target (Figure 1). We expose the board continuously to the camera system from different directions and in different distances. For simplicity we demonstrate the calibration procedure for a fixed equal number of analog integrations for each range pixel.

The parameters A and b obtained in step (iii) are not precise, if used for back-projection of reconstructed surface points into the gray-value image. Figure 2 shows all pixels detected within the gray-value image and the corresponding back-projected pixels of the range camera after the final calibration step. Figure 3 shows the corresponding distribution of the back-projection error given in Equation (22).

After the final calibration a different scene of the same calibration plate in the distance of approximately five meters was evaluated to cross-validate our calibration result. For the cross-validation we have used the data acquisition procedure described in step (i) of our calibration procedure in Section 3. Figure 4 shows the obtained pixel characteristic for one range pixel. For each calibrated range pixel and for each plane with equation $n^\top p = d$, $||n|| = 1$, reconstructed by the gray-value camera we measure the absolute range reconstruction error with respect to our calibration plate

$$n^\top w(s,l) - d. \tag{27}$$

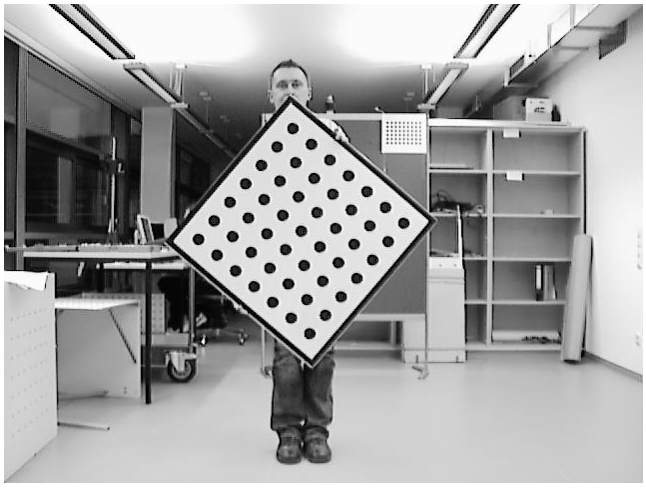


Figure 1: Calibration plate

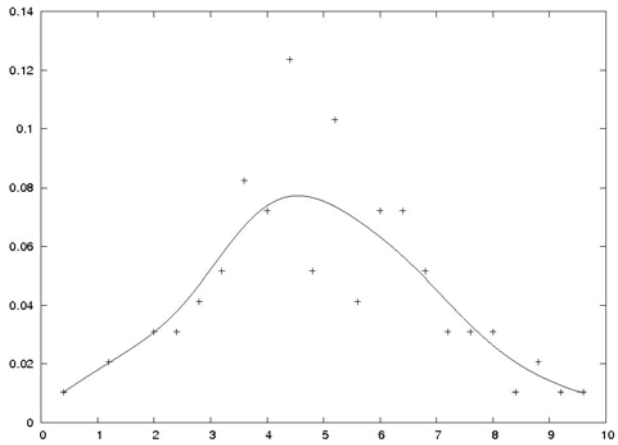


Figure 3: Distribution of the absolute back-projection error

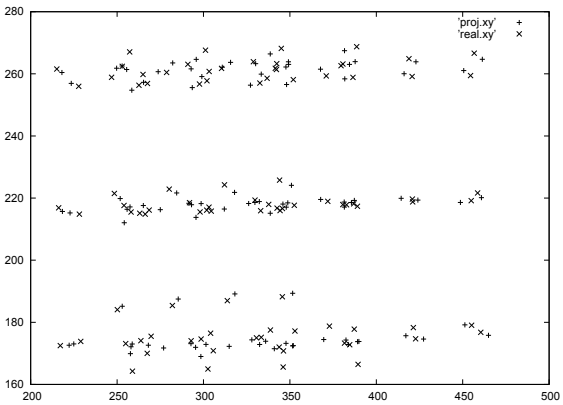


Figure 2: Detected circles and the corresponding back-projected centers of range pixels in the image space of the gray-value camera

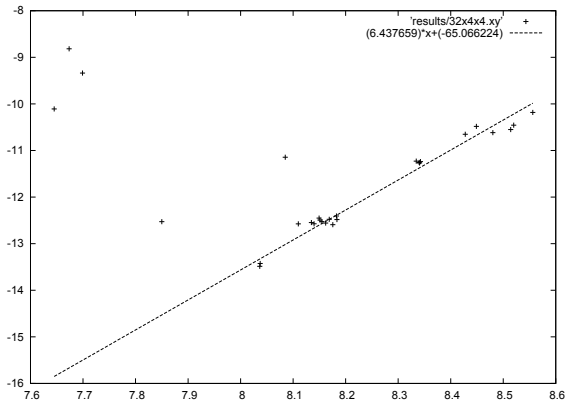


Figure 4: Example pixel characteristic obtained from noisy measurements with outliers using RANSAC

This error is the Euclidean distance between the calibration plate and the reconstructed surface point. To obtain com-

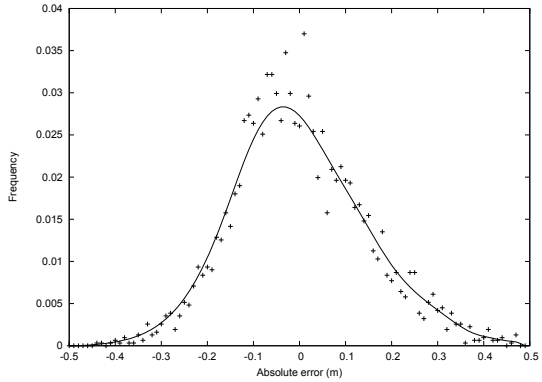


Figure 5: Distribution of the range reconstruction error

prehensive results for our experiment we have assumed the range reconstruction error of all range pixels to be identically distributed. Under this assumption we can aggregate sample data from all pixels together as being taken from a single population. Figure 5 shows the distribution of the absolute range reconstruction error for all calibrated pixels.

From the distribution in Figure 5 under the above assumptions we can conclude that errors with an absolute value of at most 16.5cm occur in 75% of all cases and errors with an absolute value of at most 29.5cm occur in 95% of all cases. The error expectation of 0.000225217m confirms the accuracy of our calibration procedure and the correctness of our model. In other words, the measured errors originate mostly from the sensor noise.

6. CONCLUSIONS

We have introduced a flexible range camera model, which is capable to provide range and surface data directly in the reference coordinate system of the gray-value camera. We have presented a cross-calibration procedure for a camera system combined of an MDSI range camera and a regular gray-value camera. Our procedure significantly extends calibration methods for MDSI cameras presented in the literature. Additionally, our calibration procedure provides also regular, gray-value camera independent, parameters for the range camera with respect to the coordinate system defined by the optical system of the range camera. We have applied the procedure to the range camera system combined of a 64×8-pixel range camera developed within the European project UseRCams and a common gray-value camera. We are able to show that our calibration procedure yields an unbiased distance- and surface reconstruction procedure. Furthermore, the calibration procedure can be applied mostly autonomously without expert interaction.

7. REFERENCES

[1] Mengel, P., Doemens, G., Listl, L.: Fast range imaging by CMOS sensor array through multiple double short time integration (MDSI). In: International Conference of Image Processing (ICIP). Volume I., Thessaloniki, Greece, IEEE (2001) 169–172

[2] Mengel, P., Listl, L., Koenig, B., Toepfer, C., Pellkofer, M., Brockherde, W., Hosticka, B., Elkahili, O., Schrey, O., Ulfig, W.: Three-dimensional cmos image sensor for pedestrian protection and collision mitigation. In: Advanced Microsystems for Automotive Applications 2006, Berlin, Springer (2006) 23 – 39

[3] Mengel, P., Listl, L., Koenig, B., Pellkofer, M., Wagner, U., Wertheimer, R.: Time-of-flight camera for pedestrian protection and collision mitigation. In: 6th European Congress and Exhibition on Intelligent Transport Systems and Services. (2007) accepted.

[4] Hanning, T., Lasaruk, A., Wertheimer, R.: MDSI range camera calibration. Advanced Microsystems for Automotive Applications, Springer (2008)

[5] Hartley, R., Zisserman, A.: Epipolar Geometry and the Fundamental Matrix. In: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)

[6] Golub, G.H., van Loan, C.F.: Matrix Computations. 3 edn. John Hopkins University Press, Baltimore (1996)

[7] Tsai, R.: A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. IEEE Journal of Robotics and Automation **3** (1987) 323 – 344

[8] Weng, J., Cohen, P., Herniou, M.: Camera calibration with distortion models and accuracy evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence **14** (1992) 965 – 980

[9] Zhang, Z.: A Flexible new technique for camera calibration. Technical report, Microsoft Research (1998) Technical Report MSR-TR-98-71.

[10] Hanning, T., Schöne, R., Graf, S.: A closed form solution for monocular re-projective 3d pose estimation of regular planar patterns. In: International Conference of Image Processing (ICIP), Atlanta, Georgia (2006) 2197 – 2200

[11] Lawson, C.L., Hanson, R.J.: Solving Least Squares Problems. Prentice-Hall, Englewood Cliffs, NJ (1974)

[12] Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T.: Numerical Recipes in C. 2 edn. Cambridge University Press (1992)

Dynamic 6DOF Metrology for Evaluating a Visual Servoing System

Tommy Chang, Tsai Hong, Michael Shneier
Intelligent Systems Division, NIST
Gaithersburg, MD 20899-8230
{hongt, tchang, shneier}@cme.nist.gov

German Holguin, Johnny Park
Robot Vision Laboratory
Purdue University
West Lafayette, IN 47907
{gholguin,jpark}@purdue.edu

Roger D. Eastman
Dept. of Computer Science
Loyola College in Maryland
Baltimore, Maryland 21210
reastman@loyola.edu

ABSTRACT
In this paper we demonstrate the use of a dynamic, six-degree-of-freedom (6DOF) laser tracker to empirically evaluate the performance of a real-time visual servoing implementation, with the objective of establishing a general method for evaluating real-time 6DOF dimensional measurements. The laser tracker provides highly accurate ground truth reference measurements of position and orientation of an object under motion, and can be used as an objective standard for calibration and evaluation of visual servoing and robot control algorithms. The real-time visual servoing implementation used in this study was developed at the Purdue Robot Vision Lab with a subsumptive, hierarchical, and distributed vision-based architecture. Data were taken simultaneously from the laser tracker and visual servoing implementation, enabling comparison of the data streams.

Keywords
computer vision, laser tracker, dynamic 6DOF metrology, performance evaluation.

1. INTRODUCTION
Real-time three-dimensional vision has been rapidly advancing over the past twenty years, leading to a number of successful laboratory demonstrations, including real-time visual servoing [7,16,17]. However, the advances have frequently not yet made the transition to commercial products, due in part to a lack of objective methods for empirical performance evaluation. To ensure a new algorithm for optical flow, stereo, visual servoing, laser Simultaneous Localization And Mapping SLAM, or other dynamic visual 3D task is valid, it would be very helpful to have a reference standard sensor system (ground truth) along with appropriate metrics for the comparison of test systems with the reference system. Standards and test procedures for dimensional metrology are well-established and highly accurate for static measurements, with coordinate measuring machines and laser trackers giving position measurements to microns. However, the theory, technology, and test procedures are not well established for dynamic dimensional measurements in uncontrolled environments.

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

In this paper we demonstrate the use of a dynamic, six-degree-of-freedom (6DOF) laser tracker to empirically evaluate the performance of a real-time visual servoing implementation, with the objective of establishing a general method for evaluating real-time 6DOF dimensional measurements of an object or assembly component under moderately constrained motion. The proposed evaluation procedure collects data simultaneously from the laser tracker and the visual servoing system under test, so the two data streams can be compared. Laser trackers produce highly accurate position and orientation data at a high data rate and for the purposes of this study will be considered ground truth.

The questions addressed in this work primarily focus on how to collect and compare the data streams. Issues here include synchronizing the data streams, so individual data points are taken at the same time; external calibration of the two sensors, so individual data points can be compared in the same coordinate system; and comparison metrics, so individual data points can be compared between the two sensors to determine how close the system under test comes to the ground truth. It would be useful to have comparison metrics that are robust to errors in synchronization and calibration, or even not dependent on calibration, to ensure accurate comparisons in the field.

The real-time visual servoing implementation used in this study was developed at the Purdue Robot Vision Lab using a subsumptive, hierarchical, and distributed vision-based architecture for smart robotics [3,6,16,17]. This is a robust, advanced dynamic visual servoing implementation with a high level of fault tolerance to non-cooperative conditions such as severe occlusions and sudden illumination changes. The Purdue system combines a ceiling mounted camera with a trinocular system mounted on the robot end-effector, and uses position based visual servoing (PBVS). The work in this paper is aimed at the evaluation of sensors for PBVS, in which the servoing system senses the position and orientation of the part in 3D coordinates, as opposed to image based visual servoing (IBVS), in which the servoing system senses the position and orientation of the part in 2D image coordinates.

2. PREVIOUS WORK
The general need for empirical evaluation has been well-recognized in the computer vision literature. Here we review articles on empirical evaluation of sensor-only 6DOF static and

dynamic pose estimation, as well as articles on evaluation of visual servoing algorithms that may combine sensor evaluation with robot control issues. In reviewing the literature, there are a number of questions that can be asked: what is under evaluation (pose estimation only, pose calculated for visual seroving, or other tasks), how measurements are taken from the system under test, how ground truth or other reference measurements are taken, what metrics are computed from the measurements, and how the metrics are interpreted to give summary judgments on system performance. We also have looked at the conditions of each test to consider the objects, motions, occlusions and environment.

For pose estimation, most evaluation papers consider static pose only [9,12] and not dynamic 6DOF sensor measurements of a part under motion. References [12,13] use Monte Carlo simulation for the evaluation of pose algorithm accuracy under noise and object orientations. In those articles, results are given for pose estimation for a complex industrial part and the error from unnamed ground truth is plotted as position or orientation error vs. the rotation of the object. The key result is to note the error as a function of part rotation varies considerably, spiking at ambiguous orientations of the object. Two papers that do consider dynamic pose are [10,15]. In [10] benchmarked video sequences are used for tests of a model-based algorithm with four parameter variations to analyze the relative contributions of subcomponents such as the edge detection operator or search technique. The results are given as deviations from the results of the one parameter set that successfully maintained track through the video sequences, but the nature and quality of this retrospective ground truth is not described in the article. In [15] three tracking approaches for 6DOF pose estimation and grasping of hand-held objects are evaluated using ground truth from an unnamed infrared marker tracking system good to 1.5 meters in position but with no rotation accuracy or measurements per second cited. The three approaches run at 8 Hz to 25 Hz. The article gives results in graphs that compare ground truth position and orientation data to robot end-effector position and tracked position, but no quantitative or summary statistics are given for the graphed data.

For visual servoing, many papers that present a new approach include an empirical evaluation, but since the paper emphasizes the development of the new approach, the evaluation section can be brief. An exception is [8] which uses sensitivity analysis and simulation to compute the contribution of image measurement errors to the calculated pose and control trajectory for PBVS and hybrid visual servoing.

The metrics used to evaluate pose estimation and visual servoing systems vary. Typical is the mean and standard deviation of a measure of error in world coordinates, including individual differences for each coordinate, a norm for position and orientation separately, or rarely a combined norm for all 6 degrees of freedom. The orientation can be compared in roll-pitch-yaw or as quaternions. In experiments without ground truth in world coordinates, or for IBVS in which pose in world coordinates is not computed, errors are computed in the image domain. [5] uses the reprojection error in the image domain. In some visual servoing evaluations, the metric is the number of cases successfully completed during the experiments.

In physical experiments in the evaluation of pose estimation or visual servoing, a mechanism must be used to generate motion, frequently a robot arm [2,4,15]. [15] uses an arm to move a camera towards a car battery through a known trajectory linear in both translation and angle, and repeats the motion 80 times to judge repeatability of the tracking algorithm.

3. PURDUE LINE TRACKING SYSTEM

Using robots to perform industrial assembly tasks is not new. In fact, robots have been used successfully in such applications over the past few decades. However, one common constraint still present in most of these applications is that the objects to be manipulated by the robot must be stationary in a known position, or moving along a well-known path with a very small amount of uncertainty. Consequently, industrial tasks such as painting, palletizing, welding, or decking have traditionally been set up in stationary locations in the assembly line where the involved parts have to come to a complete stop.

In order to eliminate inflexible, expensive stationary stations in the assembly line, robots must be able to perform their tasks on moving targets. The Purdue Robot Vision Lab addresses this goal using a subsumptive, hierarchical, and distributed vision-based architecture for smart robotics [3]. The system consists of multiple real-time control modules running in parallel, where each module is controlled by a different tracking method, with unique capabilities with respect to accuracy, computational efficiency, sensitivity to varying conditions, etc. By taking the most reliable input from all the modules, the system is able to achieve a high level of fault tolerance and robustness to non-cooperative conditions such as severe occlusions and sudden illumination changes.

In this architecture, each control module can be in a different hierarchy level. The more accurate the module is, the higher its hierarchy level. Modules from higher hierarchies can subsume functionality of modules in lower levels. Each module can run independently in a different computer over a network. Any number of modules is allowed in any given level of hierarchy, providing redundancy and fault tolerance.

Each module is composed of two main parallel processes. The first process is the visual-tracking loop that estimates the 6DOF 3D pose of the target object. This loop can be further broken down into three main parallel threads: data acquisition, data processing, and message exchange. The second process is the visual-servoing loop, which generates commands to the robot to move its end-effector to a desired position. This second loop is composed of two main threads: the message exchange and the control law calculation.

Even though every control module is able to run at its own sample rate and accuracy level in our architecture, only one module will be able to pass its generated command to the robot controller. The arbitrator listens to all the control modules and decides which module input to use to control the robot, based on module availability, estimation reliability, and module hierarchy.

Finally, the robot controller is implemented at the top of the robot controller interface. Therefore, the control modules in our system fall into the position-based look-and-move category [7]. This means that the robot controller is receiving commands directly in the Cartesian space, which makes the control plant much simpler compared to the image-based servoing category [3]. However, the position-based servoing requires camera calibration, robot calibration, and hand-eye calibration [6].

3.1 Peg and Hole Experiment

In order to demonstrate the capabilities of this architecture, we set up a variation of the traditional peg-and-hole experiment (see Figure 1). In our experiment, we attached a hollow wooden cylinder to an engine cover part, and attached a peg to the end-effector of a 6DOF robot manipulator. The engine cover is loosely attached to a linear slide, so the main motion of the engine cover is provided by the motion of the linear slide. In addition, we also attached strings to the engine cover, which a human can manipulate to generate random motions of the engine cover during the experiment. The goal of this experiment is to command the robot to insert the peg into the cylinder on the engine cover while the engine cover is in motion, resembling the automation needed to perform tasks such as glass decking or wheel decking on-the-fly [16].

Our system consists of three control modules, a system arbitrator, and a robot controller interface. The three control modules are: coarse control, model-based fine control, and stereo-based fine control.

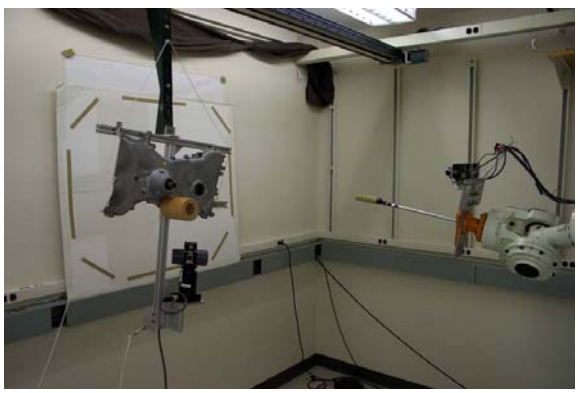


Figure 1. Peg and Hole experiment

3.2 Coarse Control

The coarse control module resides at the lowest level in the system hierarchy, meaning that it will work only when the other two modules become unavailable or unreliable. Its main purpose is to provide an initialization point for the controllers with higher hierarchy. That is, the coarse control is intended to track the target and command the robot to an approximate location in front of it. This module only requires the level of accuracy that would place the end-effector in front of the target such that cameras for the other controllers can view the target.

Coarse control visual-tracking uses a camera mounted on the ceiling with a view of the entire workspace. When the target object enters the workspace, the bar holding the target is detected and the 3D translational position of the bar in the coordinates of the robot is estimated. No rotation information is needed here. This estimated pose is passed to the visual-servoing loop that in this case would simply command the robot to move its end-effector near the target.

3.3 Stereo-based Fine Control

This is the control module that resides at the highest hierarchy level in the system, meaning that while tracking the target it subsumes the other two modules. It uses a stereo pair of cameras located on the robot's end-effector.

The visual-tracking loop in this module uses a blob analysis algorithm to detect, in both cameras, three prominent coplanar features on the engine cover. Then, using the calibration information, the 3D coordinates of those features are reconstructed, and the 6DOF 3D pose of the target is estimated in the robot coordinate system [17]. Based on the estimated pose, the visual-servoing loop then performs the peg-and-hole motion using a Proportional-Integral-derivative (PID) control law.

Since the 3D pose estimation relies on the visibility of three specific features on the engine cover, this module will only work if all three features are detected in both cameras.

Using an efficient blob analysis algorithm makes this module run very fast. It takes an average of 8.2 ms to estimate the 3D pose of the target, which is more than sufficient to process stereo images at 30 frames per second. However, this approach requires some thresholds to extract the blobs, making this module sensitive to sudden illumination changes.

3.4 Model-Based Fine Control

The main purpose of this module is to provide redundancy to the stereo-based fine controller. It uses a monocular vision system and a known wire-frame model of the target [16].

The visual-tracking loop of this module first projects the model into the input scene with respect to the initial pose that is given by the coarse control module or the stereo-based control module. Then, it sequentially matches the straight lines of the wire-frame model to the detected edges in the scene for an updated calculation of the pose of the target. For robust pose estimation, it uses a backtracking scheme for correspondence search [16].

Since this module uses extracted edges in the image, as opposed to applying thresholding to extract blobs in the stereo-based control module, it is less sensitive to illumination changes. Also, since the number of model features used in this module is typically much larger than 3 (the bare minimum number of features required to estimate the pose of an object), it is naturally robust to partial occlusions. However, it is slower than the Stereo-Based controller. It takes an average of 48 ms to estimate the 3D pose of the target, where 80 % of the computation time is taken by the edge detection phase.

4. EVALUATION EXPERIMENTS

In the experiments, the visual tracking system was set up to perform the peg and hole task using an engine part cover as the target. The engine part was either stationary or moved by an overhead linear rail at a constant velocity. The part was suspended from the linear rail to allow the experimenters to move the part back and forth by an attached string.

During each pass of the experiment, the position of the engine part was measured simultaneously by the Purdue visual tracking system and the NIST laser tracker. Data acquisition from each system was triggered by a common hardware signal, and the data time stamped and streamed to storage for off-line comparison.

The key elements of the data collection system were the laser tracker, the simultaneous data collection, and the establishment of common coordinate systems through calibration.

4.1 Laser Tracker

Figure 2. SmartTRACK sensor below engine part

The laser tracker system used consisted of two major components, a base unit and an active target, both made by Automated Precision Inc[1]. (API). The active target is a SmartTRACK Sensor (STS) capable of determining its orientation in 3D space. Weighing 1.4 kg, the STS has an angular resolution specification of ±3 arc-sec (or ±0.000833… degrees). The complete manufacturer's specification can be found in [18]. The base unit is the Tracker3™ Laser Tracking System (T3) which tracks the 3D position of the STS. The T3 system has ± 10 ppm absolute accuracy (e.g., ± 50 µm at 5 m). In addition, the base unit can be used with a passive target (a spherically mounted reflector (SMR) retro-reflective mirror) instead of the active target for single point measurements.

Together, the T3 and STS provide an accurate but limited 6DOF pose estimation system. The STS and T3 devices can take measurements independently at their own rate or they can be connected to a common external trigger.

4.2 Synchronization Issues

Both the T3/STS system and the Purdue visual-tracking system allow an external signal to trigger their data acquisition. Although the STS/T3 system is capable of handling a trigger signal up to 150 Hz, the visual-tracking system requires a 30 Hz data stream. We use a single 30 Hz trigger signal shared by both systems.

[1] Certain commercial equipment, instruments, or materials are identified in this paper in order to adequately specify the experimental procedure. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the materials or equipment identified are necessarily best for the purpose.

Although a shared trigger signal provides a solution to simultaneous measurements, the trigger itself needs to be reliable, repeatable, and controllable. We use a digital function generator that can be programmed to produce a clean squarewave signal. The digital function generator also allows us to start and stop the signal with a push of a button. The resulting signal is deterministic and free of the button-bouncing effect typically associated with inexpensive analog function generators.

The data collection programs maintain their own sequence number, which is increased each time a trigger signal is received. In addition, the program time-stamps the sequence number with a microsecond timer from the computer clock. Both systems synchronize their computer clock with a NTP (Network Time Protocol) server every 10 s throughout the entire data collection.

For data to be a matched pair, we require them to have the same sequence number. In addition, we check the difference between the corresponding timestamps and verify that the difference is small (i.e., the difference should never be greater than the period of the trigger signal shared by the two systems).

4.3 External Calibration Issues

To evaluate data collected during the experiments, we need to define metrics to compare the 6DOF pose data collected by the API T3/STS laser tracker and the Purdue line tracking system. There are eight coordinate systems involved, defined in Figure 2. We want to establish relationships between data collected in the coordinate system T of the laser tracker and one more coordinate systems of the Purdue system. We use the notation ${}_{X}H_{Y}$ to denote the homogenous transformation from coordinate system Y to X, so ${}_{T}H_{W}$ is a transformation from W coordinates to T coordinates. The 6DOF pose of the object O in the T3 coordinate system would be represented by ${}_{T}H_{O}$.

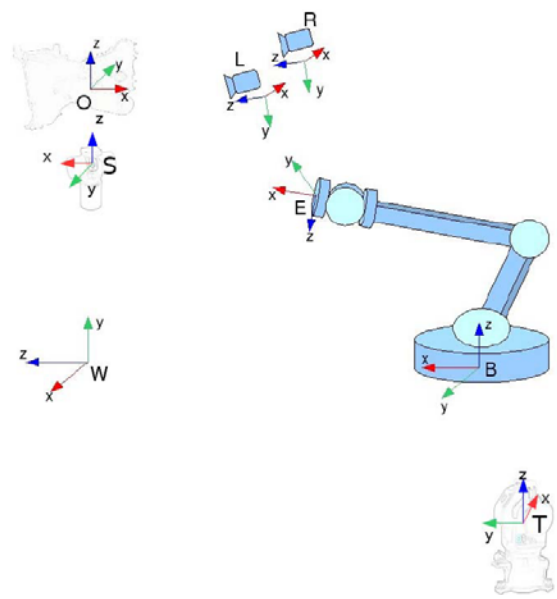


Figure 2. The coordinate systems of the Purdue Line Tracker and the STS/T3 Tracker System

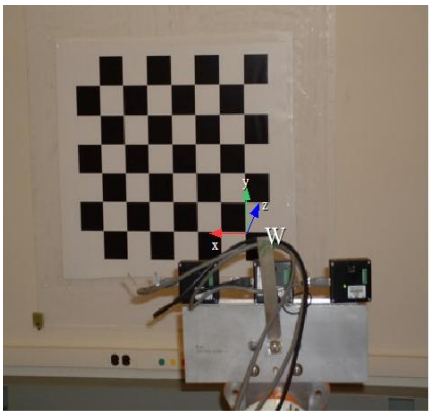


Figure 3. The world coordinate system

External calibration provides an estimate of the homogeneous transformation matrix (H) between the world W, and the robot base B, where W is defined by the calibration pattern behind the target as shown in Figure 3.

If we can successfully use the laser tracker coordinates T to establish the transformation ${}_T H_W$ or ${}_W H_T$ between the world (W) and T3, we can relate the coordinate T to the robot base B coordinate as in Equation 1.

$$ {}_T H_B = {}_T H_W \times {}_W H_B \qquad \text{Eq. 1} $$

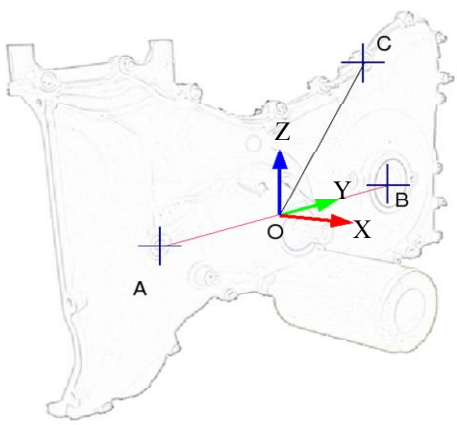


Figure 4. The object coordinate system (Y=OB, Z=up)

We don't have direct access to the origin of the engine cover O coordinates. Instead we use the Cartesian positions of three features (A,B,C), to reconstruct the coordinate frame for O with respect to coordinate T (see Figure 4).

We use the T3 laser tracker with an SMR to measure A, B and C in T coordinates. We then compute the origin O as the midpoint between A and B, and use the position of C to complete the object reference frames. Hence we construct ${}_T H_O$ or ${}_O H_T$. Similarly, we construct ${}_T H_W$ by measuring the origin of W.

Given ${}_S H_T$ from T3/STS system, we obtain ${}_S H_O$ from Equation 2.

$$ {}_S H_O = {}_S H_T \times {}_T H_O \qquad \text{Eq. 2} $$

Using Equation 1 and Equation 2, we establish transformations between the T3 base system T and the object system O in both the T3 and Purdue system, and compare those two transformations. The ground truth transformations provided by the T3/STS are on the left hand side, and the measurements provided by Purdue system and the hand-eye calibration are on the right hand side:

$$ {}_W H_T \times {}_T H_S \times {}_S H_O = {}_W H_B \times {}_B H_O \qquad \text{Eq. 3} $$

This gives the relationship ${}_W H_O(laser) = {}_W H_O(visual)$ and the inverse between the two systems in a common coordinate system, where visual stands for the Purdue tracking system.

Now we relate the two estimations of the inverse ${}_O H_W$ in the world coordinate system. In Equation 3, the calibration error in ${}_W H_T$, ${}_S H_O$ and ${}_W H_B$ can't be eliminated, so the absolute measurement approach was not used in this paper.

Since the Purdue system is a tracking system, the differential pose with respect to time is computed every cycle. It is reasonable to compare the differential pose between ground truth and the Purdue system using the world coordinates. We define ${}_{Oi} H_W(P)$ as the pose of the engine part in the world coordinate at time i for the Purdue Tracking system ${}_{Oi} H_W(G)$ as the pose of the engine part in world coordinate at time i for the ground truth system (laser tracker).

We can then obtain the following relationships for each system between time i and time i+1:

$$ {}_{Oi} H_W(P) \times {}_W H_{Oi+1}(P) = {}_{Oi} H_{Oi+1}(P) $$

$$ {}_{Oi} H_W(G) \times {}_W H_{Oi+1}(G) = {}_{Oi} H_{Oi+1}(G) $$

As one can see from the above derivation, the differential measurement will not depend on coordinate system. This approach produces the same numerical result for ${}_{Oi} H_{Oi+1}(G)$ and ${}_{Oi} H_{Oi+1}(P)$ in any coordinate system.

5. EXPERIMENTAL RESULTS

We present here preliminary results from a series of experiments conducted at Purdue in April, 2008. The protocol was to run three sets of experiments, one with the target stationary, one with the target moving with a simple linear velocity, and one with the target moving with a linear velocity but randomly displaced manually by the experimenters. The results are given below for the three sets.

The experiment had two complications. One is that the calibration of the laser tracker and Purdue system proved difficult and we were not able to establish a full set of coordinate transformations between laser tracker and visual system data. We did establish enough to compare relative poses, which are reported. Also, the

Purdue system is both a visual tracker and a robot control system. The data may at times confound the effects of both subsystems, as the robot introduces a physical motion. We only wished to evaluate the performance of the visual tracker.

5.1 Stationary Tests

The stationary tests allowed us to evaluate the basic performance of both systems and assure that the laser tracker was performing to specifications after shipping. The target was placed in four positions and data collected for 15 to 30 seconds for each. The results showed that both systems performed within specifications.

Table 1 lists the standard deviations of the stationary data set measured by the STS/T3 system. The results are consistent with the specifications [7]. The laser tracker stayed in a fixed position so the target distance varied.

Table 1. STS/T3 system: repeatability for stationary data

	Sample Size	T3/STS mean distance (mm)	2 std (mm)
Position 1	466	3550.054	0.006
Position 2	1157	3781.466	0.006
Position 3	1050	3882.787	0.005
Position 4	1018	4002.035	0.008

Table 2 lists the standard deviations of the stationary data set measured by the Purdue system. The Purdue system moves the robot end-effector near the object, so the distance remains relatively constant. The results show a consistent value near 0.6 mm at a range of approximately 2.6 m.

Table 2. Purdue system: repeatability for stationary data

	Sample Size	T3/STS mean distance (mm)	2 std (mm)
Position 1	466	2670.582	0.629
Position 2	1157	2625.820	0.582
Position 3	1050	2625.036	0.560
Position 4	1018	2636.701	0.561

For use as a reference system, a metrology technology should have an accuracy at least one order of magnitude greater than the system under test. In this case, the laser tracker is two orders of magnitude more accurate than the Purdue vision system.

5.2 Linear Motion Tests

In the linear motion tests, the target was moved about 1.5 meters left to right and tracked by both the laser tracker and the Purdue system. For each trial, the motion was repeated 30 times as the target moved, and then was quickly moved back to the start position. The backward sweep was ignored in the data analysis as the Purdue system only tracked during the forward motion.

The differential motion as measured by both systems was used to determine the consistency between laser tracker and Purdue system. In effect, the comparison is being made on the measured speed of the target in each separate coordinate (X,Y,Z, roll, pitch, yaw).

The data below are from pass 6 of the first trial, and are typical of the linear runs. The sample size is 453, with 33 ms between data points. In the graphs below the horizontal axis is frame number, and the vertical axis is the difference between the laser tracker motion change (delta) in each coordinate and the Purdue system delta. Since the laser tracker was accepted as ground truth, the difference is defined as the error in the Purdue system.

In Figure 5 the errors can be seen to be consistent and relatively independent of position along the path, although that is yet to be evaluated statistically. The error does vary by coordinate – different coordinates proved to be more sensitive to the left-to-right motion. In object coordinates the target is moving in the Y-Z plane, primarily in the Y direction, and the Z coordinate has slightly less error. Similarly, the roll angle is around the X axis, and proved an order of magnitude reduced in error over the other rotations. Table 3 quantifies the error values, while Figure 6 gives a histogram of the error values to show roughly symmetric, zero centered error distributions. For coordinate Y there appears to be a secondary peak in positive error.

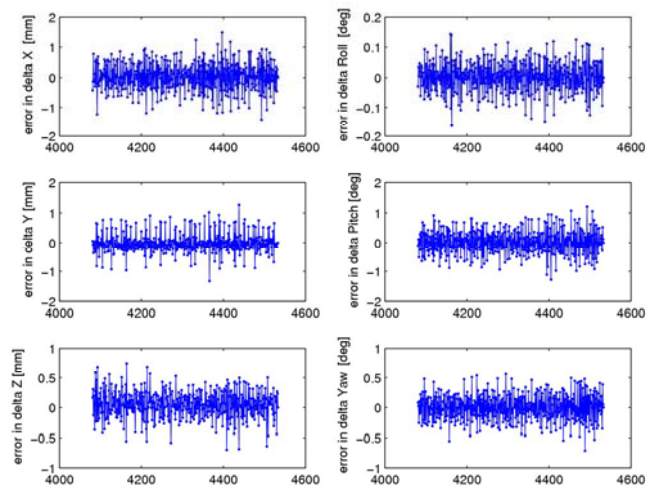


Figure 5. Pass 6 – Coordinate Errors in Delta vs. Frame Number

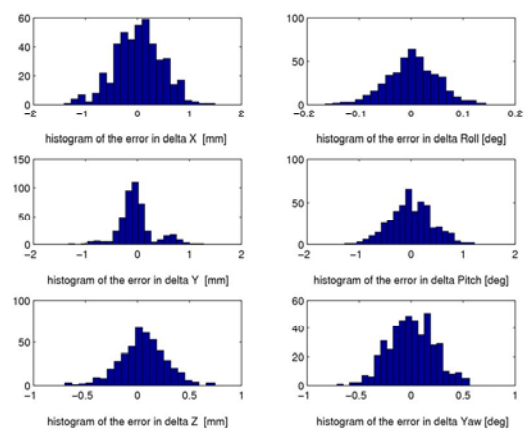


Figure 6. Pass 6 – Coordinate Errors in Delta (histogram)

Table 3. Pass 6 – Coordinate Error Statistics (n=453)

	Mean	Std	Max	Min
X in mm	0.01614	0.45376	1.49003	-1.41190
Y in mm	-0.03513	0.33355	1.27399	-1.32852
Z in mm	0.05206	0.21991	0.74652	-0.69851
distance	-0.04305	0.33273	1.21320	-1.50436
Roll in deg	0.00168	0.04915	0.14368	-0.16532
Pitch in deg	0.00088	0.40811	1.21672	-1.27938
Yaw in deg	0.00184	0.21589	0.56194	-0.71260
Total angle	-0.04798	0.05195	0.03383	-0.28635

5.3 Shaking Motion Tests

In the shaking motion tests, the basic target motion and repetitions were identical to those in the moving target test, but the experimenters could pull a string to swing the target with an impulse motion. This was done through a series of different motions – first with no extra motion to match the linear case, then with the impulse motion varying by amplitude and frequency.

The data below are from pass 3 of the first shaking motion trial. The sample size is 455, with 33 ms between data points. In the graphs below, the horizontal axis is frame number, and the vertical axis the difference between the laser tracker delta in each coordinate and the Purdue system delta.

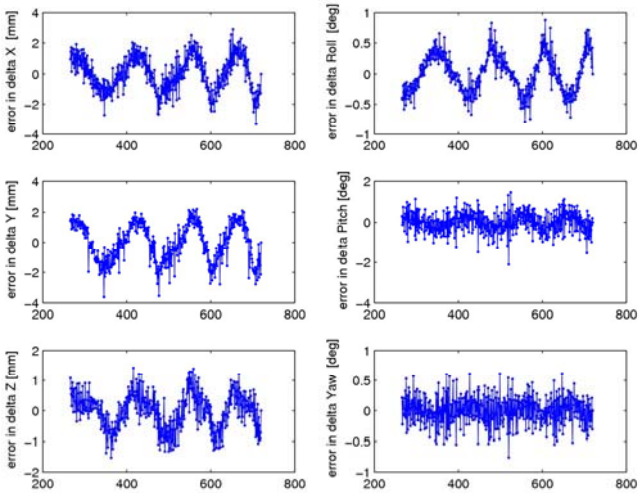


Figure 8. Pass 3 – Coordinate Errors in Delta vs. Frame Number

The graphs show four impulse motions as the target was pulled back four times, relatively smoothly and consistently. The error first goes positive as the speed of the target slows down and the Purdue system undershoots the speed, and then negative as the Purdue system overshoots the speed. The graph scales have changed from the linear motion case, as the error range has approximately doubled. The Z axis remains the one with lowest error, while the roll angle error is greater compared to the linear motion tests as the impulse motion rotated the target around the X-axis.

6. CONCLUSIONS

In this paper we have demonstrated the use of a precision laser tracker to evaluate a state-of-the-art visual servoing perception system, with the objective of establishing general techniques for evaluating 6DOF sensors. The demonstration involved a synchronized data collection system that used a hardware trigger to collect at a 30 Hz rate, while the laser tracker has the ability to collect at up to 150 Hz. The laser tracker was verified accurate enough for the approximate 1 mm range of error that needed to be measured.

Future work will involve a more detailed analysis of the data and the establishment of better calibration techniques and metrics to insure consistent comparisons between laser tracker and sensor data streams.

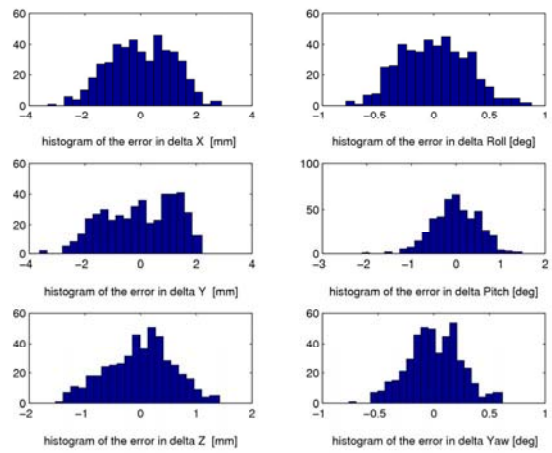


Figure 9. Run 6 – Coordinate Errors in Delta (histogram)

Table 4. Run 6 – Coordinate Error Statistics (n=453)

	Mean	Std	Max	Min
X in mm	0.00219	1.13255	2.90429	-3.32659
Y in mm	-0.04633	1.26350	2.21070	-3.62808
Z in mm	-0.01445	0.58950	1.42051	-1.53047
distance	0.90879	0.95975	3.77670	-1.84237
Roll in deg	0.00362	0.30460	0.87745	-0.78979
Pitch in deg	-0.00329	0.50645	1.48264	-2.10615
Yaw in deg	0.00103	0.23520	0.61750	-0.76112
Total angle	-0.42713	0.30752	0.15745	-1.51979

7. ACKNOWEDGMENT

We thank Elena Messina and Hui-Min Huang for the fruitful discussions. We thank Nick Scott, Alan Lytle, Chris Blackburn, Daniel Sawyer, Steve Phillips from NIST, Avi Kak from Purdue University, Jane Shi from General Motors, and Kam Lau, Alan Cui, Zaifeng Chen from API for their collaborations.

8. REFERENCES

[1] Chai, X., Shan, S., Qing, L. and Gao, W. 2006 Pose Estimation Based on Gaussian Error Models, International Conference on Biometrics, 2006: 136-143

[2] Deguchi, K., 2000 A Direct Interpretation of Dynamic Images with Camera and Object Motions for Vision Guided Robot Control. International Journal of Computer Vision 37(1): 7-20 (2000).

[3] DeSouza, G. N. and Kak, A. C., 2004, A Subsumptive, Hierarchical, and Distributed Vision-Based Architecture for Smart Robotics, IEEE Transactions on Systems, Man, and Cybernetics -- Part B: Cybernetics, 34(5): 1988-2002.

[4] Espiau, B., 1993 Effect of Camera Calibration Errors on Visual Servoing in Robotics. Experimental Robotics III, The 3rd International Symposium, Kyoto, Japan, October, 1993: 182-192.

[5] Fiore, P. D, 2001 Efficient Linear Solution of Exterior Orientation, IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23(12): 140-148, 2001

[6] Hirsh, R., DeSouza, G. N. and Kak, A. C., 2001 An Iterative Approach to the Hand-Eye and Base-World Calibration Problem, Proceedings of the IEEE International Conference on Robotics and Automation, Seoul, May 2001

[7] Hutchinson, S.A., Hager, G.D., and Corke, P.I, 1996 A tutorial on visual servo control. IEEE Transactions on Robotics and Automation, 12(5):651--670, October 1996.

[8] Kyrki, V., Kragic, D., and Christensen, H. I., 2006 Measurement errors in visual servoing. Robotics and Autonomous Systems 54(10): 815-827 (2006)

[9] Madsen, C.B, A comparative study of the robustness of two pose estimation techniques. In H.I. Christensen, W. Förstner, and C.B. Madsen, editors, Proceedings: ECVnet Workshop on Performance Characteristics of Vision Algorithms, held in conjunction with Fourth European Conference on Computer Vision, Cambridge, England, pp. 181 - 200, April 1996.

[10] Preisig, P.. and Kragic, D., 2006 Robust Statistics for 3D Object Tracking. Proceedings of the Int. Conference on Robotics and Automation, Orlando, Florida, 2006.

[11] Chai, X., Shan, S., Qing, L. and Gao, W. 2006 Pose Estimation based on Gaussian Error Models, Proceeding of International Conference on Biometrics (ICB2006), Lecture Notes on Computer Sciences (LNCS3832), D.Zhang, A.Jain (Eds.), pp136-143, Hong Kong.

[12] Reid, G. J., Tang J., and Zhi, L., 2003 A complete symbolic-numeric linear method for camera pose determination, International Symposium on Symbolic and Algebraic Computation, Philadelphia, Pennsylvania 2003: 215-223.

[13] Rivera-Ríos, A. H., Shih F. L, and Marefat, M. M. 2005 Stereo Camera Pose Determination with Error Reduction and Tolerance Satisfaction for Dimensional Measurements, Proceedings of the Int. Conference on Robotics and Automation, Barcelona, Spain, 2005: 423-428.

[14] Sepp, W., Fuchs, S. and Gerd Hirzinger. 2006 Hierarchical Featureless Tracking for Position-Based 6-DoF Visual Servoing, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2006, Beijing (China), 2006

[15] Tonko, M. and Nagel, H. 2000 Model-Based Stereo-Tracking of Non-Polyhedral Objects for Automatic Disassembly Experiments, International Journal of Computer Vision 37(1): 99-118 (2000).

[16] Yoon, Y., DeSouza, G. N. and Kak, A. C., 2003 Real-time Tracking and Pose Estimation for Industrial Objects using Geometric Features, in Proceedings of the Int. Conference in Robotics and Automation, Taiwan, 2003

[17] Yoon, Y., Park, J. and Kak, A.C. 2006 A Heterogeneous Distributed Visual Servoing System for Real-time Robotic Assembly Applications, in Proceedings of the Int. Conference on Robotics and Automation, Orlando, Florida, 2006.

[18] SMARTTRACK SENSOR, Automated Precision Inc., http://www.apisensor.com/PDF/SmartTrackeuDE.pdf, 2007.

Integrating Reification and Ontologies for Mobile Autonomous Robots

J. P. Gunderson
Gamma Two, Inc.
209 Kalamath Street, Unit 13
Denver, CO, USA
jgunders@gamma-two.com

L. F. Gunderson
Gamma Two, Inc.
209 Kalamath Street, Unit 13
Denver, CO, USA
lgunders@gamma-two.com

ABSTRACT

A significant problem that has plagued autonomous mobile robots is the need to reason about a complex and dynamic world in a timely fashion. Huge strides have been made in artificial intelligence systems that deal with symbolic representations, and the hardware for robotics has seen significant growth. However, autonomous robots have not seen equivalent improvement. In this paper we examine a technique to couple deliberative reasoning systems with mobile robotic platforms to enable the robot to reason about the world.

Mobile autonomous robots operate under extremely tight constraints in power and computational loads, and must be responsive to the dynamic environments in which they are deployed. Since they do not have the luxury of operating in a static, controlled environment, they must be capable of quickly recognizing changes to the environment, assessing the impacts of these changes, and implementing intelligent responses to those changes. All of these requirements must be met in a real-time environment, using limited computational resources. Since living organisms also operate under these same constraints we present a design that is biologically principled. This design integrates both reactive and deliberative components using a biologically principled component called a `Reification Engine`.

The `Reification Engine` acts as a bi-directional bridge between the sensor domain and the symbolic domain. It provides the ability to map the changes detected by the sensors into a symbolic representation, which is expanded by the onboard ontology to generate a semantic representation of the salient aspects of problem. This, in turn, can be analyzed by a deliberative system to generate a planned response to the changes, which can be mapped by the `Reification Engine` back into the sensor/effector space of the robot's hardware.

Keywords

robotics, ontology, reification, artificial intelligence, autonomy, biologically inspired computing

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08 August 19 - 21 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

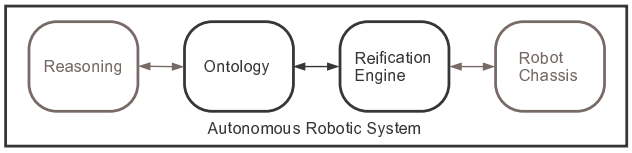


Figure 1: The four major components of an autonomous robotic system. The Reasoning component provides the ability to determine how to achieve goals. The Robot Chassis enables the system to execute the behaviors in the physical world. The middle components - the Ontology and the Reification Engine, translate between the symbolic and the sensor-based domains.

1. INTRODUCTION

This research has been driven by a single question: "Where is my Robot?" The last few decades have seen an explosion in robotic systems in many domains, including factory and laboratory automation[6, 11, 15, 22, 23], deep sea exploration[1, 12, 5], and exploration of deep space[20]. At the same time we have seen the use of teleoperated systems in search and rescue, bomb disposal, and surveillance aircraft; all of which have saved lives. However, we have not seen an equivalent explosion of autonomous systems.

An autonomous robot can be modeled as a system consisting of four major components. The first of these is the physical manifestation of the robot - the sensors, effectors, and electronics that make up the hardware. Since an intelligent, autonomous robot is being considered, the second major component is the software that makes up the 'brains' of the autonomous robot. The reasoning component operates in a purely symbolic domain: it manipulates symbols according to abstract rules. Between the reasoning and execution components lie two others - a mechanism to link meaning to those abstract symbols (the ontology), and a reification mechanism[9] that bridges the gap between these symbols and the physical domain. The ontology and the `Reification Engine` are the interface between the 'brains' and the 'brawn' - they translate between the domain of sensors and effectors into the domain of symbols and semantics. These four components are shown in figure 1.

In this paper we will focus primarily on the interaction between the ontology and the `Reification Engine`. Rather than focusing on any specific robotic chassis, we will refer to an abstract perception/action system. In the same way we will refer to a generic deliberative system, rather than

any specific architecture for the generation of behaviors or plans. We will focus on specific implementations of the ontology and the `Reification Engine`, since we are exploring the integration between the mechanisms used to represent knowledge in these two components.

2. REASONING

For most of its history, artificial intelligence has been focused on reasoning. Some of the earliest AI programs were designed as problem solvers[13, 7], planners[18], or designed to excel at games like checkers[19]. These programs manipulated symbols according to specific rules, and searched through solution spaces to find plans that would achieve the system's goals. These systems frequently relied on *ad hoc* representations of the symbols and their allowed manipulations.

Recently there has been more research into the use of ontologies as a standardized format for representing the meaning of symbols. The reasoning components of the ontology use the relationships between the concepts as rules controlling the manipulation. In some systems, the ontology fills the role of semantic memory in humans: a large diverse, persistent storage of all available knowledge. Since only a small portion of the entire knowledge base is needed for any specific problem, a subset of the ontology is used to generate the actual problem that the reasoning system processes (See [10] Chapter 9 for more information). With mobile robots, these ontologies must not only capture the abstract concepts that are the focus of much of the research, they must also capture the knowledge associated with the physical domain in which the robot functions.

2.1 Personal Rough Ontologies

Current research into artificial intelligence has addressed the need for an *intelligent* system to have a complex and detailed body of knowledge to function effectively in a complex, dynamic, and uncertain world. Unfortunately, these ontologies are not well suited to deployment into autonomous robots, since they tend to be computationally expensive. However, an autonomous robot has a critical need for the ability to reason effectively about the world into which it is deployed, especially in unconstrained environments. We demonstrate how the integration of a `Reification Engine` with an ontology tailored for a robot's needs can enable the robot to function more effectively in a dynamic world.

We utilize a biologically principled form of an ontology, designed for effective integration into an autonomous robot. The key differences lie in the area of completeness and consistency. Most living systems do not have complex truth maintenance systems analyzing every new fact or belief that is added into their cognitive systems. In spite of this, they far outperform most autonomous robotic applications. We have extended (or crippled, depending on one's viewpoint) ontologies so that they can contain inconsistent data without causing logical operations to result in invalid states. This is done by controlling the deductive reasoning to a limited distance within the ontological graph. This has two significant benefits to autonomous robots. First,the computational complexity of navigating the ontological graph is dramatically reduced, since the expansion of edges eventually tapers off to zero. Second, this mechanism acts as a *focus of attention* which enables the extraction of a small, salient subgraph to be used by the deliberative system for problem solving - thus dramatically reducing the computational burden of the planning system. Because of this lack of truth maintenance we call this a rough ontology. Since the system will modify its ontology based on personal experience, we call this a personal rough ontology (`PRO`).

The basic structure of an ontology is a directed multigraph:

$$Ontology \equiv (\mathbf{C}, \mathbf{E}) \tag{1}$$

where: $\mathbf{C}$ is a collection of nodes; and
$\mathbf{E}$ is a collection of directed edges from one node to another.

2.2 The nodes in the multi-graph

Referring back to Equation 1, we see that the multigraph consists of a collection of nodes and a collection of directed edges. Each node C corresponds to a symbolic representation of an entity. The base class for these nodes is a `Cept`, hence the non-traditional use of $\mathbf{C}$ in equation 1. Each `Cept` may have a collection of properties that describe it. These properties are used in two ways. First, the ontological system can use these properties when applying rules, and second, the deliberative system uses these properties to develop plans to achieve its goals. To facilitate these uses, the properties are represented as specific relationship arcs to other `Cepts`.

$$\mathbf{Cept}(\mathbf{C}) \equiv (symbol, \{P\}, \{R\}) \tag{2}$$

where: *symbol* is the symbolic representation of the concept (its name);
$\{P\}$ are the properties; and
$\{R\}$ are the relationships to other Cepts.

The `Cepts` in our ontology represent the things about which we have knowledge. These things correspond to the subject and object of the relationships. The common element of these things is the symbol and the list of properties. This base class is extended to represent specific types of knowledge in the ontology.

The simplest variant of a `Cept` is a `ConCept`. This is what is represented in a typical ontology. The `ConCept` extends the basic `Cept` by adding the idea that a `ConCept` can represent either an abstract concept or something more concrete. As an example, the `ConCept` of a chair represents a perceptible thing that might exist in the world; however the `ConCept` of 'laid-back' is not something that is directly perceptible. In the case of perceptible `ConCepts`, the `Reification Engine` has the responsibility of maintaining the mechanisms that allow the recognition of objects in the real world, and linking those perceived objects to the symbolic representations. This is done by adding a link from each perceptible thing in the ontology to a data structure called a `PerCept`, maintained by the `Reification Engine`, described in Section 3.

Much as the `ConCepts` correspond to nouns, the world is also full of verbs - the actions and events that cause change. Although knowledge about actions may seem to be fundamentally different than knowledge about things, they are still symbolic representations about the world. In the chair discussion above, it seemed natural that there would be a relationship between a chair and the concept of sitting, just as there seems to be a natural relationship between a bicycle and riding or a ladder and climbing. Note that the

conceptual symbol associated with sitting differs from the mechanics of lowering ones body into a seated position - that is a function of the perception/action system.

To represent the conceptual model of an action, we modify the base `Cept` class. This subclass is called an `ActCept` in the ontology. The base class is extended by adding two additional components - the conditions under which the action can be taken (e.g., to *drive_the_car* it is necessary to have a car), and the possible outcomes of acting (you might get where you want to go, or you might run out of gas). The enabling conditions and outcomes can be directly mapped onto the corresponding components of the action representation used by the deliberative system, and so the semantic memory fulfills its design goal of being the repository of knowledge for the reasoning system.

2.3 Relationships, the Edges of the Graph

Each edge in the multi-graph is a directed arc that leads from the subject node to the object node. Directed arcs are needed since not all relationships are symmetric, for example, if the type of the relationship is *part_of*, it makes sense to say that a leg is *part_of* a chair; however the reverse is not true. Each edge captures one specific type of relationship. The representational schema for the arcs is:

$$Edge(E) \equiv (C, I, N, T, S, L) \tag{3}$$

where: C is the class of the relationship;
I is the unique identifier;
N is the destination node;
T is the transitivity;
S is the strength; and
L is the likelihood that the relationship holds.

Each relationship **E** is an instance of a class of relationships. While the ontology is very generic, there seem to be a number of relationship types that are common across ontologies for systems embedded in the physical world. These include relationships like *is_a*, *has_a*, *is_part_of*, and *is_at*. To facilitate the encoding of capabilities we also add *can_do* which is the relationship between a `ConCept` and an `ActCept` that it is capable of executing.

2.4 Transient versus Persistent Knowledge

Knowledge changes over time. While many ontologies are designed to capture the relationships between permanent concepts such as what it means to be a mammal or what the definition of a prime number is; there is also a need to represent more transient information. There is a vast amount of relatively transient knowledge, far more than can be maintained in working memory at any point in time. So there must be a more permanent storage place. We include this transient knowledge in the form of properties attached to the `ConCept`, and use a specific relationship *instance_of* to link the individual `ConCept` to the framework of knowledge of more abstract concepts. In practice we have two specialized relationships for handling instance `ConCepts`: *thing_instance* and *location_instance*. These are used to provide the linkage from the specific instances to two corresponding nodes in the ontological frame work. These are labeled as *Place* and *Thing*.

In addition, there are instance `ConCepts` that appear and disappear from our knowledge base. We often encounter new people in our day to day interactions. While some of them may become friends or colleagues, many will be a single, short-term encounter. While we are interacting with them, we need to access the complex web of knowledge we have of people, and this can be done by instantiating an 'anonymous' person `ConCept`. This provides access to generic background information that allows us to reason about this person, but we do not need (or want) to clutter our ontology with references to every person we have ever encountered in our lives.

The final type of symbolic knowledge is knowledge about the current state of the world. In order for the planning system to work it needs a relatively accurate model of the state of the world. This is contained in the Current World State (CWS). The CWS is the symbolic representation utilized by the deliberative system. A more spatially detailed representation, called a `Mental Model` is used by the `Reification Engine`. When the robot encounters an object in the real world, it attempts to classify the object, so that a semantically tagged representation can be added to the CWS. The CWS includes knowledge about the permanent persistent features of the world, and it contains information about the current state of previously encountered transient objects. However, there is no pre-existing representation for an object that is encountered for the first time.

As an example, consider a robot serving drinks at a cocktail party. It knows about the general layout of the room - the walls, the big furniture, etc. This is permanent and persistent data. It also knows about a few smaller chairs, which are permanent, however, their current location may change as the chairs get moved around. It does not have instantiated representations for all the people who might arrive. Rather, when the robot encounters a new object, it will attempt to classify it. Using its sensor data, and the `PerCepts` that correspond to the currently instantiated objects, if the object can be classified, both the CWS and the `Mental Model` are updated with the new knowledge. If it fails to classify the object, it then accesses the `PerCepts` associated with more general concepts. For example, the `PerCept` for a generic `Person` has a high probability of a match, and the `Reification Engine` uses this to instantiate a new object in its CWS. This object is not permanent - rather it is transient. It comes into existence, and, after not being reinforced, it vanishes. In this way the size of the ontology is not ever increasing with references to concepts, objects, and entities that will never be accessed again.

3. REIFICATION ENGINE

The bi-directional mapping between the symbolic representations used by the ontology and the sensor based representation utilized by the perception/action system of an autonomous robot is called *reification* (See Robots, Reasoning, and Reification[10] for more detail). This process is derived from current research in cognitive science and neurophysiology. There has been research into the two areas of Symbol Grounding (mapping symbolic representations onto sensor based leaf terms) and Pattern Recognition (projecting from sensor based data onto symbolic representations). One belief is that either of these approaches, if continued sufficiently, will bridge the Sensor/Symbol gap. We believe that there is no overlap between these two research areas, and that a separate process - reification - is used by living systems when they perform these mappings.

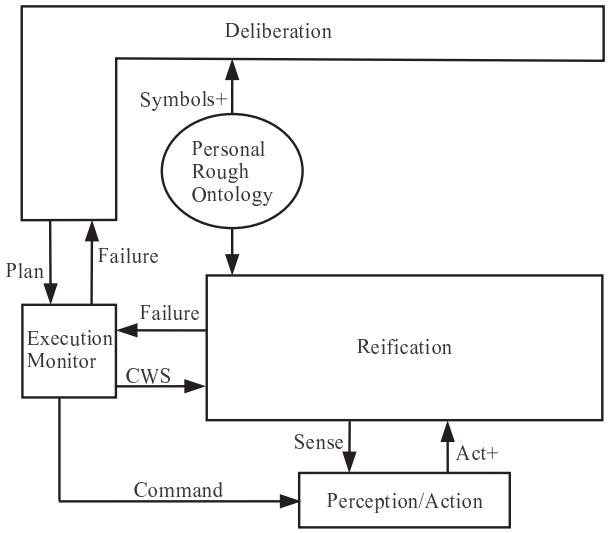


Figure 2: The structure of the Reification Engine interactions with both the PRO and the Deliberative System. Due to the tight coupling of these systems, and the central role played by the Reification Engine, the representations and services provided by this system are highly specific to the underlying sensors and actuators as well as the semantic representations needed by the deliberative system.

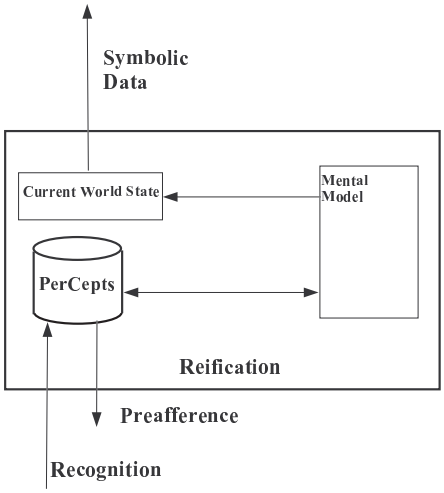


Figure 3: Using the PerCept datasets, the Reification Engine maintains a Mental Model of the things in the world, and their current state. This is used to create a symbolic representation of the Current World State, which is used, in turn, by the deliberative System. The Mental Model is used to both simplify the recognition of sensor data (e.g. turning it into symbolic information) and to generate preafference images of what the world would look like if a predicted change were to occur.

Reification provides these two services - mapping sensor data onto symbols and mapping symbols onto sensor representations. The first of these pathways corresponds closely to pattern recognition, and is treated as such in the `Reification Engine`. The latter pathway is called *preafference*, after a term used by Walter Freeman[8]. Research has shown that living systems significantly reduce the cognitive load associated with making sense out of the world by loading their sensory cortices with approximations of what they expect to perceive. If these approximations are close enough, the higher cognitive functions are left out of the loop, and the entity functions as a reactive system. However, if there is enough of a discrepancy between the expected and the actual, then the more expensive cognitive functions are brought on-line to re-evaluate the situation, and generate new solutions.

3.1 Reification Architecture

To bridge from the symbolic domain into the sensor domain and back again, the `Reification Engine` must have one foot on each shore. The architecture of the `Reification Engine` is shown graphically in Figure 3. There are three main components in the `Reification Engine`. The core component is the library of `PerCepts`, which provide the grounding in the sensor domain. Using these `PerCepts`, the `Reification Engine` maintains a model of the world, this `Mental Model` captures the sensor-derived knowledge about what things are out there, where they are, and how they are oriented. Finally, the `Reification Engine` manages a symbolic representation of the world. This CWS is provided to the deliberative system, by way of the `PRO`. The library of `PerCepts` capture the structural description of things in the world that are perceivable.

In our model a `PerCept` is the data structure used to hold the information associated with a perceived object. The `PerCept` has two components: a sensor derived signature that can be used to recognize the occurrence of the object, and a symbolic component that links to the semantic representation in the ontology. In addition to these two data components, the `PerCept` provides some basic functionality to the `Reification Engine`. Each `PerCept` is associated with a single class of perceivable object. These `PerCepts` have a symbolic tag (think of it as a name) and the sensory definition of the object.

Imagine a chair. Many of us maintain a complex hierarchy of chair types in our minds. There are kitchen chairs, office chairs, recliners, chairs with wheels, chairs without backs, chairs with padded seats, chairs that creak ominously when we sit in them. There are specialized chairs: seats in automobiles, benches, couches, dentist's chairs, and the 'naughty

chair' where children have to sit if they have misbehaved. Yet, in spite of the array of possible chairs, we seem to have a generalized concept of *chair*, at least enough to answer the question "Is that a chair?" Much as in Plato's notion of the Ideal, or the concept of a 'class' in object-oriented programming, we have a complex structure of symbols that correspond to the different types of chairs. We can utilize those symbols to reason about the types of chairs we encounter. The `PerCepts` have one foot in this complex symbolic structure. Each `PerCept` requires a symbolic tag that links into the semantic knowledge used by the deliberative system. If the ontology has information that there exists *Jim's_chair* which is a specific instance of a *short_wheeled_chair*; then there must be a `PerCept` that is identified by the semantic tag *short_wheeled_chair*, and there must be a thing in the `Mental Model` which corresponds to the current state (perhaps position and orientation) of *Jim's_chair*.

The `PerCepts` also have one foot deep in the sensor world. This sensor based representation is defined by the physical abilities of the body of the system. Each robot has its own types and placement of sensors, so the things that it perceives can and will have different sensor representations. This is true in biological systems as well. We, as humans, are very sight oriented[24], and so we attach visual representations to our semantics. We use vision as a primary metaphor for understanding - do you see what I mean? Look at it this way - at the beginning of the last paragraph, you read the words "Imagine a chair." For many of you, a visual image formed in your mind. However, other species have very different sensor modalities, and so their percepts will vary considerably. Imagine a human, a dog, and a bat encountering the same insect. While the human would focus on the appearance of the insect, the bat would more likely form an acoustically based 'image' of the bug, and the dog would generate a scent based model of great complexity.

We utilize a data-driven representation of the sensorium that corresponds to the object represented by the `PerCept`. This model is based on Brunswik's *lens model*[4]. There are two primary reasons for this approach, one is that it ensures a direct correspondence between the mental model of the thing and the perception of the thing. We create a dataset which can be used as a template. This template can be 'filled in' with the specifics of the location and orientation, and other details, to generate what the sensors would perceive. This contrasts with a more analytic representation of an object and the necessary transforms that could be used to generate the sensorium. There are several advantages to the dataset approach. Since the identifying characteristics are data, the `Reification Engine` can generate new `PerCepts` by creating new data representations. If the analytic representation were used, the system would need to generate new functions and add to its code. Of course, there is a significant disadvantage to the dataset approach: the datasets can get large, and are complex. The penalty we pay for the ability to learn is that much of the mechanism of the `Reification Engine` is dedicated to managing the `PerCepts` to provide the functions of preafference and recognition.

3.2 Recognition

The `Reification Engine` uses the `PerCepts` to recognize the state of the outside world. A simple brute force approach would be to take the sensor data and attempt to match it against every possible `PerCept` in the library. Of course there might be a large number of `PerCepts` in the library, each of which could be at any possible distance and a number of possible orientations. The computational complexity of this task is daunting to say the least. We know that living systems perform this process quickly and (in general) reliably. We also know that computer-based recognition is (in general) neither.

We take advantage of the fact that sensing does not occur in a vacuum. Sensing and perception are one step in a continuous cycle, which was preceded by an earlier perception stage. One of the core functions of reification is to maintain a (more or less) continuous model of the `PerCepts` that correspond to the state of the world[17]. The `PerCepts` from the last perception, and the `Mental Model` built from them, can be used as a guide to shorten the subsequent perceptual process. This is based on the concepts of 'thing constancy'[21] and 'object constancy'. In effect, the things that we perceive do not (in general) magically appear or disappear. Thus, we evolve the previous model of the world, based on our knowledge of what activities are occurring, and predict the ensuing state of the world. With this predicted model, we can confirm what we expect to see, at a far lower computational cost than approaching the perceived world *de nova*.

3.3 Preafference

The second function that the `Reification Engine` supports is the projection of the model onto the world. Preafference is the process of taking the `Mental Model` of the world, and generating the view that the robot would see, if everything were like the model. This process produces a *virtual reality*(VR) representation of the world as it is expected to be. In effect, the `Reification Engine` acts as a rendering engine for the mental model of the world. This rendered image is used to speed up the process of confirming that the world is pretty much what we expect. Fortunately, much of the machinery needed to perform this function is shared by the recognition task. It is only needed to use these functions in a different way to generate a preafference image, rather than to recognize the objects in the world. Just as a physical lens bends light regardless of the direction, our cybernetic lens can be used to either project the sensor domain into the symbolic, or project the symbolic domain into the sensorium.

To do this projection into the sensor domain, we take advantage of the fact that the `Mental Model` has a list of the objects that are expected to be in the world, and their positions, orientations and properties. With this information, and the machinery of the `PerCepts`, generating this rendered image is relatively straight forward. In the case of a sonar based perception system, this rendered image is a sonar image of what the sonars would return. The generation of this image is done in three stages:

1. Select the objects that are in the field of view of the robot;
2. Use the `PerCepts`, the pose of the object, and the pose of the robot to generate individual images; and,
3. Fuse these individual images into a composite image.

The first of these steps is simply done to reduce the computational burden of the full process. Rather than attempting to build the sensor images of objects that might be in

another room, or even attempting to generate a complete 360° panoramic view, we only generate images for objects that are likely to be in view. This is the same process that is used in VR rendering systems in the form of bounding surfaces[16]. The `Reification Engine` uses the sensor parameters for the specific sensors, and slightly over estimates the field of view of the robot. This angular interval is compared with each of the individual objects. We take advantage of a function of the `PerCept` that calculates the perceived angular extent of an object, based on its position and orientation, and the position of the robot. A quick test to see if the two intervals overlap is sufficient to determine if this object is in the field of view.

The final step is for the `Reification Engine` to fuse these individual views into a complete sensor image. If there is only one object generating a sensor image, this task is trivial. The difficulties arise when there are multiple objects in the field of view and there are occlusions (e.g., one object generates a return that blocks the view of another object). This is a common occurrence as the resolution of the sensor increases. However, the techniques for resolving occlusions have been well tested in visual rendering software. Utilizing the information encoded in the individual returns, and the relative positions of the robot and each item, it is fairly straight forward (if computationally intensive) to generate a final rendered sensor image. This final preafference image can be quickly compared against the actual sensor image to see if there are major discrepancies.

3.4 Updating the Current World State

The final task of the `Reification Engine` is to take the changes that have been made to the `Mental Model`, and project them into the semantic space of the Current World State. This process requires the translation from the sensor driven, numerically based information in the `Mental Model`. The key support structures for this translation are representations that link the semantic terms utilized by the deliberative system, with representations that are consistent with the `PerCepts`. For example, The `Mental Model` maintains precise locations of the objects that the robot knows about. However, the deliberative system does not model the location of objects at this resolution. It keeps track of the fact that the chair is over by the assembly bench, not that it is at coordinates X=10,100, Y=500. So the Current World State must maintain information at the level of detail needed by the deliberative system, using the symbolic names of the locations in the ontology.

As an example, imagine our robot traversing a room. At the start of the motion, the robot has a `Mental Model` of the room, its relative position to the walls, the distance to the table, the location of the chair. When the robot moves forward, the `Reification Engine` updates the expected world state, changing the position of the robot in the room. When the sensory data arrives from the perception/action system, the robot can ask the `PerCept` for the table "If I am in this position, and looking this way, am I seeing you?" Since the `PerCept` includes the necessary structural description, it can quickly confirm (or deny) whether the sensory data is consistent with the expectation. This can be continued with the chair, the walls, the picture on the wall; and if the world model is confirmed, it can be updated. This process does not require querying every possible `PerCept`, or even querying every expected `PerCept` for every possible position and orientation. Rather, it quickly attempts to confirm the expected world state.

3.4.1 Dealing with the Unexpected

Sure, but what if the world is not what the robot expects? What if someone sneaked in and moved the chair, or simply stepped into the robot's field of view? In this case the presumptive confirmation of the world fails. Now the `Reification Engine` begins to expand the range over which the search occurs. If there are people (or other moving entities) in the world model, they are added to the search "Is it possible that this person stepped to the left?" "Is it possible that the chair moved?" "Am I really where I expect to be?" If this level of relaxation of the model fails to produce a consistent world model further relaxation may occur, less likely objects are added to the search list (although it will be a while before the robot considers the possibility that a zebra has suddenly appeared in the living room). This is consistent with the results of studies on humans in disasters. People who experience dramatic destruction (such as coming out of a storm cellar after a tornado) report that for several minutes they cannot place themselves in space, the permanent features that they expected to see, the buildings, the walls, the trees, have been altered to the point where they cannot perceive a frame of reference. Nor is this limited to humans, pets can also experience disorientation due to the disruption of their sensorium, notably the changes to the landscape of scents that they use to define space[14].

3.5 Wrapping Up Reification

Reification appears to be a component necessary for autonomous systems to function effectively in a dynamic and uncertain world. Reification is the mechanism that translates between the sensor domain - the world of images, echoes, encoders, and force transducers - into the semantic domain of chairs and trees, rooms and fragile packages. In this section we sum up the various pieces, and look at how they work together to provide reification to the robot.

Reification can be viewed as a necessary middle-ware component for autonomous robots. It provides recognition services to map the sensorium into symbolic representations, and provides the complementary service preafference to map the expected state of the world into the anticipates sensor representations.

The `Reification Engine` presented here both anchors the deliberative symbols in the sensor domain, and attaches semantic tags to the patterns presented by those sensors. It does this, not by extensively hard-coding the patterns, but by associating the data derived patterns with the meanings used by the ontology. The data derived patterns are encoded as `PerCepts`, utilizing the lens model pioneered by Brunswik. This `PerCept` based approach has several benefits, as well as costs, in comparison to an analytic representation scheme. One of these benefits is that the data driven approach is based on the actual properties of the objects as viewed by the actual sensors, rather than an idealized model. As a result, if a given sensor has an idiosyncrasy (perhaps it is mounted at a slight angle), or if a specific object has a unique property (e.g., it absorbs both IR and ultrasonic pulses) that is reflected in the data derived pattern.

The second major benefit is that the patterns are simply data. There are several basic algorithms that are grounded in the physical relationships of the world: things like dis-

tance calculations, and the determination of relative angles. However, the representation of objects, sensors, and the structural descriptions of the world are simply data. This means that the cybernetic brain, and specifically the `Reification Engine`, can modify these patterns as it experiences more of the world, and can modify the parameters in response to the changes that the world will have on the robot's own structure. By using data structures for as much as possible, the robot can both stay synchronous with the dynamic world, and can update its mental representations as it experiences new things.

4. INTEGRATION

The integration of the `Reification Engine` and the `PRO` is driven by the need to effectively and efficiently share representations across a range of resolutions. The cycle can be traced from almost any point, but we will start from a change detected by the sensors in the perception/action System. Perhaps the sonars on the robot have been pinging a doorway, and the range value in the opening is 4 meters. If the door closes, the range drops to only 1.5m. The question is how does the robot interpret this change in a way that can be reasoned about?

The output from the sensor system is passed upwards to the `Reification Engine` along the Recognition pathway. Since the `Reification Engine` has a model of the CWS, it can detect the change, determine that the the change affects the doorway, and query additional `PerCepts` to conclude that the change is likely to indicate that the door has closed. The enables the `Reification Engine` to update the symbolic representation of the CWS to include a closed door. This symbolic information is passed to the `PRO`, which can extract a new problem statement. This semantics of a closed door (from the ontological relationships and properties) results in the extraction of possible actions that might affect the door state, as well as actions that can circumvent the closed door. These are packaged into a problem description which is passed to the deliberative system for solution generation.

When the deliberative system produces a plan (purely symbolic) it is passed down to the `Reification Engine` for the generation of preafference maps, which are passed down to the perception/action system for execution. Since each action has the expected sensor-based representation of the world required for its application, and the representation for the expected outcome of the action, the perception/action system can quickly detect any salient change to the environment that would require the invocation of the deliberative system. In effect, we generate a purely reactive system on the fly, and as long as things are going well, the computationally expensive operations are off-line.

Traditionally, mobile autonomous robots are built with well developed perception/action systems and fairly light weight deliberative systems. In fact much of the robotics research in the 1980's and 1990's focused on removing the deliberative systems entirely[2, 3]. This was based on the idea that deliberation is computationally expensive, and representation of a world model is both expensive and hard to maintain. Our research has shown that the correct application of representation in the `PRO` is far easier to maintain, since the need for strict truth maintenance is reduced. In addition, the use of a specialized reification mechanism can actually reduce the computational overhead, while increasing the functionality of the system.

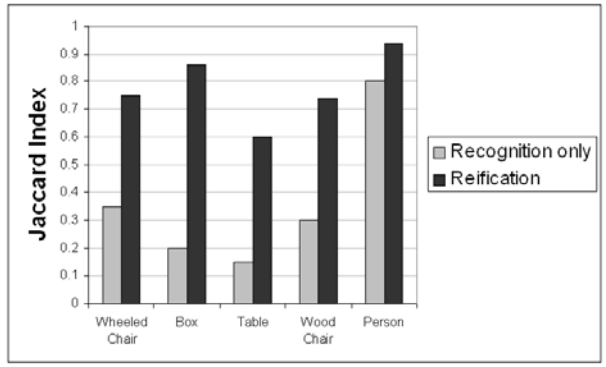


Figure 4: This chart compares the ability of the robot to classify objects using simple recognition versus a complete reification process. The full reification process first attempts to do a presumptive classification, based on the last known position and orientation of the object. If that fails, the recognition process is invoked. The quality of the classification is shown as the Jaccard Index between the classification set and the actual state.

In our tests the integration of the `PRO` and `Reification Engine` into a deployed mobile robot was shown to reduce the errors in classifying objects by 60%, and to reduce localization errors from 200mm to an average of 13mm (More comprehesive test data can be found in [10], chapter 12).

In Figure 4 we show the relative improvement provided by complete reification over simple pattern recognition alone. These comparison tests were done using exactly the same `PerCepts`, so there was no additional structural information available to the cybernetic brain. The only additional information was that the brain had a `Mental Model` that included the last position of the objects it knew about and a `PRO` that linked the `PerCept` with the symbolic knowledge of the object. Since it knew where the objects were the last time it looked, it was able to generate a preafference image - the way the world was expected to look. Using this expectation, the robot could quickly and accurately assess whether the world was generally as it was expected to be. These results were generated with small perturbations of the actual position of the objects away from their last known position. In spite of these perturbations, the preafference image enabled the `Reification Engine` to increase its ability to both model the world, and to maintain that model. This clearly shows the benefits provided by reification in robotic systems.

5. REFERENCES

[1] J. Ayers and J. Witting. Biomimetic approach to the control of underwater walking machines. *Philosophical Transactions of the Royal Society A*, 365:273–295, 2007.

[2] R. A. Brooks. Intelligence without reason. In J. Myopoulos and R. Reiter, editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 569–595, Sydney, Australia, 1991. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.

[3] R. A. Brooks. Intelligence without representation (revised). In *Mind Design II*. The MIT Press, Cambridge, MA, 2000.

[4] E. Brunswik. Representative design and probabilistic theory in a functional psychology. *Psychological Review*, 62:192–217, 1955.

[5] U. N. N. U. W. Center. Uuv, aus, and usv technology and assessment. Broad Area Announcement 01-01, 2001. http://www.npt.nuwc.navy.mil/contract/announce/baa/2001-01/#TOPIC13.

[6] J. F. Engelberger. *Robotics in Practice: Management and Applications of Industrial Robots*. Kogan Page, London, UK, 1980.

[7] G. W. Ernst and A. Newell. *GPS: A Case Study in Generality and Problem Solving*. Academic Press, London, UK, 1969.

[8] W. J. Freeman. The physiology of perception. *Scientific American*, 242:78, February 1991.

[9] J. P. Gunderson and L. F. Gunderson. Reification: What is it and why should i care? In *Proceedings of the 6th Workshop on Performance Metrics for Intelligent Systems*, page unknown, Gaithersburg, MD, 2006. National Institutes of Standards and Technology.

[10] L. F. Gunderson and J. P. Gunderson. *Robots, Reasoning, and Reification*. Springer, in press.

[11] J. Hollingum. Unmanned vehicles go to war. *Industrial Robot*, 25(6), 1998.

[12] D. B. Marco, A. J. Healey, and R. B. McGhee. Autonomous underwater vehicles: hybrid control of mission and motion. *Autonomous Robots*, 3(2–3), 1996.

[13] A. Newell and H. Simon. Gps, a program that simulates human thought. In E. A. Feigenbaum and J. Feldman, editors, *Computers and Thought*. R. Oldenbourg KG., 1963.

[14] M. Pai. Latest tornadoes bring destruction; red cross provides care, May 2008.

[15] L. E. Parker and J. V. Draper. Robotics applications in maintenance and repair. In S. Nof, editor, *Handbook of Industrial Robotics*, chapter 53. Wiley, New York, 1999.

[16] J. Pettre, P. de Heras, J. Maim, B. Yersin, J.-P. Laumond, and D. Thalmann. Real-time navigating crowds: Scalable simulation and rendering. *Computer Animation and Virtual World (CAVW) Journal - CASA 2006 special issue*, 2006.

[17] C. M. Portas, B. A. Strange, K. J. Friston, D. R. J., and C. D. Frith. How does the brain sustain a visual percept. *Proceedings of the Royal Society of London, B*, 267:845–850, 2000.

[18] E. D. Sacerdoti. Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, 5:115–135, 1974.

[19] A. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal*, 3(3):210Ű–229, 1959.

[20] H. C. Schubert and J. P. How. Space construction: an experimental testbed to develop enabling technologies. In *Proc. of the SPIE*, volume 3206, 1997.

[21] A. K. Singh. *Field Theory*, chapter 14, pages 390–410. Motilal Banarsidass Publishers, Delhi, India, 1991.

[22] T. Tsuda, D. Kato, A. Ishikawa, and S. Inoue. Automatic tracking sensor camera system. In M. A. Hunt, editor, *Machine Vision Applications in Industrial Inspection IX*. SPIE–The International Society for Optical Engineering, 2001.

[23] I. Walker, Hoover, A., and Y. Liu. Handling unpredicted motion in industrial robot workcells using sensor networks. *Industrial Robot: An International Journal*, 33:56–59(4), 2006.

[24] F. Werblin and B. Roska. The movies in our eyes. *Scientific American*, pages 72–79, April 2007.

Quantification of Line Tracking Solutions for Automotive Applications

Jane Shi, Ph.D
GM R&D Center
30500 Mound Road
Warren, MI, USA
Jane.Shi@gm.com

Rick F. Rourke
Dave Groll
Peter W. Tavora
GM Manufacturing Engineering
30300 Mound Road
Warren, MI, USA
Rick.f.Rourke@gm.com
Dave.Groll@gm.com
Peter.Tavora@gm.com

ABSTRACT
Unlike line tracking in automotive painting applications, line tracking for automotive general assembly applications requires position tracking in order to perform assembly operations to a required assembly tolerance. Line tracking quantification experiments have been conducted for a total of 16 test cases for two line tracking scenarios with three types of line tracking solutions: encoder based tracking, encoder plus static vision based tracking, and the analog sensor-based tracking for general assembly robotic automation. This paper presents the quantification results and identifies two key performance drivers fro line tracking for automotive assembly applications.

Categories and Subject Descriptors
C.4 [**Performance of Systems**]: Performance attributes.
C.3 [**Special-Purpose and Application-Based Systems**]: Process control systems; Real-time and embedded systems.

General Terms
Performance, Experimentation, Verification.

Keywords
Position tracking performance.

1. INTRODUCTION
In automotive general assembly (GA), vehicle bodies are being carried on a continuous moving assembly line through hundreds of workstations where a variety of parts are being assembled together as shown in Figure 1.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

Figure 1 Automotive General Assembly Conveyors Transport Vehicle Bodies for Assembly Operations

In order to robotically assembly parts onto a continuous moving vehicle body, the performance of current available line tracking solutions for automotive general assembly applications should be well understood. For automotive general assembly applications, the vehicle body position at the point of assembly operations on a moving conveyor needs to be tracked by encoders or other sensors for the duration of the part assembly. This means that a specific position is to be tracked for a fixed time period with a required position tolerance.

In the past several months, a collaborative study has been conducted to thoroughly quantify the performance of three types of robotic line tracking solutions, encoder-based, encoder plus static vision, analog laser sensor-based. This paper describes the quantification methodology, illustrates the experimental setup, summarizes the quantification results, and identifies two key performance drivers fro line tracking for automotive assembly applications.

2. QUANTIFYING LINE TRACKING SOLUTIONS

The operating principal of the robotic line tracking is for a robot to use a sensor's outputs to track a moving part as shown in Figure 2.

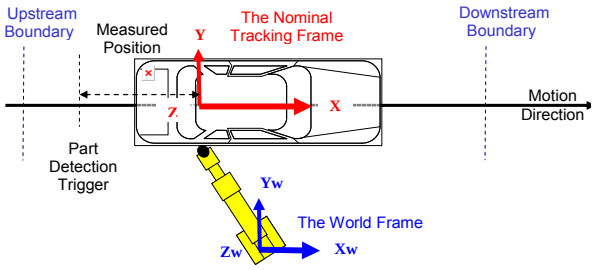


Figure 2 Robotic Line Tracking

A world frame is specified for the robot location. A nominal tracking frame is set to be rigidly attached to the moving part. Once part detection is triggered, a sensor, such as an encoder, measures the part position and robot uses the sensor's reading to track the moving part.

Several key technical issues need to be addressed when quantifying the performance of a line tracking solution:

- How to measure the part position with respect to time?
- How to measure the robot position with respect to time?
- How to synchronize above two position measurements?
- Which case of part movement should be evaluated?
- How many of part movement cases should be evaluated?

The relative position between the robot and the moving part is critical to the success of the line tracking application when robot is in a tracking mode. It is valid to combine the issue (1) and (2) to measure the relative position between the robot and the moving part. This means that only one relative position needs to be evaluated with respect to time. In the actual evaluation experiments described in the next section, the relative position is measured with Dynalog's CompuGauge. At the same time, both part position and the robot position in the tracking mode are recorded and then relative position is computed with synchronized trigger. It has been determined that the computed robot position data is accurate enough for the evaluation purpose.

Application environment plays a critical role in deciding which and how many evaluation test cases for the part movement. Several key relationships we would like to determine:

- Does the tracked position change for the same constant moving speed reached by different acceleration?
- Does the tracked position change for different constant moving speeds?
- How does the tracked position change for an unstable moving speed?

A total of sixteen evaluation test cases are designed to cover three classes of motion profile including a variety of accelerations, an emergency stop case, and an unstable conveyor bounce motion as detailed in a later section.

3. Experimental Setup and Performance Data Collection Method

In the experimental setup for the line tracking evaluation, the moving vehicle body is emulated by a robot on one dimensional rail as illustrated by Figure 3.

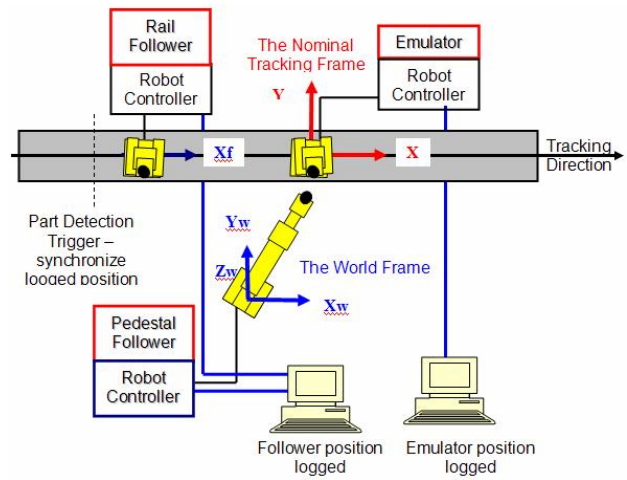


Figure 3 Quantification Experimental Setup and Data Collection Method

Three main components of the experimental setup are as followings:

- **Emulator**: it is emulating a moving vehicle body on the assembly plant floor. A variety of motion profiles have been implemented by the Emulator as summarized in next section.
- **Rail Follower**: it is a robot transport unit (RTU) that utilizes one linear rail to follow the Emulator's motion.
- **Pedestal Follower**: it is a stationary six axes robot that utilizes its arm to follow the Emulator's motion.

Two types of tracking solutions have been evaluated: the rail tracking and the arm tracking:

A. The rail tracking: the Rail Follower is tracking the Emulator with one linear rail motion.
B. The arm tracking: the Pedestal Follower is tracking the Emulator with its 6DOF arm motion.

Both Emulator and Follower's positional data are collected using a position recording software provided by the robot controller. The collected data are then calculated to derive the relative position between the Follower and the Emulator. The computed relative position was validated with Dynalog's CompuGauge at the start of project. It has been determined that the computed relative position data is accurate to be within 5%. For all evaluation tests and their associated data graph, the recorded position data by the Follower and the Emulator are analyzed.

4. QUANTIFICATION TEST CASES

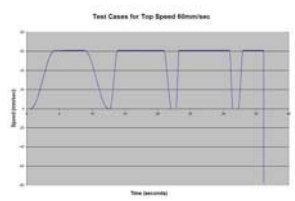
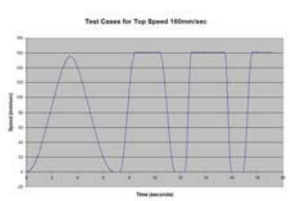
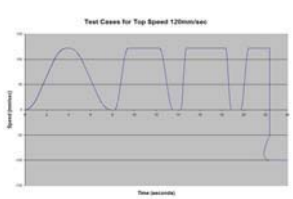
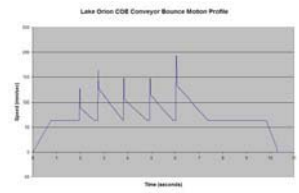
Three classes of vehicle motion profiles have been emulated by the Emulator for the quantification tests:

- Stable motion with a variety of accelerations
- E-Stop
- Actual measured conveyor bounce motion [1]

A total of 16 test cases are designed and conducted for both the rail tracking and the arm tracking scenarios. For high volume assembly plants, three line rates are used as representative cases: 40, 60, and 80 in Jobs Per Hour (JPH) which translates to conveyor speed in mm per second (mm/sec) The test cases are organized for three stable constant speeds (60mm/sec, 120mm/sec, and 160mm/sec) with four different acceleration profile (rising edge of the speed chart) plus an emergency stop cases (acceleration with very high negative number). In addition, one special test case is included that is based on the actual measured conveyor bounce motion.

Table 1 summarizes all test cases with a speed profile graph that illustrates the constant top speed and four accelerations and one emergency stop deceleration.

Table 1 Evaluation Test Cases

Line Rate (JPH) Conveyor Constant Speed (mm/sec)	Acceleration (mm/sec^2)	Speed Profile (mm/sec vs. second)
40 JPH 60 mm/sec	150	
	320	
	650	
	745	
	-11190	
60 JPH 120 mm/sec	150	
	320	
	650	
	745	
	-11190	
80 JPH 160 mm/sec	150	
	320	
	650	
	745	
	-11190	
40 JPH 60 mm/sec	Bounce with various magnitude	

5. QUANTIFICATION RESULTS OF THE ENCODER BASED LINE TRACKING SOLUTION

For the encoder based line tracking, the encoder is connected to the robot controller that drives the Follower as illustrated in Figure 4.

A set of gear boxes drive the encoder while the Emulator is

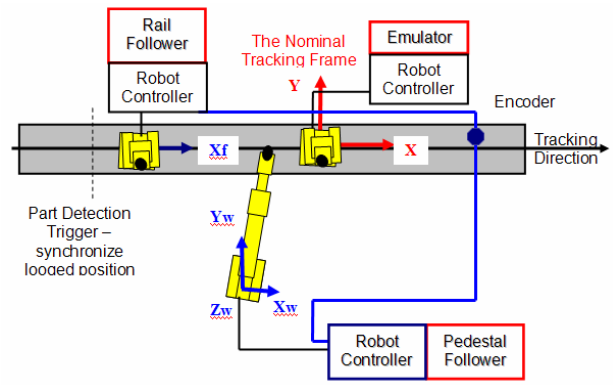


Figure 4 Encoder Connects to the Robot Controller of the Rail or Pedestal Follower

moving per the specification of the speed profile in each test case. With a properly selected gear ratio, the resolution for the encoder is set to be at least 500 counts per mm. This represents minimum resolution of 0.002mm per count. The encoder digital counts are applied to the Follower in both rail and arm tracking cases every 0.012 seconds.

A digital I/O handshaking signal between the Emulator and the Follower simulates the part detection trigger to start the line tracking. In the data collection process, the trigger signal is emulated by a digital handshaking between the Emulator and Follower to start recording robot position data at the same time. Thus the relative position between the Emulator and the Follower can be computed during the data analysis.

5.1 Rail Tracking Results

All test cases are conducted for the encoder-based rail tracking and repeated at least three times to validate the repeatability. The position lag errors are plotted in Figure 5 through 7 for the constant conveyor motion cases.

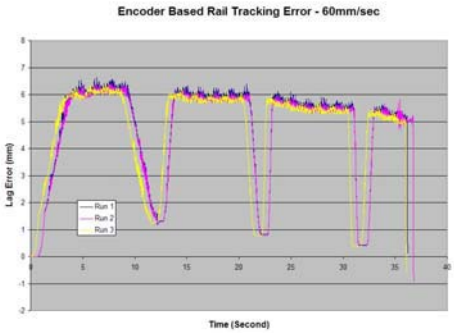


Figure 5 The Position Lag Error for the Encoder Based Rail Tracking: 60mm/sec Case

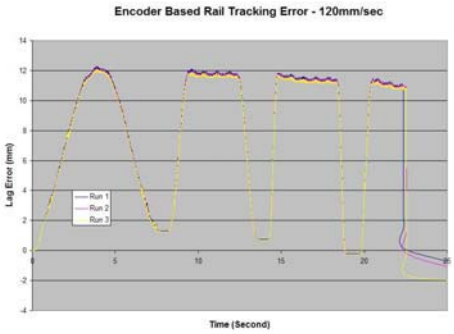


Figure 6 The Position Lag Error for the Encoder Based Rail Tracking: 120mm/sec Case

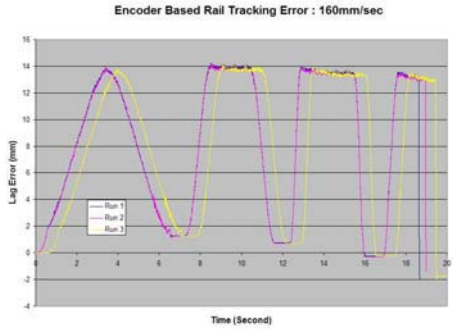


Figure 7 The Position Lag Error for the Encoder Based Rail Tracking: 160mm/sec Case

For the constant motion cases, several observations can be made of encoder based rail tracking results:

- Lag error is proportional, although not exactly linearly, to the stable top speed.
- Higher acceleration does not increase the total lag error: it seems to have reduced the total lag error slightly (about 1mm) for the rail tracking.
- And all test cases are repeatable within 1mm.

For the emulated conveyor bounce motion case, the position lag error is plotted in Figure 8.

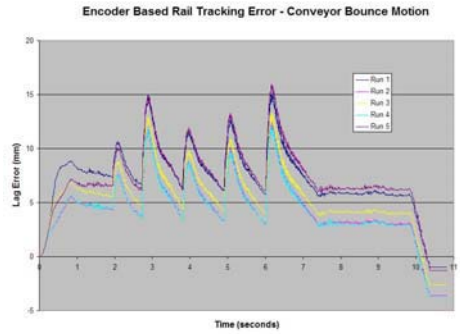


Figure 8 The Position Lag Error for the Encoder Based Rail Tracking: Emulated Conveyor Bounce Case

Table 2 below summarizes the results of the encoder based rail tracking test cases.

Table 2 Encoder (0.002mm Resolution) Based Rail Tracking Performance

Line Rate (JPH) Conveyor Constant Speed (mm/sec)	Acceleration (mm/sec^2)	Range of Position Tracking Error (mm)	Repeatability (mm)
40 JPH 60 mm/sec	150	< 6.8mm	< 0.5mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot - 1.2mm	< 0.5mm
60 JPH 120 mm/sec	150	< 12.5mm	< 0.5mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot - 2.1mm	< 0.5mm
80 JPH 160 mm/sec	150	< 14.5mm	< 0.5mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot - 2.1mm	< 0.5mm

40 JPH 60 mm/sec	Bounce with various magnitude	-3.8mm ~ 16.0mm	< 4.0mm

5.2 Arm Tracking Results

All test cases are conducted for the encoder-based arm tracking and repeated at least three times to validate the repeatability. The position lag errors are plotted in Figure 9 through 11 for the constant conveyor motion cases.

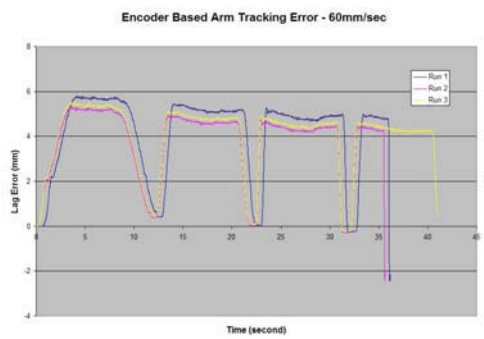


Figure 9 The Position Lag Error for the Encoder Based Arm Tracking: 60mm/sec Case

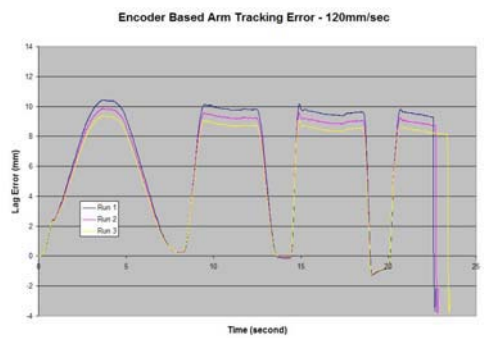


Figure 10 The Position Lag Error for the Encoder Based Arm Tracking: 120mm/sec Case

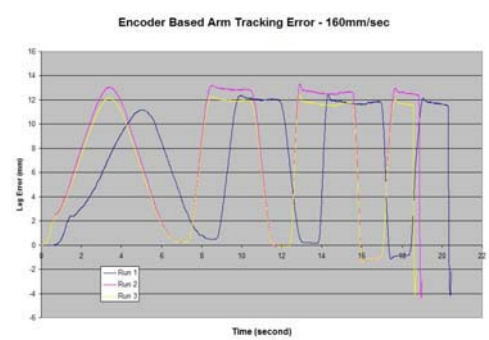


Figure 11 The Position Lag Error for the Encoder Based Arm Tracking: 160mm/sec Case

For the constant motion cases, similar observations can be made of encoder based arm tracking results:

- Lag error is proportional, although not exactly linearly, to the stable speed.
- Higher acceleration does not increase the total lag error: it seems to have reduced the lag error slightly (about 1mm) for the arm tracking.
- And all test cases are repeatable within 2mm.

For the emulated conveyor bounce motion case, the position lag error is plotted in Figure 12. The arm tracking error is within +16 and -3 mm with a repeatability of about 5 mm.

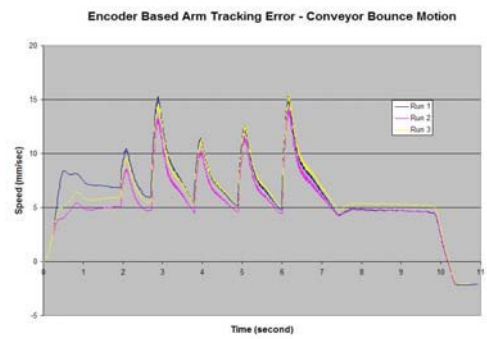


Figure 12 The Position Lag Error for the Encoder Based Arm Tracking: Emulated Conveyor Bounce Case

Table 3 below summarizes the results of the encoder based arm tracking test cases.

Table 3 Encoder (0.002mm Resolution) Based Arm Tracking Performance

Line Rate (JPH) Conveyor Constant Speed (mm/sec)	Acceleration (mm/sec^2)	Range of Position Tracking Error (mm)	Repeatability (mm)
40 JPH 60 mm/sec	150	< 6.0mm	< 1.0mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -2.5mm	< 1.0mm
60 JPH 120 mm/sec	150	< 11.5mm	< 1.5mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -4.0mm	< 1.0mm

80 JPH 160 mm/sec	150	< 13.5mm	< 2.0mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -4.5mm	< 1.0mm
40 JPH 60 mm/sec	Bounce with various magnitude	-3.0mm ~ 16.0mm	< 5.0mm

Compared with the encoder based rail tracking, the arm tracking has

- similar position lag error,
- slightly worse repeatability,
- and worse emergency stop performance.

6. QUANTIFICATION RESULTS OF THE ENCODER PLUS STATIC VISION LINE TRACKING SOLUTION

When a static vision is used for the line tracking, the Follower uses the detected object location to correct its position relative to the object. This one time correction of robot's position based on the vision results is "static" compared with the use of vision results at a fixed time period to track an object's position [2]. The encoder counts are the sensor inputs that drive the Follower to track the Emulator.

Table 4 lists the vision system setup used in the evaluation testing.

Table 4 Vision System Parameters

Camera Lens (mm)	Camera Standoff Distance (mm)	the Grid Object Dimension (mm x mm)	Pixel Resolution (mm/pixel)	Object Resolution (mm)
8	640	320 by 320	0.599	0.15

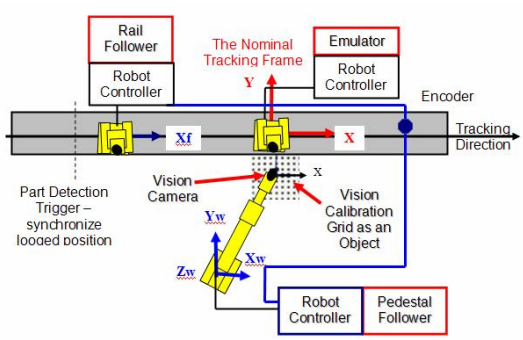


Figure 13 Setup of Vision System for Line Tracking Evaluation

Figure 13 illustrates the setup of the vision system used in the line tracking evaluation.
The vision camera is attached to the Pedestal Follower and the vision grid object is attached to the Emulator so that there is no relative motion when the Follower is tracking the Emulator at the constant speed.

The following procedures are implemented to quantify the static vision use in the encoder based line tracking:

(1) **Before** the Emulator starts to move, the Arm Follower is centered with the vision calibration grid so that the vision can use it to "set reference". This means that the vision snap and find operation will return a zero offset.
(2) Enter a static 2D offset, (x1, y1, R1), to the Emulator robot so that the robot arm moves the specified offset **before** the Emulator starts moving.
(3) **After** the Emulator reaches its **constant** speed, the camera mounted on the Pedestal Follower captures the image of the vision calibration grid. The second vision snap and find operation determines the vision offset (x2, y2, R2).
(4) Apply the vision offset (x2,y2,R2) from step 3 to the Pedestal Follower so that the Pedestal Follower compensates the offset value.
(5) Verify the applied vision offset: the camera mounted on the Follower captures the image of the vision calibration grid again. The third vision snap and find operation determines the offset (x3, y3, R3) that, in theory, should be zero if compensated correctly and perfectly.

Table 5 summarizes the test cases and the performance for the static vision with the encoder based line tracking.

Table 5 Static Vision (0.15mm Resolution) Arm Tracking Evaluation Results

Constant Speed (mm/sec)	Offset Applied			Offset Detected by Vision at Constant Speed				Vision Offset after Compensating Detected Offset		
	X1 (mm)	Y1 (mm)	R1 (deg)	X2 (mm)	X2+X1 (mm)	Y2 (mm)	R2 (deg)	X3 (mm)	Y3 (mm)	R3 (deg)
114	0	0	0	-8.4	-8.4	-0.9	0	-1.3	0	0
	0	0	0	-8.4	-8.4	-0.8	0	-1.2	0	0
	0	0	0	-8.4	-8.4	-0.8	0	-1.2	0	0
	-5	0	0	-3.3	-8.3	-0.9	0	-0.5	0.1	0
	5	0	0	-13.3	-8.3	0	0	-1.9	0	0
	-15	0	0	6.7	-8.3	-0.8	0	0.7	-0.1	0
	15	0	0	-23.3	-8.3	-1.1	0	-3.1	-0.1	0

	10	-15	3	-18.6	-8.6	13.9	3.1	-2.2	2.2	0
	10	-15	3.1	-18.6	-8.6	13.9	3.0	-2.2	2.2	0
	-10	15	-2	1.9	-8.1	-15.8	-2.0	-0.2	-2.4	0
	-10	15	-2	1.8	-8.2	-15.8	-2.0	-0.1	-2.3	0

The first three columns (X1,Y1,R1) are the static offset of the grid object applied by the Emulator in step (2). The columns (X2,Y2,R2) are the object position detected by the vision system in step (3). The last three columns (X3,Y3,R3) are the object position detected by the vision systems in step (5) after the correction has been applied in step (4).

The shaded column in Table 5, X2-X1, is the difference between the detected X offset and the applied X offset. The values in this column are the relative position lag detected by the vision when the Emulator is traveling at the steady state speed of 114mm/sec. This is this steady state position lag that is proportional to the constant speed of the Emulator. In this case (114mm/sec), it is averaged at 8.35mm with one standard derivation of 0.15mm.

Figure 14 below plots the position lag between the Emulator and the Pedestal Follower for the first seven rows in Table 5.

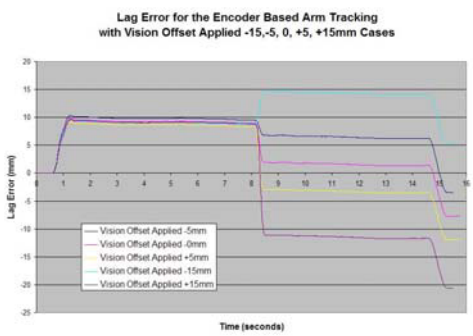


Figure 14 Position Correction is the Combination of Static Lag plus Initial Object Offset

Detailed analysis of the collected robot position data reveals that the corrected position is the combination of the static position lag (caused by the motion) and the initial object position offset. How accurately the robot can compensate the position is a different evaluation question that could be a separate investigation study. The rough indicator is the last three column (X3,Y3,R3) in Table 5.

It is worthwhile to note that the vision grid is a perfect object for the vision system to recognize and locate. In actual applications, all factors should be considered carefully to achieve the best result. It is critical to understand that the selection of the camera focal length, the camera standoff distance to the object, and the conveyor constant speed all impact the final performance of the vision use in the line tracking applications.

7. Quantification RESULTS of The Analog Laser Based Line Tracking Solution

An analog ranging laser can be used in the place of an encoder where mounting an coder is not feasible in the line tracking applications. The analog laser ranging sensor, ILD1700-100 [3], uses the optical triangulation principal to measure the distance between the laser sensor and the smooth surface target as shown in Figure 15. The measurement range is between 70mm the Start of Measuring Range (SMR) and 170mm the End of Measuring Range (EMR). The midrange (MMR) is used for the line tracking application. This means that the analog laser signal drives the Rail or Pedestal Follower to maintain a constant distance with the Emulator. Customized software has been developed to use the analog laser sensor for the line tracking function.

Figure 16 illustrates the setup for the analog laser based line

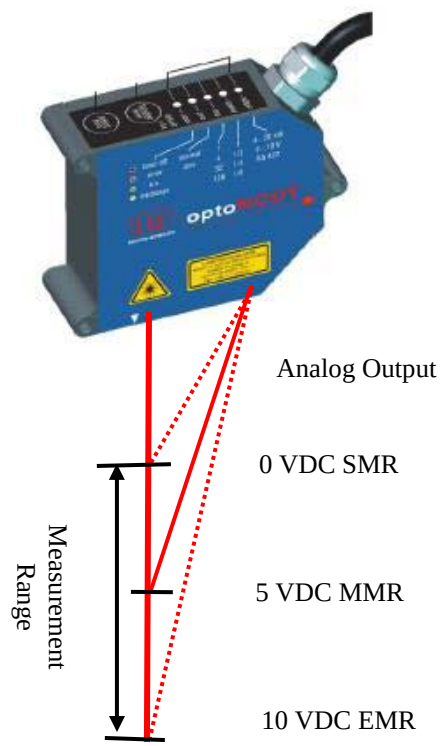


Figure 15 An Analog Ranging Laser Sensor ILD1700-100

tracking. While the Emulator is moving, the analog laser measures the distance between the Emulator and the Follower. This analog laser sensor based line tracking method is only concerned with one dimension: the conveyor main travel direction. It does not require the sensor to recognize any object geometry feature as in a traditional tracking method. Any flat reflecting surface can be used for the laser signal to return to its receiver. The goal is to keep the laser reading constant by driving the Follower to track the Emulator's position.

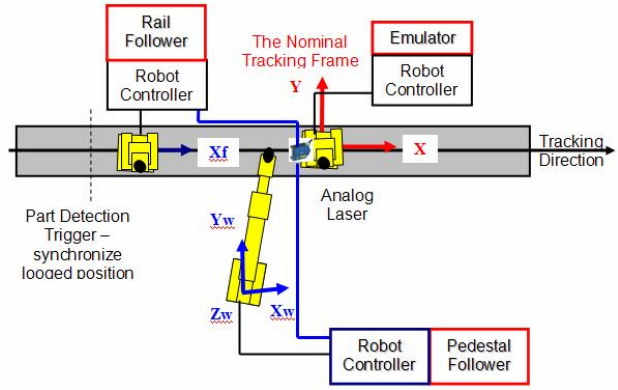


Figure 16 Analog Laser Based Line Tracking Solution

In order to use the analog laser sensor to track the Emulator, additional system components have to be set up. The analog signal is first digitized by an analog to digital (A/D) converter. Then the digital signal goes through a DeviceNet adaptor to interface with the DeviceNet module inside the robot controller as shown in Figure 17 below.

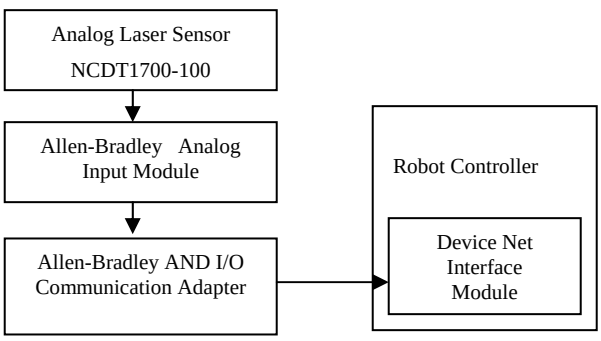


Figure 17 Analog Laser Ranging Sensor Connection to a Robot Controller

A scan rate can be set for the device net interface module at 12msec. This means that the digitized analog sensor can be read by the robot controller every 12msec. The line tracking software then utilizes the analog sensor inputs every scan period. There is an unknown system time delay between the sampled sensor reading and the use of such reading by the robot controller to track the Emulator.

The analog laser sensor has measurement resolution of 0.0006mm. Experiments have shown that the stable reading of the analog laser is around 0.0012mm when the analog sensor and the measured surface are both stationary.

7.1 Rail Tracking Results

All test cases are conducted for the analog sensor based rail tracking and repeated at least three times for validating the repeatability. The position lag errors are plotted in Figure 18 through 20 for the constant conveyor motion cases.

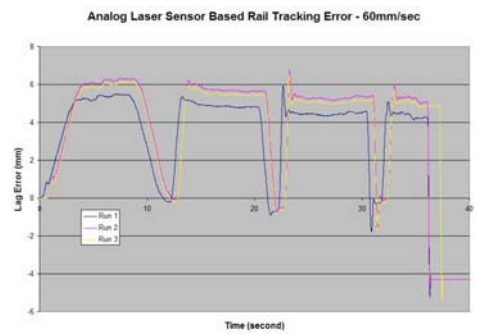


Figure 18 The Position Lag Error for Analog Sensor Based Rail Tracking: 60mm/sec Case

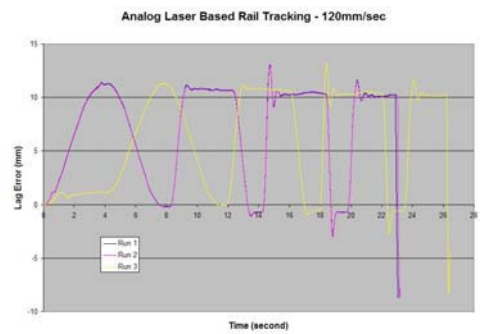


Figure 19 The Position Lag Error for Analog Sensor Based Rail Tracking: 120mm/sec Case

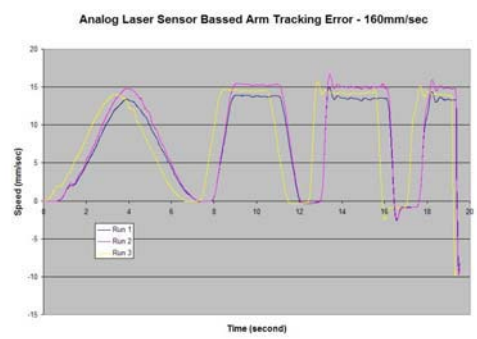


Figure 20 The Position Lag Error for Analog Sensor Based Rail Tracking: 120mm/sec Case

For the constant motion cases, several observations can be made of analog sensor based rail tracking results:

- Lag error is proportional to the top stable speed.
- Higher acceleration causes significant overshoot before the Rail Follower can stabilize its speed to match the Emulator speed as illustrated in Figure 21.

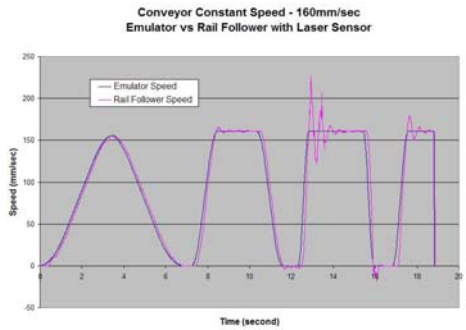


Figure 21 The Rail Follower Overshoots Before Stabilizes in High Acceleration Case

For the emulated conveyor bounce motion case, the position lag error is plotted in Figure 22.

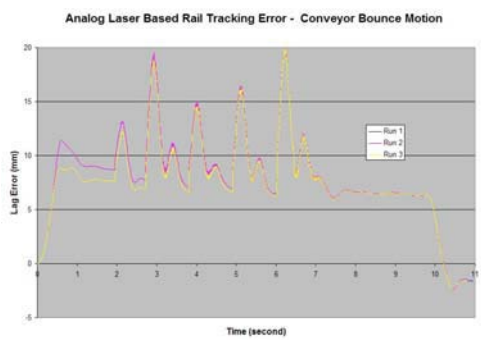


Figure 22 The Position Lag Error for the Analog Sensor Based Rail Tracking: Emulated Conveyor Bounce Case

Compared with the encoder based rail tracking result shown in Figure 8, The analog laser based rail tracking exhibits significant overshoot and oscillation.. Further examination of the Rail Follower speed identifies the root cause of the large position lag error as shown by Figure 23. With the analog sensor's inputs, the Rail Follower can not closely follow the Emulator speed.

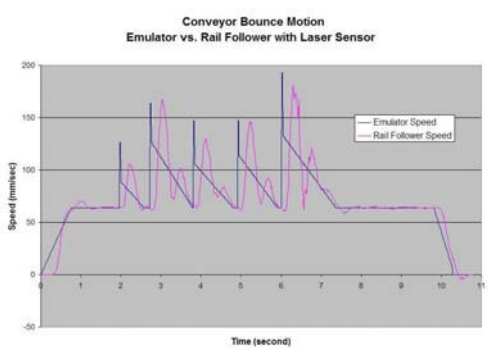


Figure 23 The Emulator Speed vs. The Rail Follower Speed with the Analog Laser

Table 6 below summarizes the results of the analog laser rail tracking test cases.

Table 6 Analog Laser (0.0012mm Resolution) Based Rail Tracking Performance

Line Rate (JPH) Conveyor Constant Speed (mm/sec)	Acceleration (mm/sec^2)	Range of Position Tracking Error (mm)	Repeatability (mm)
40 JPH 60 mm/sec	150	< 7.0mm	< 2.5mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -5.5mm	< 1.0mm
60 JPH 120 mm/sec	150	< 14.0mm	< 2.0mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -8.0mm	<1.0mm
80 JPH 160 mm/sec	150	< 18.5mm	< 4.0mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -12.0mm	< 2. 0mm
40 JPH 60 mm/sec	Bounce with various magnitude	-3.0mm ~ 21.0mm	< 5.0mm

Compared with the encoder based rail tracking of Table 2, the rail tracking with the analog laser has

- larger position lag error for constant motion speeds,
- worse repeatability,
- significantly worse emergency stop performance,
- and significantly larger position lag error for the dynamic motion speed.

7.2 Arm Tracking Results

All test cases are conducted for the analog laser based arm tracking and repeated at least three times to validate the repeatability. The position lag errors are plotted in Figure 24 through 26 for the constant conveyor motion cases.

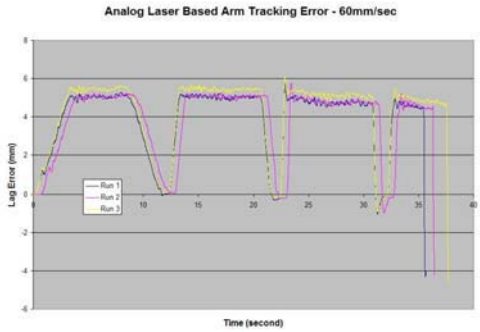


Figure 24 The Position Lag Error for the Analog Sensor Based Arm Tracking: 60mm/sec Case

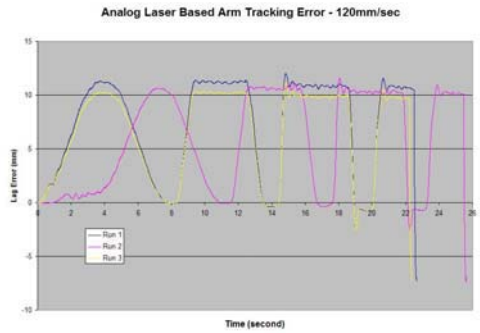


Figure 25 The Position Lag Error for the Analog Sensor Based Arm Tracking: 120mm/sec Case

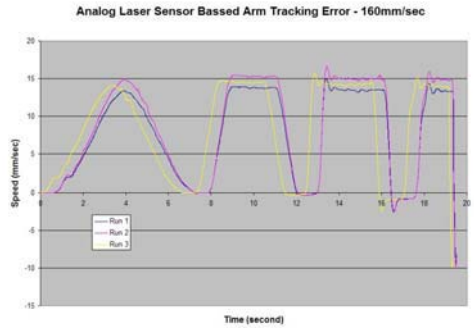


Figure 26 The Position Lag Error for the Analog Sensor Based Arm Tracking: 160mm/sec Case

For the constant motion cases, similar observations can be made of encoder based arm tracking results:

- Lag error is proportional to the top stable speed.
- Higher acceleration causes significant overshoot before the Arm Follower can stabilize its speed to match the Emulator speed as illustrated in Figure 27.

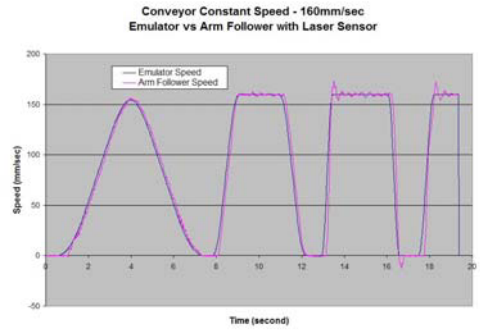


Figure 27 The Emulated Speed vs. The Arm Follower Speed with the Analog Laser Sensor

For the emulated conveyor bounce motion case, the position lag error is plotted in Figure 28.

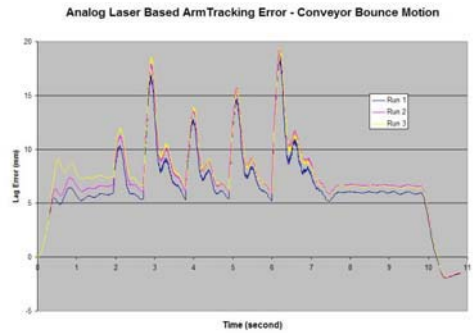


Figure 28 The Position Lag Error for the Analog Laser Arm Tracking: Emulated Conveyor Bounce Case

Compared with the encoder based arm tracking results shown in Figure 12, the analog laser based arm tracking exhibits significant overshoot and oscillation.. Further examination of the Arm Follower speed identifies the root cause of the large position lag error as shown by Figure 29 below. With the analog sensor's inputs, the Arm Follower can not closely follow the Emulator speed.

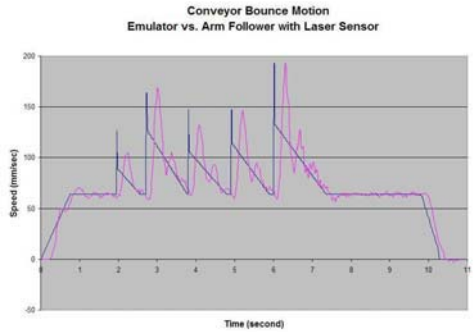


Figure 29 The Emulator Speed vs. The Arm Follower Speed with the Analog Sensor

Table 7 below summarizes the results of the analog laser sensor based arm tracking test cases.

Table 7 Analog Laser (0.0012mm Resolution) Based Arm Tracking Performance

Line Rate (JPH) Conveyor Constant Speed (mm/sec)	Acceleration (mm/sec^2)	Range of Position Tracking Error (mm)	Repeatability (mm)
40 JPH 60 mm/sec	150	< 6.5mm	< 1.0mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -5.0mm	< 1.0mm
60 JPH 120 mm/sec	150	< 12.0mm	< 2.0mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -7.5mm	< 1.0mm
80 JPH 160 mm/sec	150	< 17.0mm	< 2.0mm
	320		
	650		
	745		
	-11190 (emergency stop)	Overshoot -10.0mm	< 1.0mm
40 JPH 60 mm/sec	Bounce with various magnitude	-2.0mm ~ 19.0mm	< 6.0mm

Compared with the encoder based arm tracking of Table 3, the arm tracking with the analog laser has

- larger position lag error for constant motion speeds,
- worse repeatability,
- significantly worse emergency stop performance,
- and significantly larger position lag error for the dynamic motion speed.

8. CONCLUSIONS

Unlike the line tracking for automotive paint applications where the speed match between the robot and the vehicle body plays critical role for the paint quality, the line tracking for automotive general assembly applications requires position tracking of the system to perform assembly operations within a required assembly tolerance for a period of time. By comparing and contrasting experimental results, key performance drivers for the robotic line tracking can be concluded. Among three line tracking solutions, the encoder-based tracking method offers the tightest and most responsive robotic tracking performance. The encoder plus static vision based tracking method can be used for compensating the steady-state tracking position lag and the initial object offsets. The analog sensor-based tracking method is the least responsive due to its delayed signal propagation in the entire system.

Two key drivers have been identified for impacting the line position tracking performance:

- Sensor resolution: it limits how tight the robot system can track the conveyor line motion.
- Actual time delay of applying the sensor data: it limits how dynamically responsive the robot system can respond to the change in the conveyor line motion.

By understanding current capabilities of line tracking solutions, appropriate robotic automation systems can be developed and designed for a variety of GA applications. Key fundamental system drivers can be specified for robots to track the moving conveyor accurately enough for the specific assembly tolerance plus the environment uncertainty with adequate dynamic response at the appropriate location for the general assembly automation.

9. ACKNOWLEDGMENTS

The authors would like to acknowledge Bob Scheuerman, Marty Linn, Marjorie Winston, Thomas McGraw, Ke Zhang-Miske, Brooks A. Curtis, Ronald Reardon, Loring Dohm, Michael Poma, Neil McKay, Charles W. Wampler, and Roland Menassa. Their inputs, support, contribution, and knowledge have shaped this project from its initial concept to the final results.

10. REFERENCES

[1] J. Shi, "Preliminary Analysis of Conveyor Dynamic Motion for Automation Applications", PreMIS'08, 2008

[2] G. DeSouza, "A Subsumptive, Hierarchical, and Distributed Vision-based Architecture for Smart Robotics", Ph.D. Dissertation, School of Electrical and Computer Engineering, Purdue University, May 2002

[3] Analog Laser Ranging Sensor ILD1700 www.micro-epsilon.com

Mobile Robotic Surveying Performance for Planetary Surface Site Characterization

Edward Tunstel
Space Department
Johns Hopkins University
Applied Physics Laboratory
Laurel, MD 20723 USA
Edward.Tunstel@jhuapl.edu

ABSTRACT
Robotic systems will perform mobile surveys for scientific and engineering purposes as part of future missions on lunar and planetary surfaces. With site characterization as a task objective various system configurations and surveying techniques are possible. This paper describes several examples of mobile surveying approaches using local and remote sensing configurations. A geometric measure of area coverage performance is applied to each and relative performance in surveying a common area is characterized by expected performance trends. Performance metrics that solely express geometric aspects of the task are limited in utility as decision aids to mission operators. As such, the importance of enriching such metrics by incorporating additional attributes germane to surveying on planetary surfaces is highlighted.

Keywords
robotic surveying, planetary surface exploration, in-situ remote sensing, area coverage performance, site characterization

1. INTRODUCTION
Task-oriented algorithms that support systematic mobile surveys using science instruments are needed for planetary surface characterization on science missions. They are also needed for in-situ resource mapping on robotic missions that serve as precursors to human exploration missions. Typical objectives of site surveys include sensor coverage of designated areas. Area coverage problems for mobile robotic survey systems commonly employ sensing devices requiring close proximity to or contact with the measured phenomenon. Examples of such "local sensing" devices include ground penetrating radar, metal detectors for humanitarian de-mining, fluorescence imagers for organic molecule detection, a variety of spectrometer types, etc. Mobile robotic vehicles, or rovers, that carry survey systems comprised of local sensing devices must physically cover most, if not all, of the terrain in the designated survey area.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

Remote sensor-based area coverage contrasts with these more common area coverage problems for mobile robotic surveys. Remote sensing instruments (e.g., based on radar or optical devices such as lasers) can acquire measurements at significant distances away from the measured phenomenon. Measurements along a line-of-sight to detect airborne phenomena such as surface gas emissions, for example, account for coverage of terrain below that line-of-sight. This permits a two-dimensional search over terrain using discrete linear measurements from a distance (similar to scanning laser rangefinders).

Remote sensor-based methods are not applicable to all surveying tasks. For surveys in which they are not a better solution, they are often excellent complements to local sensor-based methods. That is, remote sensor-based surveys can serve as an efficient means to cover wide areas with the purpose of localizing smaller areas at which local sensor-based surveys of higher resolution are appropriate.

Mobility algorithms for surveying provide a means to systematically acquire measurements covering an area to be surveyed by transporting sensors and instruments to multiple locations and vantage points. Algorithms employing parallel-line transects or parallel swaths are commonly used to address robotic area coverage problems by single robots [1, 2] and multiple robots [3] when using local sensing devices. Full and partial coverage planners have also been proposed for rovers that survey terrain using local sensing devices [4]. Remote sensor-based survey approaches for rovers have recently been proposed for single- and two-rover systems performing measurements through the near-surface atmosphere [5, 6].

This paper examines examples of both mobile survey types and applies a geometric measure of their area coverage performance. It further suggests additional attributes germane to surveying tasks for planetary surface exploration. The additional attributes are intended to enrich the effectiveness and relevance of basic geometric measures or support formulation of new metrics for planetary surface domains.

2. LOCAL SENSOR-BASED SURVEYING
Due to required proximity to measured phenomena and relatively small footprints of sensitivity, local sensing devices typically necessitate dense coverage of a designated *survey region* by the host mobile platform. As such, local sensor-based mobile surveys seek to acquire measurements that cumulatively cover all or most of the survey region. Associated survey sensors or instruments are typically mounted on a rover body or deployed on a rover-

attached boom or manipulator arm. Rover mobility serves to transport the footprint of the survey instrument(s) over terrain along trajectories that fill the survey region. Fig. 1 depicts this scenario.

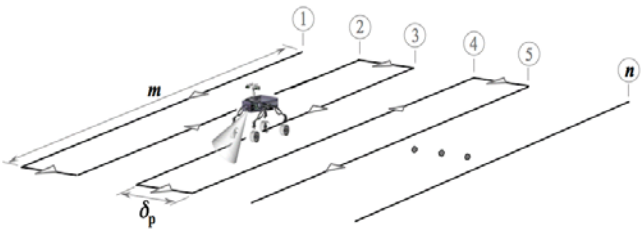


Figure 1. Local sensor-based surveying along *n* parallel transects of length *m*.

Among alternative survey trajectories for the mobile platform, parallel transects, spirals, and random walks have been proposed. Parallel transects are most commonly applied for coverage tasks although a spiral coverage approach was proposed for mobile mining of regolith on the lunar surface as part of a so-called spiral mining system [7]. Recent field tests, focused on planetary surface site characterization, used a rover operating ground penetrating radar (GPR) along parallel transects covering a 700 m x 700 m survey region [8]. This instrument was used to map the subsurface structure at the site. A total traverse distance of 20.5 km was sufficient to cover the survey region using densely spaced parallel transects (with no overlap of the GPR sensor footprint on adjacent transects) [8].

Basic geometric measures are often used to measure area coverage performance of such local sensor-based survey algorithms. Examples include measures of distance traveled and percent of total area covered [2]. Variants of the latter have been proposed based on distribution of measurement samples within the cells of a tessellated grid representation of the survey region [4]. Here, we apply a basic geometric measure comprising a combination of such attributes. It is referred to as the *quality of performance, QoP*, defined as a ratio of area covered to distance traveled [9]. Applying this metric to the recent field test result mentioned above would yield a QoP of 24 based on the survey region area and total traverse distance. In general, the QoP for a local sensor-based survey along parallel transects (Fig. 1) is computed as

$$Q_p = \frac{m(n-1)\delta_p}{[mn + (n-1)\delta_p]} \tag{1}$$

where m is the transect length, n is the number of transects traversed, and δ_p is the separation distance between adjacent transects and is assumed here to be comparable to the survey sensor footprint. The numerator and denominator of Eq. (1) respectively represent the area surveyed and total distance traversed during the survey.

In the next section we discuss mobile surveying using in-situ remote sensing and apply the same metric to area coverage performance of several types of remote sensor-based survey trajectories.

3. REMOTE SENSOR-BASED SURVEYING

Mobile remote sensor-based surveys can be performed by measurement systems whose components are separated by a distance across terrain. Such systems are comprised of an active/passive instrument component on one end and a passive/active component on the other end. One end could be stationary while the other is mobile (fixed-mobile) or both ends could be mobile (mobile-mobile). Both are considered here beginning with a fixed-mobile configuration, which is suitable for single-site surveys (unless the fixed component is also transportable to other sites).

3.1 Single-Site Remote Sensing Surveys

Consider a fixed-mobile configuration comprised of an active rover-mounted instrument, a passive receiver or retroreflector at a fixed location a distance away, a rover pan-tilt unit to point the instrument at the passive component for measurements, and the rover itself to move the instrument spatially over terrain. The passive component would remain stationary at a position central to a designated survey region. This configuration is similar to those of Total Station systems commonly used by civil engineers for land surveys and comprised of a theodolite on one end and stationary 360° retroreflector on the other. Like a civil engineer, a rover using such a survey system can acquire measurements from any radial direction when the fixed component is within line-of-sight and measurement range.

For mobile surveys, measurements are coordinated with rover mobility to survey terrain via a series of measurements across a distance *d*, which varies with rover position relative to the fixed component (Fig. 2). Such mobile robotic systems are under development for planetary surface surveying to achieve optical measurements at maximum distances of hundreds of meters [10]. The long-range measurement capability coupled with rover mobility enables wide-area surveys.

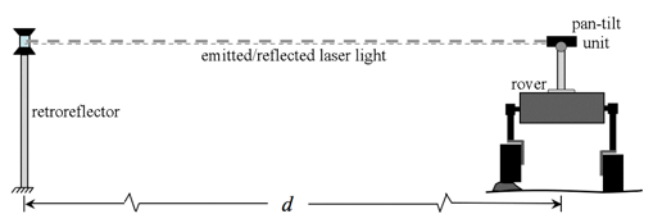


Figure 2. Example of a distributed fixed-mobile measurement configuration.

Concentric circular or spiral trajectories are compatible with distributed fixed-mobile configurations for remote sensor-based surveying. A designated survey region with a fixed instrument component at its center can be covered by traversing concentric circular trajectories as follows. The location of the fixed component is known and considered to be the origin of an inertial coordinate system in which the survey region and task is defined. Rover pose during surveys is estimated relative to this coordinate system. Beginning at a designated radial distance from the fixed component, the rover moves in arc-increments stopping periodically on the trajectory to acquire measurements. We refer to these measurement locations as *m-nodes*. Measurements along a line-of-sight between the rover-mounted active instrument and the fixed component account for 2-D coverage of terrain below the line-of-sight. Such measurement techniques are used on Earth

with laser-based spectrometers to probe for and detect gas emissions during environmental site surveys [11], and they are being developed for the same on Mars [10]. An accumulation of such linear measurements from discrete radial locations and distances achieves survey region coverage.

The following four parameters are used to configure a concentric circular trajectory covering a given survey region (Fig. 3): innermost circle radius, ρ_1; radial distance, δ_c, between circumferences of consecutive circles; arc length, s, between consecutive m-nodes on a circle; and positive integer, n, designating the nth or outermost circle including the survey region. The algorithm assumes that the rover is already within the survey region and that the fixed component is within line-of-sight from the rover [5]. If it is not, then no measurement is made. The survey completes when the nth circular trajectory is followed.

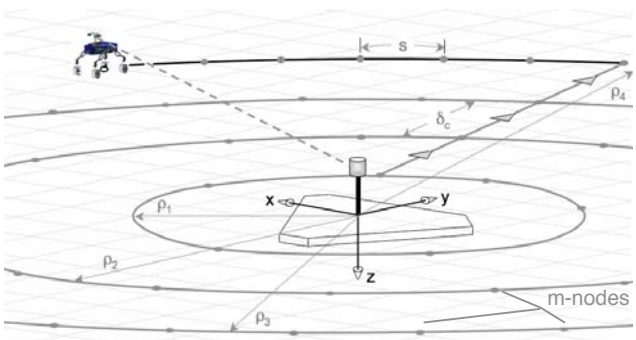


Figure 3. Concentric circular remote sensing survey and parameters.

A fixed-mobile configuration performs a spiral survey in a similar manner, differing only in that the rover drives in arcs along a trajectory of continuously increasing radius and needs no specific maneuvers to transition between successive spiral branches at larger radii [5]. Fig. 4 illustrates the spiral survey trajectory, which is parametrically similar to a circular survey.

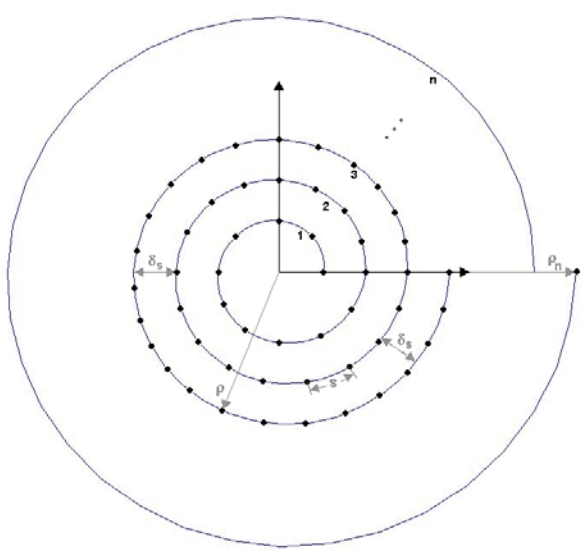


Figure 4. Overhead view of spiral remote sensing survey and parameters.

Both surveys are configured in a flexible manner to achieve desired degrees of measurement and area coverage resolution using the four parameters (ρ, δ, s, n). The surveys are primarily constrained by rover kinematic limitations, instrument maximum range, and terrain topography in the survey region whether executed radially inward or outward.

3.2 Single-Site Remote Survey Performance

Related research on distributed surveying [9] introduced the quality of performance metric defined earlier. We also apply this metric here as a basis for comparing expected performance of the concentric circular and spiral trajectories for distributed surveying.

The area of a survey region covered by either a concentric circular or linear spiral trajectory is equal to or roughly the same as $A = \pi\rho_n^2$, where ρ_n is the radius of the outermost circle or spiral branch. Areas within the survey region that are occupied by the stationary instrument component (at the origin of the survey coordinate system) and non-traversable obstacles are neglected. The total traverse distance D_c required for a concentric circular survey is the sum of distances traveled on each circumference and the radial separation distances, δ_c, between them:

$$D_c = 2\pi\left(\sum_{i=1}^{n}\rho_i\right) + (n-1)\delta_c \tag{2}$$

yielding the following QoP,

$$Q_c = \frac{\pi\rho_n^2}{2\pi\left(\sum_{i=1}^{n}\rho_i\right) + (n-1)\delta_c}. \tag{3}$$

For each linear spiral branch traversed (every $\theta = 2\pi$ radians), the spiral radius ρ increases by δ_s (Fig. 4), i.e., $\rho = (\delta_s/2\pi)\theta$. It can be shown [12] that the total traverse distance D_s required for a linear spiral trajectory is then expressed as

$$D_s = \frac{\delta_s}{4\pi}\theta_n^2 \tag{4}$$

where θ_n is the maximum spiral angle reached. The resulting QoP is then

$$Q_s = \frac{4\pi^2\rho_n^2}{\delta_s\theta_n^2}. \tag{5}$$

Based on the QoP metric the two fixed-mobile configurations for remote sensor-based surveying can be compared. With roughly the same survey region area, their QoPs are distinguished by distance traveled. If the spiral begins and ends as shown in Fig. 4, then $\theta_n = 2n\pi$, and $D_s = \delta_s\pi n^2$. For closest equivalence between the two trajectories, let the first circle radius be equal to the initial spiral radius, ρ_1, and let $\rho_1 = \delta_c = \delta_s$. Under these conditions, $\rho_2 = \rho_1 + \delta_c = 2\delta_c$, $\rho_3 = 3\delta_c$, $\rho_4 = 4\delta_c$, and so on. The summation term in Eq. (2) then becomes a function of an arithmetic series and simplifies as follows.

$$\sum_{i=1}^{n} \rho_i = \delta_c + 2\delta_c + 2\delta_c + ... + n\delta_c \tag{6}$$
$$= \delta_c(1+2+3+...+n)$$
$$= \delta_c\left[\tfrac{1}{2}n(n+1)\right]$$

Using this result in Eq. (2), we have

$$D_c = 2\pi\delta_c\left[\tfrac{1}{2}n(n+1)\right] + (n-1)\delta_c \tag{7}$$
$$= \delta_c\left[\pi n^2 + (\pi+1)n - 1\right].$$

Therefore, $D_c > D_s$ independent of an equivalent separation distance. A rover executing a concentric circular survey of n circles would need to traverse over $(4n–1)\delta_c$ meters more to cover the same area as it could with a spiral trajectory of n branches. As an example, to traverse a survey trajectory of n = 3 concentric circles separated by δ_c = 10 m, a rover would drive a linear distance of 397 m; to survey a roughly equivalent area using a spiral trajectory of n = 3 branches separated by δ_s = 10 m it would drive a linear distance of 283 m, or 29% less in this case.

3.3 Multiple-Site Remote Sensing Surveys

An example of a mobile-mobile, or tandem, survey system configuration is illustrated in Fig. 5. Both rovers could carry the active and passive components of the distributed survey instrument (e.g., for redundancy), or each rover could carry the companion component to the other's payload. Such survey configurations are suitable for multiple-site surveys due to the mobility of both platforms. The same dual mobility enables this tandem configuration to perform a number of approaches to remote sensor-based mobile surveying including, as special cases, the approaches described above for single-site surveys.

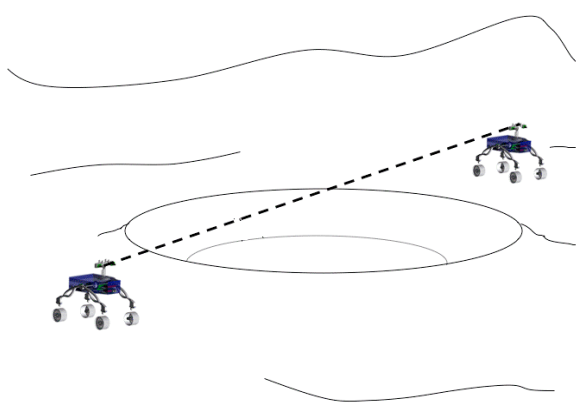


Figure 5. Tandem distributed remote sensing configuration.

3.4 Multi-Site Remote Survey Performance

A tandem-robot system was proposed in [6, 9] in which one robot carried an active instrument and the other carried the instrument's passive receiver. The robots would cooperatively perform remote sensor-based surveys using either parallel-swath or circular patterns where the width of a swath is the separation distance between the robots. The QoP metric was applied to compare several variants of these survey patterns including those illustrated below in Fig. 6. These survey approaches are referred to here as tandem-parallel and tandem-circular, and the QoPs for each are:

$$Q_{tp} = \frac{mn\delta_{tp}}{[2mn + 2(n-1)\delta_{tp}]} \tag{8}$$

$$Q_{tc} = \frac{\pi(n\delta_{tc}/2)^2}{\frac{\delta_{tc}}{2}\left(\pi n^2 + 2n - 4\right)} \tag{9}$$

where m is the length of a surveyed parallel swath (see Fig. 6), δ_{tp} and δ_{tc} are the robot separation/measurement distances during a survey, and n (in both equations) is the total number of swaths surveyed [9]. The denominators of Eqs. (8) and (9) express the total distances, D_{tp} and D_{tc}, traversed by both robots for the respective tandem survey types.

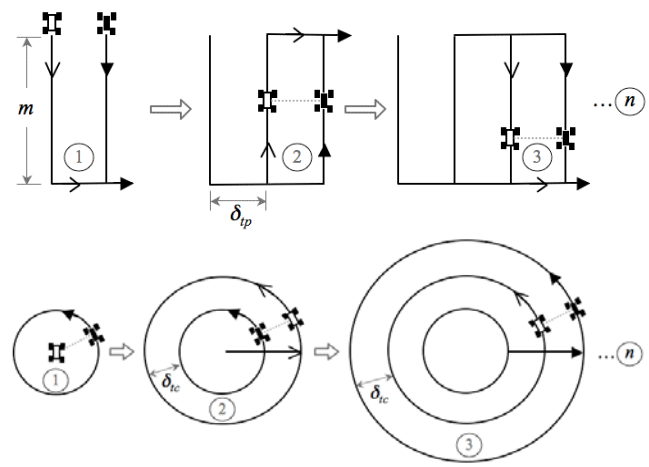


Figure 6. Tandem parallel and circular remote sensing survey trajectories.

4. CHARACTERISTIC PERFORMANCE OF MOBILE SURVEYS

The use of a common performance metric to comparatively rank a set of options provides a valuable basis or choosing the best option for a given task. However, direct one-to-one comparisons of the mobile surveying approaches discussed above are not straightforward given their respective differences in survey system configuration and survey trajectories. In fact, they can be prescribed for considerably different types of survey tasks in practice. Nonetheless, somewhat common grounds for comparison would provide a general sense for the relative performance of each approach. As a useful compromise a characteristic comparison is made here, still using a common metric, but based on assumptions that serve to equalize differences across the set of options. We use the QoP as the common metric here but any of a variety of metrics could be used instead.

Consider a mobile survey of a designated region with common area, A, and performed by each of the local and remote sensor-based methods discussed above. Recall that the metric for each survey type is the total area covered divided by the total distance traversed, D, during the survey (QoP = A/D). Since A is considered to be a common survey region size, differences in the QoP are a function of differences in D only. Further assume that each survey is designed for a comparable area coverage resolution; this would call for equal values of the parameters δ_p, δ_c, δ_s, δ_{tp}, and δ_{tc}. For the respective QoPs given by Eqs. (1), (3), (5), (8), and (9), the distance traveled is a function of the area coverage resolution. For simplicity, we assume a value of unity as the common value for the respective resolution parameters. With this assumption the expression of distance traversed for each QoP becomes a function of parameter n only, with the exception of Q_p and Q_{tp}, i.e., the local parallel and tandem-parallel survey metrics in Eqs. (1) and (8). If we assume a square survey region, then the length of a transect for a local parallel survey becomes $m = (n\text{-}1)\delta_p = (n\text{-}1)$, since δ_p is set to unity. Similarly, the length of a swath for a remote tandem-parallel survey becomes $m = n\delta_{tp} = n$, since δ_{tp} is set to unity. With all traverse distances as functions of the n-parameter for each survey type the following characteristic expressions, $D(n)$, can be created

$$D_p(n) = n^2 - 1 \tag{10}$$

$$D_c(n) = \pi n^2 + (\pi + 1)n - 1 \tag{11}$$

$$D_s(n) = \pi n^2 \tag{12}$$

$$D_{tp}(n) = 2n^2 + 2(n-1) \tag{13}$$

$$D_{tc}(n) = \tfrac{1}{2}\pi n^2 + n - 2. \tag{14}$$

Note that Eqs. (10)-(14) are only characteristic of the respective distances traversed due to the different meanings for n for each survey type, i.e., the number of parallel transects, concentric circles, spiral branches, and parallel swaths. For this discussion we will generally refer to n as the number of survey *passes*. Fig. 7 shows the characteristic trends of traverse distances required by each mobile survey configuration to cover the same area with comparable coverage resolution.

The characteristic trends are shown for up to 100 survey passes. As the number of survey passes increases the characteristic distances for the circular and spiral remote sensor-based surveys increase at the fastest and roughly the same rate. Circular remote-sensing surveys require the longest characteristic traverse distances. The inset graph in Fig. 7, which shows the same trends for up to an order of magnitude less survey passes (up to 10), reveals a more pronounced difference between expected distances required for circular and spiral surveys. The QoPs for these fixed-mobile surveys would be expected to decrease most rapidly with large numbers of survey passes, while higher QoPs would be expected for spiral surveys with low numbers of passes. For the tandem configurations, parallel circular surveys would be expected to perform better than parallel swath surveys, which would require characteristically longer traverse distances. The QoPs for tandem approaches would be expected to be higher than those of fixed-mobile approaches for the same survey region and coverage resolution. This reflects the advantage of using more than one rover for distributed remote sensing survey tasks [6]. The popular parallel transect survey approach for single rovers with local sensing devices has the slowest trend of increasing distance as the number of survey passes increases. Thus, its QoP would be expected to be high and least impacted by n, relative to the other approaches, for surveying the same area at a comparable coverage resolution.

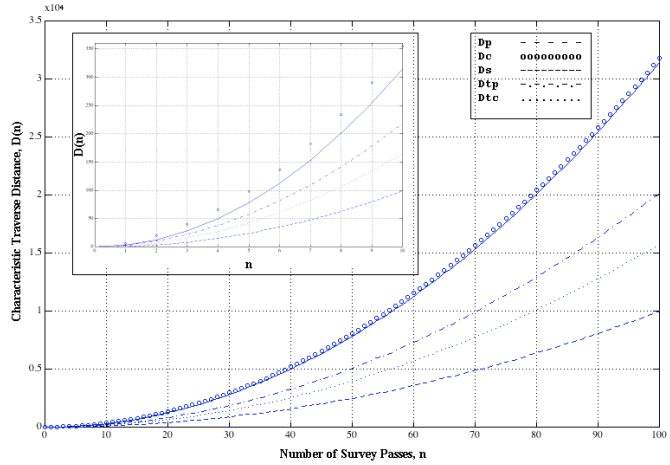


Figure 7. Characteristic traverse distances for mobile survey configurations.

5. ENRICHING METRICS FOR SURVEYS ON PLANETARY SURFACES

Thus far we have considered evaluation of coverage performance for surveys based on simple geometric measures of area and distance. Other important measures should be considered to enrich performance evaluation in the domain of planetary robotic systems and otherwise resource-constrained field robot systems. In fact, the selection of a candidate survey trajectory type for a given survey region can be based on metrics that provide some measure of relative resource usage during trajectory following (e.g., time, power, survey data storage). Selection of a survey type can also be based on any known physical constraints about the survey region such as terrain topography or the size and distribution of phenomena to be measured.

Past work provides guidance in this direction. For example, maximization of incident solar energy on rover solar arrays has been considered as a determinant for selection among several survey trajectory types [13]. Another study applied an energy efficiency metric, defined as a ratio of area covered to energy consumed, to evaluate parallel line, circular spiral, and square spiral trajectories [12]. Other attributes can be adopted that are germane to surveying for planetary surface exploration but proposed in the context of mobility/navigation or task performance. These include terrain traversability measures such as obstacle abundance [14, 15] and number or effort of rovers and humans operators involved [16]. While identifying domain specific attributes that would enrich the relevance of existing geometric measures is helpful, the manner in which they would be incorporated into a metric formulation is also worth considering. Computationally complicated metric formulations can sometimes make the use of a metric as a decision aid difficult or can obscure the interpretation of the metric. In the domain of planetary site characterization a host of system, mission, and/or environmental constraints will affect the performance of mobile surveying tasks.

Performance metrics based solely geometric aspects of the task do not capture other important performance impacts of the task, and therefore are particularly limited as decision aids. Mission operators will be better equipped to select appropriate survey methods when using metrics that account for a broader range of performance impacts that include system resources, terrain information or task constraints in addition to geometric measures like the QoP.

As one example of how a resource attribute can change and influence the effectiveness of a basic metric consider the following. In [12], energy consumed by robot wheel motors was considered based on an empirically derived model of DC motors. Differences in energy efficiency were attributed in part to the required amount of turns necessary to follow the search pattern. Depending on how a rover mobility system executes a traverse, many turns-in-place may be executed throughout a survey. Such maneuvers are not captured by a metric like the QoP and unless energy consumption is considered, the overall performance of a survey could be obscured. The study considered energy consumed during accelerations in addition to during turns, which led to a conclusion that circular spiral surveys were most efficient for larger survey areas while parallel line scans were most efficient for small survey areas [12]. This conclusion is based on the fact that the robot continuously moves without stopping and turning when executing spiral trajectories, thus consuming less energy over longer distances required for spiral surveys. Finally, consider that while distance traversed and energy consumed are correlated in most cases, if terrain traversability is ignored then a mobile surveying metric will not capture the distance or energy impacts of surveying a rough and rocky terrain cluttered with obstacles. Due to such considerations, we advocate for enrichment of metrics for mobile surveying tasks to improve their utility as decision aids for actual mission operations.

6. SUMMARY AND CONCLUSIONS

Science and engineering surveys will need to be conducted by mobile robotic systems to characterize sites on planetary surface missions. Various system configurations and surveying techniques are possible and a basis for selecting and evaluating the options is needed. Performance metrics provide some basis for decisions in this regard and a simple geometric measure is applied to five mobile survey approaches that have been proposed for robotic site characterization. The relative performance trends of the survey approaches were characterized based on a geometric assessment of their expected performance in surveying a common area at a comparable area coverage resolution. It is asserted that metrics solely expressing geometric aspects of survey task performance are particularly limited for selecting or evaluating survey options for planetary surface missions. The importance of capturing additional important attributes such as system resources, terrain information, or mission-related task constraints in performance metrics is discussed. Such enrichment is advocated to improve the utility of performance metrics as decision aids to mission operators.

7. REFERENCES

[1] Garcia, E. and Gonzalez de Santos, P. 2004. Mobile-robot navigation with complete coverage of unstructured environments. Robotics and Autonomous Syst. 46, 195-204.

[2] Wong, S.C., Middleton, L, MacDonald, B.A. 2002. Performance metrics for robot coverage tasks. In Proc. of the Australasian Conf. on Robotics and Automation (Auckland, New Zealand, Nov. 2002). 7-12.

[3] Mei, Y., Lu, Y.-H., Hu, Y.C. and Lee, C.S.G. 2006. Deployment of mobile robots with energy and timing constraints. IEEE Trans. on Robotics, 22, 3, 507-522.

[4] Fong, T., Bualat, M., Edwards, L., Fluckiger, L., et al 2006. Human-robot site survey and sampling for space exploration. In Proc. of the AIAA Space Conf. (San Jose, CA, 2006).

[5] Tunstel, E., Anderson, G., Wilson, E. 2007. Autonomous mobile surveying for science rovers using *in situ* distributed remote sensing. In Proc. of the IEEE Intl. Conf. on Systems, Man, and Cybernetics (Montreal, Oct. 2007). 2348-2353.

[6] Anderson, G.T., Hashemi, R.R., Wilson, E., Clark, M. 2000. Application of cooperative robots to search for water on Mars using distributed spectroscopy. In Proc. of the Intl. Symp. on Robotics and Applications, WAC (Maui, HI).

[7] Schmitt, H.H., Kulcinski, G.L., Sviatoslavsky I.N. and Carrier III, W.D. 1992. Spiral mining for lunar volatiles. In Proc. of the 3rd Intl. Conf. on Engineering, Construction, and Operations in Space III (Denver, CO) 1162-1170.

[8] Fong, T., Allen, M., Bouyssounouse, X., Bualat, M.G., et al 2008. Robotic site survey at Haughton Crater. In Proc. of the 9th Intl. Symp. on Artificial Intelligence, Robotics and Automation in Space (Los Angeles, CA, February 2008).

[9] Hashemi, R.R., Jin, L., Anderson, G.T., Wilson, E. and Clark, M. 2001. A comparison of search patterns for cooperative robots operating in remote environments. In Proc. of the International Symp. on Information Technology: Computing and Coding (Las Vegas, NV) 668-672.

[10] Wilson, E.W., Tunstel, E.W. and Anderson, G.T. 2007. BioGAS spectrometer for biogenic gas detection and location on the surface of Mars. In Proc. of the AIAA Infotech@ Aerospace Conf. and Exhibit (Rohnert Park, CA, May 2007).

[11] Hashmonay, R.A. and Yost, M.G. 1999. Innovative approach for estimating fugitive gaseous fluxes using computed tomography and remote optical sensing techniques. Journal of Air Waste Management Assoc. 49, 8, 966-972.

[12] Mei, Y., Lu, Y.-H., Hu, Y.C. and Lee, C.S.G. 2004. Energy-efficient motion planning for mobile robots. In Proc. of the IEEE Intl. Conf. on Robotics and Automation (New Orleans, LA) 4344–4349.

[13] Shillcutt, K.J. 2000 Solar Based Navigation for Robotic Explorers. Doctoral Thesis. Robotics Institute, Carnegie Mellon University (Pittsburgh, PA) CMU-RI-TR-00-25.

[14] Tunstel, E. 2007. Operational performance metrics for Mars Exploration Rovers. Journal of Field Robotics: Quant. Perf. Eval. of Robotic and Intelligent Systems, 24, 8-9, 651-670.

[15] Sukhatme, G.S. and Bekey, G.A. 1996. Multicriteria evaluation of a planetary rover. In Proc. of the IEEE Intl. Conf. on Robotics and Automation (Minneapolis, MN).

[16] Elfes, A., Dolan, J., Podnar, G., Mau, S. and Bergerman, M. 2006. Safe and efficient robotic space exploration with tele-supervised autonomous robots. In Proc. of the AAAI Spring Symposium (Stanford, CA, March 2006) 104-113.

Evaluating Situation Awareness of Autonomous Systems

Jan D. Gehrke
Center for Computing Technologies (TZI)
Universität Bremen
Am Fallturm 1, 28359 Bremen, Germany
jgehrke@tzi.de

ABSTRACT

Autonomous systems proved to be very successful in specialized problem domains. But their perception, reasoning, planning and behavior capabilities are generally designed to fit special purposes. For instance, a robotic agent perceives its environment in a way that was defined in advance by a human designer. The agent does not exhibit a certain perception behavior because it actually thinks it would be reasonable to do so. But with an increasing level of autonomy as well as a larger temporal and spatial scope of agent operation higher-level situation analysis and assessment become essential. This paper examines criteria for evaluating situation-awareness of autonomous systems and proposes methods to satisfy them. An example application scenario is presented that provides initial results for evaluating situation-aware systems.

1. INTRODUCTION

Autonomous systems are being developed for numerous application areas. These systems proved to be very successful in specialized problem domains, e. g., road driving, area exploration, or robot soccer. Nevertheless, in many cases, the perception, reasoning, planning and behavior capabilities of autonomous systems are *designed* to fit a special purpose. For instance, a robotic agent perceives its environment in a way that was defined in advance by a human designer. Therefore, the agent does not show a certain perception behavior because it actually *thinks* it would be reasonable to do so. It is a reflex rather than a deliberate action. For a lot of applications this might be sufficient. But with an increasing level of autonomy as well as a larger temporal and spatial scope of agent operation higher-level situation analysis and assessment become essential. We will focus on knowledge-based autonomous systems that require up-to-date knowledge to decide on their next course of action. Knowledge-based systems are particularly challenging if they require knowledge of distant locations and/or prediction of future events. These systems cannot rely on their own sensory capabilities only. They need to infer future states and communicate with other agents to share information.

Situation assessment depends on the current goals of the autonomous system and knowledge about the state of the world. The *relevance* of a specific piece of information or some kind of information for situation assessment is determined by the current plan and other potential plans under consideration. Information relevance leads to the pragmatic dimension of knowledge-based systems, i. e., information on how to use information. Usually, this is implicitly specified by the decision system. For instance, the decision system may use behavior rules whose rule body implies the information needed to evaluate the rule head. Problems arise if the decision-relevant information is not available to the agent. Depending on the applied knowledge representation and reasoning mechanism, missing information is simply considered undefined, default-valued, or false. The latter applies for predicates when using negation as failure [7] as in logic programming. This seems inadequate for autonomous systems in partially observable environments. Even the Open World Assumption turns out to be insufficient for situation assessment. While it prevents to assume wrong states of the unknown world and to base inferences on them, it does not directly enable to reason about what is unknown. As a consequence, the system would not be able to assess a situation correctly, e. g., to detect a harmful risk, because it has a lack of information which it is not aware of.

Hence, autonomous systems doing situation assessment have to be enabled to detect unknown information, thereby becoming *known unknowns*. This detection process must be governed and prioritized by information relevance. If the agent's sensory capabilities cannot provide information needed, other agents or information sources have to be inquired. As an alternative, the agent has to accept its lack of information and address it, e. g., by more cautious behavior.

Only autonomous systems possessing such higher level reasoning capabilities are able to have true *situation awareness* [10]. It even becomes more complicated if agents have to reason about future events. Such agents would need prediction abilities including knowledge on how the world evolves in order to qualify the probability that some information does not change in a given time interval. In spacious environments with mobile robots, additional spatial reasoning about information dynamics is required. This paper outlines a classification of criteria for situation awareness of autonomous systems and suggests knowledge representation and reasoning methods to address them.

The remainder of this paper is structured as follows. Sec-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS '08, August 19–21, 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

tion 2 discusses issues of situation awareness in established autonomous system architectures. Section 3 defines and examines awareness criteria and possible techniques to implement them. A corresponding example scenario for a preliminary study is presented in section 4). The paper concludes with a discussion of this survey (Sect. 5) and a summary (Sect. 6).

2. AUTONOMOUS AGENTS AND SITUATION AWARENESS

Intelligent agents are a fundamental concept in Artificial Intelligence for autonomous decision-making. For most applications domains of agents up-to-date and precise knowledge on the state of the world is crucial for system performance. But surprisingly a lot of usual architectures do not explicitly consider the acquisition of information needed. This limitation might lead to suboptimal, wrong, or even disastrous decisions. Thus situation awareness for autonomous systems intends to evaluate and establish the basis for decision making depending on the agent's current tasks and goals.

Williams et al. [22] have a similar goal. They evaluate the so-called *groundedness* of representations in autonomous systems (mainly those applied in *RoboCup* competitions). The approach defines a measure for the capability of creating and maintaining correct associations between representations and their (physical) real-world counterparts in the system's knowledge base. Proposed qualities of groundedness include, e. g., relevance, accuracy/precision, uncertainty management, and self-awareness w. r. t. the state of the robot body, location, and sensors. The corresponding system evaluation is rather qualitative and performed offline by humans. Thus, the approach provides useful criteria for system evaluation but does not enable the system to reason about itself in order to improve its groundedness.

The simple reflex agent, as the most basic kind of agent, is the worst example for groundedness or situation awareness. Such agents are governed by condition/action rules and always do the same thing given the same perception. In contrast to the model-based reflex agent it has no internal state influencing its decisions [20]. Both reflex agents cannot be considered situation-aware. The simple reflex agent only takes into account instantaneous knowledge; the model-based variant has no notion of information relevance because it has no explicit goals. Anyway, there may be simple but useful tasks that are successfully handled by reflex agents.

The most important term in AI is the rational agent. Wooldridge defines an agent to be "rational if it chooses to perform actions that are in its own best interests, given the beliefs it has about the world" [23, p. 1]. But this definition could also consider an agent rational if it chooses an action without knowing the state of the world. Thus, the situation-aware agent extends and substantiates the classical rational agent definition. The belief about the world is no longer taken for granted but actively controlled by knowledge acquisition as an additional reasoning process.

The belief-desire-intention (BDI) model has become a prevalent approach in academia for deliberative software agent architectures (cf. [23, 15]). It is based on a theory of human practical reasoning developed by Bratman [6]. Human practical reasoning, according to BDI, consists of deliberation, i. e., deciding what state should be achieved, and means-ends reasoning, i. e., deciding how to achieve it. In the BDI model, an agent is represented by its subjective knowledge about the world (*beliefs*) and persistent goals that should be achieved (*desires*). Desires and current beliefs result in achievable goals and possible actions towards them. Finally, in a process of deliberation, the agent commits to a goal and a corresponding plan (*intention*). The fundamental BDI model does not consider the assessment of beliefs in terms of completeness, correctness/uncertainty, or being up-to-date with respect to the goals to be achieved. Additionally, the model does not take into account different levels of decision making with respect to real-time requirements or temporal scope of action and decision making.

The Real-time Control System (RCS) developed at National Institute of Standards and Technology (NIST) models an intelligent system as a hierarchy of goal-directed sensory-interactive control processes [2] representing organizational levels as well as temporal scopes of decision making. The process hierarchy in RCS enables the decomposition of subtasks to different agents as well as different planning intervals within a single agent. Each level contains computational elements for sensory processing, world modeling, value judgment, and behavior generation [1]. Situation awareness could be assigned to the higher-level RCS world modeling components with a tight connection to behavior generation and sensory processing. That is, RCS could be augmented in world modeling by goal-oriented pro-active knowledge acquisition that is governed by behavior generation demands and may provide a focus of attention in sensory processing.

3. CRITERIA FOR SITUATION AWARENESS

Situation awareness is a field of research that commonly examines information requirements of humans for special jobs such as facility monitoring or flying aircraft [10]. Endsley [9, p. 97] describes situation awareness as "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future". This leads to Endsley's three levels of situation awareness:

- **Level 1 – Perception:** Basic perception of important information.
- **Level 2 – Comprehension:** Correct interpretation and integration of perceptions as well as relevance assessment.
- **Level 3 – Projection:** The ability to predict future situations based on current perceptions and background knowledge.

Although the definition and the three levels of awareness are intended for human situation awareness they can be adopted for autonomous systems, too. Nevertheless, there are a lot of technical requirements that are partially taken for granted regarding humans but much more challenging for technical systems. Thus, this section proposes the following criteria for situation awareness of autonomous systems and possible methods to fulfill them:

1. **Reasoning about ignorance:** The agent's knowledge base can be queried for unknown or uncertain information.

2. **Model of perception abilities:** The agent is aware of its sensors and the kind of information they may generate.
3. **Model of information relevance:** Based on the current set of goals or a general utility function as well as applied decision rules, the agent can identify information needed and qualify its importance for its performance measure.
4. **Model of information dynamics:** Knowledge on information dynamics that provides stability information for prediction of future states of the world.
5. **Spatio-temporal reasoning:** The model of information dynamics is applied for temporal reasoning on change over time and spatial reasoning on neighborhood and hierarchies of regions
6. **Social ability:** If agents need information beyond their sensory capabilities they have to cooperate with other agents for external information acquisition.

3.1 Reasoning about Ignorance

To enable an autonomous system to measure awareness in a given situation the system needs to know what it does *not* know (*known unknowns*). That is, it has to detect a possible lack of knowledge in its knowledge base. This is particularly important and challenging in environments that are highly dynamic and only partially observable. Unfortunately, many logic-based systems use *negation as failure* in reasoning, i. e., propositions or predicates are assumed to be false if there is no fact or prove stating the opposite. This *closed world assumption* (CWA) is opposed to the *open world assumption* (OWA) which does not make any assumptions about missing knowledge. Instead, logical inference only relies on given facts.

Open world reasoning is particularly applied in ontologies and description logics [3] for concept subsumption in ontology TBoxes, i. e., the schema level of a knowledge base. Situation awareness is rather focused on the instance or assertional level (ABox) of a knowledge base. While the open world assumption also applies for ABox reasoning, it does not provide inferences that would directly provide information on unknown facts. In contrast, a query for the truth of facts that are neither known nor deducible will just return *false*.

As a consequence, CWA and OWA reasoning systems will create a biased view of the state of the world that is very likely to be wrong. If decisions rely on that wrong beliefs system performance is jeopardized. But a strictly logical approach to reason about agent ignorance will raise several difficulties for the logical foundations of representation and reasoning as well as computational efficiency. A structural approach that keeps track of knowledge base changes and instantly replies to queries on ignorance is probably preferable. Anyway, a three-valued logic with OWA reasoning would be required, too.

In general, a strictly logic-based knowledge representation in dynamic environments is debatable. These approaches do not sufficiently handle the uncertainty that is an inevitable consequence in such domains. Though logics are still very useful and powerful for representing background knowledge of an autonomous system, dynamic environmental properties should rather be represented by probabilistic approaches with aleatory variables. In particular, Bayesian inference provides powerful means to reason about uncertainty. A corresponding approach for value measurement of missing information is presented in Sect. 3.3.

3.2 Model of Perception Abilities

While it is important to be aware of ignorance, agents might deliberately or of necessity choose to decide instantly although having incomplete or uncertain knowledge. This might be due to limited resources or perception abilities, i. e., sensors. An autonomous system that desires a certain kind of information may not be able to acquire this information from its own sensors or data sources. In such cases, it is obviously not reasonable to wait for that information before the decision for which it is considered helpful.

But deliberate ignorance requires background knowledge on what information can be obtained or not. This knowledge can be provided by a sensor model that describes sensors by the type of information they may deliver as well as precision, accuracy, spatial range, and response time. A related ability, the projection of future events, is discussed in Sect. 3.4.

For situation-aware agents we created an OWL-DL [4] ontology of possible sensors and their properties. This ontology does not describe concrete sensor, e. g., some special LIDAR product. Instead, it represents

- types of sensor devices (simple and complex),
- the (usually physical) quantities that are measured,
- the unit of measurement (e. g. SI units),
- the method of measurement (mechanical, electrical, optical, chemical etc.), and
- the sensor quality in terms of measurement errors, precision/accuracy, range, response time etc.

Examples of modeled sensors are those for electrical quantities (i. e. ohmmeter, galvanometer, voltmeter, etc.), electromagnetic radiation sensors for a given spectrum (i. e. infrared, visible light, etc.), acoustic sensors, thermal sensors and range sensors etc. The basic sensors were called *detectors*. But sensors as well as measured quantities can also be more complex. For instance, a weather "sensor" provides more than just one quantity and may aggregate raw measurements to some qualitative information.

The general advantage for agents using this ontology is that they can logically infer whether they have a sensor (or other information source) that will provide a desired information on some quantity or other environmental feature. Additionally, an ontology-grounded value description will allow for better interpretation of sensor data. The ontology also enables agents to communication about desired information abstracting from the specific sensors used (cf. Sect. 3.6).

3.3 Model of Information Relevance

In large and dynamic environments there will be a lot of information that is inaccessible or unknown to the autonomous system. Other information is available but there are no sufficient resources for processing. Obviously this also holds for humans but they perform very good in a lot of tasks anyway. This is because the major part of the state of the world is simply irrelevant. Humans have the basic

cognitive capability to focus on the relevant stimuli and ignore the others. For low-level tasks this human capability is not based on a deliberation of relevance but basic cognitive functions, e. g., within *superior colliculus* for visual, auditory and somatosensory stimuli. Only relevant stimuli arrive at short-term memory [16].

When trying to imitate this human capability in a cognitive approach to autonomous systems it is practically impossible to design a generic approach for all purposes. Low-level attention control will require low-level implementation because of real-time constraints. But there are also other kinds of information that allow for higher-level assessment of information relevance for determining a focus of attention. Such information is related to decisions with temporal scopes beyond the servo level with 0.05 seconds plans (cf. [2]).

Based on the current set of goals or a general utility function as well as applied decision rules, the agent could identify information needed and qualify its importance for its performance measure. For goal-oriented systems with rule-based decisions, information (as logical predicate) is relevant if it is contained in the body of some rule whose head is going to be evaluated. While this could be identified in a quite straightforward way on syntax level, such a system would require to determine predicates with unknown truth value. This again, has to apply some reasoning on ignorance (Sect. 3.1).

Utility-based systems with probabilistic knowledge representation need to assess information relevance differently. Here we consider a decision-theoretic agent that tries to maximize its utility based on expected utility of each action A with possible outcomes O_1 to O_n of different probabilities given a set of evidences E. In simple cases there is a single decision (not a sequence) and a corresponding action with a discrete distribution of results. Then, the expected utility of action A is defined by:

$$EU(A|E) = \sum_{i=1}^{n} P(O_i(A)|Do(A), E) \cdot U(O_i(A))$$

Here the best action $\alpha = \max_A EU(A|E)$ depends on the given evidence E. Thus α will potentially change given additional evidence E_j:

$$\alpha_{E_j} = \max_A EU(A|E, E_j)$$

Intuitively, the relevance (or value) V_{E_j} of a new evidence E_j is then defined as the difference in expected utility of the chosen action after and before knowing E_j. But because the agent needs to evaluate the value before acquiring the information it will need to average over possible assignments e_{jk} of E_j, i. e.,

$$V_{E_j} = \left(\sum_k P(E_j = e_{jk}|E) \cdot EU(\alpha_{e_{jk}}|E, E_j = e_{jk}) \right) - EU(\alpha|E)$$

This approach is proposed in *information value theory* by Howard [13]. If the acquisition of information E_j is costly it is only worthwhile if its price is less than its value (presupposing utility is measured in same quantity as price). Note that the information value formula presupposes perfect information. Uncertain information can be included by modeling a probabilistic relation from the uncertain variable to the actual variable.

In order to apply this relevance measurement to assess and potentially increase situation awareness an agent will need to proceed in a systematic way. That is, there is no sense in evaluating relevance of all possible information (i. e., aleatory variables). The agent needs some background knowledge on utility-influencing factors.

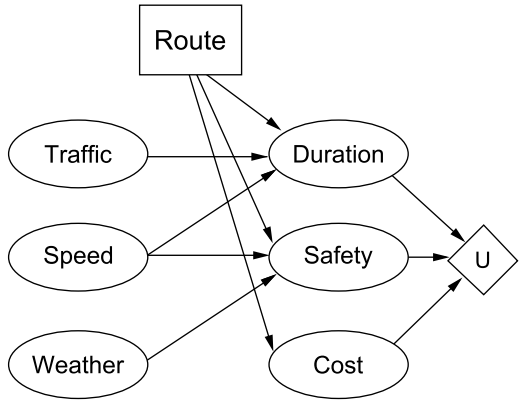


Figure 1: Simple decision network example for a route selection problem (adapted from [20, p. 598])

This knowledge can be provided as a simple list or, more elaborately, as a decision network (or decision diagram) [14]. A decision network combines Bayesian inference with actions and utilities as additional types of nodes besides the usual chance nodes. The factors directly influencing agent utility are represented as parent chance nodes. Possible outcomes of actions are modeled as child chance nodes of each action respectively. There may be additional chance nodes that influence other chance nodes as in Bayesian networks. An example is given in Figure 1.

This decision problem representation enables a systematic ordering of information evaluation and acquisition by traversing the network from the utility node(s) along the path with highest influence on utility. For this purpose, it is necessary to distinguish sensor nodes as special chance nodes since only evidence from sensor nodes is actually available.

Unfortunately, the design of decision networks features the same difficulties as in Bayesian networks: one needs to assign prior probabilities as well as conditional probabilities based on sufficient experience. Because this is not given in general, the agent should be able to update the Bayesian influence model based on ongoing experience. Updates should primarily concern the assignment of probabilities. Creating new Bayesian links between chance nodes is likely to fail due to insufficient statistical evidence in complex and dynamic environments causing possible post-hoc or cum-hoc fallacies.

3.4 Model of Information Dynamics

Whereas decision-theoretic information-gathering agents can be implemented based on decision networks there are special requirements in dynamic environments with decisions of long temporal and spatial scope. As an example, we will use a simple route planning task with a truck that aims at finding the fastest route between two locations in a road network. A decision network for that problem could include chance nodes for influencing factors such as construction work, weather, traffic, time of day, and others (cf. Fig. 1).

The special challenge is that these factors describe future states of the world as well as properties of distant locations.

Consequently, the relevant information is not available from own sensors (distant information) or even theoretically unavailable (future states). Anyway, background knowledge on information dynamics can help infer or estimate such information. If the agent knows that some environment property (e. g., weather) holds in an area of certain extension (e. g. a particular stretch of a freeway) it is easily able to infer from a property of one location to the same property at another location within that area (cf. Sect. 3.5). It gets much more complicated when the agent has to infer this property in an neighboring area. There will be no general probabilistic relation here. It may depend on the property, location, time, and possibly several other factors.

On the other hand, an environment property that is stated to hold for a larger area should allow to deduce this property for sub-areas based on known region partonomies. Again, this inference will provide uncertain information. The statement for a large area will usually only average all locations therein. Thus, single locations might be very different.

Similar considerations also hold for the temporal dimension. A situation-aware system needs prediction abilities to estimate probabilities of variable assignments at future states or it has to acquire reliable predictions from other sources. Obviously, this becomes less achievable when the predicted state is far in the future. On the other hand, there are variables that will not change or only change slowly or rarely.

Re-iterating the above examples, weather or traffic feature some expected stability and transition probability from one state to another, respectively. Such dynamics are often modeled with Hidden Markov Models (HMMs), Kalman filters, or, more general, dynamic Bayesian networks (DBN). That is, one defines an influence or sensor model and a time transition model for the property. Classical DBNs presuppose a fixed discretization of time from one state to another. But in general a reasonable time discretization for a Markov transition model will depend on an average stability of the modeled property. Additionally, information stability may again depend on other factors and change from state to the following.

While DBNs try to infer from sensor evidence and a variable assignment at one state to that of a following (or previous) state, situation-aware agents might not need such kind of prediction for rather distant decisions (i. e. several hours). In some cases these predictions, if possible, will tend to be very uncertain anyway. The most naïve autonomous system will just assume that everything that holds presently will stay that way all along. A situation-aware system should at least consider that the world is dynamic and have some idea about these dynamics. It might be sufficient to know the expected stability (or half-life-period) of each property/variable in order to evaluate the probability that the current value will persist until the desired time in the future. If such persistence is evaluated to be unlikely (i. e. below a predefined threshold) the system is forced to rely on prior probabilities.

In general we advocate the inclusion of dynamic Bayesian networks in decision networks, i. e., dynamic decision networks.

3.5 Spatio-temporal Reasoning

If the situation-aware autonomous system evaluates relevance of missing information it will also need to qualify time and space when/where the desired property value holds. In the truck agent routing example the weather or traffic at some distant freeway stretch will influence the expected driving time. When trying to acquire the information from some external source or sensor for that region, the agent also has to consider the expected time when it plans to arrive there. Unfortunately, the arrival gets more and more uncertain with increasing distance of that location.

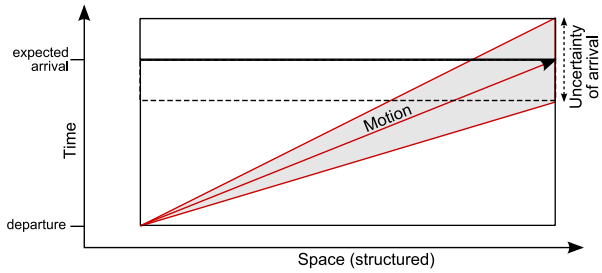


Figure 2: Spatio-temporal region defined by vehicle motion on a road stretch with uncertain driving time.

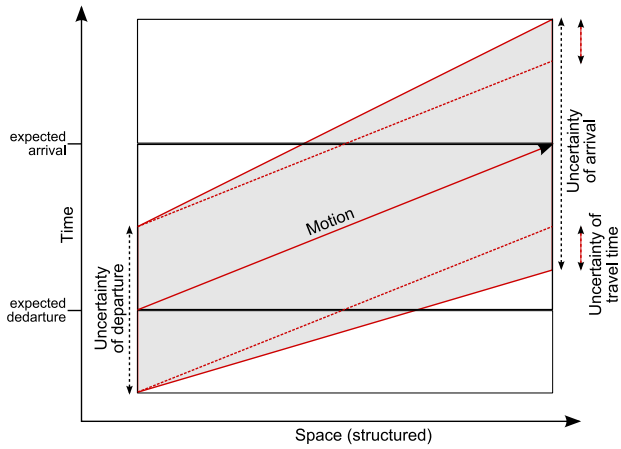


Figure 3: Spatio-temporal region with uncertain departure time.

Figure 2 depicts the situation for a road stretch at the beginning of a truck tour. While the departure time is known arrival time is uncertain. The uncertainty of arrival time or driving time can be provided by a decision network (or regular Bayesian network). But the agent needs to determine the expected spatio-temporal region it will move in to qualify the information for that freeway in space in time. We will call this a *region of relevance* for that agent.

We propose to choose a threshold as a quantile in probability distribution of arrival time to specify a confidence interval of arrival time. The upper bound of that interval will determine the upper temporal bound of the region of relevance for that freeway stretch. All spatio-temporal information overlapping that region is considered potentially relevant w. r. t. value of information (Sect. 3.3).

These regions of relevance will expand in temporal directions for distant locations because uncertainty of departure time increases with uncertainty of arrival time for previous regions. A region of relevance for distant locations is shown in Figure 3.

For qualitative topological inference on this regions the agent can apply usual spatial calculi such as *Region Connection Calculus* (RCC) [19]. This is of particular interest when agents want to share information and need to determine whether there is potential for cooperation. The regions of relevance are then complemented by regions of expertise, i. e., regions other agents can provide information for.

3.6 Social Ability

Social ability is one of the minimal criteria for intelligent agents defined by Wooldridge and Jennings [24]. In this context, social ability refers to communication with other agents. For situation-awareness communication is needed when there is relevant information that is neither available from own sensors nor inferable by reasoning. For instance, information on distant locations needs to be acquired from external sources.

In practice, the mere ability of communication will not suffice. A situation-aware system must determine that some information cannot be acquired from own sensors (Sect. 3.2) but is available from other sources instead. This includes the discovery of such sources, a common communication language with agreed content semantics, and price negotiations if information is subject to charges.

4. APPLICATION SCENARIO

One particular application domain in our research has been *autonomous logistics* [21]. That means, there are agents as autonomous decision makers that represent one or more logistic objects (e. g. truck, container, or pallet) and control logistic processes locally. This usually requires cooperation with other logistic agents as well as continuous acquisition of information for decision-making. This information is provided by local sensors (e. g., humidity sensors for monitoring of perishable goods) as well as external data sources (e. g. for weather and traffic reports).

When developing an autonomous logistics system, we evaluate agent behavior by stochastic simulation before deployment. That is, the developed agents as well as test mocks representing external agents and other entities run in a simulated logistics environment before they are released to a real-world environment. For this purpose we implemented the multiagent-based simulation system PlaSMA (*Platform for Simulations with Multiple Agents*). PlaSMA is based on the FIPA-compatible multiagent development environment JADE [5]. Multiagent-based simulation (MABS) applies object-based modeling [18, 8] as well as discrete time, distributed simulation techniques [11]. In MABS, the agents are parallel logical simulation processes that, in general, represent objects of the simulation model at the same time. Thus, MABS is a natural way to test agent behavior and interaction.

We applied this simulation platform for studies of agent situation awareness in vehicle route planning (similar to the example in section 3). Uncertainty in vehicle route planning is an everyday problem in transportation over longer distances. The shortest route is not always the fastest. Even when considering maximum allowed or average expected speed on single roads the planned routes may prove to be suboptimal. While suboptimal solutions are a natural property in dynamic, open environments with partial observability, usual route planning does not make use of much up-to-date or background information that would be obtainable and correlates with travel time.

We conducted simulation studies [12] that help find the utility or relevance of environmental information on travel time and its application in route planning cost functions. Furthermore, the experiments should provide evidence for the time in the future when incoming information is useful and the robustness when working with uncertain or wrong information.

Inductive machine learning was applied to pre-processed traffic information in order to learn traffic models for specific roads depending on weather, time of day, and day of week [12]. The data was based on German and Austrian traffic censuses. The learned traffic model includes prediction rules like

```
[Speed=60]
    <= [Day=Mo] [Time=morning]
       [Weather=moderate..good]
```

That is, average possible speed on the corresponding road is expected to be 60 kmph on Monday mornings if weather conditions are from moderate to good. Such predictions were integrated in a truck agent's utility function for best routes in a transportation process. As part of the agents knowledge acquisition component, the planning system initiates the gathering of weather information within a certain distance towards its destination. This information from external weather services together with time and date information is used to consult the rule base for a travel time prediction. The time and date in these rule base queries again depends on prediction of travel time to reach the location of interest (cf. discussion in Sect. 3.5).

In several stochastic simulations these predictions turned out to be valuable. Situation-aware vehicle agents were up to 6.3 % faster on average than regular, non-predictive agents. Additionally, the standard deviation of travel time was reduced by 28 %. With significance level $\alpha = 0.05$ the simulated results to not differ more than 0.0016 % from the actual values. Thus, dedicated means to increase situation-awareness are actually shown to be of advantage in the presented logistics scenario.

5. DISCUSSION

The survey on situation awareness in this paper identifies criteria for knowledge representation and reasoning of autonomous systems in dynamic environments. In particular, the paper focuses on domains with decisions that may have a larger spatio-temporal scope. We do not claim that the means proposed for implementing these criteria are imperative. Furthermore, a reference implementation of the described approach is work in progress. The presented application example still relies on some pre-designed information acquisition behaviors instead of fully goal-oriented and deliberate acquisition. An extended conceptual architecture that includes explicit information acquisition for situation assessment was proposed in [17].

The general question arises whether situation awareness turns out to be necessary or even futile for particular autonomous systems. Furthermore, most systems will probably only fulfill a subset of all criteria. Some criteria are probably not imperative for some degree of situation awareness. Endsley [10] also distinguishes three levels for human situation awareness that could match a particular subset

of the criteria proposed here. Relevance assessment is certainly mandatory for level 2 (*comprehension*). Representing and reasoning with information dynamics pertains to level 3 (*projection*). But social ability (i. e. information exchange) is beyond the scope of Endsley's approach. This criteria could establish a fourth level of awareness. On the other hand, a system could be able to share information but not be capable of projection. So it does not fit in as an extension of level 3.

But assuming an agent that is matching all criteria: Is such an agent capable of actually *measuring* (i. e. quantifying) its situation awareness? We think that information value theory has the answer. The difference between the expected utility with current sensory evidence and the expected utility given complete sensory evidence (considering perception abilities) could provide a corresponding measure. But future research has to be done for a complete theory of situation awareness and its assessment.

6. CONCLUSION

This paper provides a survey on situation awareness for autonomous systems by analyzing features and limitations of existing architectures and proposing a set of criteria to be satisfied by situation-aware agents. The proposed criteria build upon the definition of human situation awareness by Endsley [9, 10] but take into account the special requirements and needed capabilities of technical systems. The proposed criteria are *reasoning about ignorance*, *an agent perception model*, *information relevance assessment*, *a model of information dynamics* and associated *spatio-temporal reasoning capabilities*, and *social ability* for acquiring information beyond own sensors.

The presented logistics routing scenario shows that seemingly barely relevant information on environmental properties can significantly increase performance of autonomous agents. Future work will address a theoretical foundation of situation awareness and its measurement based on dynamic decision networks and information value theory.

7. ACKNOWLEDGMENTS

This research is funded by the German Research Foundation (DFG) within the Collaborative Research Center 637 "Autonomous Cooperating Logistic Processes: A Paradigm Shift and its Limitations" (SFB 637) at the University of Bremen, Germany.

8. REFERENCES

[1] J. Albus. A reference model architecture for intelligent unmanned ground vehicles. In *Proceedings of the SPIE 16th Annual International Symposium on Aerospace / Defense Sensing, Simulation and Controls*, pages 303–310, Apr. 1–5 2002.

[2] J. Albus, T. Barbera, and C. Schlenoff. RCS: An intelligent agent architecture. In R. M. Jones, editor, *Intelligent Agent Architectures: Combining the Strengths of Software Engineering and Cognitive Systems*, number WS-04-07 in AAAI Workshop Reports. AAAI Press, 2004.

[3] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook. Theory, Implementation and Applications*. Cambridge University Press, 2nd edition, 2007.

[4] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, L. A. Stein, and F. W. Olin. OWL web ontology language reference. Available from `http://www.w3.org/TR/owl-ref/`, Feb. 10 2004.

[5] F. Bellifemine, A. Poggi, and G. Rimassa. Developing multi-agent systems with a FIPA-compliant agent framework. *Software-Practice and Experience*, 31(2):103–128, Feb. 2001.

[6] M. E. Bratman. *Intention, Plans, and Practical Reason*. Harvard University Press, Cambridge, MA, USA, 1987.

[7] K. L. Clark. Negation as failure. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 293–322, New York, 1978. Plenum Press.

[8] P. Davidsson. Multi agent based simulation: Beyond social simulation. In *Multi-Agent Based Simulation, Second International Workshop (MABS 2000)*, number 1979 in LNCS, pages 97–107, Boston, MA, USA, July 8–9 2000. Springer-Verlag.

[9] M. R. Endsley. Design and evaluation of situation awareness enhancement. In *Proceedings of the Human Factors Society 32nd Annual Meeting*, volume 1, pages 97–101, Santa Mocica, CA, USA, 1988. Human Factors Society.

[10] M. R. Endsley. Theoretical underpinnings of situation awareness. A critical review. In M. R. Endsley and D. J. Garland, editors, *Situation Awareness Analysis and Measurement*, pages 3–32. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 2000.

[11] R. Fujimoto. *Parallel and Distributed Simulation Systems*. John Wiley & Sons, New York, NY, 2000.

[12] J. D. Gehrke and J. Wojtusiak. Traffic Prediction for Agent Route Planning. In *International Conference on Computational Science (ICCS 2008)*, number 5103 in LNCS, pages 692–701, Kraków, Poland, June 23–25 2008. Springer-Verlag.

[13] R. A. Howard. Information value theory. *IEEE Transactions on Systems Science and Cybernetics*, SSC-2(1):22–26, Aug. 1966.

[14] R. A. Howard and J. E. Matheson. Influence diagrams. *Decision Analysis*, 2(3):127–143, Sept. 2005.

[15] S. Kirn, O. Herzog, P. Lockemann, and O. Spaniol, editors. *Multiagent Engineering: Theory and Applications in Enterprises*. Springer-Verlag, 2006.

[16] E. I. Knudsen. Fundamental components of attention. *Annual Review of Neuroscience*, 30:57–78, July 2007.

[17] M. Lorenz, J. D. Gehrke, J. Hammer, H. Langer, and I. J. Timm. Knowledge management to support situation-aware risk management in autonomous, self-managing agents. In *Self-Organization and Autonomic Informatics (I)*, volume 135 of *Frontiers in Artificial Intelligence and Applications*, pages 114–128, Amsterdam, 2005. IOS Press.

[18] H. V. D. Parunak, R. Savit, and R. L. Riolo. Agent-Based Modeling vs. Equation-Based Modeling: A Case Study and Users' Guide. In *MABS 1998*, pages 10–25, Paris, France, 1998. Springer-Verlag.

[19] D. A. Randell, Z. Cui, and A. G. Cohn. A spatial logic

based on regions and connection. In B. Nebel, C. Rich, and W. Swartout, editors, *Proceedings of the 3rd International Conference on Principles of Knowledge Representation and Reasoning*, pages 165–176. Morgan Kaufmann, 1992.

[20] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River, NJ, USA, 2nd edition, 2003.

[21] B. Scholz-Reiter, K. Windt, and M. Freitag. Autonomous logistic processes - New demands and first approaches. In *Proceedings of the 37th CIRP-International Seminar on Manufacturing Systems*, pages 357–362, Budapest, May 19–21 2004.

[22] M.-A. Williams, P. Gärdenfors, A. Karol, J. McCarthy, and C. Stantom. A framework for evaluating groundedness of representations in systems: From brains in vats to mobile robots. In *IJCAI-05 Workshop on Agents in Real-Time and Dynamic Environments*, pages 17–24, Edinburg, UK, July 30 2005.

[23] M. Wooldridge. *Reasoning about Rational Agents*. The MIT Press, 2000.

[24] M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–152, 1995.

NIST/IEEE Virtual Manufacturing Automation Competition: From Earliest Beginnings to Future Directions

Stephen Balakirsky
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD 20878
stephen@nist.gov

Raj Madhavan
National Institute of Standards and Technology
Gaithersburg, MD 20878
Oak Ridge National Laboratory
Oak Ridge, TN 37831
raj.madhavan@nist.gov

Chris Scrapper
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD 20878
chris.scrapper@nist.gov

ABSTRACT
This paper provides an overview of the National Institute of Standards and Technology (NIST)/IEEE Virtual Manufacturing and Automation Competition (VMAC). Detailed information will be provided on the competition's objectives, history, and operation. In addition, the supporting software infrastructure will be discussed.

Categories and Subject Descriptors
I.2.9 [**Robotics**]: Autonomous vehicles; commercial robots and applications; kinematics and dynamics,; workcell organization and planning.

General Terms
Performance, Experimentation, Verification, Management.

Keywords
Competition, simulation, manufacturing, USARSim.

1. INTRODUCTION
Automated Guided Vehicles (AGVs) represent an integral component of today's manufacturing processes. Major corporations use them on factory floors for jobs as diverse as intra-factory transport of goods between conveyors and assembly sections, parts and frame movements, and truck trailer loading/unloading. The National Institute of Standards and Technology (NIST) is currently running a program to develop test method standards for the next generation of AGVs. These test methods specifically target small- and medium-sized manufactures and the problems that they face in applying automation to legacy manufacturing facilities.

In order to assess future automation needs, focus on solutions, and ease entry of new features into AGV product lines, NIST is striving to form a coalition that brings together the three communities of industrial end-users, researchers, and manufacturers of AGV equipment. This coalition will then focus on moving from real-world problems, to research activities, to system features. As the first thrust of this effort, NIST performed outreach to several small- and medium-sized manufactures. Phone interviews and site visits were conducted, and a common automation need emerged. This need is for both unit loaders and forklifts to operate in unstructured cramped environments to deliver parts and materials for use in the manufacturing process.

Automating these systems to operate in unstructured environments presents an exciting area of current research in robotics and automation. Unfortunately, the traditional entry barrier into this research area is quite high. Researchers need an extensive physical environment, robotic hardware, and knowledge in research areas ranging from mobility and mapping to behavior generation. An accepted approach to lowering this entry barrier is through the use of simulation systems and open source software. The use of simulation significantly reduces the amount of infrastructure required, the number of AGV systems needed, and the risk to both humans and robots. In addition, hardware-in-the-loop (HIL) or real/virtual testing allows for the exploration of scenarios that would be unsafe to pursue with actual robotic equipment.

NIST turned to the IEEE to build researcher involvement in this new coalition by submitting a successful proposal to the Washington section of the Robotics and Automation Society of IEEE to create a new Virtual Manufacturing and Automation Competition (VMAC).

2. VMAC PRELIMINARIES
By soliciting proposals from IEEE members and local student branches and robotic clubs in the Greater Washington area (Maryland, Virginia, and the District of Columbia), the organizers were able to bring together eight teams from regional universities consisting of faculty mentors, undergraduate students, and graduate students. The original VMAC participants are shown in Figure 1. The educational institutions that participated in this effort were: two teams from Hood College (Frederick, MD), one team from George Mason University (Fairfax, VA), and two

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

teams from University of Maryland Eastern Shore (Princess Anne, MD). Two teams from George Washington University (Washington, DC) and one team from Morgan State University (Baltimore, MD) dropped out of the competition due to lack of interest and other timing conflicts.

It has been NIST's experience that competitions are an effective means of stimulating interest and participation among students. Competitions tend to get the students excited about the technologies and encourage larger participation in the research community. As mentioned earlier, one of the main motivations for the VMAC was to lower the entry barrier for teams to participate in the competition. As such, it was decided that the teams would be provided with all of the hardware and software necessary to compete. This included the Unified System for Automation and Robot Simulation (USARSim) [2] system, the Mobility Open Architecture Simulation and Tools (MOAST) robotic control framework [1], and the hardware to run the controller (a Linux PC). In addition to the hardware and software, two workshop/tutorials were held and a user's support group was established. The first workshop was a 2-day event that took place in October 2007 at NIST in Gaithersburg, MD. During this workshop, participants were acquainted with the software base (USARSim and MOAST) and participated in the design of the competition events. A follow-up workshop was held at Hood College in February 2008 to track progress and provide hands-on assistance with any outstanding issues.

Figure 1 : Members of the teams from the inaugural VMAC workshop.

2.1 USARSim

The USARSim project was originally created in 2002 by Carnegie Mellon University (CMU) and the University of Pittsburgh to study human-robot interaction in the area of Urban Search and Rescue (USAR); hence the original acronym USARSim [8, 13]. The first release of USARSim was built by creating modifications to Epic's Unreal Engine 2 game engine[1]. It supported models for a few differential drive robotic platforms, a restricted set of sensors, and a small set of USAR specific test arenas. In addition, the robotic platforms could only be controlled through the use of the RETSINA [11] multi-agent system software.

In 2005, USARSim was selected as part of the base infrastructure for the RoboCup Rescue Virtual Robot Competition. The virtual robot competition is an annual international event that highlights research in diverse areas such as multi-agent cooperation, communications networks, advanced mobility, mapping, and victim search strategies. In addition to the competition, USARSim management was taken over by NIST and an international development community was established on the open source sourceforge.net website. While much of the original structure of the code was maintained, the code was reorganized and interfaces were standardized around SI units. The first official release (Version 1.0) was produced in October 2005.

A large-scale development effort accompanied the transition to sourceforge and the involvement of the Robocup community. Version 3.31, released in July 2008 offers 15 different sensors, from odometry to an omnidirectional camera. 23 different robotic platforms are now available; these include wheeled robots, cars, tracked vehicles and flying robots. In addition, many of the sensors and robots have undergone rigorous validation in order to prove their similarities and difference from the real devices [6, 9, 12]. With a growing community of users, urban search and rescue applications were no longer the only target area of interest for USARSim. In 2007, NIST provided additional modifications to USARSim to make it compatible with the manufacturing domain. These additions included a robotic forklift and unit loader AGV as shown in Figure 2. In order to introduce dynamics into the factory simulation, a "factory controller" was also introduced. This controller allows for the external control of conveyors, box chutes, and sensors that are able to interact with the vehicles under test. Due to the broader range of use that USARSim now enjoys, its acronym was maintained while its name was changed to Unified System for Automation and Robotics Simulation.

Figure 2: Forklift and unit loader AGVs created for USARSim.

[1] Certain commercial software and tools are identified in this paper in order to explain our research. Such identification does not imply recommendation or endorsement by the authors, nor does it imply that the software tools identified are necessarily the best available for the purpose.

2.2 MOAST

The MOAST framework builds upon several decades of robotic architecture and control system work that has been performed by NIST. The framework is designed to be a sample implementation of the 4D/RCS reference model architecture [4,5]. The RCS reference model architecture is a hierarchical, distributed, real-time control system architecture that decomposes a robotic system into manageable pieces while providing clear interfaces and roles for a variety of functional elements.

Figure 3 depicts the general structure of each echelon (level) of the 4D/RCS hierarchy. Each echelon in 4D/RCS contains a systematic regularity and is composed of control nodes that follow a sense-model-act paradigm. These control nodes are sensory processing (SP), world modeling (WM), value judgment (VJ), and behavior generation (BG). Sensory processing is responsible for populating the world model with relevant facts. These facts are based on both raw sensor data and the results of previous SP (in the form of partial results or predictions of future results). WM must store this information, information about the system self, and general world knowledge and rules. Furthermore, it must provide a means of interpreting and accessing this data. BG computes possible courses of action to take based on the knowledge in the WM, the system's goals, and the results of plan simulations. VJ aids in the BG process by providing a cost/benefit ratio for possible actions and world states.

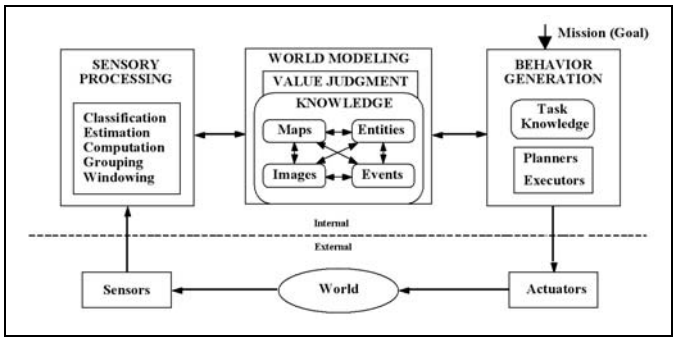


Figure 3: Generic 4D/RCS control node.

All of the echelons in 4D/RCS follow the sense-model-act paradigm with the principal difference between echelons being in the knowledge requirements and the fidelity of the planning space. This regularity in the structure enables flexibility in the system architecture that allows scaling of the system to any arbitrary size or level of complexity [7].

In 2005, MOAST was officially released on sourceforge as a public domain robotic control system. Since that time, it has attracted an international developer community and now contains control modules for robotic vehicles, manipulators, and end effectors. Figure 4 depicts the currently implemented echelons of the control system and their responsibilities. The central idea behind the VMAC competition is that research groups would be able to select their area of expertise and modify a small subset of control modules in order to improve the overall system performance. After the competition, each research group is required to contribute their modified modules back to the community. Thus, all teams benefit from each other's strengths.

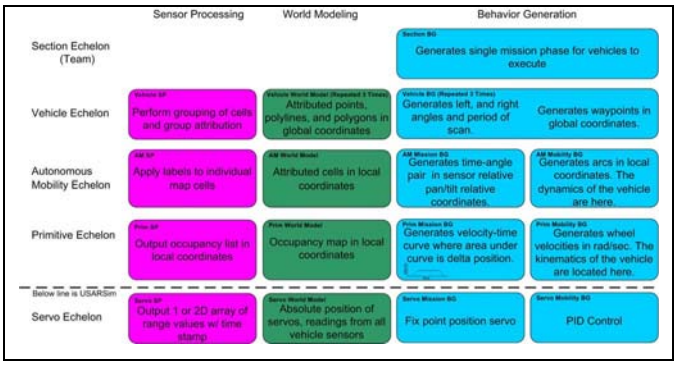


Figure 4: Control module responsibilities in the MOAST implementation of 4D/RCS.

2.3 Competition Design

This competition design was based on the successful RoboCup Rescue Virtual Robots Competitions [3]. Since all code used in these competitions is open source, participants are able to learn from their competitors and concentrate their research in their particular areas of expertise. It was envisioned that researchers from multi-agent cooperation, mapping, communications networks, and sensory processing backgrounds would all be interested in participating.

The initial competition design was formulated by using the SCORE framework [10]. This framework specifies that an overall system scenario be defined, and then basic elemental skills that allow for the successful completion of the scenario be extracted. Systems are then evaluated on both their ability in the elemental tasks as well as the overall scenario.

From the outset, the competition was to be based on real-world scenarios. Based on NIST's industry outreach effort, the scenario chosen was a factory setting that had significant clutter, maze-like passageways of various widths, and dynamic obstacles. The objective was to have several Ackerman-steered AGVs pick-up packages at a central loading station, and deliver these packages to one of several unloading stations. The package destinations were encoded in a Radio Frequency IDentification (RFID) Tag on each package.

Utilizing the SCORE framework, this scenario was decomposed into elemental tasks that included traffic management, route planning, accurate path following, and docking with loading/unloading stations. While the baseline code provided to the teams was capable of performing the objectives, it was far from optimal.

As time progressed, it became apparent that performing a complete SCORE evaluation of all of the elemental tasks and the overall scenario was an overly ambitious set of tasks for the first year of the competition. A decision was made to break the competition into several discrete best-in-class elemental tasks that would contribute to the scenario, and to leave the end-to-end objective and remaining elemental tasks for next year. The tasks of accurate route following and docking with loading/unloading stations were chosen as the VMAC tasks for the 2007-2008 competition year.

For the competition, the teams examined the BG modules at the prim and AM echelons in order to improve upon the existing baseline operation of the control system.

3. VMAC COMPETITION

The regional competition was held on April 18th 2008 at NIST. The competition was broken down into two tasks: path following and docking. For both tasks, the teams were required to accept a standard command message as input. They were free to use as much of the provided infrastructure as they desired (e.g. groundtruth).

For path following, the input consisted of a set of constant curvature arcs that were to be followed. In order to allow observers to examine the system's real-time performance, the target trajectory's arcs were shown in the simulated world. These arcs were represented as a red centerline with a blue error radius as shown in Figure 5. The vehicle could then be observed while driving in the simulation. In addition to the simulated view, as shown in Figure 6, the passageway boundaries and vehicle centerline trajectory were plotted in real-time and statistics were automatically generated. As the path progressed, the arc complexity was increased and the error radius was reduced. Hood College produced the best-in-class algorithm for path following followed by George Mason University in a close second. The scoring criteria included the distance traveled before the first crossing over the error radius and the average path and speed deviation. The team's score, as shown in Figure 7, was displayed in real-time during the competition.

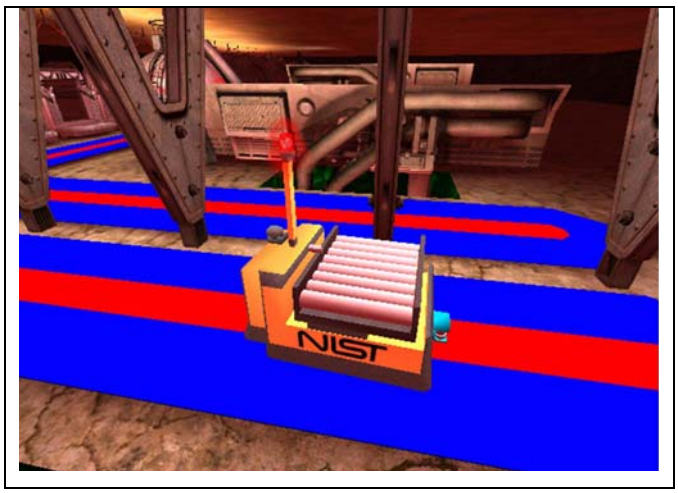


Figure 5: Example of the path following task. The red line is the centerline that is to be followed. The blue line represents the allowable error radius for the center of the vehicle.

Docking ability was measured by placing the AGV in a room with a docking station. The input for the docking behavior consists of the upper right and lower left corners of both the room and docking table, as well as the desired location of the vehicle's center point when docked. As shown in Figure 8, the AGV was required to successfully dock without a collision, where docking amounts to parallel parking a rear steered Ackerman vehicle. Upon successful docking, the AGV was placed in a smaller room and the procedure was repeated. The size of the room ranged in size from 10m x 10m to 4m x 4m. Once again, Hood College received top honors in this competition. They were able to successfully dock in every room in which they were placed.

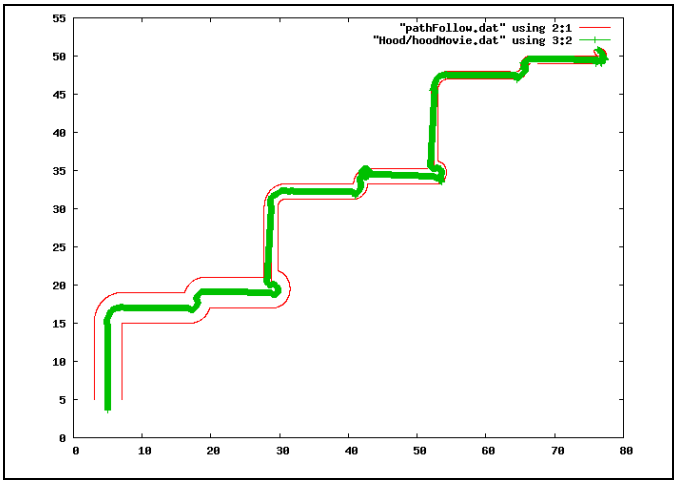


Figure 6: Plot of robot centerline trajectory and error boundaries. Note that allowable path deviation decreases as with distance traveled.

File Information			
	Path data file	VMAC_CompDay1/pathPts.dat	
	Tean's data file	VMAC_CompDay1/team_logs/team1_v4.dat	
Path Information			
	Commanded Speed	0.4	
	Path Length	116.795183	
	Vehicle Path Deviation Normalizing Factor	1.167	
Performance Evaluation of Deviations Over Entire Run			
	Number of Deviation Outside Error Bound	7	
	Total Time of Deviation Outside Error Bound	6.1981	
	Completed Entire Path	TRUE	
Performance Evaluation of Path Following Over Entire Run			
	Total Time of Run	401.0019	
	Distance Traveled During Run	153.004972	
	Distance Traveled Along Path During Run	116.102756	
	Mean of Speed During Run	0.381603	
	Standard Deviation of Speed During Run	0.123377	
	Mean of Deviation During Run	0.266681	
	Standard Deviation of Deviation During Run	0.284552	
Performance Evaluation of Path Following Until First Deviation			
	Total Time of Run	216.8005	
	Distance Traveled During Run	81.185515	
	Distance Traveled Along Path During Run	64.84049	
	Mean of Speed During Run	0.374628	
	Standard Deviation of Speed During Run	0.1308	
	Mean of Deviation During Run	0.304629	
	Standard Deviation of Deviation During Run	0.327008	
Summary of Results			
0.7	Percentage Completed	55.51640773	38.86149
0.1	Deviation of Speed from Commanded Speed	-39.043	-3.9043
0.1	Deviation of Vehicle from Commanded Path	-54.12485004	-5.412485
1	Completion Bonus	10	10
total			39.5447

Figure 7: Sample of statistics gathered for each run of the path following elemental evaluation.

3.1 Follow-on with a Real Platform

One of the main criticisms of simulation is that algorithms that work in virtual reality do not translate well to the real-world. The VMAC competition tried to address this concern by allowing the winning team's code to run on the NIST surrogate AGV. This transition was made possible due to USARSim and MOAST providing validated models of real-world sensors and robotic platforms. In addition, the MOAST implementation provides for standardized interfaces that are followed by both our real platform and the simulation.

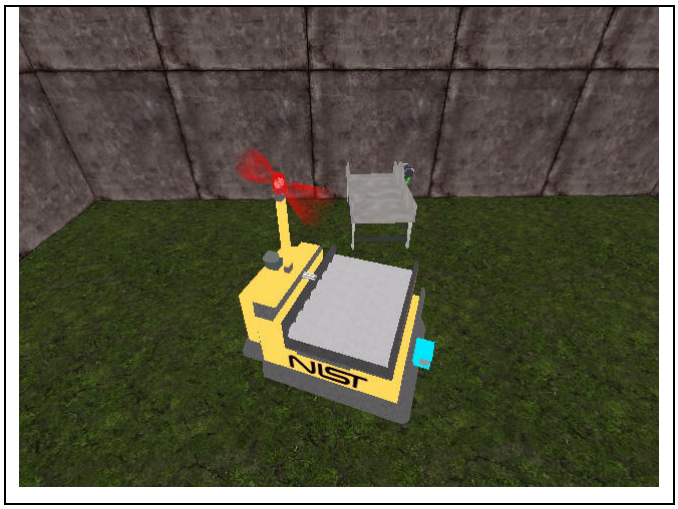


Figure 8: Example of the docking task. The vehicle was required to dock with a loading station without experiencing any collisions. Upon success, the vehicle was moved to a space with less maneuver room and the test was repeated.

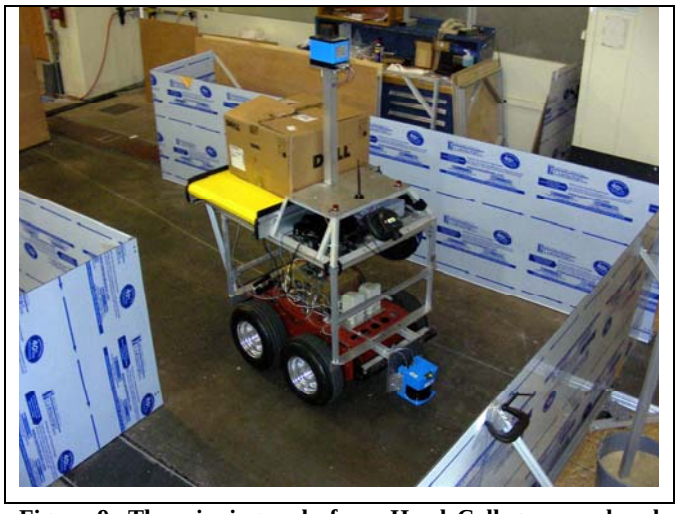


Figure 9: The winning code from Hood College was placed without change on NIST's surrogate AGV. The vehicle was able to successfully navigate and dock with a loading area.

Significant differences exist between the NIST surrogate vehicle and the simulated AGV. For example, the simulated AGV is a rear-steered Ackerman vehicle while the NIST surrogate is skid steered. However, the modularity of the 4D/RCS architecture makes many system specific details irrelevant to higher echelons in the architecture. A constant curvature arc sent from the AM echelon to the PRIM echelon must be accurately followed. AM is unaware of how PRIM will follow that arc. It is up to PRIM to account for the individual vehicle differences.

Figure 9 shows the NIST AGV driving through a performance test metric that allows for vehicle side clearance and free space for turns to be precisely controlled. The Hood College code was run without alteration on the NIST vehicle and docking procedures were successfully performed.

3.2 ICRA Participation

The successful National VMAC has been used as a model for a virtual manufacturing competition that is now an annual event at the robot challenge which is part of the IEEE International Conference on Robotics and Automation (ICRA). The first step to making this a reality occurred recently when with funding from IEEE, the organizers and the winning team from Hood College were able to demonstrate their code to all attendees for the first two days of the ICRA conference that was held in Pasadena, California from May 19-23, 2008.

4. FUTURE

The organizers are continuing this highly successful joint RAS/NIST effort with already approved follow-on funding from RAS. The teams from this year's competition are demonstrating and presenting their research as team-description papers during a special session at the annual Performance Metrics for Intelligent Systems (PerMIS'08) Workshop held at NIST. The seeds from last year's funds are already bearing fruit. Several of the Universities are planning courses based on the material and expertise that they have gained as a direct result of their participation in this competition. In addition, a public forum and website (www.vmac.hood.edu) has been created where participants are able to post and answer questions and initiate discussions with other members. The continued funding will facilitate expanding the user base and building a self-sustaining community that is able to pool resources and create improvements to the simulation system. Additional corporate interest is also being generated with a major manufacturer being interested in expanding the competition to include the autonomous handling of pallets by a forklift in the factories buffer zone area (the area where shipments are first received and unloaded).

In terms of the actual competition, a national competition is being planned for the 2008-2009 year and the organizers have submitted a proposal requesting that the VMAC be included as an event in the robot challenge at ICRA'09 to be held in Kobe, Japan. As a lead-in for next year's national competition, all of the Hood code has now been checked-in to the CVS repository of MOAST.

5. ACKNOWLEDGMENTS

The authors would like to thank NIST and the Robotics and Automation Society (RAS) of IEEE for their continued support of this effort. This work was made possible through a grant received under the RAS new initiatives competition.

6. REFERENCES

[1] MOAST Homepage. http://moast.sourceforge.net . 2007.
[2] USARSim Homepage. http://usarsim.sourceforge.net . 2007.
[3] RoboCup Rescue Virtual Robots Competition. http://www.robocuprescue.org/wiki/index.php?title=Virtualrobots . 2008.
[4] Albus, J., "Outline for a Theory of Intelligence," *IEEE Transactions on Systems Man and Cybernetics*, Vol. 21, 1991, pp. 473-509.
[5] Albus, J., Bostleman, R., Madhavan, R., Scott, H., Barbera, A., Szabo, S., Hong, T., Chang, T., Shackleford, W. P., Shneier, M., Balakirsky, S., Schlenoff, C., Huang, H., and Proctor, F., "Intelligent Control of Mobility

Systems," *Computational Intelligence in Automotive Applications*, Vol. 132/2008, Studies in Computational Intelligence, Springer Berlin, Heidelberg, 2008, pp. 237-274.

[6] Balaguer, B. and Carpin, S., "Where am I? A Simulated GPS Sensor for Outdoor Robotic Applications," *Proceedings of the First International Conference on Simulation, Modeling, and Programming of Mobile Robots*, 2008.

[7] Balakirsky, S. and Scrapper, C., "Planning for On-Road Driving through Incrementally Created Graphs," *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, 2004.

[8] Lewis, M., Sycara, K., and Nourbakhsh, I., "Developing a Testbed for Studying Human-Robot Interaction in Urban Search and Rescue," *Proceedings of the 10th International Conference on Human Computer Interaction (HCII'03),* 2003.

[9] Pepper, C., Balakirsky, S., and Scrapper, C., "Robot Simulation Physics Validation," *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop,* 2007.

[10] Schlenoff, C., Steves, M., Weiss, B., Shneier, M., and Virts, A., "Applying SCORE to Field-based Performance Evaluations of Soldier Worn Sensor Technologies," *Journal of Field Robotics*, Vol. 24, No. 8-9, 2006, pp. 671-698.

[11] Sycara, K. and Pannu, A. S., "The RETSINA multiagent system (video session): towards integrating planning, execution and information gathering," *Proceedings of the Second International Conference on Autonomous Agents*, ACM, New York, NY, 1998.

[12] Taylor, B., Balakirsky, S., Messina, E., and Quinn, R., "Design and Validation of a Whegs Robot in USARSim," *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, 2007.

[13] Wang, J., Lewis, M., and Gennari, J., "A Game Engine Based Simulation of the NIST Urban Search & Rescue Arenas," *Proceedings of the 2003 Winter Simulation Conference*, 2003.

Analysis of a Novel Docking Technique for Autonomous Robots

Henson, G., Maynard, M., Liu, X., Dimitoglou, G.
Department of Computer Science
Hood College
Frederick, MD 21701, USA
{gah1, mmaynard, xliu, dimitoglou} @hood.edu

ABSTRACT

Autonomous robot docking requires accurate path following and ultimately, accurate alignment with the target location, typically a docking station. In this paper we develop a novel, partially heuristic, technique that allows accurate docking for Ackerman-steered vehicles. The technique is validated via simulation. Multiple experiments were performed to better understand and analyze the heuristic element of the technique. The results underscore the impact of the vehicle's steering characteristics in docking precision and may prove valuable in attempting to remove the heuristic element.

Categories and Subject Descriptors

I.2.9 [**Robotics**]: Autonomous vehicles; I.6 [**Simulation and modeling**]: Applications

General Terms

Robotics, Simulation

Keywords

Mobile robot motion-planning, simulation

1. INTRODUCTION

Autonomous mobile robots are designed to perform certain tasks. Typically to perform these tasks, robots have to plan their path, avoid objects and navigate in the operational environment. Often task completion requires approaching a target location where some additional final operation is required, such as moving an actuator. In manufacturing environments docking to a target location (e.g. a palette shelf or a conveyor belt) is a common final operation. The docking process itself can be complex, requiring maneuvering and precision which can be affected by the size and shape of the workspace and the physical dimensions and maneuverability of the robotic platform.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08 August 19-21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

In this paper we examine docking of an autonomous robotic platform by knowing only two locations in the workspace: the point of origin and the target location. We propose an algorithm that allows docking using only numerical calculations considering, besides the robot's physical geometry, the origin and target location. We presume the robot has no proximity detection capability or any other relevant sensor payload and the workspace is obstacle-free.

The following section provides a description of the environment and robotic platform assumptions and constraints for the docking problem. In Section 3, the proposed technique is presented. Section 4 describes the simulation experiments conducted to confirm the validity of the proposed algorithm and to collect some performance metrics with respect to docking accuracy.

2. OPERATIONAL ENVIRONMENT

We assume the operational environment to be a rectangular workspace (W). In the workspace there is a docking location (D) and an arbitrary starting location (L_1) for the robot (A). The robot, an Ackerman-steered, Automatic Guided Vehicle (AGV) is fitted with a roller deck.

The docking location also has a roller deck which should enable the AGV to unload cargo if it is properly docked. An area in front of the dock is also the target location (L_D) the robot must reach by maneuvering in the obstacle-free workspace (Figure 1). Since there are no obstacles in the workspace, there are many more paths [3] in the configuration space (C-space) that can be expressed as a continuous map:

$$\tau\colon S \in [0,1] \mapsto \tau(s) \in C \tag{1}$$

where $\tau(0) = L_i$ is the initial configuration and $\tau(0) = L_D$ is the target configuration of the path. However, Equation 1 does not take into account the additional requirement of arriving at the dock in the correct orientation (pose).

For task completion, the robot needs to arrive at the target location L_D and also to park in such a way that the roller decks of both the vehicle and the dock are aligned. This excludes any docking approach that does not conclude with the left wide side of the vehicle aligned to the dock's roller deck.

In the environment, the following is known:

1. the coordinates of the four corners defining the workspace and the room boundary
2. the coordinates of the corners defining the dock
3. the initial (L_1) and target (L_D) robot locations

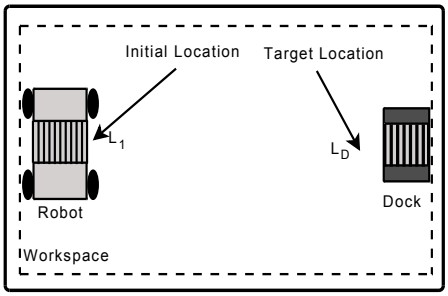


Figure 1: A generalized view of the workspace with the robot in its initial position.

4. the initial orientation (pose) of the robot and its physical dimensions.

The point we will use as the location of the robot in the world during the simulation is between the front wheels.
The side to side location is calculated by centering the loading site on the UnitLoader to the center of the docking station.

3. APPROACH

The problem of docking the robot is the same as parallel parking a vehicle next to a fixed location.
We begin by calculating the required maneuvers in reverse order. This simplifies the problem as we obtain both the final position and the required orientation. We can then calculate the points that form a traversable path to the robot's position.
The key point in this path is the point that the robot travels to immediately before arriving at the docked position. We call this this the *penultimate point* (p) and it is chosen to provide a simple traversable path to the docking position as well as being easily reachable from the robot's current location.
The final docking position the robot needs to obtain is

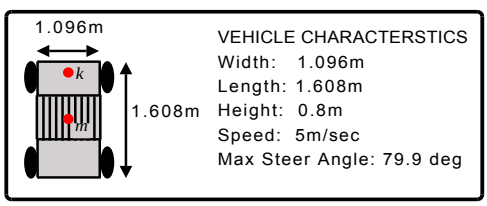


Figure 2: Robot physical dimensions.

calculated using the physical dimensions of the robot and the location of the docking station. Because the navigation coordinates are calculated using a point (k) in the front mid-line of the robot, another point (m) is calculated to determine the point of alignment between the center of the dock and the center of the vehicle's roller deck (Figure 2).
The docking target location is half of the robot's width from the docking side of the docking station. Once this docking point is calculated, a penultimate point can also be calculated.
The penultimate point also needed to be located in such a place as to allow the robot to easily reach the docking location.
To take advantage of the vehicle's rear steered Ackerman design, we chose to approach the docking position using the robot's reverse direction. This put the steering wheels in the front and allowed a smoother motion to pivot to the docking location.
To accomplish this, the penultimate point can be on either side (left or right) of the docking location to allow the robot to reverse into the correct position. The optimal L_D point is determined according to the physical dimensions of the docking station and the physical geometry of the vehicle.
Algorithm 1 illustrates in pseudocode the required steps in order to calculate the x and y coordinates for both the penultimate (p) and goal point (L_D). The terms $half_robot_length$ and $half_robot_width$ are measurements that are used to properly align the vehicle and its roller deck to the dock and its roller deck.
Once these calculations are made, function FollowPath performs the steering and direction maneuvers for the vehicle to begin moving first towards the calculated penultimate point and then to the goal point.
While the p is a single reachable point in the workspace, we consider it to have a much larger footprint indicated by a *tolerance*, which is an arbitrary radius around the center of p.
The goal point, however, does not have such tolerance as any deviation may indicate a misalignment with dock, rendering the unloading of any package impossible.

Algorithm 1 Calculate and Find the penultimate point p

$p.x \leftarrow dock.center.x + robot_length$;
$p.y \leftarrow dock.center.y + half_robot_length$;
$L_D.x \leftarrow dock.center.x + rollerdeck_nav_point_offset$;
$L_D.y \leftarrow dock.center.y + half_robot_width$;
repeat
  FollowPath towards p
  $distance = current.location - p$
until distance $< p_t$
repeat
  FollowPath towards L_D
  $distance = current.location - L_D$
until distance $< p_t$

4. SIMULATION

4.1 Experimentation

The developed docking algorithm was exercised, tested and verified using a simulation infrastructure comprised of USARSim [5, 7], a Gamebots [2] variant of the Unreal Engine [6] and the Mobility Open Architecture Simulation and Tools (MOAST)[4] framework [1].

4.1.1 Design

The simulation environment provides a number of configurable parameters for the vehicle that can be adjusted to modify the navigation performance.
For the experiments a Unit Loader AGV was used and all parameters were kept at their default settings. As we were more concerned with docking accuracy rather than the path following speed or reducing task completion, none of these parameters were altered. Velocity, in particular, remained constant at 5m/sec.

The critical element to the success of a docking attempt is the determination of the penultimate point.
The single success factor in this experiment is the accurate docking of the vehicle to the dock. As the penultimate point (p) is arbitrarily determined with respect to the dock, we examined the possibility of moving p to examine if different locations would yield more accurate results. Through experimentation, we determined the optimal (fully aligned) goal point (L_D) to be at xy-location (-0.26, -2.36).
Assuming a constant vertical distance from the initial location (L_1) to the p, multiple points were examined on the horizontal axes by increasing and decreasing the distance to the dock.
The intention of the experimentation was to measure the performance of the vehicle with respect to its accuracy in arriving at the optimal location by changing the proximity of p.
The expectation was that a more distant p may result in more accurate docking. This expectation was based on how coarse Ackerman-steered vehicles are when maneuvering, therefore more distance from the dock should have resulted in a more "comfortable" steering environment.

4.1.2 Execution & Analysis

In Table 1, the coordinates of seven executions illustrate the different p positions and their corresponding L_D locations. In the simulated workspace, the optimal docked L_D is (-0.57, -2.39).
The results indicate a considerable spread of values in the horizontal distance from the optimal L_D x-location of -0.57 with deviations as large as 0.62m (ID 2).

Table 1: x,y coordinates of various p points and their corresponding final docking locations

ID	Penultimate (p) Location	Final Location (L_D)
1	(-1.57, -2.39)	(-0.03, -2.40)
2	(-1.37, -2.39)	(0.05, -2.39)
3	(-1.17, -2.39)	(0.04, -2.39)
4	(-0.97, -2.39)	(-0.26, -2.36)
5	(-0.77, -2.39)	(-0.15, -2.17)
6	(-0.67, -2.39)	(-0.18, -2.04)
7	(-0.57, -2.39)	(-0.29, -1.96)

These effects are most prominent in Figure 3. Superimposed on an xy-axis, seven penultimate points ($P_1 - P_7$) are examined on the horizontal axis.
Penultimate point P_1 enables the vehicle to reach and dock as L_{D4} which is the optimal target location L_D. Subsequent executions with p closer to the dock (e.g. $P_5 - P_7$) result in far L_D's. Similarly, executions with p further away from the dock (e.g. $P_1 - P_3$) also end up in points distant from the dock L_D's.
As expected, L_{D4} is aligned with the center of the dock (D_{center}) but not adjacent to it. Because L_{D4} is calculated to be in the middle of the vehicle, a distance is required to ensure that at least half of the vehicle's width and some additional space is allotted to ensure proper proximity but not collision with the dock.

5. CONCLUSION AND FUTURE WORK

Autonomous robot docking is an interesting problem with many practical applications.

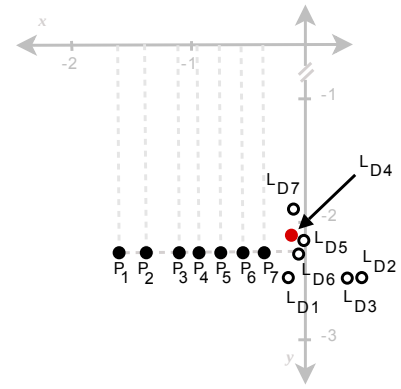


Figure 3: Graphical depiction of multiple simulation executions with varying penultimate points.

However, specific vehicle characteristics that may affect steerability may render the application of the technique more challenging and far less accurate than expected. Using an Ackerman-steered vehicle provided such a challenge.
To address the problem, a simple technique was developed that enables Ackerman-steered vehicles to accurately dock. The technique is based on the assumption that certain environment and workspace information is known. The key element of the techniques is heuristically determining a penultimate point (p) off the final docking destination that enables the vehicle to maneuver and essentially parallel park next to the docking station. The technique was verified using a simulation environment and the proposed algorithm performed as expected in the synthetic workspaces. As the determination of p is based on heuristics, numerous different p points on the horizontal axes were examined to assess their viability. The results were somewhat unexpected as docking accuracy varied for both closer and more distant p points.
Further examination is required to better understand the reasons for the variation in accuracy, particularly for distant points, as well as more work for the further analysis of the collected data.
Beyond the simulated experimentation, the proposed docking technique was also tested using a physical platform, an actual AGV. Albeit the vehicle was skid-steered, therefore it had a greater degree of maneuverability, the results ware congruent with the expected behavior and the simulations presented in this paper: the vehicle docked perfectly next to the dock.
We anticipate that a deeper examination of the observed vehicle behavior will enable the development of intelligent, adaptive and non-heuristic-based methods for the calculation of one or more penultimate points.

6. ACKNOWLEDGMENTS

The authors are grateful to S. Balakirsky (NIST), R. Madhavan (NIST/ORNL) and C. Scrapper (NIST) for their guidance with the Virtual Manufacturing & Automation Competition (VMAC). The authors also appreciate the continuous and generous support provided by Dean Bands and the Hood College Graduate School.

7. REFERENCES

[1] S. Balakirsky and E. Messina. MOAST and USARSim -

a combined framework for the development and testing of autonomous systems. In *Proceedings of the 2006 SPIE Defense and Security Symposium*, 2006.

[2] G. A. Kaminka and M. M. Veloso et al. Gamebots: a flexible test bed for multiagent team research. *Commun. ACM*, 45(1):43–45, 2002.

[3] J-C. Latombe. *Robot Motion Planning*. Springer, 1990.

[4] MOAST project. *http://moast.sourceforge.net*, accessed June 1 2008.

[5] S. Carpin M. Lewis S. Balakirsky, C. Scrapper. USARSim: providing a framework for multi-robot performance evaluation. In *PerMIS 2006*, 2006.

[6] Unreal Engine (UNR). *http://www.epicgames.com*, accessed June 1 2008.

[7] USARSim. *http://sourceforge.net/projects/usarsim*, accessed June 1 2008.

Partitioning Algorithm for Path Determination of Automated Robotic Part Delivery System in Manufacturing Environments

Payam Matin
Department of Engineering & Aviation Sciences
University of Maryland Eastern Shore
Princess Anne, MD-21853
001 (410)-621-3020
phmatin@umes.edu

Ali Eydgahi
Department of Engineering & Aviation Sciences
University of Maryland Eastern Shore
Princess Anne, MD-21853
001 (410)-651-7559
aeydgahi@umes.edu

Ranjith Chowdary
Department of Mathematics & Computer Science
University of Maryland Eastern Shore
Princess Anne, MD-21853
001 (601)-818-4491
rkpoldas@umes.edu

ABSTRACT
This paper presents a student research project as a supplement to the undergraduate curriculum for students in the area of Science, Technology, Engineering and Mathematics. The goal of the project presented in this paper is to provide students with experience in design of a routing path for an automated robotic part delivery system in manufacturing environments. Despite all advancement in automation of manufacturing processes, still there are many challenges that need attentions in automating part transformation between manufacturing stations on an unstructured plant. In this study, first a mathematical model of a plant layout containing static obstacles is constructed. A partitioning algorithm is then introduced that partitions the layout into obstacle-free regions. Next, a searching algorithm is utilized to yield all possible combinations of regions that can connect a starting point to a destination throughout the plant layout. Finally, physical paths are constructed by drawing line segments within the obstacle-free regions and through the intersections between the regions. A program in C language has been developed that accommodates the algorithm introduced. Excellent level of success has been measured in the performance of the designed algorithm and program code. This study could be potentially beneficial to the industry since it tends to remove the labor costs.

Categories and Subject Descriptors
I.2.9 [**Artificial Intelligence**]: Robotics – *autonomous vehicles*

General Terms
Algorithms, Design.

Keywords
Automated Guided Vehicle (AGV), Mobile Robot, Autonomy and Manufacturing.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

1. INTRODUCTION
Automated Guided Vehicles (AGVs) represent an integral component of today's manufacturing processes. They are widely used on plant floors for intra-factory transferring of parts between conveyors and assembly lines, loading, and unloading. The use of Automated Guided Vehicles aligns to the philosophy of flexible manufacturing systems, where 24-hour-a-day use of machines rather than human is utilized for performing repetitive tasks. This approach helps reduce manufacturing cost and increase efficiency in a manufacturing system [1]. Although the use of AGVs is helpful, operating them in an unstructured manufacturing environment could be technically challenging and economically very costly. Challenges associated with design and operation of AGVs may be named as routing, navigation and guidance, traffic management, load transfer, and system management [2]. The first challenge in design of an AGV is routing, which refers to system's ability to make decision in order to select optimum routs to specific destination. If optimum routs are selected using a routing system, the guided vehicle is supposed to follow the predetermined rout. Navigation and guidance system is designed to make this possible. As an Automated Guided Vehicle navigates through a path, it should be able to avoid collisions with other operating vehicles and obstacles on the field. Traffic management system addresses such important task. Furthermore, since the system should be able to load and unload goods, a load transfer system should manage these operations. Finally, a system management is needed to have full provision over all components of the systems for efficiency in operation. Despite all advancement made in AGV systems, many challenges still remain unsolved, which could be viewed as an opportunity for more advancement and development in this area. Also, as these systems are becoming more popular in today's industry, any developmental activity by students in this area that addresses existing challenges of various parts of the AGV systems would contribute to the educational and learning experience of students.

A comprehensive effort has started by a team of engineering students at UMES to design and prototype a Robotic System that will completely automate part transformation between different manufacturing stations. To accomplish this task, real-world data from industry part transformation scenarios are used to design and prototype a Robotic Part Delivery System. This project helps

students get exposed to a number of emerging areas in Mechanical Engineering, Electrical Engineering, and Computer Science. They also gain experience in making a working prototype system, doing engineering analysis, and creating engineering documents / drawings as required for manufacturing of a Robotic System.

This paper focuses more on the modeling approach that has been taken to design the routing algorithm of a part delivery system.

The objectives of this work include:

- To make a realistic model of a manufacturing environment layout,
- To develop an algorithm for routing purposes,
- To implement the algorithm to a number of realistic scenarios for validation purposes.

2. MODELING APPROACH

The routing system is an algorithm that makes decision in order to select optimum path to a specific destination from a starting location. Before the routing system can be discussed, a model of plant layout should be considered. A plant layout may be generally viewed as what has been depicted in Figure 1. The plant boundary is considered as a rectangle. The obstacles such as partitioning walls or machines are considered as vertical and horizontal lines located within the boundary. All obstacles are assumed to be static. No dynamic obstacle has been considered at this stage of the study.

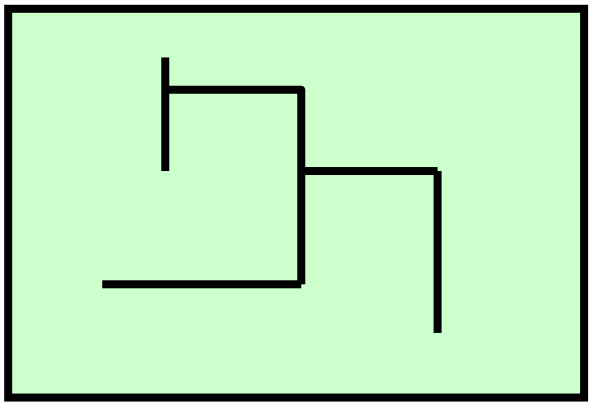


Figure 1. Plant layout model with static obstacles such as partitioning walls and machines

The routing system is an algorithm that makes decisions on what path the part delivery system should take to avoid the obstacles and reach the final destination from the starting location. For the plant layout model presented, such path may be shown in Figure 2 from the starting location to the final destination.

The path selection problem would be fairly simple if no obstacles were located within the boundary. In such case, the path would simply be the straight line connecting the starting to the ending point as shown in Figure 3.

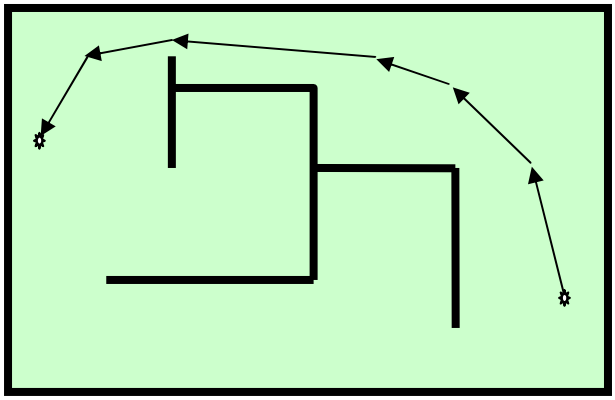


Figure 2: Obstacle-avoiding path from the staring location to final destination

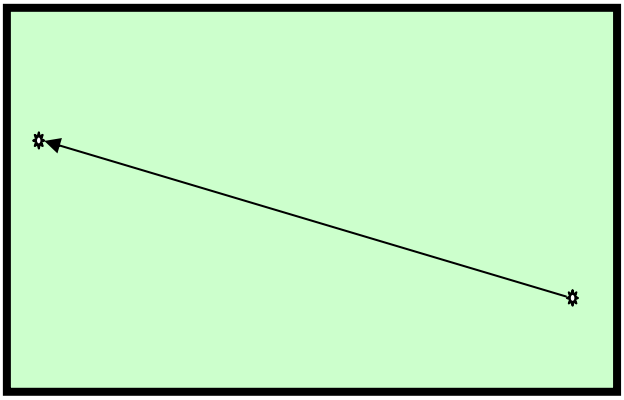


Figure 3: Path to the destination from the starting location in an obstacle-free layout

If the layout is partitioned to obstacle-free regions, the path selection process will be as simple as straight line selections within each region. Traversing between the regions will be conducted through the intersections between the obstacle-free regions. Figure 4 shows the plant layout partitioned into obstacle-free regions based on the locations of the existing obstacles in the layout.

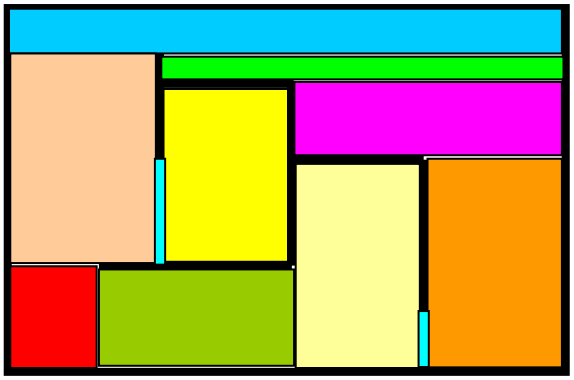


Figure 4: Plant layout partitioned to obstacle-free regions based on the locations of obstacles

The layout partitioning algorithm is discussed in details. Generally, the idea is to block the areas between the plant boundaries and the wall obstacles such that no obstacle is left within.

To accommodate the partitioning algorithm, the plant layout should be discritized and meshed into rectangular grid as shown in Figure 5. The robotic system can only operate through the grid points on the layout excluding the ones on the boundaries and wall obstacles. These points may be referred as free points as opposed to close points on the boundaries and obstacles. Mathematically, free points may be viewed as 1s as opposed to close points that may be viewed as 0s. Thus, mathematical representation of the layout is achieved by constructing a binary matrix called layout matrix using grid information. As shown in Figure 6, in such matrix, 1s are placed in the corresponding locations of the open points on the layout. In contract, 0s are placed in the corresponding locations of the close points on the layout.

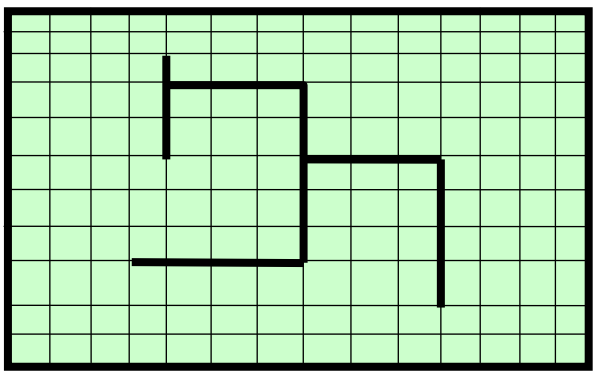


Figure 5: Plant layout meshed into rectangular grid

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	0	1	1	1	1	1	1	1	1	1	0
0	1	1	1	0	0	0	0	1	1	1	1	1	1	0
0	1	1	1	0	1	1	0	1	1	1	1	1	1	0
0	1	1	1	0	1	1	0	0	0	0	1	1	1	0
0	1	1	1	1	1	1	0	1	1	0	1	1	1	0
0	1	1	1	1	1	1	0	1	1	0	1	1	1	0
0	1	1	0	0	0	0	0	1	1	0	1	1	1	0
0	1	1	1	1	1	1	1	1	1	0	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 6: Plant layout matrix;
0s on the sides represent the plant boundary;
0s within the boundaries represent obstacles;
1s represent free grid points

Layout matrix contains all the information about the boundaries, obstacles, and free points on the plant layout. From now on, rather than working with physical geometry of the layout, the algorithm is managed by processing such binary matrices leading to more simplicity. In order to partition the layout matrix to free-obstacle regions, the areas which only contain 1s and are restricted by 0s on sides should be selected and blocked as shown in Figure 7. To accomplish this, following steps should be taken:

1- For the very top available row (the first row below the plant top boundary), successive 1s are blocked into one dimensional horizontal blocks called region-heads that are free of obstacles. If there is no zero on the row, the whole row is blocked into just one single region-head (as shown in Figure 7 by red dash block). If there were 0s on the row, multiple region-heads would be obtained that would belong to this particular row and be separated by the 0s.

2- Once the region-heads of the very first available row are obtained, in order to construct the free-obstacle regions, they have to be extended downward to the levels that restrict them by 0s. (In Figure 7, the very first available row can not be extended any downward since there is a 0 in the row below; Thus the region-head itself is selected as a region)

3- The information on the regions selected is stored using column and row numbers of the sides of the regions (that belong to the first available row). The regions are also numbered globally.

4- Steps 1 through 3 are repeated for the next row below the very top available row and so on and so forth as shown in Figure 7. It should be noted that every time that step 1 is repeated for a new row, the algorithm escape from the regions that have been already blocked, and only construct the region-heads where is still free of regions.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	0	1	1	1	1	1	1	1	1	1	0
0	1	1	1	0	0	0	0	1	1	1	1	1	1	0
0	1	1	1	0	1	1	0	1	1	1	1	1	1	0
0	1	1	1	0	1	1	0	0	0	0	1	1	1	0
0	1	1	1	1	1	1	0	1	1	0	1	1	1	0
0	1	1	1	1	1	1	0	1	1	0	1	1	1	0
0	1	1	0	0	0	0	0	1	1	0	1	1	1	0
0	1	1	1	1	1	1	1	1	1	0	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 7: Layout partitioning algorithm;
Region heads depicted in red dash-lines;
Free-obstacle regions depicted in rectangular block;

After all the steps explained are implemented, the layout matrix is partitioned into obstacle-free regions with global numberings shown in Figure 8. This partitioned matrix is the mathematical representation of the plant layout shown in Figure 4.

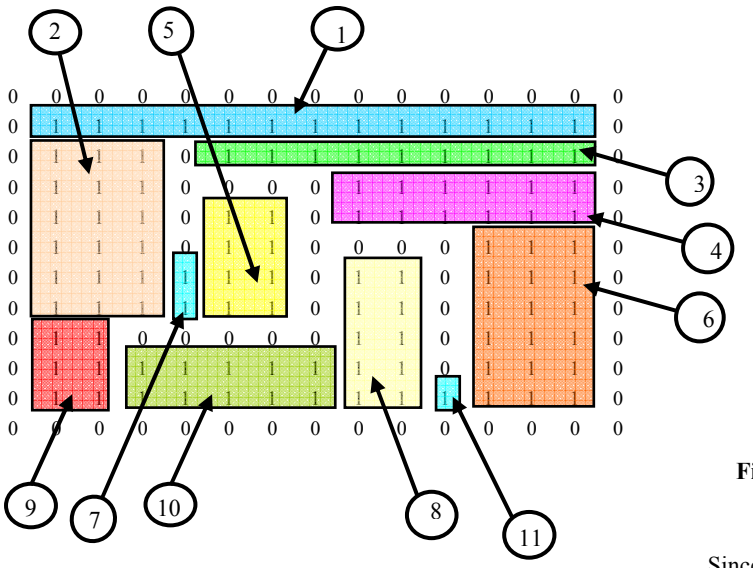


Figure 8: Layout matrix partitioned with free-obstacle regions along with their global numberings

Since the robotic system traverses from a region to another through the intersections between the regions, it is important to extract the information on the intersecting regions from the layout matrix. For this purpose, the possibility of intersection is considered for each and every region in the layout matrix with the other regions. For instance, an intersection on the lower side of region # i may be identified if there is a region such as region # x that satisfies the following conditions:

(The lower row number of region # i) +1 =
(The upper row number of region # x)

And

Left column number of region # i <
(left column number of region # x) <
right column number of region # i

OR

Left column number of region # i <
(right column number of region # x) <
right column number of region # i

Similar conditions may be set up to identify the intersection on all other 3 sides of region # i. The information about intersection is completed as theses conditions are checked one by one for all the existing regions in the layout. At the end, it will be known what regions intersecting what other regions and where the intersecting points are as far as row and column numbers are concerned. A part of this information is stored in a binary matrix called intersection matrix shown in Figure 9. Intersection matrix indicates whether or not each region intersects with the other obstacle-free regions in the layout.

In intersection matrix, 1 in ij location is the indication of intersection between region # i and region # j. In contrast, 0 in ij location indicates that there is no intersection between these regions.

	1	2	3	4	5	6	7	8	9	10	11
1	0	1	1	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	1	0	1	0	0
3	1	0	0	1	0	0	0	0	0	0	0
4	0	0	1	0	0	1	0	0	0	0	0
5	0	0	0	0	0	0	1	0	0	0	0
6	0	0	0	1	0	0	0	0	0	0	1
7	0	1	0	0	1	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	1	1
9	0	1	0	0	0	0	0	0	0	1	0
10	0	0	0	0	0	0	0	1	1	0	0
11	0	0	0	0	0	1	0	1	0	0	0

Figure 9: Intersection matrix representing the info on the intersecting free-obstacle regions

Since the intersection matrix contain the information on the intersecting regions, and since the only way of traversing between the regions is through the intersections, the intersection matrix may be used to find all the possible combinations of regions that construct a path from a starting region to a destination region. That could be accomplished by repeated searching of 1s from a starting region to a destination region as shown in Figure 10.

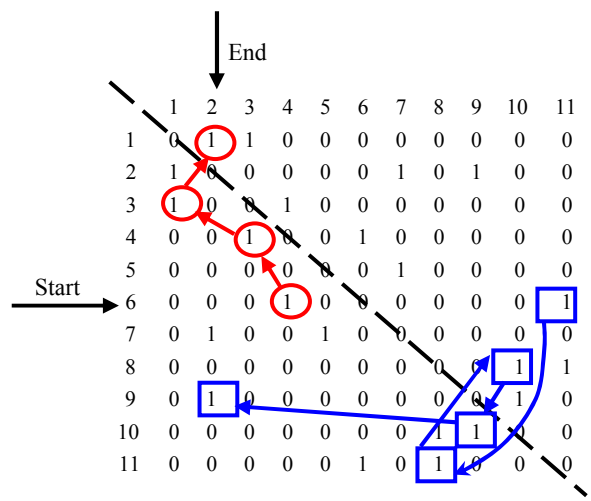


Figure 10: Repeated search algorithm to find all possible combinations of regions constructing a path from a starting region to a destination region

In this figure, the intension is to start from region # 6 and end up at region # 2. Starting from row # 6, two 1s are identified. The first one is on the 4th column and the other one is on the 11th column. These 1s are indicating that region # 6 has intersection with region # 4 and # 11. (This can also be seen in the partitioned layout matrix shown in Figure 8.) Each of these regions may be viewed as a starting region of an independent path that may be preceded by other regions to the final destination region. For instance, let's consider the path starting with region # 4. The repeated search should continue by looking at row # 4 identifying region # 3 as the next intersection region (which can be also

observed in Figure 8 again). The searching algorithm continues all the way to the final destination, region # 2. The order of the regions found for both independent possible paths is shown in Figure 10. Alternatively, these paths are shown as a tree in Figure 11.

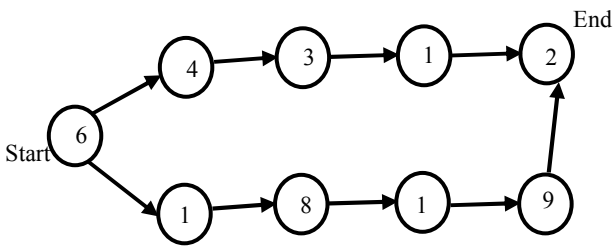


Figure 11: Path tree summarizing all possible paths from a staring region to destination

Now that the order of traversing the regions from the starting region all the way to the final destination is obtained, the physical path that connects the regions from a starting point in the starting region to the destination point in the destination region may be constructed. For this purpose following notes should be considered for path construction:

1- The Path is constructed of different line segments
2- Line segments are either within the free-obstacle regions or crossing over the intersections
3- Line segments are straight lines connecting only 1s (open points on the grid)

Considering these points, following steps should be taken to construct a path from the starting point to the destination as shown in Figure 12:

1- A line segment is constructed from the starting point located at the starting region to a targeted point on the intersection with the following region. The targeted point on the intersection could be the point having shorter distance to the destination relative to the other points on the intersection. (Targeted point still belongs to the starting region).

2- A line segment is constructed by crossing over the intersection to a new targeted point in the following region. This new targeted point is located immediately next to the point (on the intersection) which corresponds to the previous targeted point.

3- Steps 1 and 2 are repeated for the following regions until getting into the region immediately before the final destination region.

4- A line segment is constructed to the point on the intersection that provides shortest distance to the final destination.

5- Final line segment to the final destination is obtained.

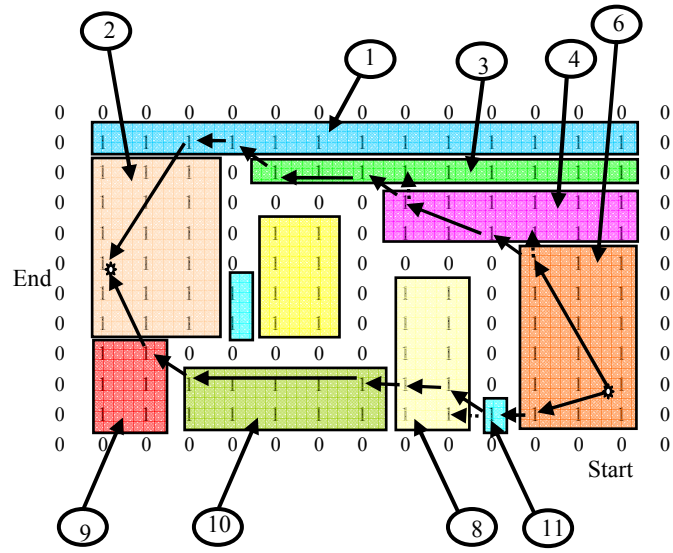


Figure 12: Paths constructed from the starting point to the destination

The path construction algorithm is repeated for all possible paths summarized in path tree. Once all physical paths are determined, the optimum path may be selected. The selection process could be based on shortest distance criterion. In such case, the total path length could be calculated by calculating the length of each line segment. The selection process could also be based on shortest time criterion. That involves more complexity. The average velocity of the robotic system along with starting acceleration, and ending deceleration should be considered for each line segment. The angles between the line segments and their subsequent effects on the velocity should also be considered. More complexity may even come into the play if the path selection process is based on both distance and time. That involves solving an optimization problem with a clearly defined objective.

3. RESULTS

A program in C language has been developed that accommodates the algorithm discussed. The plant layout, the starting point and the destination point are the inputs to the program. The program partitions the layout, and comes up with all possible combinations of regions that a path could be constructed through the regions. The path is routed and outputted by the program. A number of layout containing different shapes of obstacles have been tried out. Excellent level of success has been measured in the performance of the designed algorithm and program code.

Figure 13 and 14 show the path determined by the code for the layout shown in these figures and different starting points and ending points.

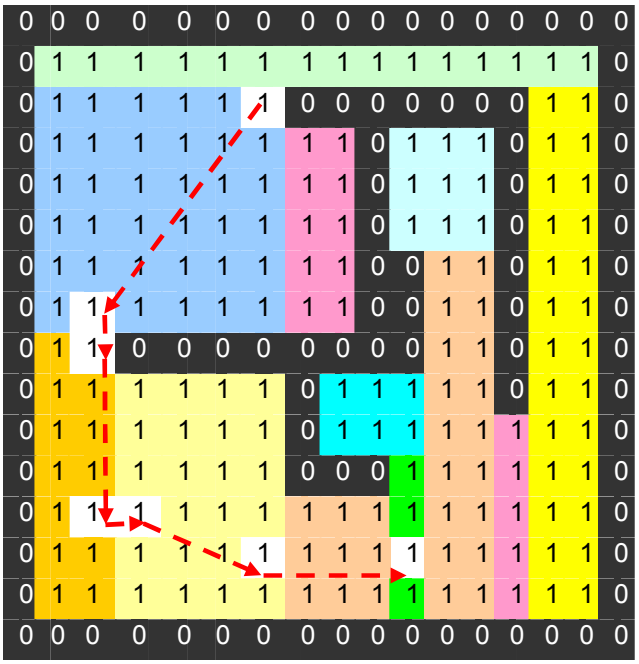


Figure 13: Path determined by the code written

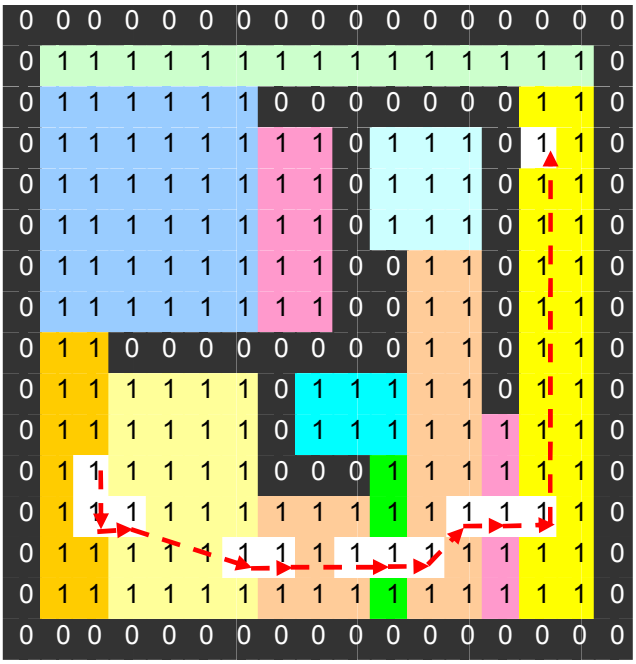


Figure 14: Path determined by the code written

4. FUTURE WORK

Future work involves incorporating in-angle, round, and dynamic obstacles in the algorithm. It also involves improving path construction algorithm such that the path can be updated and smoothened between each line segment.

5. CONCLUSION

A comprehensive effort has started by a team of engineering students to design and prototype a Robotic System that will completely automate part transformation between different manufacturing stations. A routing system has been designed by layout partitioning algorithm that partitions the layout to obstacle-free regions. The algorithm has delivered all possible combinations of regions that can connect the starting point to the destination. The algorithm also constructs the physical paths by drawing line segments within the obstacle-free regions and through the intersections. A program has been developed in C language to implement the algorithm introduced. The program exhibited an excellent amount of success.

6. ACKNOWLEDGMENTS

The Authors acknowledge the support of National Institute of Standards and Technology (NIST) through Virtual Manufacturing Automation Competition workshop. Special thanks go to Dr. Raj Madhavan and Dr. Stephen Balakirsky.

7. REFERENCES

[1] The basic of Automated Guided Vehicles, AGV System, Seimens, March 2006.

[2] Flexible manufacturing systems, University of Kentucky, March 2006.

Algorithms and Performance Analysis for Path Navigation of Ackerman-Steered Autonomous Robots

G. Henson, M. Maynard, G. Dimitoglou, X. Liu
Department of Computer Science
Hood College
Frederick, MD 21701
{gah1, mmaynard, dimitoglou, liu}@hood.edu

ABSTRACT
Autonomous robot navigation is one of the most important areas of robotics research. While waypoint-based navigation is well understood, particular design characteristics (e.g. steering mechanisms) of a robotic platform may render general navigation approaches inaccurate or ineffectual. In this paper, we develop a novel, deliberative-planning, waypoint-based robot navigation algorithm that allows accurate path following for Ackerman-steered vehicles. The algorithm is validated via simulation. Collected performance data provide insight on the relation of the algorithm to other mobility factors such as velocity.

Categories and Subject Descriptors
I.2.9 [**Robotics**]: Autonomous vehicles; I.6 [**Simulation and modeling**]: Applications

General Terms
Algorithms, Performance, Experimentation

Keywords
Mobile robot motion-planning, simulation

1. INTRODUCTION
In terrestrial navigation, waypoints are longitude and latitude-based coordinate locations used to construct routes. In the context of robotic navigation, waypoints can serve as the guides for a robot to follow a specific path. While following waypoints is trivial, idiosyncrasies of the robotic platform and the operating environment may render this type of navigation inaccurate and under some circumstances, ineffectual. An example of such a case is with Ackerman-steered vehicles. The limitation of such vehicles is their inherent inability to perform sharp turns to reach waypoints in very close proximity. Specifically, waypoints located within the vehicle's instantaneous center of rotation (ICR).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

This case presents two navigation challenges. First, it may cause the vehicle to deviate considerably from the prescribed path. Second, it may cause the vehicle to enter an endless spiral path attempting to reach the practically unreachable waypoint. To address these challenges we propose an algorithm that visits all waypoints on an arbitrary projected path, minimizing path deviation, avoiding spiral movements and continuing navigation on the projected path.

The following section includes some background and related work on robot path navigation and Ackerman steering. In Section 3, the proposed algorithm is presented. Section 4 describes the simulation experiments conducted to confirm the validity of the proposed algorithm and to collect some performance metrics. Finally, the performance analysis section provides an analysis of the performance metrics with respect to path accuracy.

2. BACKGROUND
Autonomous navigation of a mobile robot is an interesting challenge since it brings together numerous aspects of a robotic platform such as sensing and actuation while requiring additional computations for planning or dynamic problem solving [11]. The navigation method can be categorized in two categories: one based on reactive behavior and one based on deliberative planning. With reactive behavior, the robot navigates according to up-to-date sensor information, whereas with deliberative methods, a priori knowledge of the environment is used to plan one or more obstacle-free paths towards a goal location. In dynamic environments, reactive behavior methods can on the path) and do not require complete knowledge of the working environment. Deliberative planning methods, on the other hand, assume complete knowledge of the robot's initial and goal points along with any static obstacles in the environment. With such knowledge, the navigation task is to plan paths without collision. Path planning is also an interesting problem and can be roughly categorized into three methods [8]: roadmap, cell decomposition [5, 7, 9] and potential field [3] and [17].

The work presented in this paper is based on the roadmap approach where the path finding problem is reduced to the connection of the start and goal location to a waypoint-based roadmap. While in the literature [16] there have been multiple ways and approaches to generate and follow waypoints, ranging from using pure deliberative or reactive navigation methods to combining them or applying genetic algorithms, our approach is simplified. A complete knowledge of the world exists and a collision-free path is known. After generating waypoints across the path, the challenge is to accurately navigate from the initial

to the goal location. Accuracy is determined by the least deviation of the vehicle from the path. One aspect that narrows down our examination is the use of an Ackerman-steered vehicle which has unique characteristics in terms of its degrees of steerability and maneuverability[13].

3. APPROACH

To address the problem, it is assumed that a path is comprised of one or more path segments as illustrated by points *a*, *b* and *b*, *c* in Figure 1, consisting of constant curvature arcs, with tangential interconnects between the arcs.

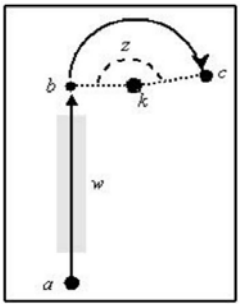


Figure 1. Path segments for navigation.

These path segments are defined a priori and the interconnections are defined by start and end points, generating a full path from an initial point *a* to a goal point *c*. The constant curvature arcs are defined by the center point *k*, the end point *c*, and an angle *z* defining how long the arc extends along the circle. In addition, path segments have *tolerance* (*w*), a virtual width, providing lateral boundaries across the path, indicating how closely the robot is traversing each segment. In terms of the autonomous vehicle, it is assumed that it can travel with velocity (*v*). For a robot to follow a path, algorithm GenerateWaypoints (Figure 2) converts the path segments into waypoints and stores them in a list. Line segments are converted to an initial and a final waypoint. The arcs are also converted to several points along the circle defined by the arc.

```
GenerateWaypoints (in: arcs[];
                   out: intermediate_waypoints[])
while arcs to process do
  Read an arc
    if the arc is a line then
      Calculate distance to goal_point
      Find subgoal_point in space 1m short of goal_point
      Add subgoal_point to intermediate_waypoint list
    else // not a line
      Calculate number of goal_points in arc
      Calculate radial distance between goal_points
      for each radial increment do
        Compute goal_point on the goal curve
        Add goal_point to intermediate_waypoint list
      end for
    end if
end while
```

Figure 2. Pseudocode for waypoint generation.

The GenerateWaypoints algorithm allows the vehicle to generate waypoints and follow them, getting closer to the goal path, and approach the next waypoint in any given arc. However this algorithm is agnostic of the the idiosyncratic steering of Ackerman-steered vehicles that have very coarse turning capabilities and may also enter an infinite spiral around waypoints that are located within the vehicle's ICR.

1.1 The Naive Navigation Algorithm

To address the coarse turn granularity problem and proactively try to maintain some accuracy to the path, the NaiveNavigation algorithm (Figure 3) traverses the list of waypoints and navigates the path while attempting to keep the vehicle's path within the prescribed acceptable deviation *tolerance* (*w*).

```
NaiveNavigation (in: intermediate_waypoints[])
while intermediate_waypoints remain do
  Compute current_location
  Read goal_point from intermediate_waypoints
  Compute distance: current_location to goal_point
  if distance < tolerance then
    Select next goal_point
  end if
  Compute direction: current_location to goal_point
  Steer robot in direction of goal_point
end while
```

Figure 3. NaiveNavigation algorithm pseudo code.

The path to the first waypoint on the list becomes the current goal point and the path and angle to this point is calculated. The vehicle moves and if it passes over the goal point or navigates within the defined path deviation tolerance, the waypoint is considered reached. It gets removed from the list and the next waypoint on the list becomes the new current goal point.

1.2 Enhancing the Naive Navigation Algorithm

This NaiveNavigation algorithm is simple but has a limitation. With rear-steered Ackerman vehicles, it is possible that a subsequent next waypoint, for example waypoint p2 in Figure 4, is located within the vehicle's ICR, making it practically unreachable. Using the NaiveNavigation algorithm, when such unreachable waypoint is encountered, the vehicle gets stuck in an infinite cyclical movement around this waypoint in an attempt to reach it.

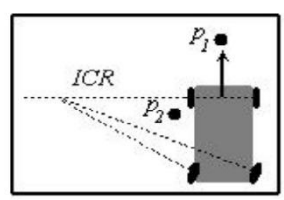


Figure 4. Case of a waypoint (P_2) within the vehicle's ICR.

One approach to remedy this infinite cyclical movement anomaly is to calculate waypoint reachability with respect to the vehicle's ICR. Once the ICR distance is found, it is used to determine the ICR location on one side of the vehicle. This location is then used as the point of origin to an alternate reference frame. The goal point is then translated into this alternate reference frame

and the Euclid distance between the ICR and goal is computed. If this value exceeds the length of the ICR, the point is deemed reachable and the algorithm continues as described above. If the distance between the ICR and the goal point is less than the length of the ICR, the point is deemed unreachable. The same calculations are made with respect to the other side of the vehicle.

There are two approaches to deal with an unreachable waypoint. One approach is to ignore the unreachable waypoint by skipping it and attempting to reach the subsequent waypoint assuming that the increased distance to the subsequent waypoint will increase the likelihood of it being reachable. While skipping waypoints may not have a negative effect towards reaching the final destination waypoint, it causes some deviations from the path, most likely outside the acceptable tolerance levels, resulting in inaccurate path following.

The second approach is to perform corrective actions to make the unreachable waypoint reachable. Since the steering granularity of Ackerman-steered vehicles is coarse, it is best that no turning takes place, but rather that the corrective action is a controlled and limited motion in the reverse direction. The reverse direction actions also causes deviations but it is expected they are limited and would have no negative effect on reaching the final destination waypoint.

1.3 The Skip-Path Algorithm

The Skip-Path algorithm (Figure 5) assumes that even if an unreachable waypoint is skipped, the vehicle would still be able to steer towards the overall (final) goal location by acquiring a subsequent waypoint after the one that was skipped. This optimistic self-correction is performed by continuously calculating the vehicle's ICR as it navigates to the next waypoint.

```
1.  SkipPath (in: intermediate_waypoints[])
2.  while intermediate_waypoints remain do
3.     Compute current_location
4.     Read goal_point from intermediate_waypoints
5.     Compute distance: current_location to goal_point
6.     if distance is less then tolerance then
7.        Read next goal_point from intermediate_waypoints
8.     end if
9.     compute location of right_ICR
10.    if goal_point is within ICR radius of right_ICR then
11.       Read next goal_point from intermediate_waypoints
12.    end if
13.    compute location of left_ICR
14.    if goal_point is within ICR radius of left_ICR then
15.       Read next goal_point from intermediate_waypoints
16.    end if
17.    compute direction: current_location to goal_point
18.    Steer robot in direction of goal_point
19. end while
```

Figure 5. The Skip-Path algorithm pseudo code.

When an unreachable waypoint is encountered, it is skipped and the vehicle navigates to the next waypoint on the list. This technique prevents the vehicle from the infinite spiral movement since the vehicle always continues to advance along the path.
The main disadvantage of this approach is that once the vehicle skips a waypoint, it is often impossible to reclaim an accurate position and direction on the path. In most cases, the vehicle gets close to the expected path but continues to skip waypoints until it reaches the beginning of a different path segment.

The constant skipping of waypoints is due to the vehicle's corrective steering while attempting to reach the next waypoint after skipping one and inadvertently placing it within its ICR, hence rendering it also unreachable. Therefore, while the Skip-Path approach solves the infinite cycling problem, it results in highly inaccurate path navigation.

1.4 The Reverse-Path Algorithm

The described algorithmic approaches developed thus far assumed forward-only vehicle motion. By relaxing this assumption in the Skip-Path algorithm, the highly inaccurate path navigation is no longer a problem.

```
1.  Reverse-Path (in: intermediate_waypoints[])
2.  while intermediate_waypoints remain do
3.     Compute current_location
4.     Read goal_point from intermediate_waypoints
5.     Compute distance: current_location to goal_point
6.     if distance is less then tolerance then
7.        next goal_point
8.     end if
9.     compute location of right_ICR
10.    if goal_point is within ICR radius of right_ICR then
11.       Steer robot reverse to left
12.       continue
13.    end if
14.    compute location of left_ICR
15.    if goal_point is within ICR radius of left_ICR then
16.       Steer robot reverse to right
17.       continue
18.    end if
19.    compute direction: current_location to goal_point
20.    Steer robot in direction of goal_point
21. end while
```

Figure 6. The Reverse-Path algorithm pseudo code.

Figure 6 illustrates the pseudocode for the Reverse-Path algorithm that incorporates reverse motion to the Skip-Path algorithm. The vehicle's reverse motion is used to ensure goal point reach-ability along with accuracy within the tolerance measures. In this approach, once an unreachable waypoint is detected using the ICR calculations, rather than skipping the unreachable point and moving to the next one, the vehicle reverses to a position in which the unreachable waypoint is located outside the vehicle's ICR and therefore becoming reachable. The vehicle then proceeds to the waypoint and then continues to progress through the list. This approach maintains the benefits of the prior solution with respect to the infinite cycling problem and ensures accurate path navigation without skipping any waypoints.

4.Simulation & Experiments

The developed navigation algorithms were exercised, tested and verified using a simulation infrastructure comprised of USARSim [2, 4, 15], a Gamebots [1, 6] variant of the Unreal

Engine [14] and the Mobility Open Architecture Simulation and Tools (MOAST) [10] framework [12].

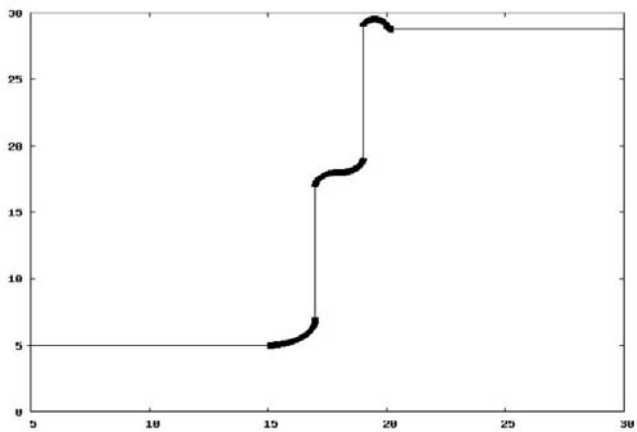


Figure 7. An example of a simulated navigation path of lines and arcs.

Using an arbitrary navigation path of multiple segments (Figure 7) the algorithms were executed using a simulated, Ackerman-steered, automated guided vehicle (AGV) to navigate the path. The navigation path or the "world" along with an arc file, are provided as input, and the simulated vehicle generates the waypoints to be followed to complete the path from the initial to a goal waypoint.

1.5 Simulation Executions

The simulation environment provides a number of configurable parameters (Table 1) for the vehicle that can be adjusted to modify the navigation performance. For the experiments, a Unit Loader AGV was used and all parameters but velocity were kept at their default settings.

Table 1. Unit Loader Default Navigation Parameters.

Vehicle (Unit Loader) Parameters	
MAX_TRAN_VEL	5
MAX_TRAN_ACC	50
MAX_ROT_VEL	600
MAX_ROT_ACC	6000
V_CUTOFF_ANGLE	210
W_CUTOFF_ANGLE	10
CONTROL_POINT	0.3 0 0 0 0 0

The path segments from Figure 7 become the navigation path for the vehicle to traverse. Figure 8 illustrates the generated path and the executed path of the REVERSE-PATH algorithm.

From Figure 8 it is obvious that the vehicle deviates on certain locations from the prescribed path in order to reverse and correct its course.

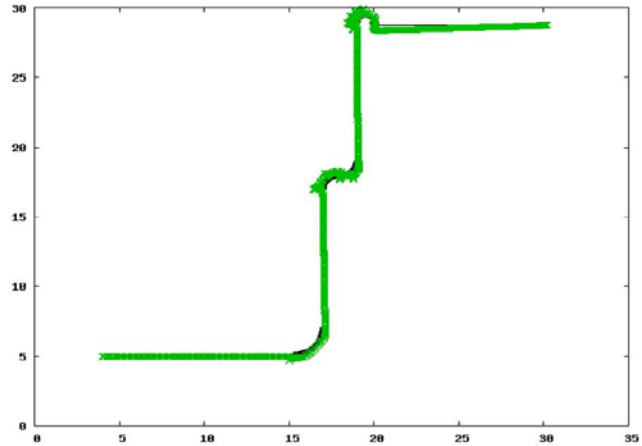


Figure 8. Graph of the simulated execution of the Reverse-Path algorithm. The more pronounced areas of the graph indicate reverse actions the vehicle performed to correct its orientation and reach one or more waypoints.

1.6 Performance Analysis

With the REVERSE-PATH algorithm the focus is on path navigation accuracy and speed of path completion. To examine the range of the results, six simulation executions were performed, with different vehicle velocity parameters (Table 2). By examining the results and plotting the velocity with respect to the time it took to complete the path (Figure 9), it is shown that as velocity increases, the time required to complete the path decreased.

Table 2: The REVERSE-PATH algorithm execution results with variable velocities.

Execution ID	Velocity (m/sec)	Time (sec)	Number of Reverse Actions
001	1.0	443	1
002	2.0	235	1
003	2.5	215	7
004	3.0	220	11
005	4.0	182	16
006	5.0	173	20

Intuitively, this is an expected result, however, it does not reveal any information about the accuracy of the followed path.

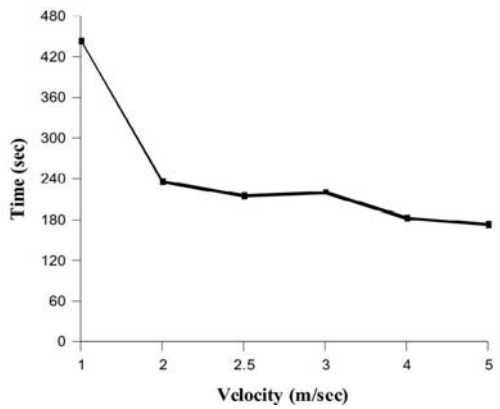


Figure 9: Plot of the time over variable velocity of multiple executions of the same navigation path using the ENHANCED SKIP-PATH algorithm.

In fact, it hides the effect that the higher velocities have on accuracy in following the path. Figure 10 reveals that while the time for path completion is reduced with higher speeds, the amount of corrective actions, in this case, the number of times the vehicle must reverse to continue reaching all of the path waypoints also increases dramatically.

For velocities that do not exceed 2.0 m/sec the number of reverse corrective actions is the same and negligible. However, even a 0.5 m/sec velocity increase causes the number of corrective actions to increase by a factor of seven. Doubling the velocity from 2.0 m/sec to 4.0 m/sec results in a sixteen-fold increase of the number of corrective actions. The steep increases in the number of corrective actions indicate the vehicle must maneuver, and therefore deviate, from the original path to compensate for poor orientation from waypoints.

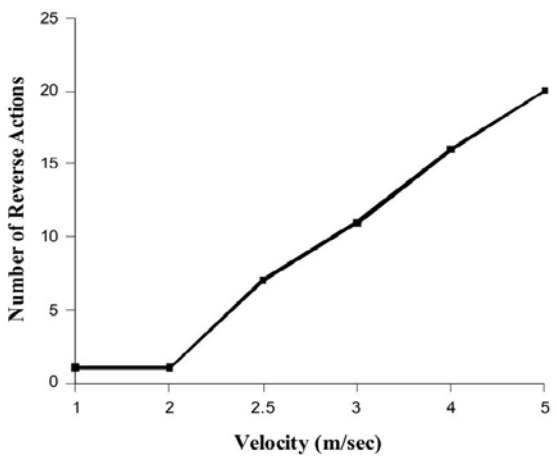


Figure 10: Plot of the number of reverse actions over variable velocity of multiple executions of the same navigation path using the ENHANCED SKIP-PATH algorithm.

5. Conclusion and Future Work

Waypoint navigation is an intuitive and well-studied technique. However, specific vehicle characteristics that may affect steer-ability may render the application of the technique more challenging and far less accurate than expected. Using an Ackerman steered vehicle provided such a challenge. Significant anomalies, such as infinite cycling around waypoints or very coarse accuracy in following a prescribed path, became quickly obvious with naive algorithm implementations. Through an iterative approach, a naive, coarse and often unsuccessful waypoint navigation algorithm evolved to an accurate and robust technique. By computing the vehicle's ICR in real time, it became trivial to determine if the vehicle would be able to reach an upcoming waypoint and stay within the expected navigation tolerance. When this was not possible, a self-corrective reverse-motion action would align the vehicle enough to exclude the waypoint from the vehicle's ICR, thus making it reachable.

This approach provided a solution to the infinite cycling problem and enabled a much finer approach in limiting deviations from the expected path tolerance. The results and the proposed technique were verified via simulation. The application of the REVERSE-PATH algorithm in the simulation environment performed as expected and provided a number of performance metrics. During the design and implementation of the algorithm, the vehicle's velocity was assumed to be constant and consistent throughout each path. In the simulation, different vehicle velocities were used that revealed how dependent the proposed technique is on the vehicle's velocity. The performance analysis results indicate that the accuracy in path navigation of the REVERSE-PATH algorithm is also dependent on the vehicle's speed. Specifically, as the vehicle's velocity increases, the number of corrective reverse-maneuvers also increase, causing path deviations just by attempting to re-establish the correct vehicle orientation.

Further work on the impact of the variation of velocity is warranted to examine the effect of variable speeds on different path segments. Preliminary results indicate that slower velocities on curves and higher velocities on straight lines is a promising approach but require more elaborate computations in determining the curvature and length for each segment before traversing it.

6. ACKNOWLEDGMENT

The authors are grateful to S. Balakirsky (NIST), R. Madhavan (NIST/ORNL) and C. Scrapper (NIST) for their guidance with the Virtual Manufacturing & Automation Competition (VMAC). The authors also appreciate the continuous and generous support provided by Dean Bands and the Hood College Graduate School.

7. REFERENCES

[1] Adobbati, R., Marshall, A.N., Scholer, A., Tejada, S., Kaminka, G.A., Schaffer, S., Sollitto, C. “GameBots: A 3D virtual world test bed for multiagent research,” in Proceedings of the Second International Workshop on Infrastructure for Agents, MAS, and Scalable MAS, Montreal, Canada, 2001.

[2] S. Balakirsky, C. Scrapper, S. Carpin, M. Lewis (2006). "USARSim: providing a framework for multi-robot performance evaluation". Proceedings of PerMIS 2006.

[3] J. Barraquand, B. Langlois, J.-C. Latombe, “Robot Motion Planning With Many Degrees of Freedom and Dynamic Constraints,” in Fifth International Symposium of Robotics Research, pp. 435-444, 1989.

[4] S. Carpin, M. Lewis, J. Wang, S. Balarkirsky, C. Scrapper. ”USARSim: a robot simulator for research and education”.

Proceedings of the IEEE 2007 International Conference on Robotics and Automation, 1400-1405.

[5] V. Hayward, "Fast Collision Detection Scheme by Recursive Decomposition of a Manipulator Workspace," in IEEE International Conference on Robotics and Automation, pp. 1044-1049.

[6] Kaminka, G. A., Veloso, M. M., Schaffer, S., Sollitto, C., Adobbati, R., Marshall, A. N., Scholer, A., and Tejada, S. 2002. GameBots: a flexible test bed for multiagent team research. Commun. ACM 45, 1 (Jan. 2002)

[7] K. Kondo, "Motion Planning with Six Degrees of Freedom by Multistrategic Biderectional Free-Space Enumeration," IEEE Transactions on Robotics and Automation, vol. 7, no. 3, pp. 267-277, 1991.

[8] J.-C. Latombe, "Robot Motion Planning". Kluwer Academic Publishers, 1991.

[9] T. Lozano-Perex, "A Simple Motion Planning Algorithm for General Robot Manipulators, IEEE Journal of Robotics and Automation," vol. RA-3, pp. 224-238, June 1987.

[10] "MOAST project, " http://moast.sourceforge.net, accessed June 1, 2008

[11] Murphy, R.R.: "Introduction to AI Robotics". MIT Press, Cambridge, MA, USA (2000)

[12] C. Scrapper, S. Balakirsky, E. Messina, "MOAST and USARSim – A Combined Framework for the Development and Testing of Autonomous Systems," in Proceedings of the 2006 SPIE Defense and Security Symposium, 2006.

[13] M.A. Sotelo, "Lateral control strategy for autonomous steering of Ackerman-like vehicles,"Robotics and Autonomous Systems, Volume 45, Number 3, 31 December 2003, pp. 223-233(11).

[14] UNR, Unreal engine, http://www.epicgames.com, accessed June 1, 2008.

[15] USARSim, http://sourceforge.net/projects/usarsim, accessed June 1, 2008.

[16] Wang, Y., Mulvaney, D., Sillitoe, I., and Swere, E. "Robot Navigation by Waypoints". J. Intell. Robotics Syst. 52, 2 (Jun. 2008), 175-207.

[17] T.S. Wikman. W.S. Newmanm A Fast On-line Collision Avoidance Method for Kinematically Redundant Manipulator Based on Reflex Control, in IEEE International Conference on Robotics and Automation pp. 261-266, 1992.

Wireless Communications in Tunnels for Urban Search and Rescue Robots

Kate A. Remley, Galen Koepke, Dennis G. Camell, Chriss Grosvenor
NIST Electromagnetics Division
325 Broadway
Boulder, CO, USA 80305
+1 303 497-3652
remley@boulder.nist.gov

Lt. George Hough
New York City Fire Department
New York City, NY USA
gch9@columbia.edu

Robert T. Johnk
Institute for Telecommunications Sciences
325 Broadway
Boulder, CO, USA 80305
bjohnk@its.bldrdoc.gov

ABSTRACT

We report on propagation tests carried out in a subterranean tunnel to support improved wireless communications for urban search and rescue robots. We describe single-frequency and ultrawideband channel-characterization tests that we conducted, as well as tests of telemetry and control of a robot. We utilize propagation models of both single-frequency path loss and channel capacity to predict robot performance. These models can also be used for optimizing wireless communications in tunnels of various sizes, materials, and surface roughness.

Categories and Subject Descriptors

Defining and measuring aspects of an intelligent system. Evaluating components within intelligent systems

General Terms

Measurement, Performance, Reliability, Experimentation, Standardization, Verification.

Keywords

Multipath; radiowave propagation; RMS delay spread; robot; urban search and rescue; wireless communication; wireless system

INTRODUCTION

Researchers from the Electromagnetics Division of the National Institute of Standards and Technology (NIST) and the Fire Department of New York (FDNY) conducted tests of radiowave transmission and detection in tunnels at the Black Diamond Mines Regional Park near Antioch, California on March 19-21, 2007. Our goal was to investigate propagation channel characteristics that affect the reliability of wireless telemetry and control of Urban Search and Rescue (US&R) robots in tunnels and other weak-signal environments. We describe measurement methods that we used to study parameters relevant to robot performance. We also use the measured data to verify models of radiowave propagation in tunnels. These models can be used to predict robot performance in tunnels having characteristics different from the ones we measured, such as subways and utility tunnels. This work supports the development of technically sound standards for US&R robots [1-3].

We used both time- and frequency-domain techniques to study issues such as channel multipath and loss that may impede successful wireless communications in tunnels. We tested both video and control of a robot inside a mine tunnel. We also implemented propagation models of path loss and channel capacity and compared our measured results to these models. We summarize below the data we collected and interpret the key findings from the study, which is described in its entirety in [4].

Recently, the wireless field has seen a renewed interest in studies of signal propagation in both mine and subway tunnels, following a good deal of study on mine communications in the 1970s. A seminal work on mine tunnel propagation by Emslie et al. [5], studied radiowave propagation in small underground coal tunnels (4.3 m wide x 2.1 m high) for frequencies ranging from 200 MHz to 4 GHz. Emslie developed a model for propagation in tunnels that is used today. Recently, Rak and Pechak [6] applied Emslie's work to small cave galleries for speleological applications, confirming Emslie's findings that once a few wavelengths separate the transmitter and receiver, the tunnel acts as a waveguide that strongly attenuates signals below the waveguide's cutoff frequency. Because the walls of the tunnel are not perfectly conducting, signals operating above the cut-off frequency also experience significant loss. In a recent paper, Dudley, Lienard, Mahmoud, and Degauque [7] performed a detailed assessment of operating frequency in a variety of tunnels. They found that as frequency increases, the lossy waveguide effect decreases.

Our measurements, covering a much wider frequency range than [7], and implementation of the model of [6] also confirm the lossy waveguide effect in the tunnels we studied. This effect can have a significant impact on choice of frequency for critical applications such as US&R operations, where typically infrastructure such as a repeater network is not available and lives may be at stake.

Another factor in tunnel communications is multipath caused by reflections off the walls, floor, and ceiling of the tunnel. This was clearly seen in the work of Dudley, et al. [7]. Multipath can have a pronounced effect on successful transmission of wideband data. Multipath interference may affect certain frequencies in a wideband signal while simultaneously having little impact on other frequencies. This frequency selectivity can make decoding signals difficult for the demodulator in a receiver.

We studied the severity of multipath in the tunnel environment by measuring the RMS delay spread, as well as the success or failure of wideband data transfer by use of a commercially available robot. We compare our measured results to a model of channel capacity based on Shannon's theory of channel capacity

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08,August19-21,2008,Gaithersburg,MD,USA.
ACM ISBN 978-1-60558-293-1/08/08.

[6]. This theorem provides a basis for predicting the success of wireless communication in multipath environments.

We first describe the measurements we carried out and interpret the results. We then discuss the models we implemented and how they can be used to optimize radio communications for US&R applications.

1. TEST ENVIRONMENT

The Black Diamond Mines are part of an old silica mine complex that was used early in the 1900s to extract pure silica sand for glass production. As such, the walls of the mine shafts are rough and consist of sandy material.

Two tunnels were studied, the Hazel-Atlas North (here called the "Hazel-Atlas" tunnel) and Hazel-Atlas South (here called the "Greathouse" tunnel). The tunnels are located beneath a mountain and are joined together deep inside, as shown in Figure 1(a). The dimensions of the Hazel-Atlas tunnel varied from approximately 1.9 m (6', 3") x 1.9 m to as much as 2.6 m (8', 5") x 2.4 m (8', 0"). The dimensions of the Greathouse tunnel were somewhat bigger, up to approximately 3 m square in places. The Hazel-Atlas tunnel contained railroad tracks spaced 61 cm (24") apart. Both tunnels consisted of a straight section followed by a 90 degree turn around a corner, as shown in Figure 1(a). Below we report on results for the Hazel-Atlas mine tunnel, shown in more detail in Fig. 1(b). The complete set of data on both tunnels can be found in [4].

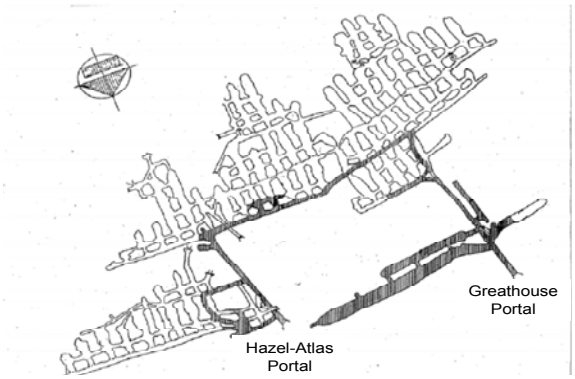
Figure 2 shows photographs of the tunnel, including: 2(a) the portal (entrance) of the Hazel-Atlas mine; 2(b) approaching the right-angle turn shown in Figure 1(b); and 2(c) past the turn. The photos show the rough, uneven walls in the tunnels, some with wooden shoring, and the railroad tracks.

2. MEASUREMENTS

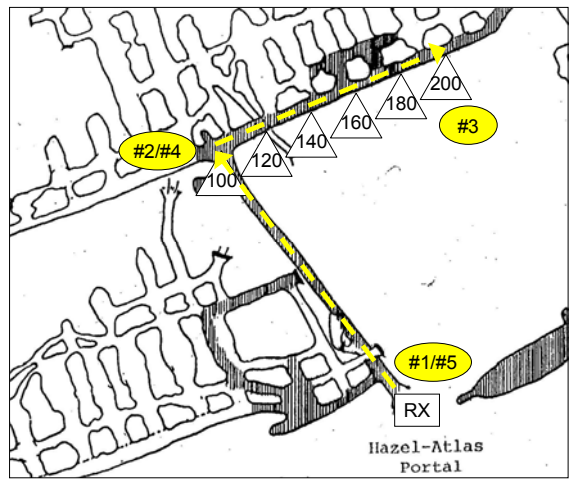
2.1 Narrowband Received Power

We measured the power received outside the tunnel from a transmitter placed at various locations inside the tunnel. We collected single-frequency (unmodulated, carrier only) received-power data at frequencies near public-safety bands (approximately 50 MHz, 150 MHz, 225 MHz, 450 MHz). Gathering information at these frequencies helps to provide a choice of optimal frequency for the US&R community for this environment, both for robot communications and for other types of radio communication. These data provide insight into the lossy waveguide effect mentioned in the Introduction.

The handheld transmitters we used were radios similar to those of first responders, except they were placed in ruggedized cases and were modified to transmit continuously. Each radio transmitted a signal of approximately 1 W through an omnidirectional "rubber duck" antenna mounted on the case. During the tests, the radio antennas were approximately 0.75 m from the floor, a height similar to that of the robot we studied.



(a)



(b)

Figure 1: (a) Overview of the Hazel-Atlas mine tunnel complex. The network of mines is located deep within a mountain. The dark-shaded areas are accessible. (b) Close-up view of the Hazel-Atlas tunnel. The dashed line shows the path along which we took measurements, including the right-angle turn at 100 m. The triangles indicate the distance in meters, the ovals correspond to locations shown in Fig. 3, and the receiving equipment is labeled RX.

We carried the radio transmitters from the entrance to locations deep within the tunnels while continuously recording the received signal. From the Hazel-Atlas tunnel portal, we moved the transmitter approximately 100 m down a straight tunnel, then turned a corner and proceeded another 100 m, as shown in Fig. 1(b).

The receiving equipment was located just outside the portal. Omnidirectional discone receiving antennas were mounted on tripods, as shown in Fig. 2(a). We used a narrowband communications receiver to convert the received signal to audio frequencies, where it was digitized by a computer sound card and recorded on a computer. This instrument, when combined with NIST-developed post-processing techniques [1, 8], provides a high-dynamic-range measurement system that is affordable for most public-safety organizations. Part of the intent of this project

was to demonstrate a user-friendly system that could be utilized by US&R organizations to assess their own unique propagation environments.

Figure 3 shows representative measured received-power data at frequencies of 50 MHz, 162 MHz, and 448 MHz acquired while the transmitters were carried by foot through the tunnel. The signals were sampled at approximately 48 kHz and the power averaged over one-second intervals. The left and right halves of the graph show measurements made walking into and out of the tunnel, respectively, and thus mirror each other. The vertical dashed lines on the graph correspond to the entrance (#1, #5), turn (#2, #4), and turn-around point (#3) in the measurement path, as shown in Fig. 1(b).

The cut-off frequency for this type of tunnel is difficult to define since the walls behave as lossy dielectrics rather than conductors. These conditions are discussed in [9], where the attenuation constant is found to vary as the inverse of frequency squared [Section 2.7, pp. 80-83]. Hence, we would expect higher attenuation at the lower frequencies but no sharp cut-off. Further complications in this tunnel are the axial conductors (cables, water pipes, rails) that may support a TEM-like mode of propagation, the irregular cross-section, and the side chambers and tunnels.

For the Hazel-Atlas mine tunnel, we see in Fig. 3(a), strong attenuation of the 50 MHz signal and in Fig. 3(b), the received power of the 162 MHz signal also decreases rapidly as the transmitter moves into the tunnel. This rapid attenuation is due to the lossy waveguide effect described in references [4-7]. The signal for the 448 MHz carrier frequency (Fig. 3(c)) exhibits less attenuation and this is where the models of [5] may apply. Signals may travel even further at higher frequencies, as discussed in [5-7]. This frequency dependence may play a significant role in deciding which frequencies to utilize in US&R robot deployment applications, as will be discussed in Section 3.

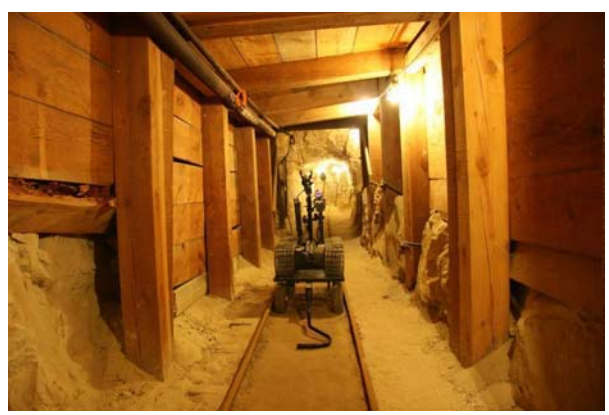
2.2 Excess Path Loss and RMS Delay Spread

We also conducted measurements at several stationary positions covering a very wide frequency band. These "excess-path-loss" measurements provide the received signal power relative to a direct-path signal over a frequency band. When transformed to the time domain, the wide frequency band yields a short-time-duration pulse. This pulse can be used to study the number and duration of multipath reflections in an environment.

Our synthetic-pulse, ultrawideband system is based on a vector network analyzer (VNA). Our measurements covered frequencies from 25 MHz to 18 GHz. The post-processing and calibration routines associated with it were developed at NIST [10]. In the synthetic-pulse system, the VNA acts as both transmitter and receiver. The transmitting section of the VNA sweeps over a wide range of frequencies a single frequency at a time. The transmitted signal is amplified and fed to a transmitting antenna. For this study, we used omnidirectional discone antennas for frequencies between 25 MHz and 1.6 GHz, and directional horn-type transmitting and receiving antennas for frequencies between 1 GHz and 18 GHz.

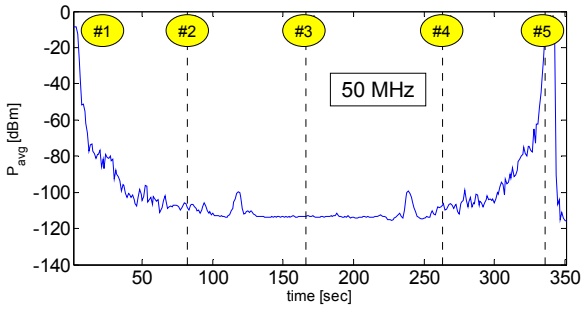
(a)

(b)

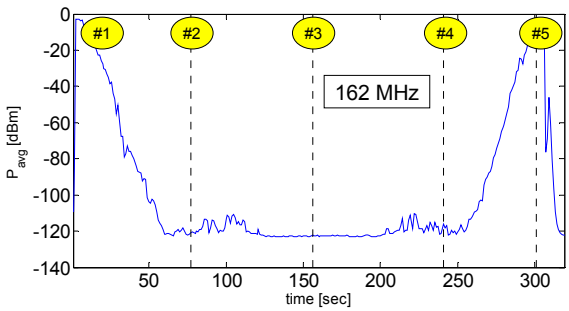


(c)

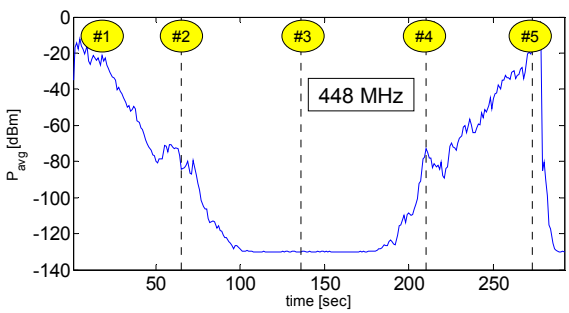
Figure 2: (a) Portal into the Hazel-Atlas mine tunnel. (b) 90 m inside showing the bend depicted in Fig. 1(b) and the rough, sandy wall material. (c) Wood shoring approximately 150 m into the tunnel. The robot we tested can be seen on the cart between the railroad tracks.



(a)



(b)



(c)

Figure 3: Received-power data in the Hazel-Atlas Mine for three carrier frequencies: (a) 50 MHz, (b) 162 MHz, (c) 448 MHz. In each case the #2 and #4 vertical dashed lines correspond to the turn at 100 m: once on the way into the tunnel and once on the way out. The #3 dashed line represents the end point at 200 m, shown in Fig. 1(b).

The received signal was picked up over the air in the tunnel by the receiving antenna and was relayed back to the VNA via a fiber-optic cable. The fiber-optic cable phase-locks the received signal to the transmitted signal, enabling post processing reconstruction of the time-domain waveform associated with the received signal. The broad range of frequencies and time-domain representation provide insight into the reflective multipath nature of the tunnel that cannot be captured by use of single-frequency measurements. The receive antenna must remain fixed during each measurement, so these tests are carried out at discrete locations, unlike the single-frequency tests.

We measured excess path loss every 20 m starting approximately 10 m into the tunnel. The VNA was located at the Hazel-Atlas portal. The transmitting antenna was located at the portal as well. The graphs show data starting from 0 Hz, however the valid (calibrated) measurement range is stated for each graph.

Figures 4 and 5 show measured excess path loss over a wide frequency band measured 50 m and 120 m, respectively, in the Hazel-Atlas tunnel. Note that at 120 m, we have passed the right-angle turn in the tunnel. The top curve in each graph represents the received power level, referenced to the calculated free-space path loss at that location. The bottom curve represents the noise floor of the measurement system.

Figure 4 shows that even in a line-of-sight condition approximately 50 m from the tunnel entrance, the spectrum of the received signal displays significant frequency dependence. At frequencies between 25 MHz and 1.6 GHz (Fig. 4(a)), the lossy waveguide effect is shown by the rapidly decreasing signal on the left-hand side of the graph. We see that a carrier frequency higher than approximately 700 MHz would suffer less loss compared to lower frequencies in this particular tunnel. Figure 4(b) shows frequencies from 1 GHz to 18 GHz. In this case, we see frequency dependence in received power caused by strong reflections, as shown by the deep nulls and peaks in the top curve of Fig. 4(b).

Once the receiving antenna turns the corner, see Fig. 5, the signal takes on a more random variation with frequency since transmission consists of reflected signals only. For frequencies from 25 MHz to 1.6 GHz (Fig. 5(a)), the received signal power is near the noise floor of the receiver since the two curves almost overlay. For the higher frequencies (Fig. 5(b)), we see that the average received signal level is relatively constant with frequency, but the peaks and nulls are still significant.

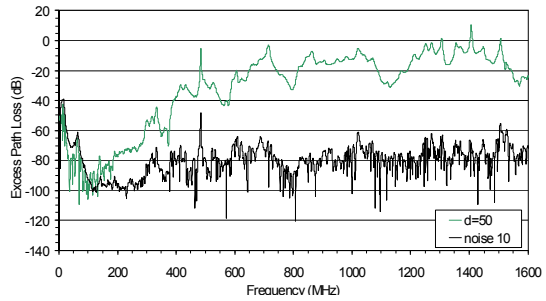
Finally, we present the RMS delay spread for the Hazel-Atlas mine tunnel in Table 1 for frequencies from 25 MHz to 1.6 GHz and 1 GHz to 18 GHz. We see that the shortest delay spreads are found by use of the directional antennas. The complete set of UWB excess-path-loss data is given in [4].

2.3 Tests of Robot Communications

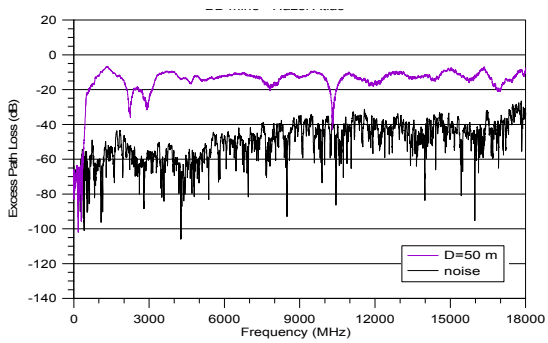
We carried out tests on a commercially available robot. Control and video were as-built for the commercial product. We used the omnidirectional antennas that came with the system for all tests in order to assess the default capabilities of this robot.

The robot we used is controlled with a 2.4 GHz spread-spectrum, frequency-hopping protocol, which was configured to transmit in the unlicensed 2.4 GHz industrial, scientific, and medical (ISM) band. The control channel utilizes a modulation bandwidth of approximately 20 MHz. The output power of the bidirectional control link is nominally 500 mW.

The robot transmits video by use of one of ten channels between 1.7 GHz and 1.835 GHz. The robot we tested transmitted at 1.78 GHz by use of an analog modulation format that was non-bursted and non-frequency-agile. The video channel utilized approximately 6 MHz of modulation bandwidth. The output power was nominally 2 watts.

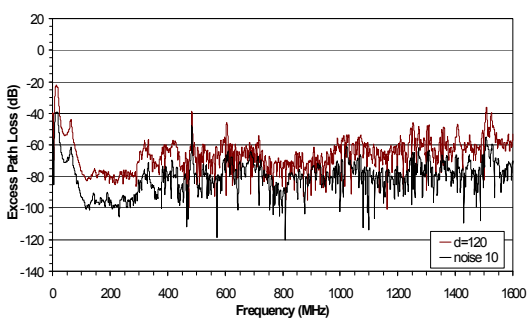


(a)

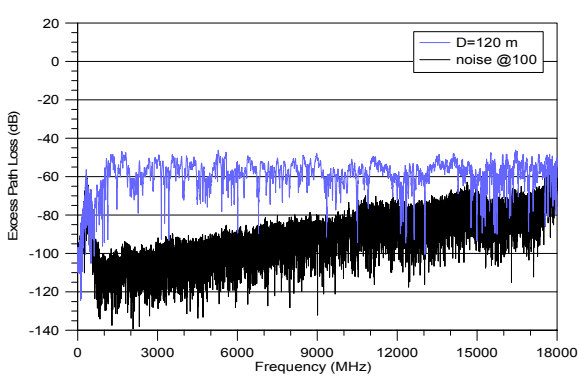


(b)

Figure 4: Excess path loss measurements over a wide frequency band carried out 50 m from the portal of the Hazel-Atlas mine. (a) 25 MHz to 1.5 GHz. (b) 1 GHz to 18 GHz.



(a)



(b)

Figure 5: Excess path loss measurements carried out 120 m from the portal of the Hazel-Atlas mine. (a) 25 MHz to 1.6 GHz. (b) 1 GHz to 18 GHz.

Table 1: RMS Delay Spread for the Hazel-Atlas mine tunnel. Center column: Frequencies from 25 MHz to 1.6 GHz measured with omnidirectional antennas. Right column: Frequencies from 1 GHz to 18 GHz measured with directional antennas. The gray-shaded areas represent a non-line-of-sight propagation condition.

Distance (m)	RMS Delay Spread Low Freqs. (ns)	RMS Delay Spread High Freqs. (ns)
0 31.0		14.4
10 25.3		17.6
20 18.5		7.6
30 15.9		15.0
40 17.0		11.5
50 15.5		13.1
60 19.7		20.6
70 17.2		11.1
80 15.2		10.0
90	15.2	8.4
100	15.7	9.6
110	x	7.5

The robot controller was located at the entrance to the tunnel, shown in Fig. 6. We positioned the robot inside the tunnel after the first bend in a non-line-of-site condition. The robot was moved through the tunnel on a cart, as shown in Fig. 2(c), so that we could check the control link even after video was lost. Every 10 m, the video quality and control link were checked. Video was rated qualitatively by the robot operator, and control was checked by the ability of the operator to move the robot arm, and verified by a researcher in the tunnel. No attempt was made to provide more granularity in these tests. That is, we assumed that moving the arm up was equivalent to moving it down or rotating it.

Table 2 shows the results of our tests. We were able to communicate with the robot in a non-line-of-sight condition deep within the tunnel. This is consistent with the results of Fig. 5(b), which indicates that signals in the low gigahertz range should propagate farther than those at lower frequencies.

Table 2 also shows that control of the robot was possible much deeper into the tunnel than we were able to receive video, even though the output power of the video channel is higher (2 watts for video vs. 0.5 watt for control). However, a much higher data rate is necessary to maintain high-quality video transmission, as opposed to the relatively small amount of data needed to control the robot. Transmitting this large amount of data requires a more stringent success rate than for the control channel; therefore, failure of the video before the control is not unexpected. The delay experienced in controlling the robot when it was deep in the mine indicates packet loss and resend for error correction under weak-signal conditions.

Figure 6: Robot operator positioned at the entrance to the Hazel-Atlas mine tunnel. The robot was operated in a non-line-of-sight condition more than 100 m inside the tunnel.

Table 2: Results of wireless communication link tests carried out inside Hazel-Atlas tunnel at Black Diamond Mines Regional Park.

Distance in tunnel (m)	Video quality (1.7 GHz)	Control of arm (2.4 GHz)
100 good		yes
110 good		yes
120 poor	(intermittent)	yes
130 poor	(intermittent)	yes
140 very	poor	yes
150 none		yes
160 none		delay experienced
170 none		intermittent control
180 none		delay experienced
190 none		delay experienced
200 none		delay experienced
205 none		none

3. MODELED RESULTS

3.1 Single-Frequency Path Gain Models

To study the extent of waveguiding in these tunnels, we implemented an analytical model that simulates signal propagation in tunnel environments having various physical parameters [5, 6, 11]. Briefly, the model assumes a dominant EH_{11} mode in a lossy rectangular waveguide with the attenuation α in dB/m expressed for vertical polarization as

$$\alpha = \alpha_{\text{TUNNEL}} + \alpha_{\text{ROUGHNESS}} + \alpha_{\text{TILT}}, \quad (1)$$

where

$$\alpha_{\text{TUNNEL}} = 4.343\lambda^2\left(\frac{1}{a^3\sqrt{\varepsilon_R - 1}} + \frac{\varepsilon_R}{b^3\sqrt{\varepsilon_R - 1}}\right), \quad (2a)$$

$$\alpha_{\text{ROUGHNESS}} = 4.343\pi^2 h^2 \lambda\left(\frac{1}{a^4} + \frac{1}{b^4}\right), \quad (2b)$$

$$\alpha_{\text{TILT}} = 4.343\frac{\pi^2\theta^2}{\lambda}, \quad (2c)$$

and λ is the wavelength, a is the width of the tunnel, b is the height of the tunnel, and h is the roughness, all in meters. Other parameters include ε_R, the dielectric constant of the rock walls, and θ, the angle of the tunnel-floor tilt in degrees.

We set the parameters of the model to approximate the Hazel-Atlas tunnel, given below in Table 3. This model works well only for frequencies will above the cut-off frequency, that is, for wavelengths significantly less than the dimensions of the tunnel [5, 6]. Hence, in Fig. 7 we show results for 448 MHz only. At distances around 80 m, the signal was able to propagate through an air vent as well as through the tunnel, so the overall received signal level increases. The good agreement between the measured and modeled data led us to conclude that waveguiding plays a significant role in radio propagation in these tunnels.

Table 3: Parameters used in tunnel model.

Parameter	Value
Width	2 m
Height	2 m
Wall roughness	0.3
ε_r	6
tilt	1°

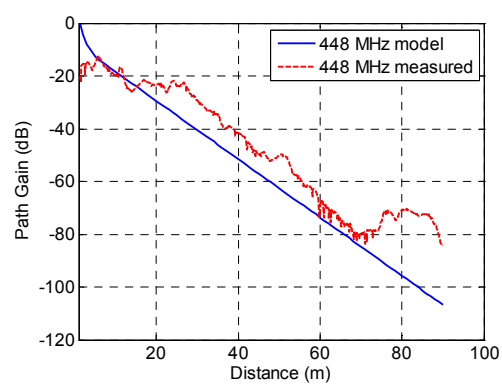


Figure 7: Comparison of measured and modeled data for the Hazel-Atlas tunnel. The carrier frequency is 448 MHz. The modeled data simulate waveguide propagation for a waveguide whose physical parameters approximate those of the tunnels.

The model also lets us explore which frequencies may be optimal for robot or other wireless communications in the tunnel. Figure 8 compares a number of commonly used emergency

responder frequencies as a function of distance within the tunnel. As discussed in [5, 6], the frequency-dependent behavior of the tunnel leads to a "sweet spot" in frequency. Below the sweet spot, signals do not propagate well, due to the effect of waveguide-below-cutoff attenuation and wall loss. Above the sweet spot, free-space path loss (which increases with frequency) and α_{TILT} dominate and signals do not propagate well. Again, models such as these may enable a choice of appropriate frequency for US&R robot communications in tunnel environments. Note that these results are valid only for a tunnel with these dimensions, wall materials, and surface roughness. The curves would need to be recalculated for other types of tunnels.

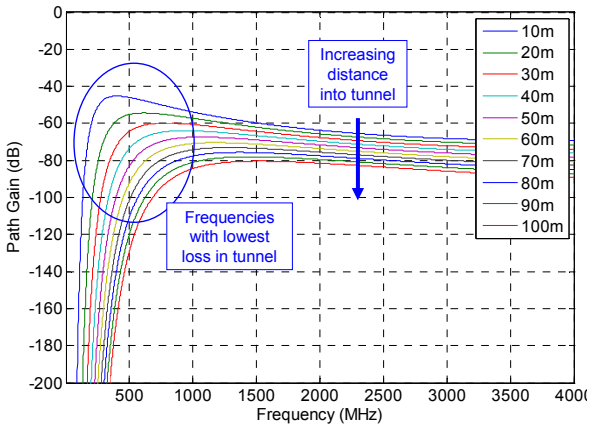


Figure 8: Path gain versus frequency for various distances in a tunnel having physical characteristics similar to those of the Hazel-Atlas tunnel. Frequencies around approximately 400 MHz to 1 GHz propagate better than either lower or higher carrier frequencies.

We also used the model to investigate the video performance of the robot, described in Section 3.c. The frequency-hopping control channel would need to be modeled by use of other methods, since it consists of several narrowband channels frequency hopping within a wide modulation bandwidth. In Fig. 9, we plot the estimated path gain at a carrier frequency of 1.78 GHz for the tunnel environment with a right-angle turn 100 m from the receiver. We used the parameters in Table 3 for the model. A path gain of −40 dBW was used as an approximation for the turn in the tunnel at 100 m, based on work done by Lee and Bertoni in [12]. We plot the flat earth path gain for comparison.

Figure 9 also shows the theoretically computed excess link margin (ELM). The ELM is the difference between the received signal strength and the minimum receiver sensitivity. The receiver sensitivity is determined by the thermal noise of the receiver and the receiver's front-end amplifier noise (5 dB, as a rule of thumb). The thermal noise is given by $N = kTB$, where k is the Boltzmann constant, T is the temperature in Kelvin, and B is the bandwidth of the receiver. In order for a wireless link to be maintained, the ELM must be greater than zero dB.

The ELM plotted in Fig. 9 agrees well with the measured results from Table 2, which show that the video completely drops out between approximately 140 m and 150 m. Given the fluctuation in signal strength due to multipath fading in this tunnel environment, once the link margin drops below 10 dB at approximately 120 m, the video quality degrades and the picture becomes intermittent.

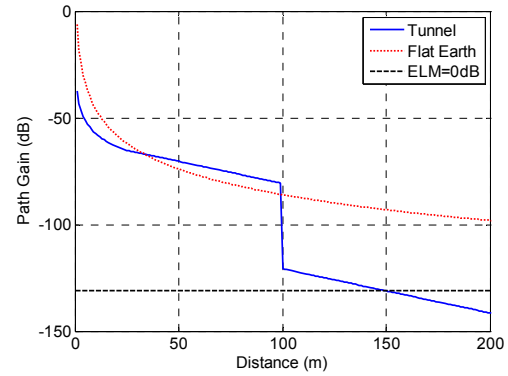


Figure 9: Path gain curves for tunnel with a right-angle turn at 100 m (solid) and flat earth (dashed) environments. The curve labeled "ELM = 0" indicates where the excess link margin calculation predicts loss of signal. As shown, this occurs approximately 150 m into the tunnel.

3.2 Channel Capacity Model

In general, received RF power and bandwidth effectively place an upper bound on the capacity of a communications link. The Shannon channel capacity theorem [13] can be used to predict the approximate maximum data rate for tunnel communications, even though the Shannon theorem is based on the assumption of a Gaussian noise (low multipath) environment. For a given modulation bandwidth, the received signal power relative to the noise power determines the theoretical upper limit on the data rate (channel capacity). The Shannon capacity theorem is given by

$$C = B\log_2\left(1 + S/N\right), \tag{3}$$

where C is the channel capacity in bits/second, B is the channel bandwidth in hertz, S is the received signal power in watts, and N is the measured noise power in watts. The capacity represented by this equation is the upper limit, and in reality the capacity would be difficult to attain with real hardware.

For an analog transmission, Shannon's limit gives us a way to estimate the channel capacity. The National Television System Committee (NTSC) analog video channel that our robot used has a video bandwidth of 4.2 MHz and a transmission rate of approximately 30 frames per second, where each frame consists of 525 scanning lines, giving a line rate of 15.734 kHz [14]. We can place an upper bound on the amount of data that could be transmitted in each line by considering a typical implementation of NTSC, where each line is digitized into 768 pixels. This gives a digital scanning rate of approximately 12 MHz. The specification of 768 pixels per line is used in studio environments. We expect the potential channel capacity to be lower in the analog transmission case.

Figure 10 shows simulations of the Shannon limit for our robot's 4.2 MHz video bandwidth, 1.78 GHz video channel. Table 4 shows the distance into the tunnel where 12 Mb/sec transmission rate occurs assuming our maximum possible channel

capacity to be various fractions of the Shannon limit. Based on this information, we would expect to encounter video problems somewhere between 120 m and 130 m into the tunnel, which Table 2 shows is indeed where we started to experience signal degradation. Thus, we are able to form a rough estimate of the distance into the tunnel where we expect the video to fail based on a simple implementation of the Shannon theorem.

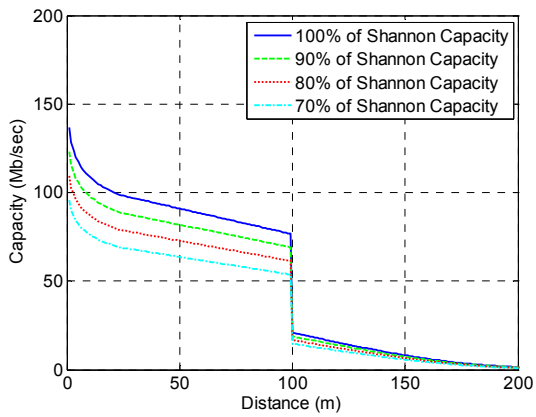


Figure 10: Channel capacity predicted by the Shannon theorem for a carrier frequency of 1.78 GHz and a video modulation bandwidth of 4.2 MHz. At 120 m, where we experienced intermittent video, 80 % of the Shannon limit is 15.4 Mb/sec and 70 % of the Shannon limit is 13.5 Mb/sec.

Table 4: Distance into tunnel for a channel capacity of approximately 12 Mb/sec.

Fraction of Shannon capacity (%)	Distance into tunnel (m)
100 133	
90 129	
80 122	
70 114	

4. CONCLUSION

We have presented measured data collected in a subterranean tunnel environment. Results show ed waveguide-below-cutoff and wall attenuation effects. We saw frequency -dependent peaks and nulls in the channel due to strong m ultipath reflections and attenuation in the tunnel. In non-line-of-sight conditions, we saw classic Rayleigh-distributed noise-like signals.

We implemented models that m ay be used to predict radiowave propagation and modulated-signal performance within tunnels for robots or other wire less devices. Using the models, it was possible to ascertain the optim al carrier frequency range for a robot within this tunnel environment. The intent of this work is to improve radio communications for urban search and rescue robots when they transmit wideband, digitally modulated signals. We hope that these data will prove us eful in standards development, as well as improved technology and sy stem design for the emergency-responder community.

5. ACKNOWLEDGEMENT

The authors thank Alex Bordetsky of the Naval Postgraduate School for facilitating the m easurements during recent interagency m arine interdiction operation sy stem tests. We also thank Bill Dunlop, Steve MacLaren, and Dave Benzel of Lawrence Livermore National Laboratory for logistical and technical support. We are indebted to Frederick M. Remley for details on the NTSC video standard.

6. REFERENCES

[1] Elena Messina, " Performance Standards for Urban Search and Rescue Robots," *ASTM Standardization News,* August 2006, http://www.astm.org/cgi-bin/SoftCart.exe/SNEWS/AUGUST_2006/messina_aug06.html .

[2] National Institute of Standards and Technology, "Statement of Requirements for Urban Search and Rescue Robot Performance Standards-Preliminary Report" http://www.isd.mel.nist.gov/US&R_Robot_Standards/Requirements%20Report%20(prelim).pdf

[3] K. A. Rem ley, G. Koepke, E. Messina, A. Jacoff, G. Hough, "Standards development for wireless com munications for urban search and rescue robots," *Proc. Int. Symp. Advanced Radio Technol.,* pp. 66-70, Boulder, CO, Feb. 2007.

[4] K.A. Remley, G. Koepke, C.L. Holloway, C. Grosvenor, D. Camell, J. Ladbury, D. Novotny, W.F. Young, G. Hough, M.D. McKinley, Y. Becquet, J. Korsnes, "Measurements to support broadband modulated-signal radio transmissions for the public-safety sector," *Natl. Inst. Stand. Technol. Note 1546*, Apr. 2008.

[5] A.G. Emslie, R.L. Lagace, and P.F. Strong, "Theory of the propagation of UHF radio waves in coal mine tunnels," *IEEE Trans. Antennas and Prop.*, vol. 23, no. 2, Mar. 1975, pp. 192-205.

[6] M. Rak and P. Pechac, "UHF propagation in caves and subterranean galleries," *IEEE Trans. Antennas and Prop.,* vol. 55, no. 4, April 2007, pp. 1134-1138.

[7] D. G. Dudley, M. Lienard, S.F. Mahmoud and P. Degauque, "Wireless propagation in tunnels," *IEEE Antennas Prop. Mag.,* vol. 49, no. 2, April 2007, pp. 11-26.

[8] M. Rütschlin, K. A. Remley, R. T. Johnk, D. F. Williams, G. Koepke, C. Holloway, A. MacFarlane, and M. Worrell, "Measurement of weak signals using a communications receiver system," *Proc. Intl. Symp. Advanced Radio Technol.*, Boulder, CO, March 2005, pp. 199-204.

[9] P. Delogne, "Leaky Feeders and Subsurface Radio Communications, IEE press, 1982.

[10] B. Davis, C. Grosvenor, R.T. Johnk, D. Novotny, J. Baker-Jarvis, M. Janezic, "Complex permittivity of planar building materials measured with an ultra-wideband free-field antenna measurement system," *NIST Journal of Research*, Vol. 112, No. 1, Jan.-Feb., 2007, pp. 67-73.

[11] G. Hough, "Wireless robotic communications in urban environments: issues for the fire service," Thesis, Naval Postgraduate School, March 2008. http://theses.nps.navy.mil/08Mar_Hough.pdf

[12] J. Lee and H. Bertoni, "Coupling at cross, T, and L junctions in tunnels and urban street canyons," *IEEE Trans. Antennas Prop.,* vol. 51, no. 5 , May 2003, pp. 926-935.

[13] C.E. Shannon, "Communication in the presence of noise," *Proc. IRE*, vo. 37, no. 1, Jan. 1949, pp. 10-21.

[14] C.A. Poynton, "A Technical Introduction to Digital Video," Wiley, 1996, pp 26-27.

A Performance Assessment of Calibrated Camera Networks for Construction Site Monitoring

Itai Katz, Nicholas Scott, Kamel Saidi
National Institute of Standards and Technology
100 Bureau Drive Stop 8611
Gaithersburg, MD, 20899-8611, USA
{itai.katz, nicholas.scott, kamel.saidi}@nist.gov

ABSTRACT

Monitoring the status of construction activity is a key factor in maximizing efficiency and reducing cost on capital construction sites. In contrast to human inspection, an automated technology would provide more frequent updates. However, real-time construction site monitoring remains an elusive goal because of many factors including the environment's large size and dynamic behavior. Calibrated camera networks have recently been developed to enable smart environments, where the location of people and objects in the scene are updated in real-time. Despite a growing interest in this technology, a detailed performance analysis has not been conducted. In this paper, empirical results on the calibration and static 3D localization error are presented for a calibrated camera network prototype using four cameras. Using a total station to provide ground truth, the cameras are calibrated and points in the workspace are reconstructed in 3D by combining observations from each image plane. Results show that performance is well suited for localizing objects typically found on a construction site (although scaling errors have not yet been characterized).

Keywords

construction automation, calibrated camera network, performance evaluation

1. INTRODUCTION

1.1 The Problem

Situation awareness on a construction site is a key element in maintaining a safe and productive project. In industries such as manufacturing, where the work environment is highly controlled and predictable, situation awareness is relatively easy to achieve. However, the dynamic and cluttered nature of a construction site makes a similar degree of control virtually impossible. Nevertheless, in order for project managers and others to make informed and timely decisions, they must have accurate information of the current status of all project activities. Decisions made based on old or inaccurate information could have wide-reaching ramifications on the schedule, budget and safety of a project, particularly on large capital projects.

Critical information needed for proper decision-making includes:

- Locations of workers and equipment
- Locations of bulk materials and manufactured construction components
- Level of completion for construction activities
- Locations of potential safety hazards

1.2 Camera Networks as a Solution

In this paper, we propose a solution to the situation awareness problem on construction sites using calibrated camera networks. Rather than simply providing raw video feeds, calibrated camera networks provide a method to abstract the data into a form that is more readily usable. While these systems are well known in the academic community [3], their use in the construction industry is novel. This paper extends previous results in [4] on localization error for long-baseline methods using an alternative calibration technique. The data set is 50 % larger and better characterizes the work volume.

1.3 Outline

Below we introduce the calibrated camera network (CCN) concept and describe a potential role in the construction domain. Section 2 provides an overview of CCNs and illustrates how they could potentially be applied to the construction domain. Section 3 demonstrates the setup and testing of a camera network and reports on experimentally-derived errors for 3D localization. Conclusions are provided in Section 4.

2. CALIBRATED CAMERA NETWORKS IN CONSTRUCTION

2.1 System Description

A CCN is a collection of three or more vision sensors whose positions and orientations in 3D space have been determined. In contrast to a collection of isolated cameras, CCNs have two key differences:

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08 August 19-21, 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1/08/08.

1. Each camera in the network is a "smart" node that combines an image sensor, on-board computing, and the capability to network with other nodes (typically wirelessly). This allows each node to independently carry out low-level image processing tasks (e.g., background subtraction) or more sophisticated vision algorithms (e.g., pedestrian detection). By communicating only the output of these algorithms, rather than the raw video streams, bandwidth is conserved.

2. Each camera is calibrated with respect to a common coordinate system (i.e., the pose and internal parameters – intrinsic and extrinsic – of the cameras are known). This step is performed before the nominal operating mode and allows the 2D image planes to be combined in order to recover 3D structures.

An example of a CCN is shown in Figure 1. In principle, a CCN can be used to detect and track objects in 3D, and to perform geometric reconstruction.

Figure 1: Photo of a calibrated camera network with four cameras. The checkerboard pattern in the center is a calibration target.

2.2 Benefits Over Existing Sensors

There are various manufacturers of video cameras designed specifically for remote monitoring of construction sites. Although these systems could be of benefit for large sites where continuous, in-person monitoring is impractical, the effort required by the human operator increases with the number of video feeds to be monitored. As the number of feeds increases, visual clutter for the human operator also increases and the ability to discriminate interesting or anomalous events can actually decrease [4].

A CCN has additional advantages over other 3D tracking sensors such as laser trackers, ultra wideband locators, laser-based localization systems, and high-frame rate 3D imaging systems. The advantages include:

- Robustness to noise: Individual localization errors can be averaged together. Localization error can be reduced by incorporating additional nodes into the network.

- Occlusion handling: CCNs are inherently redundant. By placing cameras around the perimeter of the workspace, occluded objects can still be tracked as long as they remain visible in two or more cameras.

- Arbitrarily-sized workspace: Adding coverage in a CCN can be achieved by incorporating additional nodes into the network.

- Token-less operation: Tracking an object or a person does not require placing a token such as a Radio-Frequency Identification (RFID) tag, optical receiver, or other cooperative target on the object being tracked.

- Passive operation: As long as multiple cameras in a CCN can "see" an object within a workspace, no supplementary illumination source is necessary to track that object.

- Decentralized communication: Since the nodes in a CCN can form an ad-hoc mesh network in order to share and transmit data, as long as a node can see at least one other node within the network, the CCN can function in noisy environments where a central device acting as the bridge between multiple wireless sensors can experience range and interference problems.

2.3 Site Deployment

Prior to installation at a construction site, a suitable camera configuration must be determined that will provide the desired tracking performance. The number and layout of cameras will depend on many factors including the workspace volume, camera field of view and resolution, and the desired spatial and temporal resolution. An exploration of the tradeoff between camera resolution and the number of cameras is discussed in [2]. For a given point in space, two cameras are sufficient to recover the 3D position, however, additional cameras will tend to improve the system performance by reducing the uncertainty.

Once a configuration has been chosen and the camera network has been installed on-site, the system must be calibrated. Calibration requires characterization of both intrinsic and extrinsic parameters for each camera. Intrinsic calibration determines a set of internal parameters (focal length, image center, pixel size, and lens distortion) that define how points in 3D are projected onto the 2D image plane of the camera. Extrinsic calibration determines the relative position and orientation of each camera. Both calibrations are commonly performed by taking images of calibration targets and then exploiting the known target dimensions to determine the unknown parameters [5].

2.4 Applications

The real-time 3D localization capabilities of CCNs can assist with a number of problems relevant to the construction industry. The following are some examples of CCN construction applications:

- Personnel, equipment and material tracking

- Automated project status monitoring

- Productivity assessment

- Automated safety and security alerts

3. EXPERIMENT

The majority of CCN literature focuses on short-baseline (< 5 m) configurations. Despite growing interest, a detailed performance analysis of long-baseline (> 5 m) calibrated camera networks has not been conducted.

3.1 Procedure

Four cameras were mounted indoors on tripods roughly 4 m high and were arranged in a rectangle with dimensions of approximately 12 m x 6 m. Each camera had a resolution of 1024 pixels x 768 pixels, and the height, angle, and field of view were adjusted to provide an overlapping workspace volume of roughly 4 m x 4 m x 2.4 m. To provide ground truth points for determining camera poses, the positions of 36 points distributed throughout the work volume were observed with a total station. Twelve coplanar points were taken at three different heights (roughly 0 m, 1 m, and 2 m). A black 3.81 cm retroreflector was used both as a cooperative target for measurement with a total station as well as a visible target in the camera images. The retroreflector was placed either directly on the ground or on a tripod set to the desired height. The model of total station used provides 0.2 mm spatial uncertainty, an order of magnitude better performance than estimated for the camera network.

For each observation, the 3D position of the retroreflector was found with the total station, while the corresponding 2D pixel coordinates in each image were manually identified. An example of an image from one of the cameras is shown in Figure 2. Note the retroreflector mounted on the tripod in the center of the image.

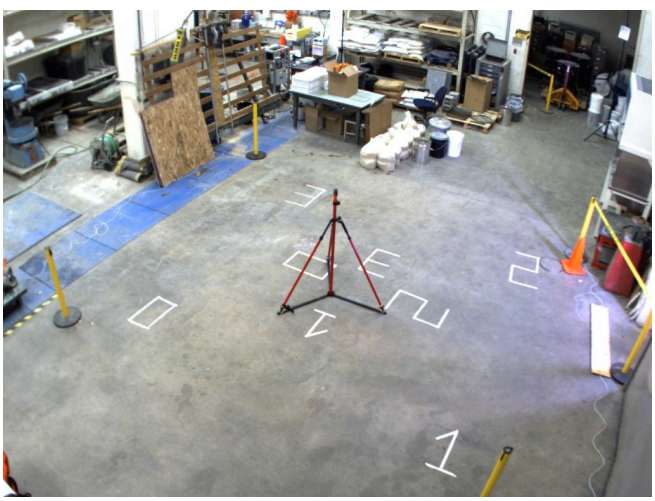


Figure 2: A typical view from one camera. The tripod in the center holds a retroreflector.

Intrinsic and extrinsic calibration, including lens distortion parameters, were obtained using the well-known Direct Linear Transformation (DLT) algorithm. First described in 1971 [1], the DLT method has become the gold-standard algorithm for camera calibration. For convenience, we briefly describe its derivation below. A thorough explanation is beyond the scope of this paper, however the interested reader is directed to [3] for a complete derivation.

The projective model for a pinhole camera maps 3D world coordinates to 2D pixel coordinates by way of a projection matrix. The basic equation is given by

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{1}$$

where $[x\ y\ 1]^T$ is the homogenous pixel coordinate vector, $[X\ Y\ Z\ 1]^T$ is the homogenous world coordinate vector, P is the 3 x 4 projective matrix which encompasses both intrinsic and extrinsic parameters, and λ is an arbitrary scaling factor. Given a pixel coordinate vector $[x_i\ y_i]^T$ and world coordinate vector $[X_i\ Y_i\ Z_i]^T$, the objective is to solve for P. Through straightforward manipulation, (1) can be rewritten in linear form as

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -x_1X_1 & -x_1Y_1 & -x_1Z_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -y_1X_1 & -y_1Y_1 & -y_1Z_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_N & Y_N & Z_N & 1 & 0 & 0 & 0 & 0 & -x_NX_N & -x_NY_N & -x_NZ_N \\ 0 & 0 & 0 & 0 & X_N & Y_N & Z_N & 1 & -y_NX_N & -y_NY_N & -y_NZ_N \end{bmatrix} \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_{11} \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ \vdots \\ x_N \\ y_N \end{bmatrix} \tag{2}$$

where the elements of P can be solved for using the method of least mean squares. When lens distortion parameters are taken into consideration, (2) takes on a non-linear form. In this case, an initial estimate for P can be obtained linearly, and a non-linear, iterative solver can be used to jointly refine the estimate for P and the distortion parameters. This process is depicted in Figure 3.

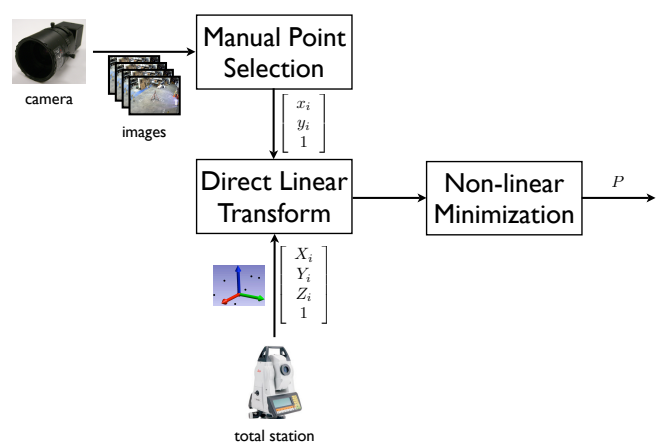


Figure 3: Process for calibrating a CCN

The approach to reconstructing 3D points from multiple camera images follows directly from (1), where P and $[x_i\ y_i]^T$ are known, and $[X_i\ Y_i\ Z_i]^T$ are the unknown values to be solved for. Geometrically, this is equivalent to projecting a line from the focal point of each camera through the pixel viewing the retroreflector. The reconstructed point is that which minimizes the perpendicular distances to these projected lines.

To test performance, a subset of the 36 observed points was reserved for calibration and the remaining points were

used to measure error. The number of points in the calibration subset varied from 10 to 35 and were chosen at random. This process was repeated 1000 times. Each of these calibration subsets was used to calibrate three camera networks consisting of 2, 3, and 4 cameras, respectively. The error metric used is the average Euclidean distance between the reconstructed point and the ground truth point as observed by the total station. The results are discussed in Section 3.2.

3.2 Results

Several trends are evident from the analysis presented in Figure 4. Reconstruction error tends to decrease as the number of calibration points increases. Reconstruction error also decreases with additional cameras. With four cameras and 35 calibration points, the CCN can resolve points to within 7 cm of ground truth, on average. This level of performance is inadequate for precision control applications, but is well-suited for tracking movements of relatively large objects such as personnel and heavy equipment.

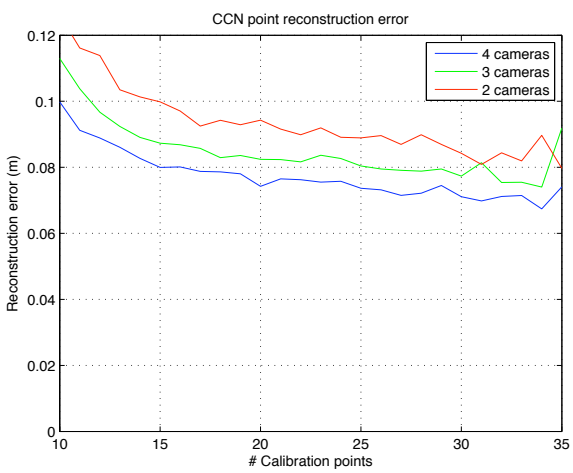


Figure 4: The localization error of the calibrated camera network decreases as the number of calibration points is increased.

4. CONCLUSION

Situation awareness is a critical factor in maintaining a safe, efficient construction site. As the quantity of information increases, intelligent sensors will become increasingly important. Calibrated camera networks are well suited to this role by providing 3D imaging capabilities that are complementary to existing technologies. In this paper, we described the characteristics and design considerations of camera networks, and empirically showed that errors within 7 cm are achievable.

As an in-depth experimental verification of calibrated camera networks has largely been unexplored, the experiment described in this paper can be considered a pilot study. Future experiments will characterize the effect of calibration schemes, workspace volume, and lighting variations on camera network error.

5. ACKNOWLEDGMENTS

The authors would like to thank Nicolas Philippon for his invaluable support of the data collection that made this work possible.

6. REFERENCES

[1] Y. I. Abdel-Aziz and H. M. Karara. Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In *Proceedings of the Symposium on Close-Range Photogrammetry*, Falls Church, Virginia, 1971. American Society of Photogrammetry.

[2] P. Carr, P. J. Thomas, and R. Hornsey. Performance benefits and limitations of a camera network. In *Proceedings of SPIE Digital Wireless Communications VII and Space Communication Technologies*, volume 5819, pages 328–339, March 2005.

[3] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

[4] R. Rosenholtz, Y. Li, and L. Nakano. Measuring visual clutter. *Journal of Vision*, 7(2):1–22, 2007.

[5] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.

A Queuing-Theoretic Framework for Modeling and Analysis of Mobility in WSNs

Harsh Bhatia
Dhirubhai Ambani Institute of of ICT
Near Indroda Circle, Gandhinagar
Gujarat - 382007, India
harsh_bhatia@daiict.ac.in

R.B. Lenin
Department of Computer Science
University of Arkansas at Little Rock
Little Rock, AR 72204, U.S.A.
rblenin@ualr.edu

Aarti Munjal
Department of Mathematical and Computer Science
Colorado School of Mines
1500 Illinois St, Golden, CO 80401, U.S.A.
amunjal@mines.edu

S. Ramaswamy
Department of Computer Science
University of Arkansas at Little Rock
Little Rock, AR 72204, U.S.A.
sxramaswamy@ualr.edu

Sanjay Srivastava
Dhirubhai Ambani Institute of of ICT
Near Indroda Circle, Gandhinagar
Gujarat - 382007, India
sanjay_srivastava@daiict.ac.in

ABSTRACT

In this paper, we present a complete framework for modeling and analysis of Mobility in Wireless Sensor Networks using OQNs with $GI/G/1$ nodes and single-class customers. We formalize and present three variations - gated queues, intermittent links and intermittent servers. We suitably modify and use the Queuing Network Analyzer (QNA) to study performance measures including: throughput, average waiting time (end-to-end delay), and packet loss probability. The results are verified by simulation in OMNeT++.

General Terms

Performance, Theory

Keywords

Unreliable servers, squared coefficient of variation, Poisson process, Rayleigh and exponential distributions

1. INTRODUCTION

Open queuing networks (OQNs) have been widely used as efficient tools to analyze the performance measures in computer and communication systems ([2], [6], [8], [9]). For many classes of OQNs, elegant and efficient solution methods exist. Well known closed product-form solutions are available for simplified networks such as Jackson and BCMP networks under a number of restrictions ([14],[5]). These networks, however, do not always apply in practice.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS' 08, August 19-21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

In wireless sensor networks with mobile nodes – we refer them as Mobile Wireless Sensor Networks (MWSNs) – the link between any two nodes will be either available or unavailable due to node mobility. Node unavailability is caused by poor node signal strengths due to low battery, weather conditions or both, or when the node is out of radio-range. In [11], the authors have developed a random walk based mobility model for a MWSN and have derived the probability distributions of link availability between any two nodes. In [4], the authors had used a queuing network for delay analysis of wireless ad hoc networks in which static nodes are distributed uniformly and independently over a torus of unit area. In this network, the transition probabilities for forwarding packets from one node to another node are considered as functions of communication area of nodes of ad hoc networks.

The mobility of nodes in a MWSN can be captured in terms of gated queues, intermittent links or intermittent servers of the queuing networks under investigation with immobile nodes. Hence, in this paper we analyze three types of OQNs with $GI/G/1$ immobile nodes and single-class customers - one with gated nodes, second with intermittent links and third with intermittent servers. We use appropriate distributions for the time durations for gate open/close, link up/down, and server up/down. We also suitably modify the Queuing Network Analyzer (QNA), a method proposed by Kuhn [10], and later expanded by Whitt [16] to approximate performance measures of large OQNs with general inter-arrival and general service distributions; to study the performance measures of networks under investigation with general distributions for the time duration of gate open and close, link up and down, and server up and down. The performance measures include throughput, average waiting time (end-to-end delay), customer loss probability and path availability.

The paper is structured as follows. Section II introduces a general OQN and its performance measures. In section III, the QNA method is discussed. In sections IV, V and VI, we modify the QNA method for OQNs with gated nodes, with

interrupted links and with interrupted servers, respectively. In section VII, we briefly outline the validation of our results and present out conclusions.

2. GENERAL OQN

A queuing network is a natural extension of a collection of interactive queuing systems, referred to as nodes. Consider an OQN with M single-server and infinite buffer nodes. Let μ_i denote the mean service rate and λ_{0i} denote the mean external arrival rate at node i. Let $p_{ij},\ i,j = 1,2,\ldots,M$, denote the transition probability by which customers finishing service at node i will join node j. The matrix $\boldsymbol{P} = (p_{ij})$ is the sub-stochastic matrix such that with probability $1 - \sum_{j=1}^{M} p_{ij}$ customers finishing service at node i will leave the network. The average arrival rate λ_j and the departure rate λ_{j0} of customers at node j are given as follows for $j = 1,2,\ldots,M$,

$$\lambda_j = \lambda_{0j} + \sum_{i=1}^{M} p_{ij}\lambda_i;\ \lambda_{j0} = \left(1 - \sum_{i=1}^{M} p_{ji}\right)\lambda_j. \tag{1}$$

For an acyclic network (where customers are not allowed to revisit the nodes) with Poisson external arrivals and exponential service times, the total arrivals at node i follow a Poisson process with rate λ_i given by (1). For cyclic networks, due to dependency amongst the arrival streams, the total arrivals at node i do not follow Poisson process though the external arrivals and service times are Poisson and exponential, respectively, in all nodes. The Jackson and BCMP networks are examples of cyclic networks [14]. For these networks, the steady-state joint probability distribution of number of customers in all the nodes of the network admits product-form solution and hence leading to closed-form solutions for performance measures like average wait times for customers in the network.

For networks with $GI/G/1$ nodes, finding closed-form solutions for performance measures is more subtle. To overcome this difficulty, the QNA method is used to find approximate solutions for the performance measures of OQN with $GI/G/1$ nodes. We discuss this method in more detail in section 3.

For the queuing networks under investigation in sections 4, 5 and 6, one can analyze the performance measures like *throughput*, *customer loss probability*, and *average waiting time*. These measures are defined as follows.

The throughput $\mathcal{T}$ of the network is given by

$$\mathcal{T} = \sum_{j=1}^{M} \lambda_{j0}, \tag{2}$$

where λ_{j0} is given by (1).

The customer loss probability P_L is given by

$$P_L = \frac{\gamma - \mathcal{T}}{\gamma},\ \text{with}\ \gamma = \sum_{j=1}^{M} \lambda_{0j}. \tag{3}$$

where γ is the total inflow to the network. The average waiting time W_s of a customer in the network (end-to-end delay) is given by

$$W_s = \sum_{j=1}^{M} W_{sj},\ \text{with}\ W_{sj} = W_{qj} + \frac{1}{\mu_j}, \tag{4}$$

where W_{sj}, W_{qj} and μ_j are the average waiting time, average queuing time and average service rate at node $j,\ j = 1,2,\ldots,M$. In this paper, we find analytical formulas for W_s of the proposed queuing-theoretic framework.

3. QNA METHOD

The QNA is an approximation technique and a software package developed at Bell Laboratories to calculate approximate congestion measures of a network of queues [16]. It is a powerful tool to analyze general queuing networks. The most important feature of the QNA is that the external arrival processes need not be Poisson, and the service-time distributions need not be exponential. The QNA can provide a fast approximate solution for large networks.

The QNA has been used extensively in many theoretical and practical applications and the results have been compared with simulation results and/or the results of other techniques ([15], [7], [13]). The low relative error percentage makes QNA one of the most important tools for analyzing general networks.

The input to QNA comprises of the number of nodes M, mean arrival rate λ_{0i} and squared coefficient of variation (SCV) c_{0i}^2 for external arrivals at node $i,\ i = 1,2,\ldots,M$, mean service rate μ_i and SCV c_{si}^2 for service time at node $i,\ i = 1,2,\ldots,M$, and the transition (routing) probabilities $p_{ij},\ i,j = 1,2,\ldots,M$.

The total arrival rate at node i is given by (1). The traffic intensities or utilization of node i is given by $\rho_i = \lambda_i/\mu_i$. The arrival rate from node i to node j is given by $\lambda_{ij} = p_{ij}\lambda_i$. The proportion of arrivals to node j from node i is given by $q_{ij} = \lambda_{ij}/\lambda_j,\ i = 0,1,\ldots,M$. The SCV of the total arrivals at node j is calculated as follows [16]:

$$c_{aj}^2 = a_j + \sum_{i=1}^{M} b_{ij} c_{ai}^2, \tag{5}$$

where a_j and b_{ij} are derived after considering merging and splitting of traffic streams and are given as follows:

$$\begin{aligned} a_j &= 1 + w_j\{(q_{0j}c_{0j}^2 - 1) \\ &\quad + \sum_{i=1}^{m} q_{ij}\left[(1-p_{ij}) + (p_{ij}\rho_i^2 x_i)\right]\} \end{aligned} \tag{6}$$

and $b_{ij} = w_j p_{ij} q_{ij}(1-\rho_i^2)$, $x_i = \max_{1\le i\le M}(c_{si}^2, 0.2)$, $w_j = \left[1 + 4(1-\rho_i)^2(v_j - 1)\right]^{-1}$, and $v_j = \left[\sum_{i=0}^{m} q_{ij}\right]^{-1}$. The SCV c_{aj}^2 can also be calculated as follows [16, eqn. (41)]:

$$c_{aj}^2 = 1 - w_j + w_j \sum_{i=1}^{M} p_{ij} c_{ij}^2, \tag{7}$$

where c_{ij}^2 is the SCV of the traffic flow from node i to node j and is given by

$$\begin{aligned} c_{ij}^2 &= q_{ij}[1 + (1-\rho_i^2)(c_{ai}^2 - 1) \\ &\quad + \rho_i^2(c_{si}^2 - 1)] + 1 - q_{ij}. \end{aligned} \tag{8}$$

The expressions in (6) and (8) are obtained by setting $m = 1$ (number of servers in each node) and $v_{ij} = 0$ in equations (25) and (41) of [16], respectively. The approximate formula for the average waiting time of a customer at node j is then given by [16]

$$W_{qj} = \frac{\lambda_j(c_{aj}^2 + c_{sj}^2)g_j}{2(1-\rho_j)}, \tag{9}$$

where g_j is a function of ρ_j, c_{aj}^2 and c_{sj}^2, such that $g_j = 1$ for $c_{aj}^2 \geq 1$. On substituting (9) in (4) we get the end-to-end delay W_s.

4. OQN WITH GATED NODES

We consider an OQN as discussed in section 2 with the following modifications: each node has a gate which goes on and off with rates α and β with variances v_{on} and v_{off}, respectively. When the gate is on, the customers are allowed to enter the queue, otherwise they are lost including the external arrivals as shown in Figure 1.

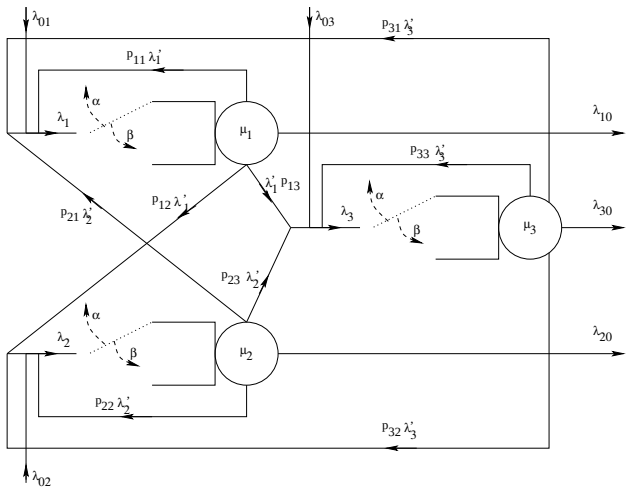


Figure 1: 3-node OQN with gated nodes

Let λ'_j denote the mean arrival rate of the interrupted arrival process, due the presence of on-off gate, at node j. The total arrival rate λ_j at node j due to external and internal traffic flows at node j is given by $\lambda_j = \lambda_{0j} + \sum_{i=1}^{M} p_{ji}\lambda'_i$.

THEOREM 1. *The mean arrival rate $\lambda'_j = p_{on}\lambda_j$ and SCV $c_{aj}^{'2}$ of the interrupted arrival process at node j, due to presence of the on-off gate, is given by*

$$c_{aj}^{'2} = c_{aj}^2 + k\lambda_j, \tag{10}$$

where p_{on} is the probability that the gate of node j is on and is given by

$$p_{on} = \frac{\beta}{\alpha+\beta}, \text{ and} \tag{11}$$

$$k = \frac{\alpha(v_{on}\alpha^2 + v_{off}\beta^2)}{(\alpha+\beta)^2}. \tag{12}$$

Proof: In [3], the author has analyzed switched general process (SGP), denoted by A, wherein the arrivals to the queuing system switches between two general renewal processes A_1 and A_2 with rates λ_1 and λ_2 according to a general renewal switching (on-off) periods V_1 and V_2 with rate α and β, respectively. The author has derived the effective mean arrival rate λ' and SCV $C(A)$ of A as

$$\begin{aligned}
\lambda' &= \frac{\lambda_1 E[V_1] + \lambda_2 E[V_2]}{E[V_1] + E[V_2]}, \\
C(A) &= \frac{\lambda_1 C(A_1) E[V_1]}{\lambda'(E[V_1] + E[V_2])} + \frac{\lambda_2 C(A_2) E[V_2]}{\lambda'(E[V_1] + E[V_2])} \\
&\quad + \frac{(\lambda_1 - \lambda_2)^2 \left[E[V_1]^2 Var[V_2] + E[V_2]^2 Var[V_1]\right]}{\lambda'(E[V_1] + E[V_2])^3},
\end{aligned}$$

where $E[V_i]$ and $Var[V_i]$ are the mean and variance of $V_i,\ i = 1, 2$, respectively.

Since in our network, arrivals are not allowed to enter a node when its gate is closed, we have $\lambda_2 = 0 = C(A_2)$. Setting $\lambda_1 = \lambda,\ E[V_1] = 1/\alpha,\ E[V_2] = 1/\beta,\ Var[V_1] = v_{on}$ and $Var[V_2] = v_{off}$, we get

$$\lambda' = \frac{\beta}{\alpha+\beta}\lambda, \tag{13}$$

$$C(A) = C(A_1) + \lambda\frac{\alpha(v_{on}\alpha^2 + v_{off}\beta^2)}{(\alpha+\beta)^2}. \tag{14}$$

Since λ_j is the total arrival rate at node j of the queuing network with gated nodes, the mean arrival rate λ'_j and SCV $c_{aj}^{'2}$ of the effective arrival process at this node j are obtained by replacing λ' by λ'_j, λ by λ_j, $C(A)$ by $c_{aj}^{'2}$ and $C(A_1)$ by c_{aj}^2 in (13) and (14). Hence the theorem. Q.E.D

The utilization at node j is given by $\rho_j = \lambda'_j/\mu_j$ and the arrival rate from node i to node j is given by $\lambda'_{ij} = \lambda'_i p_{ij}$. The proportion of arrivals from node i to node j is given by $q_{ij} = \lambda'_{ij}/\lambda_j,\ i \geq 0$. Customers from node j leave the network with rate

$$\lambda_{j0} = \left(1 - \sum_{i=1}^{M} p_{ji}\right)\lambda'_j,\ j = 1, 2, \ldots, M. \tag{15}$$

Substituting (15) in (2) and in (3), the throughput $\mathcal{T}$ and customer loss probability $P_L(N)$ can be calculated, respectively.

The c_{aj}^2 in (10) is computed as follows: For $j = 1, 2, \ldots, M$,

$$c_{aj}^2 = a_j + \sum_{i=1}^{M} b_{ij}c_{ai}^2 + k\sum_{i=1}^{M} b_{ij}\lambda_i, \tag{16}$$

where k is given by (12). By replacing λ_j by λ'_j and c_{aj}^2 by $c_{aj}^{'2}$, we can use the QNA method (modified) presented in section 3 to compute the $W_{qj},\ j = 1, 2, \ldots, M$ of this gated OQN.

5. OQN WITH INTERMITTENT LINKS

We consider an OQN as discussed in section 2 with the following modifications: the link connecting between any two nodes goes on and off with rates α and β with variances v_{on} and v_{off}, respectively. When the link is on between nodes i and j, the customers departing node i are allowed to enter the queue of node j with probability p_{ij} as shown in Figure 2. In the model discussed in previous section, when the gate of a node is off, then the node is disconnected from the entire network since no other node can send customers to this node. This is, however, an unrealistic assumption model as for as its application to MWSNs are concerned because in MWSNs, a link between two nodes may be down, but these nodes may still be connected to other nodes in the network.

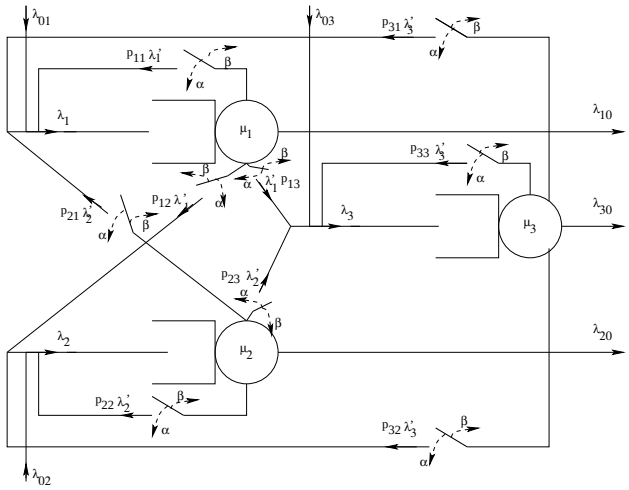


Figure 2: 3-node OQN with intermittent links

THEOREM 2. *The mean arrival rate λ_j and SCV $c_{aj}^{'2}$ of the interrupted arrival process, due to presence of the on-off links, at node j of the network is given by*

$$\lambda_j = \lambda_{0j} + p_{on} \sum_{i=1}^{M} \lambda_{ij}, \tag{17}$$

$$c_{aj}^{'2} = 1 - w_j(1 - \sum_{i=1}^{M} p_{ij} c_{ij}^2 - k \sum_{i=1}^{M} p_{ij} \lambda_{ij}), \tag{18}$$

where p_{on} is the probability that the link between two nodes is on and is given by (11) and w_j is the same as the one given in (6) and k is given by (12).

Proof: Due to the presence of on-off links between nodes i and j, the mean arrival rate λ'_{ij} between these nodes is obtained by simply replacing λ by λ_{ij} and λ' by λ'_{ij} in (14). That is,

$$\lambda'_{ij} = p_{on} \lambda_{ij}. \tag{19}$$

Similarly, the SCV c_{ij}^2 of the traffic flow between node i and node j is obtained by replacing $C(A)$ by $c_{ij}^{'2}$, $C(A_1)$ by c_{ij}^2 and λ by λ_{ij} in (14). That is,

$$c_{ij}^{'2} = c_{ij}^2 + k\lambda_{ij}, \tag{20}$$

where k is given by (12).

The total arrival rate at node j is given by $\lambda_j = \lambda_{0j} + \sum_{i=1}^{M} \lambda'_{ij}$. On substituting for λ'_{ij} from (19) in this equation, we get (17).

The SCV c_{aj}^2 of the total traffic flow to node j is obtained by replacing c_{ij}^2 by $c_{ij}^{'2}$ in (7). That is,

$$c_{aj}^2 = 1 - w_j + w_j \sum_{i=1}^{M} p_{ij} c_{ij}^{'2}, \tag{21}$$

On substituting (20) in (21) yields (18). Hence the theorem. Q.E.D.

Using (17), (2) and (3), we can compute the $\mathcal{T}$ and P_L. Using (18) and (9) we can compute W_{qj} at node j for $j = 1, 2, \ldots, M$.

6. OQN WITH INTERMITTENT SERVERS

We consider an OQN as discussed in section 2 with the following modifications: the server in each node goes on and off with rates α and β, respectively. When the server of node j is on, the customers are served at the rate μ_j and the server is off, the customers will wait in the queue until the server becomes on. A 3-node network with intermittent servers is shown in Figure 3.

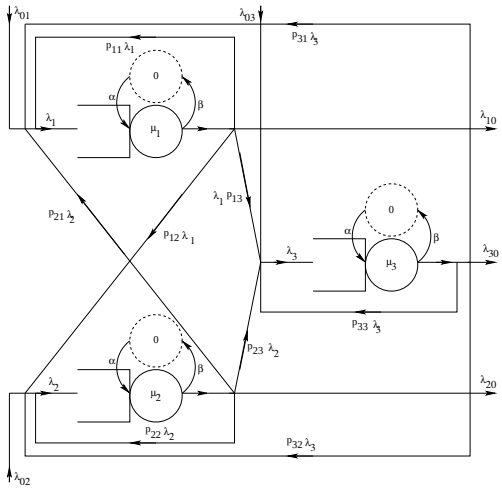


Figure 3: 3-node OQN with intermittent servers

THEOREM 3. *For the network under investigation, the mean service rate $\mu'_j = p_{on}\mu_j$ and the SCV of the effective service time distribution is given by*

$$c_{sj}^{'2} = c_{sj}^2 + k\mu_j, \tag{22}$$

where p_{on} is the probability the server at node j goes on and is given by (11) and the constant k is given by $k = \alpha p_{on}^2 \frac{(1+c_{off}^2)}{\beta^2}$ with c_{off}^2 being the SCV of the server off distribution.

Proof: This model falls under the category of the unreliable server models. In [1, section 10.2.2], the authors have given the first two moments of the effective service time distribution for an unreliable server with general service and general up and down time distributions as follows[1]:

$$E[G] = E[B](1 + \eta E[D]) \tag{23}$$

$$E[G^2] = E[B^2](1 + \eta E[D])^2 + E[B]\eta E[D^2], \tag{24}$$

where G is the random variable denoting the effective service time of an interrupted server whose down time is given by the random variable D. $E[B]$ and $E[B^2]$ are the first and second moments of the processing time of this unreliable server when it is up and η is its expected up time. Though the second moment result is proved with the assumption that the arrivals follow Poisson process, it can be proved that it holds even for general arrival process when the arrivals are renewal processes. Dividing (24) by square of (23) yields

$$c_s^{'2} = c_s^2 + \frac{\eta(1 + c_{off}^2)}{E[B]E[D]^2(1 + \eta E[D])^2}, \tag{25}$$

[1]Reproduced here for ease of reading and completeness.

where $c_s^{'2}$, c_s^2 and c_{off}^2 are the SCVs of the effective service time, processing time and server off time distributions, respectively.

For node j of the queuing network under investigation we have $E[B] = 1/\mu_j$, $E[D] = 1/\beta$, $E[G] = 1/\mu'_j$, $\eta = \alpha$, $c_s^{'2} = c_{sj}^{'2}$, $c_s^2 = c_{sj}^2$. Hence the theorem is proved on substituting these in (23) and (25). Hence the theorem. Q.E.D.

Since there are no changes in the arrival process and its moments, the inflow of customers is equal to the outflow as we do not lose any customers and hence $P_L = 0$. We can get other performance measures by simply replacing μ by μ', c_{sj}^2 by $c_{sj}^{'2}$ in the standard QNA.

7. NUMERICAL RESULTS

To verify the analytical results, we simulated these three types of queuing network models. The stopping criterion for the simulation guaranteed a maximum relative error of 5%. The relative error in the simulation was computed from the associated confidence interval, which was obtained through the usual normal distribution approximation.

We used OMNeT++ [12], a discrete-event simulation package to perform all the simulations. In the simulation, the following inputs were given depending on the type of networks. For all the three types of networks: External arrival distributions and the corresponding rates λ_{0j}, $j = 1, 2, \ldots, M$. Service time distributions and the corresponding rates μ_j, $j = 1, 2, \ldots, M$. For the gated nodes network: On-off distributions of M gates and the corresponding rates α and β. For the intermittent links network: On-off distributions of links and the corresponding rates α and β. For the intermittent server network: On-off distributions of servers and the corresponding rates α and β.

Node j	$1/\mu_j$	λ_{0j}	W_s		Error
			QNA	Simulation	
1	0.0400	2.0	0.0711	0.0682	4.0925
2	0.0400	2.0	0.1422	0.1365	4.0127
3	0.0400	2.0	0.1196	0.1137	4.9038
4	0.0400	2.0	0.2915	0.2965	1.7307
5	0.0400	2.0	0.1030	0.0979	4.9087
6	0.0400	2.0	0.2248	0.2219	1.2865
7	0.0400	2.0	0.0756	0.0723	4.3541
8	0.0400	2.0	0.1025	0.0996	2.8141
9	0.0400	2.0	0.0722	0.0690	4.3974
10	0.0400	2.0	0.0648	0.0622	3.9498

Table 1: Network with 10 gated nodes

We did not use any other information like SCV and distribution of internal arrivals. In each simulation we computed the following statistics: mean and standard deviation of inter-arrival times, W_{sj} for each node, the customer loss probability and W_s of the network.

In Tables 1 - 3, we have compared both analytical and simulation results for queuing networks with specific distributions for on-off durations of gates, links and servers and service time of servers. In all the examples, the external arrivals to the networks are assumed to follow Poisson processes.

In Table 1, we considered a network with 10 gated nodes, deterministic service, and Rayleigh on and off times with rates $\alpha = 0.7181$, $\beta = 0.0319$. In Table 2, we considered a network with 10 nodes connected by intermittent links, exponential service, and exponential on and off times with rates $\alpha = 10/9$, $\beta = 25$. In Table 3, we considered a network with 10 intermittent server nodes, deterministic service, Rayleigh on and off times with rates $\alpha = 0.7181$, $\beta = 0.0319$.

Node j	$1/\mu_j$	λ_{0j}	W_s		Error
			QNA	Simulation	
1	0.0400	2.0	0.0724	0.0719	0.7026
2	0.0400	2.0	0.1029	0.1026	0.3236
3	0.0400	2.0	0.0962	0.0950	1.2475
4	0.0400	2.0	0.1222	1.1207	1.2520
5	0.0400	2.0	0.0902	0.0885	1.8765
6	0.0400	2.0	0.1173	0.1159	1.2080
7	0.0400	2.0	0.0750	0.0746	0.4724
8	0.0400	2.0	0.0896	0.0881	1.6272
9	0.0400	2.0	0.0781	0.0762	2.3851
10	0.0400	2.0	0.0679	0.0675	0.5775

Table 2: 10-node Network with intermittent links

The simulation values match with the analytical values given by QNA with very less relative error. In most of the cases, the relative error (last columns in all tables) is approximately 1%.

Node j	$1/\mu_j$	λ_{0j}	W_s		Error
			QNA	Simulation	
1	0.0511	2.0	0.0504	0.0485	5.0933
2	0.1415	2.0	0.1423	0.1544	9.0869
3	0.1036	2.0	0.1043	0.1047	1.0618
4	0.0653	2.0	0.6283	0.0626	4.1172
5	0.0832	2.0	0.0833	0.0812	2.3861
6	0.4423	2.0	0.4494	0.4051	8.4119
7	0.0546	2.0	0.0544	0.0524	4.0811
8	0.0823	2.0	0.0825	0.0808	1.7634
9	0.0516	2.0	0.0514	0.0491	4.8271
10	0.0450	2.0	0.0449	0.0432	3.8929

Table 3: Network with intermittent server nodes

8. CONCLUSION

The research work presented in this paper provides a queuing-theoretic framework to model and analyze MWSNs by means of a mechanism that leverages the power of QNA. While QNA is an approximation technique, the results show that it results in very less error; and its use is highlighted by its applicability to model and analyze the vagaries of complex real-world networks. The paper is an attempt to analytically address this emerging need in this area; and hence we have made some realistic assumptions. Our current research is directed towards relaxing some of these assumptions, finding various other parameters - such as incorporating finite buffers, varying battery levels of a node and prevailing weather conditions; and identifying appropriate statistical distributions for such parameters.

Acknowledgment

This work is based in part, upon research supported by the National Science Foundation (grant nos. CNS-0619069, EPS-0701890 and OISE 0650939), National Board of Higher Mathematics, India (grant no. 48/5/2004/R&D-II/2120), NASA EPSCoR Arkansas Space Grant Consortium (grant

no. UALR 16804), and Acxiom Corporation (contract no. 281539). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agencies.

9. REFERENCES

[1] I. Adan and J. Resing. Course note on queueing theory. http://www.win.tue.nl/ iadan/que/, 2003.

[2] D. Bertsekas and R. Gallager. *Data Networks*. Pearson education, 2006.

[3] V. N. Bhat. Renewal approximations of the switched Poisson processes and their applications to queueing systems. *Journal of the Operational Research Society*, 45(3):345–353, 1994.

[4] N. Bisnik and A. Abouzeid. Queueing network models for delay analysis of multihop wireless ad hoc networks. In *Proceeding of the 2006 international conference on Communications and mobile computing*, pages 773–778, Vancouver, British Columbia, Canada, 2006. IWCMC'06.

[5] E. Gelenbe and G. Pujolle. *Introduction to Queueing Networks*. John Wiley and Sons, 1987.

[6] G. Giambene. *Queueing Theory and Telecommunications – Networks and Applications*. Springer, 2005.

[7] B. R. Haverkort. Approximate analysis of networks of $PH/PH/1/K$ queues with customer loss: Test results. *Annals of Operations Research*, 9:271–291, 1998.

[8] B. R. Haverkort. *Performance of Computer Communication Systems – A Model-Based Approach*. Wiley, 1999.

[9] F. J. Hayes and T. V. Ganesh Babu. *Modeling and analysis of telecommunications networks*. Wiley-Interscience, Hoboken, New Jersey, 2004.

[10] P. Kuhn. Approximate analysis of general queueing networks by decomposition. *IEEE Transactions on Communications*, 27(1):113–126, 1979.

[11] A. B. McDonald and T. Znati. A mobility-based framework for adaptive clustering in wireless ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 17(8):1466–1487, 1999.

[12] OMNeT++. http://www.omnetpp.org/index.php, 2007.

[13] G. Schneider, M. Schuba, and B. R. Haverkort. QNA-MC: A performance evaluation tool for communication networks with multicast data streams. In R. P. et al., editor, *Tools'98*, 1469, pages 63–74, Spain, 1998. Universitat de les Illes Balears, Lecture Notes in Computer Science, Springer-Verlag.

[14] J. Walrand. *An Introduction to Stochastic Modeling*. Academic Press, 3rd edition, 1998.

[15] W. Whitt. Performance of the queueing network analyzer. *The Bell System Technical Journal*, 62(9):2818–2843, 1983.

[16] W. Whitt. The queueing network analyzer. *The Bell System Technical Journal*, 62(9):2779–2815, November 1983.

Towards Information Networks to Support Composable Manufacturing

Mahesh Mani
Design and Process Group
Manufacturing System Integration Division
National Institute of Standards and Technology
100 Bureau Drive, Gaithersburg, MD USA
(301) 975 -5219
mahesh@nist.gov

Albert. T. Jones
Enterprise Systems Group
Manufacturing System Integration Division
National Institute of Standards and Technology
100 Bureau Drive, Gaithersburg, MD USA
(301) 975 - 3554
albert.jones@nist.gov

Junho Shin
Enterprise Systems Group
Manufacturing System Integration Division
National Institute of Standards and Technology
100 Bureau Drive, Gaithersburg, MD USA
(301) 975 -3654
junho.shin@nist.gov

Ram D. Sriram
Design and Process Group
Manufacturing System Integration Division
National Institute of Standards and Technology
100 Bureau Drive, Gaithersburg, MD USA
(301) 975 -3507
sriram@nist.gov

ABSTRACT
Rigid, supply-chain organizational structures are giving way to highly dynamic collaborative partnerships. These partnerships will develop rapidly by composing global manufacturing resources in response to open market opportunities and they will disband just as rapidly when those opportunities disappear. Cooperation, coordination, and distributed decision-making will be critical to the success of these dynamically composable systems. That success, in turn, will depend on the creation of a manufacturing information network that automates as much as possible, the identification, formalization, encoding, and sharing of appropriate manufacturing- and business-related knowledge. In this paper we present some of the issues and requirements associated with the creation of such networks.

Categories and Subject Descriptors
C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design- *Distributed networks.*

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

General Terms
Algorithms, Management, Performance, Design, Standardization, Languages

Keywords
Manufacturing information networks, interoperability, common vocabulary, standards, composable manufacturing

1. INTRODUCTION
For most of the 20th century, competitive advantage was defined by the production and labor capabilities of individual original equipment manufacturers (OEMs). In the mid 90s, OEMs sought to reduce production and labor costs by outsourcing those capabilities globally. Thus, the rigid supply chain structures of the past are slowly giving way to virtual supply networks of collaborative partnerships. These partnerships will develop rapidly by composing global resources in response to market opportunities and they will disband just as rapidly when those opportunities disappear. These resources now include designers, engineers, planners, transporters, suppliers, fabricators, assemblers, buyers, vendors, and service providers, among others. The only way to improve competitive advantage in such a networked system is to improve the orchestration of all of these resources so that they behave as if they were a single virtual factory.

A major concern in these networked systems is the availability of the right product, process, and business information when and where it is needed. We believe that this requires a separate integration infrastructure, which we call the manufacturing

information network (MIN). A MIN will enable the discovery, coordination, and automated exchange of information regardless of where it resides in the system. The reason that discovery, co-ordination, and automated exchange is so important is that the system orchestrator must have (1) intimate, real-time knowledge of the capabilities and capacities of all potential partners (2) accurate predictions of customer demands and product requirements, (3) the ability to match supplier capabilities to those requirements, and (4) the skills to manage the entire, global set of production and delivery schedules.

In this paper we present our vision for a MIN, which is based on the concepts of the internet of services (IOS) and service-oriented architecture (SOA). The rest of the paper is organized as follows: Section 2 first presents the modeling of composable manufacturing systems followed by section 3 briefly discussing information exchange in composable manufacturing systems. Section 2 and 3 aim to present the diverse information exchange in MIN. Section 4 briefly presents the commonly used standards, reference models and business processes. Sections 5 and 6 present our proposed MIN approach and related research issues respectively. Finally Section 7 presents our conclusions.

2. MODELING COMPOSABLE MANUFACTURING SYSTEMS

Bringing complex products to customers requires close collaborations among a number of functions including design, engineering, manufacturing, and logistics. As noted above, these functions are performed by resources that are geographically distributed around the world. As described in [1,2] these resources are modeled frequently as autonomous, interacting software agents with each agent executing one function. Agents are implemented as intelligent web-based applications that wrap functional applications and manage electronic information exchange. The communications protocols used by these software agents have been developed by FIPA (Foundation of Intelligent and Physical Agents)[1], an IEEE Computer Society standards organization that promotes agent-based technology and the interoperability of its standards with other technologies. These simple protocols govern the information data flow through the participating agents, thus ensuring seamless data and information transfer. Their potential benefit is that the agents can exchange the same information as their real-world counterparts. The principal drawback is that they are not powerful enough to allow the agents to understand most of that information. Therefore, interoperability remains a major impediment for software agents, just like it does for their real-world counterparts.

3. INFORMATION EXCHANGE IN COMPOSABLE MANUFACTUING SYSTEMS

Information exchange is critical for collaborations and management at every phase of the product lifecycle. In current supply chains, these exchanges take place typically through numerous message-based transactions between the various partners. The granularity and complexity of the information exchanged depends on the type of transaction and the partners involved. The type of transaction is based on an agreed upon understanding of the business process being executed (see section 4). This understanding is possible because (1) the partners are known in advance, that is the organizational structure is somewhat rigid and (2) the negotiation is dictated by the OEM, who sits at the top of a command/control hierarchy.

Several recent reports on the "Future of Manufacturing" predict two major changes in these supply chains [3,4]. First, OEMs' former command-and-control business model will evolve into more of a collaborative and negotiated partnership model. Second, the supply base will no longer be static and known in advance; rather, it will change regularly. In this new environment where systems are composed dynamically, the existing transaction-based approach will not be adequate for all information exchanges. Before describing the MIN approach, we briefly review some commonly used information standards and reference models governing message transactions and business processes.

4. COMMONLY USED STANDARDS, REFERENCE MODELS AND BUSINESS PROCESSES

A number of supply-chain Common Business Processes (CBP), information standards, messaging standards, and reference models have been defined. Business process specification is a declaration of the partners, roles, collaborations, choreography and business document exchanges that make up a business process. Some are industry specific and some are not. CBPs are industry neutral and re-usable business processes. Various components of a common business process specification can be re-used to create new business processes (See Figure 1]. Re-use will typically occur at the business process, business collaboration, business transaction, and business document model components.

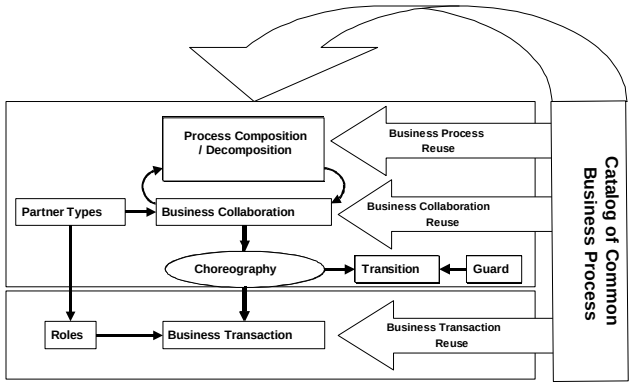


Figure 1. Catalog of Common Business Processes[2]

ANSI X12 (also known as ANSI ASC X12) is the official designation of the U.S. national standards body for the development and maintenance of Electronic Data Interchange (EDI) standards. X12 has an underlying syntax, which is an ANSI standard. Within that syntax, there are directories of data elements, composite data elements, segments, and messages. There are conventions for placing messages in an "envelope"

[1] http://www.fipa.org/specifications/index.html

[2] http://www.ebxml.org/specs/bpPROC.doc

which identifies the sender and receiver and other attributes of a transmission[3].

EDIFACT (Electronic Data Interchange for Administration, Commerce, and Transport) is the international Electronic Data Interchange developed under the United Nations. EDIFACT has an underlying syntax, which is an ISO standard. Within that syntax, there are directories of data elements, composite data elements, segments, and messages[4]. It also provides interactive exchange protocols and standard messages.

EIDX represents the Electronics Industry Data Exchange Group. As part of the Computing Technology Industry Association (CompTIA), EIDX is committed to advancing industry growth through the development of standards, best practices, accreditations, professional education and development, tools and business solutions[5]. EIDX provides implementation guidelines for EDI and XML-based vocabulary standards.

RosettaNet is an independent, self-funded, non-profit consortium dedicated to the development and deployment of standard electronic business interfaces. These standards form a common e-Business language, aligning processes between supply chain partners on a global basis. RosettaNet's Partner Interface Processes (PIP) standards allows trading partners of all sizes to connect electronically to process transactions and move information within their extended supply chains[6].

The Open Applications Group Integration Specification (OAGIS) is an effort to provide a canonical business language for information integration. It uses XML as the common alphabet for defining business messages, and for identifying business processes (scenarios) that allow businesses and business applications to communicate. OAGIS provides the definition of business messages in the form of Business Object Documents (BODs) standards and example business scenarios that provide example usages of the BODs[7].

Table 1 presents possible mappings between several of these standards.

Table 1. Example mapping of common business processes

Common Business Processes	Normative Category	Normative Sub Category	EDIFACT including sub-sets	X12 including sub-set	RosettaNet Partner Interface Process	CII (HWSW 001A)	OAG BODs
Distribute Dispatch Instructions	**Procurement Management**	**Transportation and Distribution**	**INSDES**	**862, 858**	**PIP3B1**		**165_sync_shipschd_001**
Notify Of Advance Shipment	**Procurement Management**	**Transportation and Distribution**	**DESADV**	**856, 869**	**PIP3B2**	**0520**	**165_sync_shipschd_001**

Besides the above, SCOR and DCOR present process reference models for any supply chain.

SCOR: Supply-Chain Operations Reference is a process reference model that has been developed and endorsed by the Supply-Chain Council (SCC). The SCC is an independent, not-for-profit, global corporation with membership open to all companies and organizations interested in applying and advancing the state-of-the-art in supply and design chain management systems and practices. SCOR has been adopted as the cross-industry de facto standard diagnostic tool for supply chain management[8]. It is a hierarchical model that has five major building-block processes: plan, make, source, deliver, and return.

DCOR: The Design Chain Operations Reference-model (DCOR)[9]. The newest model from the SCC, the DCOR-model captures the SCC's Technical Development Steering Committee's consensus view of design chain management. The structure is based on the same hierarchical philosophy as SCOR, but with five different building-block processes: plan, research, design, integrate, and amend.

5. OUR APPROACH: MANUFACTURING INFORMATION NETWORKS (MINS)

As noted above, future manufacturing networks will be dynamically created from global resources (see Figure 2). Orchestration, cooperation, coordination, and distributed decision-making will be critical to their success. The execution of these functions, in turn, will depend on the creation of an infrastructure that automates, as much as possible, the exchange of required manufacturing- and business-related knowledge. We call such an infrastructure a manufacturing information network (MIN). A MIN (see Figure 3) is created every time an application needs to find or exchange information.

Conceptually, the MIN will constitute another "layer" of open cyberspace, sitting atop the Internet and other evolving Web technologies. As such, it will be independent of any particular enterprise software applications. Such an infrastructure will enable the complete virtualization of and ubiquitous access to global manufacturing resources allowing information to be exchanged anywhere, anytime, on any device. The primary feature of the MIN is composability.

Composability is a system design principle that deals with component inter-relationships. A high degree of composability means the system has recombinant components that can be selected and assembled in arbitrary ways to satisfy user requirements. A composable component must be self-contained (modular), self-descriptive, and be able to integrate easily with other components in the system.

The key to achieving this capability is to link these various components together in arbitrary ways to exchange the information necessary to meet business needs.

The primary components of the MIN include services, repositories, brokers, and registries (See Figure 2).

[3] http://www.x12.org

[4] http://www.stylusstudio.com/edifact

[5] http://eidx.comptia.org

[6] http://www.rosettanet.org

[7] http://www.openapplications.org

[8] http://www.supply-chain.org/cs/root/scor_tools_resources/scor_model/scor_model

[9] http://www.supply-chain.org/cs/root/scor_tools_resources/designchain_dcor/dcor_models

Services: The goal of the services is to be discovered and used as frequently as possible and by as many different actors. A service must be engineered for interoperability and designed to live in a completely open environment where they will not know who their potential partners will be in advance. This means, at a minimum, they must publish complete (register with the Service Registry), semantically rich descriptions and representations of what they do (their capabilities), the information they need to do it (their inputs), and the information they provide when they are done (their outputs). Ultimately, if the descriptions and representations are not easily understood, a service will not get used.

A number of different services are envisioned:

- Services that facilitate real-time information sharing and collaboration between enterprises, such as reasoning, searching, discovery, composition, assembly, and delivery of semantics automatically.
- Services that leverage emerging Web technologies for enabling a new generation of information-based applications that can self-compose, self-declare, self-document, self-integrate, self-optimize, self-adapt, and self-heal.
- Services that support knowledge creation, management, and acquisition to enable knowledge sharing between virtual organizations.
- Services that help connect islands of interoperability by federating, orchestrating, or providing common e-business infrastructural capabilities such as digital signature management, certification, user profiling, identity management, and libraries of templates and interface specifications.
- Services that support the use of mashup technologies such as verification of credentials; reputation management; assessment of e-business capabilities; assessment of collaboration capabilities; facilities for data sourcing, integrity, security and storage; contracting; registration and labeling; and payment facilities, among others.

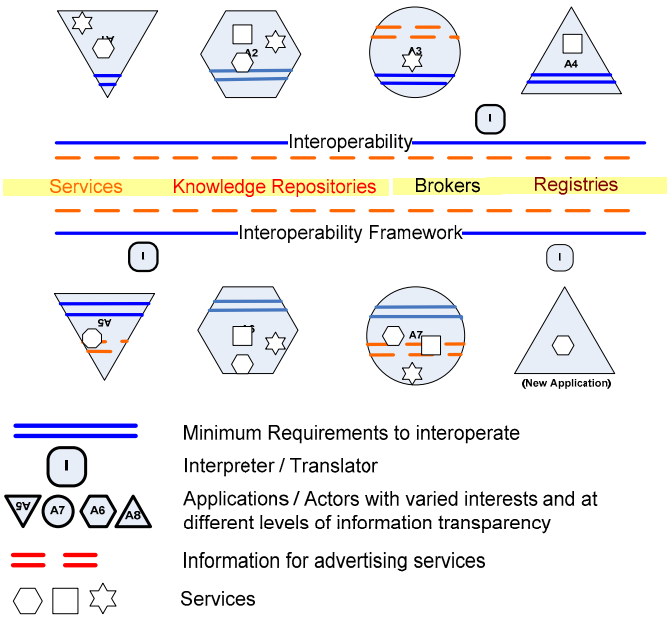


Figure 2. MIN Concept

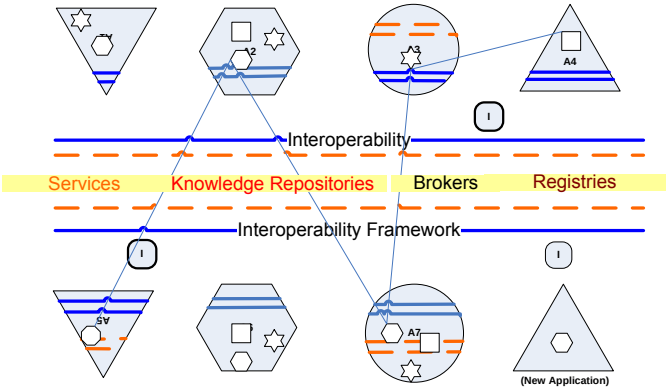


Figure 3. Composable-On-Demand MINs

Knowledge Repositories: Knowledge Repositories continuously capture, store, and analyze all manufacturing-related information, for example, the knowledge of assets, manufacturing facilities/ capabilities of an organization or enterprise. Participating applications can then query or browse both structured and unstructured information in order to retrieve and update information. Besides being dynamic, knowledge repositories must cater for explicit form of knowledge to be able to retrieve context sensitive data.

Brokers: Brokers can also be referred to as the MIN infrastructure bureau services. Depending on the demand and request, the broker identifies the service component that fills the need, locates it, and plugs it into the framework. The broker's function is to select and assemble components belonging to different applications into integrated processes; for example, for order fulfillment. This integration is analogous to the formation of virtual enterprises from separate enterprises. Component technology provides seamless communication between applications residing in the different supply-chain partners.

Registries: Registries can be thought of as advertisement services or yellow pages. They can also be useful to locate knowledge repositories and services.

6. RESEARCH ISSUES

To address the development of the above discussed composable MIN and its corresponding infrastructure, we need structured research into the conditions/requirements for interoperability in these networks. The following research questions will have to be investigated:

- an interoperability framework in the context of MIN
- specific roles of a MIN component like brokers, services, registries, will have to be investigated as part of the framework
- minimum requirements for an application to participate in MIN (for service provider and service requester)
- level of information transparency required to participate in MIN
- roles of an interpreter/ translator for new application services

- common vocabulary requirements for interoperability
- level of intelligence that will make these application systems adaptive and self configuring (implies an application to be either context-aware, adaptive or anticipated based on experience)
- specific use cases of manufacturing and business processes (scenarios or transactions)

There are undertakings at NIST to develop and demonstrate an open, standards-based, testing and integration infrastructure that enables the automated exchange of manufacturing information among suppliers [3]. This infrastructure will provide the foundation for the new types of collaboration and management described above as a number of NIST-supported economic impact studies claim that such an infrastructure does not exist today [4].

7. CONCLUSION

In this paper, we presented an approach towards information networks that supports composable manufacturing. We identified a need for a MIN infrastructure that provides a mechanism for information acquisition, sharing, delegation of tasks, and decision making between the various entities. We speculate on how the future of MINs may evolve as composable on demand MINs. Correspondingly we presented and discussed the issues and prerequisites that need to be addressed to support interoperability in such MINs.

Semantic interoperability may be the key to MINs of the future. Adopting semantic interoperability for product-process automates information flow in an extended enterprise among different levels of information resources. Future research is on composable MIN and architectures to enable the way manufacturing organizations use information technology (computers, networks, information systems, data, algorithms, and decision support) to make their manufacturing processes more effective.

8. ACKNOWLEDGEMENTS

The authors wish to acknowledge Steven Fenves, Eswaran Subramanian and Nenad Ivezic for participating and contributing to discussions pertaining to the Manufacturing Information Networks.

DISCLAIMER

No approval or endorsement of any commercial product by the National Institute of Standards and Technology is intended or implied.

9. REFERENCES

[1] Mahesh, M., Ong, S.K., Nee, A.Y.C., Fuh, J.Y.H., and Zhang, Y.F. 2007. Towards A Generic Distributed and Collaborative Digital Manufacturing. Robotics and Computer Integrated Manufacturing, International Journal of Manufacturing and Product and Process Development. 23, 3, (June 2007), 267-27. DOI=http://dx.doi.org/10.1016/j.rcim.2006.02.008

[2] Sriram, R.D. 2002. Distributed and Integrated Collaborative Engineering Design. Sarvan Publishers, Maryland.

[3] Li, M.S., Jones, A. 2008. An information infrastructure for next generation Global enterprises, International Multi-Conference on Engineering and Technological Innovation: IMETI 2008. Orlando. Florida. USA.

[4] Comparisons and Analyses of U.S. and Global Economic Data and Trends: Globalization: Trends and Impact Factors of Globalization of Manufacturing Input Factors on Future U.S. Manufacturing Capability. National Institute of Standards and Technology. U.S. Department of Commerce 2004. http://www.sri.com/policy/csted/reports/nist/NIST_SUBD_Globalization.pdf

3D Reconstruction of Rough Terrain for USARSim Using a Height-map Method

G. Roberts
Intelligent Systems Division, National Institute of Standards & Technology, 100 Bureau Drive, MS 8230, Gaithersburg, MD 20899 USA.
1-301-975-3434,
gael.roberts@nist.gov

S. Balakirsky
Intelligent Systems Division, National Institute of Standards & Technology, 100 Bureau Drive, MS 8230, Gaithersburg, MD 20899 USA.
1-301-975-4791,
stephen.balakirsky@nist.gov

S. Foufou
Laboratoire Electronique, Informatique et Image (LE2i Lab). UMR CNRS 5158, University of Burgundy, 21000, Dijon, FRANCE.
00-333-8039-3805,
sfoufou@u-bourgogne.fr

ABSTRACT

In this paper, a process for a simplified reconstruction of rough terrains from point clouds acquired using laser scanners is presented. The main idea of this work is to build height-maps which are level gray-scale images representing the ground elevation. These height-maps are generated from step-fields which can be represented by a set of side-by-side pillars. Although height-maps are a practical means for rough terrain reconstruction, it is not possible to represent two different elevations for a given location with one height-map. This is an important drawback as terrain point clouds can show different zones representing surfaces above other surfaces.

In this paper, a methodology to create several height-maps for the same terrain is described. Experimental results are shown using the high-fidelity physics-based framework for the Unified System for Automation and Robot Simulation (USARSim).

Keywords

3D Reconstruction, height-map, step-field, point cloud, USARSim.

1. INTRODUCTION

The usefulness of simulation systems for developing agent control systems is well established. The role of the simulation is to provide convincing sensor measurements as input to the control system and to accurately model the system's response to actuator outputs (physics, dynamics, and statics). An important enabler that allows the simulation system to meet these requirements is the existence of an accurate model of the real world.

One aspect of this world model is its representation of the ground surface, or a height-map. The required complexity of the ground surface representation is directly related to the agent's application and target environment. For example, an agent that is required to operate in a flat-floored environment with no overhanging obstacles may only require occupancy information in its model. More complex environments that include uneven terrain may require a 2.5D model (a single elevation value per unit area). Even more complex environments that include overhanging obstacles or multi-level terrain may require a full 3D model (multiple elevation values per unit area).

This paper addresses the automatic creation of height-maps from data collected with laser scanners. The overall objective is to be able to recreate complex 3D terrains such as those that may be found at disaster sites (partially destroyed buildings, large amounts of debris, etc.) for use in simulation and in reconstructed physical representations. While the general techniques developed in this paper should be applicable to the general problem, the specific output formats have been tailored for the Unified System for Automation and Robotics Simulation (USARSim) framework [1].

USARSim was initially developed in 2002 as a low cost (under $40) robotics simulator at Carnegie Mellon University (CMU) and the University of Pittsburgh [2]. In 2005, USARSim management was handed off to the National Institute of Standards & Technology (NIST) and it became an open source project hosted on the Sourceforge network (www.sourceforge.net/projects/usarsim). USARSim is based on the UnReal Game Engine from Epic Games[1] and provides a high-fidelity, physics based simulation environment for robotic development. One of the unique features of USARSim lies in its validated sensor and robot models [3]. In addition to its use as a research tool, USARSim is also used as the basis for the RoboCup Rescue Virtual Competition as well as the

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

[1] Certain commercial software and tools are identified in this paper in order to explain our research. Such identification does not imply recommendation or endorsement by the authors, nor does it imply that the software tools identified are necessarily the best available for the purpose.

IEEE Virtual Automation and Manufacturing Competition. More information on USARSim may be found in [4].

The remainder of this paper is laid out as follows: Section 2 recalls some preliminary definitions regarding the dataset, step-fields, and height-maps. The relation between our work and other works is explained in Section 3. Section 4 gives the general approach adapted in this work. Section 5 details the proposed algorithm for terrain reconstruction using height-maps. Some experimental results are shown and discussed in Section 6. Section 7 concludes and discusses about future extensions of this work.

2. PREREQUISITE

This section indicates the type of data used in this project and gives some fundamental definitions.

The purpose of this work is to reconstruct models of rough terrains for use in a simulator. Laser based scanners are used to acquire 3D numerical data (point clouds) representing the ground. Point clouds may be complex, containing ground, surfaces, and heaps of objects. Point clouds are muddled, contain no texture, and their points are not uniformly distributed in space.

In this work, the complexity of point clouds representing real-world scenes is intentionally avoided by only considering point clouds generated artificially. These point clouds are simpler but keep approximately the same characteristics (muddled, not textured, not uniform) as those acquired from real-world scenes. These artificial point clouds are generated as follows: a scene is created with different simple objects (planes, spheres, cylinders …) and points are randomly put onto the surface of each object using their equation. At the end, noise is added to make the clouds more realistic.

Two important models are calculated for each point cloud: a step-field model and a height-map model. A step-field can be represented by a set of side-by-side pillars (see Figure 1) and is computed as follows: starting from a horizontal plane, side by side pillars are placed with predefined dimension (height * width * depth). These pillars will all have identical width and depth, but their height will vary according to the configuration and properties of the point cloud. More details regarding the construction of these pillars are given later.

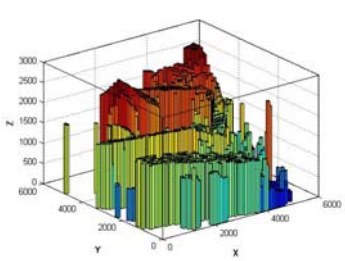


Fig. 1: Example of step-field

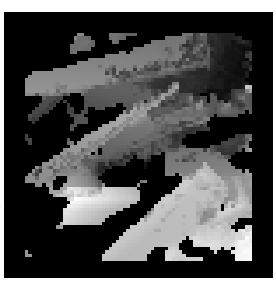


Fig. 2: Example of height-map

The height-map model, also called CEM (Cartesian Elevation Map) or elevation map, is a gray scale picture (based on 256 levels of gray) which represents the height information of the studied terrain (see Figure 2). The lighter the color of a pixel is, the higher the elevation of the terrain is for the location represented by this pixel.

3. RELATED WORK

Since the beginning of mobile robot development, researchers have been striving to find ways for the platforms to navigate through various terrains. Work has even been performed on classifying the difficulty of terrain for robotic traversal. The general idea of our work is to find an adequate representation of rough terrains that will allow simulations of robots to be used to create and evaluate algorithms that will allow mobile robots to be more or less autonomous on rough terrains.

Because robots come in many different shapes, sizes and abilities, simulation is a useful tool that allows one to predict which types of robots will be best-suited for particular types of terrain under consideration. Simulation is also an efficient way to test the impact of algorithm changes on the effectiveness of a given platform. There are two main types of ground mobility: walking robots [5, 6] and rolling robots (with tracks or wheels). Despite a large effort in the area of walking robots, they are not completely adequate for the ground studied for this paper. The reason for this is the rocky and uneven nature of a disaster site. Consequently, rolling robots which are also the subject of studies as regards their efficiency for rough terrains are used [7, 8, 9].

The terrains are created from a cloud of points. These reconstructions are an important area of computer graphics and there are many methods of 3D reconstruction from point clouds. Among these methods, which have the objective of creating a mesh, there are the Delaunay triangulation (which is joined to the Voronoi diagram) [10, 11, 12], related methods such as Ball-Pivoting (pivot of a sphere) [13] or other methods such as the one from Chang et al. [14]. All these methods give very good constructions, but the results are often very costly in memory.

To resolve this problem, other research aims at simplifying the obtained meshing. These simplifications consist in reducing the number of polygons in a meshing by grouping polygons according to certain criteria such as the orientation of polygons or their configurations [15, 16, 17]. Although these mesh simplifications give reasonable results, the processes to do this are very costly in memory compared to the creation of height-maps as discussed in the next paragraph.

Other methods for creating ground representations use height-maps. The concept of height-map is used for several studies including the crossing possibilities of grounds or the construction of obstacles on a ground [18, 19]. Height-maps are useful for path planning and collision avoidance [20], they are also used to represent surfaces having a low-level of detail such as landscapes (e.g., video games).

4. THE APPROACH TO THE PROBLEM

The purpose of this work is to create a 3D terrain model that will be used in a simulator, consequently the simplest possible construction is necessary in order to reduce the computation costs during simulation. Therefore, in order to have a good simulation, it is very important to have a very simple scene because the Frame Per Second (FPS) is taken into consideration. For a complex scene with big meshed objects, the FPS will be too low for effective rendering and the simulation will certainly be too slow.

Our simulator utilizes the Unreal 3D engine, thus the editor of this engine is used to construct the 3D models either by importing them as meshed scenes or by generating a height-map from point clouds. However, it is only considering the use of height-maps to avoid having bad FPS during the simulation.

5. ALGORITHM DESCRIPTION

The algorithm operates in two steps: (i) compute the step-field from the point cloud, (ii) and then compute the height-map. Several height-maps may be created for the same scene in order to have various layers representing surfaces (or objects) because these surfaces can be positioned one on the top of the other.

5.1. Step-fields elaboration

Having acquired the point cloud, a virtual horizontal plane P(x,y) is created:

$$P(x,y) = Z_{min} - \alpha. \quad (1)$$

Where Z_{min} represents the coordinates of the lowest point of the cloud on the Z axis, and α represents a small distance defined by the user. Therefore, the plane will be "α units" below the point cloud. This plane is bounded by the cloud on the X and Y axes.

Then a grid on this plane is created. Each compartment of this grid represents the possible base of a future pillar. The side sizes of each compartment will be equal to the dimension of the pillar sides. The size of a compartment is given by the user. Thus, the dimensions of the grid (length * width compartments) are a function of the plan size and the compartments size.

A pillar is created in each compartment situated under a non-empty subset of points. The height of this pillar is the average of the heights (Z-components) of the points of the subset. The obtained set of pillars defines the step-field of the scene (terrain), and will be used to construct the height-map.

5.2. Height-map construction

The height-map is built from the step-field. To this end, each pillar created is correlated with a pixel of a gray-scale image. If a pillar has a width of 10 cm, a pixel of the image will also represent a zone of 10 cm in reality. One should notice that if there was beforehand a grid of 64*64 compartments on the initial plane, there will be a height-map of 64*64 pixels.

Height-map being an image of 256 levels of gray, the maximal height (H_{max}) of the pillars is put in correspondence to the highest level of the scales of gray (that is 256). Other heights (h) of pillars will be represented by a proportional gray level g, such that:

$$g = h * 256 / H_{max}. \quad (2)$$

We note that the simulator constrains the height-map to have a number of pixels equal to a power of 2 for each of its sides. For that purpose, padding is added around the height-map obtained previously. This padding will have a value equal to 0 on the scale of the gray-levels. (see Figure 2).

5.3. Construction of different layers

A layer can be assimilated to one height-map or to one surface in the 3D reconstruction.

5.3.1. Why different layers?

The purpose of creating different layers is to rebuild surfaces which are above other surfaces. Effectively, in a point cloud, several objects heaped upon each other can be represented. A perspective scan of a table is a perfect simple example of this situation, because in the associated point cloud, there is the surface of the ground and the top surface of the table.

A height-map is a gray scale picture and with only one gray color. It is not possible to represent different elevations. Thus, different surfaces (or elevations) cannot be represented for a given location with only one height-map.

5.3.2. The process

With one step-field, one height-map is constructed, and with one height-map, one surface is constructed for the simulator. So, in order to have different surfaces in the final 3D model, it is necessary to construct different step-fields for the same scene.

As presented in Section 5.1, a plane is placed below the point cloud and it is subdivided into a grid. Each compartment of the grid is potentially the base of several pillars. Since the height of a pillar is equal to the average of the Z-coordinates of the points above it, to construct different pillars at the same location the points above the compartment are portioned according to their elevations into different subsets. Then, the height of each pillar will be equal to the average of the Z-coordinate of the points of one particular subset. Grouping the obtained pillars into different sets gives different step-fields. The pillars of the same compartment cannot be placed in the same set (every step-field is composed of pillars from different compartments).

Consequently, if there are X sets of points, there will be X pillars for the same location. The number of pillars can be different from one compartment to another. The number of layers equals the maximum number of pillars created on the grid for one compartment.

5.3.3. Classification of pillars

When pillars are created for one compartment of the grid, the first created pillar is placed in the first layer, the second pillar in the second layer and so on. This placement of this pillar leads to layers containing pillars not according to their heights, but according to their creation rank. A pillar which should be in step-field number N can be in step-field number N-1. As a consequence, the built surfaces will not be regular and a pillar replacement process is needed to classify each pillar in the adequate layer.

To classify the pillars, the average elevation of each layer is computed. Then, pillars having heights close to the average of a particular layer are moved to this layer. This process is repeated several times until stability is reached.

6. EXPERIMENTAL RESULTS

Some experimental results are presented and discussed in this section,. Figures 3, 4, 5, and 6 show different reconstruction cases. Figures 3.a, 4.a, 5.a, and 6.a show the point clouds and Figures 3.b, 4.b, 5.b, and 6.b show the associated computed reconstruction. Textures that appear on these reconstructions represent the layers found on the reconstructed surfaces.

In the first example (see Figure 3.a and Figure 3.b), there are 3 horizontal overlaid planes. A point in a plane does not have the same elevation as a point in another plane. By this fact, at the creation of the step-field, it is easy to classify the pillars according to their elevations for each layer, and one may notice remark that the associated height-maps are very homogeneous (there is one height-map for each surface). Of course this is the perfect case where the reconstructed layers (Figure 3.b) correspond to the surfaces of the initial planes.

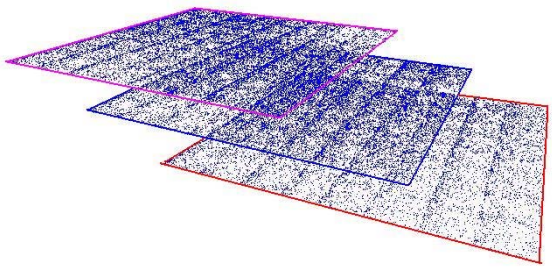


Fig. 3.a: Point cloud representing 3 planes

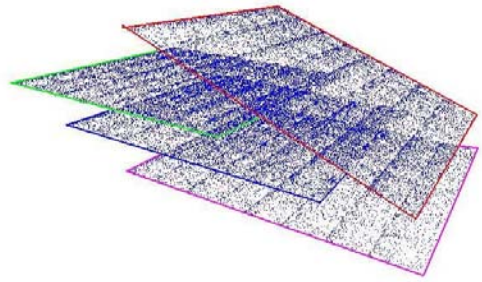


Fig. 4.a: Point cloud representing 4 planes

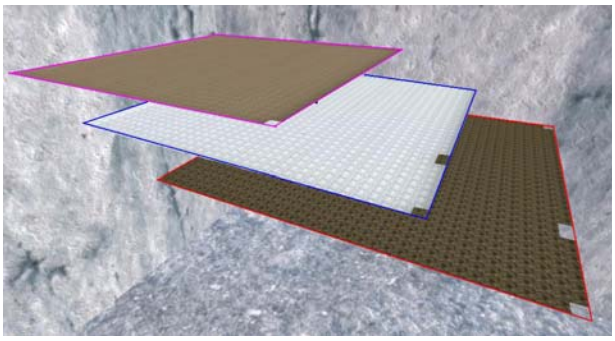


Fig. 3.b: Construction representing 3 planes

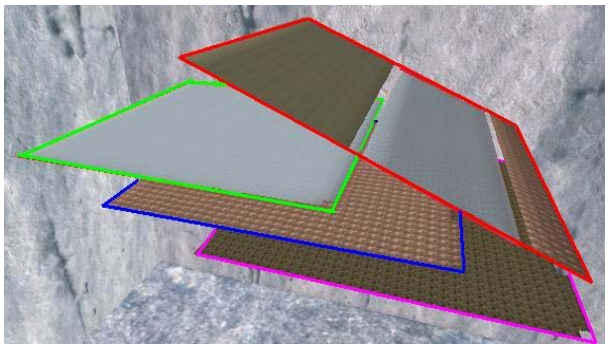


Fig. 4.b: Construction representing 4 planes

The second example (Figure 4.a) represents a scene that is a little more complicated. There are the three planes from Figure 3.a above which a fourth inclined plane is positioned. This fourth plane crosses the elevation of the two highest horizontal planes. Figure 4.b displays the associated construction, where one can see that the three parallel planes are constructed correctly, but the inclined plane presents some discrepancies. This inclined plane is constructed with three different layers because the pillars are classified according to their elevations. Since the inclined plane crosses the elevations of the two other planes, the pillars for the inclined plane are classified in different layers. This classification may lead the pillars of the inclined plane to be positioned in the same layer as the pillars of the other planes with approximately the same elevation.

Notice that the reconstruction of the inclined surface is not continuous. That is explained by the fact that the inclined plane is reconstructed with three layers, and although no data are lost, the continuity of the surface at this location is lost, which creates a discontinuity in the 3D model. The 3D model is used to manipulate a robot in a simulator, and the created holes have the size of the pillars (a small size). Thus the robots can cross over these holes and there is no real impact for the simulation.

The third example (see Figure 5) is more complicated than the second example where the inclined plane does not only cross the elevation of others surfaces, but also crosses one of these planes. The same problem as previously is noticed (the holes because of the discontinuity) and the construction is more complicated too. Actually, even the horizontal planes are built with several layers but the 3D reconstruction is still correct.

The fourth example (see Figure 6) presents a more complex scene where a cylindrical object is positioned on an horizontal plane. The relevance of this scene is to show how one object (the cylinder) which presents a surface above itself is reconstructed. The algorithm cuts the object surface into two parts (see Figure 6.b) in order to create two different layers for the same object.

During the process of creating the ground, there are edge effects. It is during the process of creation of a 3D scene from height-maps that there is a loss of information. This is due to the fact that a pixel of the height-map represents a vertex and not a compartment of the surface in the simulator. This fact has the effect of cutting down the side of the surfaces. Consequently, if there is a ground (or a height-map) which is strewed with holes, these holes will be increased in the final reconstruction, and if the surface is too thin, it will totally disappear.

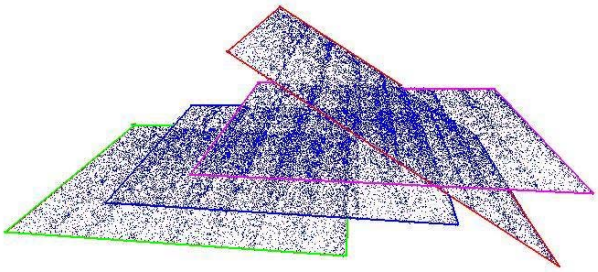


Fig. 5.a: Point cloud representing 4 planes with intersection

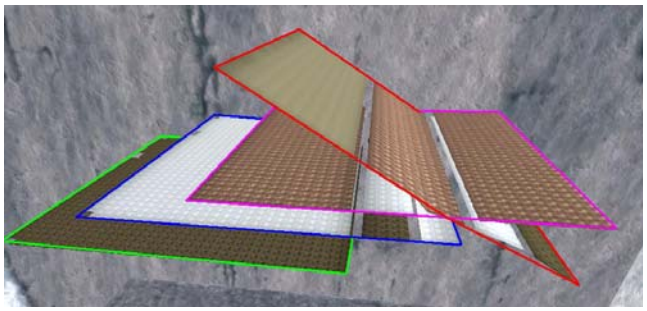


Fig. 5.b: Construction representing 4 planes with intersection

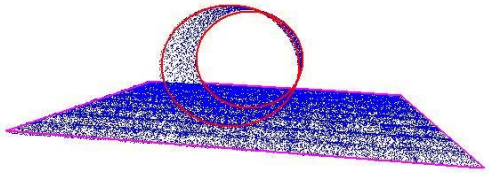


Fig. 6.a: Point cloud representing a cylinder onto a plane

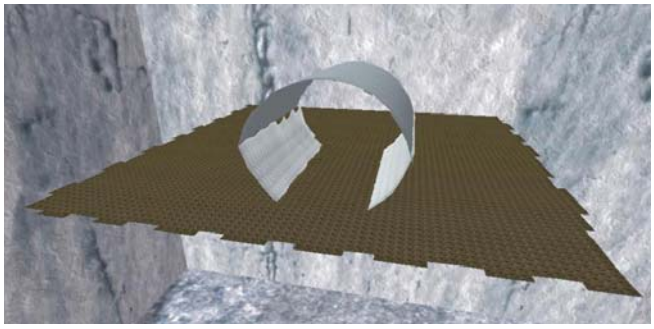


Fig. 6.b: Construction representing a cylinder onto a plane

7. CONCLUSION

We have presented a method to build a simplified 3D scene from point clouds using height-maps. The use of several height-maps allows us to reconstruct in a very simplistic way different scenes, which could have had a heavy and high cost mesh based reconstruction. Furthermore, the elaboration of overlapped height-maps representing grounds to be analyzed permits one to simplify the reconstruction of overlapped surfaces. This new reconstruction method gives satisfactory results for relatively simple scenes. However, height-maps based reconstruction methods are still limited for ground representations, which become complex by the presence of several kinds of objects. The height-maps method works very well for the generation of grounds having few discontinuities, but fails with complex grounds with many discontinuities or with important variations. A future work is to validate this method in two different ways. The first is to create a terrain as similar as possible with the reality. The second way is to compare the similarity between the behaviors of a robot on the terrain in the simulation and in the real world. Namely, the second way is more important than the first one.

The objective is to reconstruct real rough terrains for a simulation, thus we can have very complex scenes. In these scenes, we can differentiate what we call the simple parts of the scene and the complex parts of the scene. To differentiate these two sorts of parts, the local density of points and the shape of the representation of the points are taken in consideration. In this way, a smooth shape will be assimilated to a simple part, and a small area with much variation will be assimilated to a complex part. In order to improve our algorithm, we envisage combining the overlapped height-maps method (for the simple parts of the ground) with a simple mesh-based method (for the complex parts of the ground). Actually, we can also define the boundary between a simple and a complex part with the effectiveness of local reconstructions using the overlapping height-maps method.

8. REFERENCES

[1] Carpin, S.; Lewis, M.; Wang, J.; Balarkirsky, S.; Scrapper, C.; USARSim: a robot simulator for research and education. Proceedings of the IEEE 2007 International Conference on Robotics and Automation, Page(s):1400-1405

[2] Balaguer, B.; Balakirsky, S.; Carpin, S.; Lewis, M.; Scrapper, C.; "USARSim: A Validated Simulator for Research in Robotics and Automation", Workshop on "Robot Simulators: Available Software, Scientific Applications, and Future Trends" at IEEE/RSJ 2008.

[3] Carpin, S.; Stoyanov, T.; Nevatia Y.; Lewis, M.; Wang, J.; Quantitative Assessments of USARSim Accuracy. Proceedings of PerMIS 2006, NIST Special Publication 1062, August 2006.

[4] Usarsim homepage: http://sourceforge.net/projects/usarsim (Accessed: 2008)

[5] Pongas, D.; Mistry, M.; Schaal, S.; "A Robust Quadruped Walking Gait for Traversing Rough Terrain", Robotics and Automation 10-14 April 2007 Page(s):1474 – 1479.

[6] Rebula, J.R.; Neuhaus, P.D.; Bonnlander, B.V.; Johnson, M.J.; Pratt, J.E.; "A Controller for the LittleDog Quadruped Walking on Rough Terrain", Robotics and Automation, 2007 IEEE International Conference on 10-14 April 2007 Page(s):1467 – 1473.

[7] Fukuda, T.; Zhang, X.; Hasegawa, Y.; Matsuno, T.; Hoshino, H.; "Preview posture control and impact load control of rough terrain vehicle with interconnected suspension", Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004

IEEE/RSJ International Conference on Volume 1, 28 Sept.-2 Oct. 2004 Page(s):761 - 766 vol.1,
[8] Udengaard, M.; Iagnemma, K.; "Kinematic analysis and control of an omnidirectional mobile robot in rough terrain", Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on Oct. 29 2007-Nov. 2 2007 Page(s):795 – 800.
[9] Nakamura, S.; Faragalli, M.; Mizukami, N.; Nakatani, I.; Kunii, Y.; Kubota, T.; "Wheeled robot with movable center of mass for traversing over rough terrain", Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on Oct. 29 2007-Nov. 2 2007 Page(s):1228 – 1233.
[10] Ledoux, H.; "Computing the 3D Voronoi Diagram Robustly: An Easy Explanation", Voronoi Diagrams in Science and Engineering, 2007. ISVD '07. 4th International Symposium on 9-11 July 2007 Page(s):117 - 129.
[11] Tse, R.; Gold, C.; Kidner, D.; "Using the Delaunay Triangulation/ Voronoi Diagram to extract Building Information from Raw LIDAR Data", Voronoi Diagrams in Science and Engineering, 2007. ISVD '07. 4th International Symposium on 9-11 July 2007 Page(s):222 – 229.
[12] Labatut, P.; Pons, J.-P.; Keriven, R.; "Efficient Multi-View Reconstruction of Large-Scale Scenes using Interest Points, Delaunay Triangulation and Graph Cuts", Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on 14-21 Oct. 2007 Page(s):1 – 8.
[13] Bernardini, F.; Mittleman, J.; Rushmeier, H.; Silva, C.; Taubin, G.; "The ball-pivoting algorithm for surface reconstruction", Visualization and Computer Graphics, IEEE Transactions on Volume 5, Issue 4, Oct.-Dec. 1999 Page(s):349 – 359.
[14] Ming-Ching, C.; Leymarie, F.F.; Kimia, B.B.; "Surface Reconstruction from Point Clouds by Transforming the Medial Scaffold", 3-D Digital Imaging and Modeling, 2007. 3DIM '07. Sixth International Conference on 21-23 Aug. 2007 Page(s):13 – 20.
[15] Hua-Hong, C.; Xiao-nan, L.; Ruo-tian, L.; "Surface Simplification Using multi-edge mesh collapse", Image and Graphics, 2007. ICIG 2007. Fourth International Conference on 22-24 Aug. 2007 Page(s):954 - 959.
[16] Qu, L.; Meyer, G. W.; "Perceptually Guided Polygon Reduction", Visualization and Computer Graphics, IEEE Transactions on Volume 14, Issue 5, Sept.-Oct. 2008 Page(s):1015 – 1029.
[17] In Kyu, P.; Sang Wook, L.; Sang Uk, L.; "Shape-adaptive 3D mesh simplification based on local optimality measurement", Computer Graphics and Applications, 2002. Proceedings. 10th Pacific Conference on 9-11 Oct. 2002 Page(s):462 – 466.
[18] Asada, M.; "Building a 3D world model for mobile robot from sensory data", Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on 24-29 April 1988 Page(s):918 – 923, vol.2.
[19] Asada, M.; "Map building for a mobile robot from sensory data", Systems, Man and Cybernetics, IEEE Transactions on Volume 20, Issue 6, Nov.-Dec. 1990 Page(s):1326 – 1336.
[20] Gutmann, J.S.; Fukuchi, M.; Fujita, M.; "A Floor and Obstacle Height Map for 3D Navigation of a Humanoid Robot", Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on 18-22 April 2005 Page(s):1066 – 1071.

Using Virtual Scans to Improve Alignment Performance in Robot Mapping

Rolf Lakaemper
Temple University
Philadelphia, PA, USA
lakamper@temple.edu

Nagesh Adluru
Temple University
Philadelphia, PA, USA
nagesh@temple.edu

ABSTRACT

We present a concept and implementation of a system to integrate low level and mid level spatial cognition processes for an application in robot mapping. Feedback between the two processes helps to improve performance of the recognition task, in our example the alignment of laser scans. The low level laser range scan data ('real scans'), are analyzed with respect to mid level geometric structures. The analysis leads to generation of hypotheses (Virtual Scans) about existing real world objects. These hypotheses are used to augment the real scan data. The core mapping process, called Force Field Simulation, iteratively aligns the augmented data set which then in turn is re analyzed to confirm, modify, or discard the hypotheses in each iteration. Experiments with scan data from a Rescue Robot Scenario show the applicability and advantages of the approach.

1. INTRODUCTION

This article demonstrates how the integration of low level spatial cognition processes (LLSC) and mid level spatial cognition processes (MLSC) can help to improve the performance of spatial recognition tasks in robotics. We are using the area of robot perception for mobile rescue robots, specifically alignment of 2D laser scans, as a showcase to demonstrate the advantages of the integration of LLSC and MLSC processes. Alignment of data, e.g. acquired from multiple robots from different positions, is a required task for environment mapping. The basic task of mapping is to combine spatial data usually gained from laser range devices, called 'scans', to a single data set, the 'global map'. The global map represents the environment as scanned from different locations, even possibly scanned by different robots ('multi robot mapping'), usually without knowledge of their pose (= position and heading).

We apply our LLSC/MLSC system to map an environment from Urban Search and Rescue Robots. Mapping is a requirement for these robots to perform their main task: to autonomously detect victims in disaster areas (e.g. tsunami, earthquake or terrorist attack). The goal is to assist the rescue teams generating location and best access path. Typically multiple robots are deployed in the area to cooperatively map the disaster area. Currently their applicability is limited due to the extreme constraints to robot mapping in such an environment: single scans have little overlap, no additional sensor information, e.g. odometry, is reliably available, and none or just a few landmark features are present.

In robot cognition, MLSC processes infer the presence of mid level features from low level data based on regional properties of the data. In our case, we detect the presence of simple mid level objects, i.e. line segments and rectangles. The MLSC processes model world knowledge, or assumptions about the environment. In our case, we assume the presence of (collapsed) walls and other man made structures. If possible wall-like elements or elements somewhat resembling rectangular structures are detected, our system generates the most likely ideal model as a hypothesis, called 'Virtual Scan'. Virtual Scans are generated from the ideal, expected model in the same data format as the raw sensor data, hence Virtual Scans are added to the original scan data indistinguishably for the low level alignment process; the alignment is therefore performed on an augmented data set.

In robot cognition, LLSC processes usually describe feature extraction based on local properties like spatial proximity, e.g. based on metric inferences on data points, like edges in images or laser reflection points. In our system laser scans (virtual or real) are aligned to a global map using mainly features of local proximity using the LLSC core process of 'Force Field Simulation' (FFS). FFS was recently introduced to robotics [20]. In FFS, each data point can be assigned a weight, or value of certainty. It also does not make a hard, but soft decision about the data correspondences as a basis for the alignment. This is achieved by computation of a correspondence probability to multiple neighboring points, based on weight, distance and direction of underlying linear structures. Mainly this feature makes FFS a natural choice over its main competitor, ICP [2, 24], for the combination with Virtual Scans (however, the general idea of Virtual Scans is applicable to both approaches). The weight parameter can be utilized to indicate the strength of hypotheses, represented by the weight of virtual data.

FFS is an iterative alignment algorithm. The two levels (LLSC: data alignment by FFS, MLSC: data augmentation) are connected by a feedback structure, which is repeated in each iteration:

- The FFS-low-level-instances pre-process the data. They

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for prof t or commercial advantage and that copies bear this notice and the full citation on the f rst page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specif c permission and/or a fee.
*PerMIS'*08 August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08...$5.00.

find correspondences based on low level features. The low level processing builds a current version of the global map, which assists the mid-level feature detection

- The mid level cognition module analyzes the current global map, detects possible mid level objects and models ideal hypothetical sources possibly being present in the real world. These can be seen as suggestions, fed back into the low level system by Virtual Scans. The low level system in turn adjusts its processing for re-evaluation by the mid level systems.

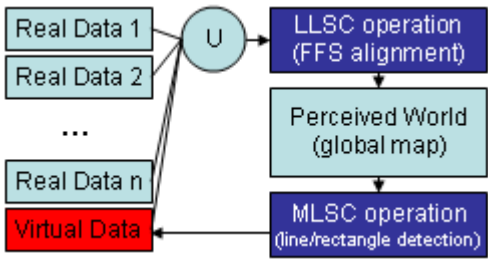


Figure 1: LLSC/MLSC feedback. The LLSC module works on the union of real scans and the Virtual Scan. The MLSC module in turn re-creates a new Virtual Scan based on the result of the LLSC module.

In such a system, MLSC processes steer LLSC processes introducing higher knowledge to enable spatial inferences the LLSC system is not able to draw by itself. However, the MLSC system also needs assistance of the LLSC for two reasons: MLSC systems concentrate on higher information which needs LLSC pre-processed data (e.g. a set of collinear points is passed to the MLSC as a single line segment). But also LLSC processes have to support the suggestions stated by the MLSC. Since MLSC introduces higher knowledge, it is dangerous to focus on spatial mid level inferences too early. Feedback with the LLSC system enables more careful evaluation of plausibility (see also final note in section 'Experiments'). The following example will illustrate the

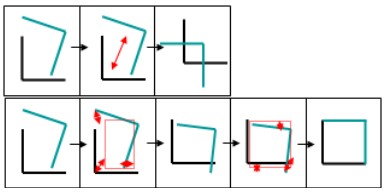


Figure 2: Alignment using Virtual Scans (VS). Top row: Alignment without VS. The LLSC misalignes the scene due to lack of global information. Bottom row: two iterations using VS. The hypothetical rectangle createa a the VS add global information, leading to correct alignment. The VS are newly created in each iteration.

approach: Figure 2, top left, assumes 2 scans, e.g. taken from robots in 2 different positions. A typical LLSC alignment process like ICP or FFS tries to find correspondences between single points in the scans, and to superimpose the scans in an optimal way, e.g. minimizing the summed distances between corresponding points. These approaches assume the existence of overlapping features in scans. Like in this example, such a constraint is often not met, and the assumption leads to wrong results. The LLSC alignment process works on a local basis, hence it is not able to detect a possible rectangular feature. Figure 2 illustrates the result of an alignment process with (bottom row) and without (top row) Virtual Scans. In the illustration of our approach in fig.2, bottom, an MLSC process detects a rectangular structure and adds an optimal model to the data set. The LLSC module aligns the augmented data. The hypothesis now directs the scans to better location. In each iteration, the relocated real scans are analyzed to adjust the MLSC hypothesis, LLSC and MLSC assist each other in a feedback loop.

2. RELATED WORK IN SPATIAL COGNITION AND ROBOT MAPPING

The potential of MLSC has been largely unexplored in robotics, since recent research mainly addressed LLSC systems. They show an astonishing performance: especially advances in statistical inferences [5, 10, 13] in connection with geometric modeling of human perception [6, 9, 25] and the usage of laser range scanners contributed to a breakthrough in robot applications, with the most spectacular results achieved in the 2005 DARPA Grand Challenge where several autonomous vehicles were able to successfully complete the race [26]. But although the work on sophisticated statistical and geometrical models like extended Kalman Filters (EKF),e.g. [12], Particle Filters [10] and ICP (Iterative Closest Point) [2, 24] utilized in mapping approaches show impressive results, their limits are clearly visible, e.g. in the aforementioned rescue scenarios. These systems are still based on low level cognitive features, since they construct metric maps using correspondences between sensor data points. However, having these well-engineered low level systems at hand, it is natural to connect them to MLSC processes to mutually assist each other.

The knowledge in the area of MLSC in humans, in particular in spatial intelligence and learning, is advancing rapidly [7, 14, 27]. Research in AI models such results to generate generic representations of space for mobile robots using both symbolic, e.g. [16], and non symbolic, e.g. [8], approaches. Each is trying to identify various aspects of the cognitive mapping process. Naturally, SLAM (Simultaneous Localization and Mapping [4] is often used as an application example [23]. In [28], a spatial cognition based map is generated based on High Level Objects. Representation of space is mostly based on the notion of a hierarchical representation of space. Kuipers [16] suggests a general framework for a Spatial Semantic Hierarchy (SSH), which organizes spatial knowledge representations into levels according to ontology from sensory to metrical information. SSH is an attempt to understand and conceptualize the cognitive map [15], the way we believe humans understand space. More recently, Yeap and Jefferies [29] trace the theories of early cognitive mapping. They classify representations as being space-based and object-based. Comparing to our framework, these classifications could be described being related to LLSC and High Level Spatial Cognition (HLSC), hence the supposed LLSC/MLSC system would relate closer to space-based sys-

tems.

In [1], the importance of 'Mental Imagery' in (Spatial) Cognition is emphasized and basic requirements of modeling are stated. Mental Images invent or recreate experiences resemble actually perceived events or objects. This is closely related to the "Virtual Scans" described in this proposal. Recently, Chang et al. [3] presented a predictive mapping approach (P-SLAM), which analyzes the environment for repetitive structures on the LLSC level (lines and corners) to generate a "virtual map". This map is either used as a hypothesis in unexplored regions to speed up the mapping process or as an initialization help for the utilized particle filters when a region is first explored. In the second case the approach has principles similar to the presented Virtual Scans. The impressive results of P-SLAM can also be seen as proof of concept of integrating prediction into robot perception.

The problem of geometric robot mapping is based on aligning a set of scans. On the LLSC level the problem of simultaneous aligning of scans has been treated as estimating sets of poses [22]. The underlying framework for such a technique is to optimize a constraint-graph, in which nodes are features, poses and edges are constraints built using various observations and measurements.

There are numerous image registration techniques, the most famous being Iterative Closest Point (ICP)[2], and its numerous variants to improve speed and converge basins. Basically all these techniques do search in transformation space trying to find the set of pair-wise transformations of scans by optimizing some function defined on transformation space. The techniques vary in defining the optimization functions that range from being error metrics like "sum of least square distances" to quality metrics like "image distance". 'Force Field Simulation' (FFS), [20], minimizes a potential derived from forces between corresponding data points. The Virtual Scan technique presented in this paper will interact with FFS as underlying alignment technique.

3. SCAN ALIGNMENT USING FORCE FIELD SIMULATION

The understanding of FFS is crucial to the understanding of the presented extension of the FFS alignment using Virtual Scans. We will give an overview here. FFS aligns single scans S_i obtained by robots, typically from different positions. We assume the scans to be roughly pre-aligned (see fig.7), e.g. by odometry or shape based pre-alignment. This is in accord with the performance comparison between FFS and ICP described in [19]. FFS alignment, in detail described in [20] is able to iteratively refine such an alignment based on the scan data only. In FFS, each single scan is seen as a non-deformable entity, a 'rigid body'. In each iteration, a translation and rotation is computed for each single scan simultaneously. This process minimizes a target function, the 'point potential', which is defined on the set of all data points (real and Virtual Scans: FFS can *not* distinguish). FFS solves the alignment problem as optimization problem utilizing a gradient descent approach motivated by simulation of dynamics of rigid bodies (the scans) in gravitational fields, but "*replaces laws of physics with constraints derived from human perception*" [20]. The gravitational field is based on a correspondence function between all pairs of data points, the 'force' function. FFS minimizes the overlaying potential function induced by the force and converges towards a local minimum of the potential, representing a locally optimal transformation of scans. The force function is designed in a manner that a low potential corresponds to a visually good appearance of the global map. As scans are moved according to the laws of motion of rigid bodies in a force field, single scans are not deformed.

Fig. 3 shows the basic principle: forces (red arrows) are computed between 4 single scans (the 4 corners). FFS simultaneously transforms all scans until a stable configuration is gained.

With S_1, S_2 being two different scans, the force between two single data points $v_i \in S_1$ and $u_j \in S_2$ is defined as a vector

$$V(v_1, v_2) = M(v_1, v_2)\frac{v_i - u_j}{\|v_i - u_j\|} \tag{1}$$

Its magnitude $\|V(v_i, u_j)\| = M(v_i, u_j)$ is defined as:

$$M(v_i, u_j) = \frac{1}{\sigma_t\sqrt{2\pi}} e^{\left(-\frac{\|v_i, u_j\|^2}{2\sigma_t^2}\right)} w_i w_j \cos(\angle(v_i, u_j)) \tag{2}$$

with parameters σ_t, w_i, w_j, $\angle(v_i, u_j)$ defined as follows: $\angle(v_i, u_j)$ denotes the angle between the directions of points, which is defined as the angle between directions of assumed underlying locally linear structures. See fig. 4, left, for an example, which especially shows the influence of the cosine-term in eq.2: forces are strong between parallel structures only. In eq.2, the forces are strongly depending on σ_t, which is a parameter steering the radius of influence. With σ_t decreasing during the iterative process, FFS changes the influence of each data point from global to local. In addition, the weight w_i, w_j (or mass) determines the influence of points v_i, u_j. The weight is a parameter which can e.g. express the certainty about a point, or it can model the feature importance. We utilize this feature of FFS to model the strength of hypothesis in the Virtual Scans. Hence in eq.2 the interfacing between LLSC and MLSC can be seen directly: distance and cosine term refer to LLSC, while the weights are derived from MLSC (in case of the Virtual Scans).

To compute the resulting movement from the forces of all point pairs between different scans, FFS re-assigns a constant mass to all data points and applies Newton's law of movement of rigid bodies in force fields. Constant mass causes data points participating in stronger force relations to influence the transformation stronger than those responding to weaker forces. For a single transformation step see fig. 4, right.

The step width Δ_t of FFS is determinded by a 'cooling process'. Δ_t It is monotonically decreasing, allowing the system in early iterations to jump out of local minima, yet to be attracted by local features in later steps. The interplay between σ_t and Δ_t is an important feature of FFS. See figures 7 and 8 for an example of the performance of FFS on a laser range data set.

FFS is closely related to simultaneous ICP. A performance evaluation of both algorithms [19] showed similar results. In general, FFS can be seen as more robust with respect to global convergence with non near optimal initialization, since the point relations are not built in a hard (nearest neighbor) but soft(sum of forces) way. Also the inclusion of weight parameters makes it a natural decision for our purposes of extension using Virtual Scans.

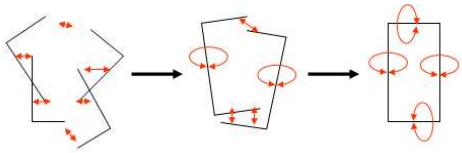


Figure 3: Basic principle of FFS. Forces are computed between 4 single scans. Red arrows illustrate the principle of forces. The scans are iteratively (here: two iterations) transformed by translation and rotation until a stable configuration is achieved.

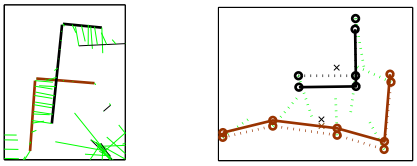


Figure 4: FFS example. Left: Forces (green) between two rigid structures (brown, black). The black and brown lines connect the actual data set for display reasons only. The figure shows a magnification of the upper left corner of fig. 7, right. Right: example of force and movement. Dotted lines show 2 scans (black, brown) and their forces (green) in iteration t. Solid lines show the resulting transformed scans at iteration $t+1$.

4. CREATING VIRTUAL SCANS: MID LEVEL ANALYSIS

The analysis module detects line segments and rectangles in each iteration of the FFS alignment. Both detection steps work on the entire point set of the current global map, i.e. the union $\bigcup_i S_i$ of all points of the real scans S_i.

4.1 Lines

The usage of lines for our Virtual Scan approach is motivated by the world knowledge assumption of scanning a man made environment (e.g. a collapsed house): although these environments often locally don't show major linear elements any longer, a global view still often reveals an underlying grid structure. In our line detection, we use the classic line detection approach of Hough transform [11], since it detects *globally* present linear structures. Hough transform does not only show location and direction of a line, but also the number of participating data points. We use this value to compute a certainty-of-presence measure, i.e. the strength of the line hypothesis. We only use lines above a certain threshold of certainty. We will specify below how the detected lines are utilized to create the Virtual Scan.

4.2 Rectangles

For rectangle detection, we use an approach described in [17]. This approach needs line segments as input, which model the underlying point data. Each line segment is translated into the 'S,L,D space' (Slope,Length,Distance), which simplifies the detection of appropriate (rectangular like) configurations of four near parallel and near perpendicular segments. To use this approach, the laser data has to be modeled by underlying line segments. Since line segments rely on local linearity of the underlying data points, global Hough detection is not feasible. A recently published technique [21] from robot mapping, using a statistical approach, 'Extended Expectation Maximization', is specifically tailored to model laser scan data with line segments. To speed up the rectangle detection process, we detect line segments in each *single* scan once before the FFS process. Hence, for rectangle detection, we superimpose the data represented the pre-computed line segments line segments according to the FFS transformation of the underlying scan, rather than data points. Due to yet imprecise single scan alignment during the early FFS process, single lines in the environment are frequently presented by clusters of line segments, rather than the required single line segments. We merge similar lines in a cluster to a single prototype using a line merge approach described in [18], see figure 5. The rectangle detection module then predicts location, dimension and certainty-of-presence of hypothetical, ideal rectangles present in the data set of line segments. The certainty, or strength of the hypothesis is derived from properties (segment length, perpendicularity) of participating rectangle-generating line segments. This value is used to create the weight of the ideal rectangle in the Virtual Scan.

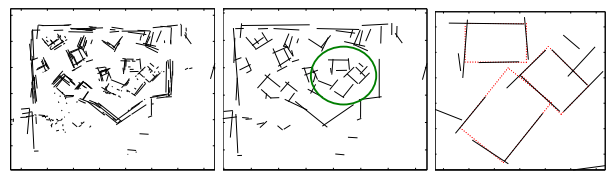


Figure 5: Rectangle detection. a) Global map built by line segments of all single scans. b) result of global line merging c) red: detected rectangles (magnification of area encircled in b)

4.3 Creating a Virtual Scan

A Virtual Scan is a set of virtual laser scan points, superimposed over the entire area of the global map. The detected line segments and rectangles are 'plotted' into the Virtual Scans, i.e. they are represented by point sets as if they would be detected by a laser scanner. We assume a virtual laser scanner that represents each line and rectangle by a set of points, sub sampled equidistantly according to the point density of the underlying point data in the original data set.

An important feature of the Virtual Scan is, that each point in the Virtual Scan is assigned a weight, being the strength of hypothesis of a virtual structure it represents. Utilizing this feature, we benefit from the weights steering the FFS alignment. As defined in eq.2, the weight w_i, w_j directly influences the alignment process; stronger points, i.e. points with higher value w_i, have a stronger attraction. Hence, a strong hypothesis translates into a locally strongly attractive structure. The hypothesis value reflects the belief into the hypothesis relative to the real data; all data points of the real data are assigned a 'normal' weight of 1.

5. ALIGNMENT USING VIRTUAL SCANS: ALGORITHM

The algorithm describes the interplay between LLSC (FFS) and the MLSC analysis. S_i, $i = 1..n$, denotes the real scan

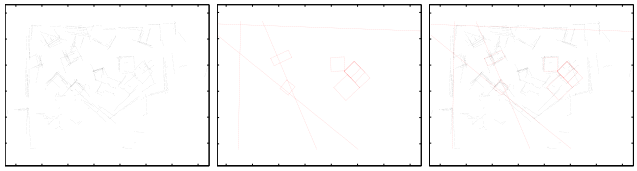


Figure 6: Virtual Scans in an early stage of FFS. a) global map b) the Virtual Scan consisting of points representing detected lines and rectangles c) superimposition of real data and Virtual Scan. This is the data used in the next FFS iteration.

data, consisting of n scans. $V^{[t]}$ is the Virtual Scan in iteration t. Init: $t = 1$, $V^{[0]} = \emptyset$, create set of line segments L_i for each scan S_i

---

1) Perform FFS on $\bigcup_{i=1..n} S_i \cup V^{[t-1]}$, resulting in transformations $T_i^{[t]}$ for each scan $S_{i=1..n}$

2) Form global map G of points and GL of line segments, superimposing the transformed scans and their line segment representation: $G = \bigcup_{i=1..n} T_i^{[t]}(S_i)$, $GL = \bigcup_{i=1..n} T_i^{[t]}(L_i)$

3) Detect set of lines $\mathcal{L}$ in G, set of rectangles $\mathcal{R}$ in GL

4) Create Virtual Scan $V^{[t]}$. $V^{[t]}$ contains scan points representing the elements of $\mathcal{L}$ and $\mathcal{R}$.

5) Compute parameters σ_t and Δ_t for the FFS process

6) Loop: goto 1, or end if FFS converged (stable global map).

---

6. EXPERIMENTS AND RESULTS

6.1 NIST Disaster Area

This data set consists of 60 single laser scans, taken from 15 different positions in 4 directions (N,W,S,E) with 20° overlap. It can be interpreted as a scene scanned by 15 robots, 4 scans each. No order of scans is given. The scans resemble the situation of an indoors disaster scenery, scanned by multiple robots. The scans have little overlap and no distinct landmarks. The initial global map was computed using a shape based approach described in [19]. See fig. 7 for example scans and the initial map. We used the initial global map for two different runs of FFS, once with Virtual Scans, once without. The experiment was performed to demonstrate the increase in alignment performance using Virtual Scans. The increase in performance was evaluated by visual inspection, since for this data set no ground truth data is available. Comparing the final global maps of both runs, the utilization of Virtual Scans leads to distinct improvement in overall appearance and mapping details, see fig.8. Overall, the map is more 'straight' (compare e.g. the top wall), since the detection of globally present linear structures (top and left wall in fig.8) adjusts all participating single segments to be collinear. These corrections advance into the entire structure. More objectively, the improvements can be seen in certain details, the most distinct encircled in fig.8, bottom right. There especially the rectangle in the center of the global map is an excellent example for a situation where correct alignment is not achievable with low level knowledge only. Only the suggested rectangle from the Virtual Scan (see fig.8, top left) can force the low level process to transform the scan correctly. Without the assumed rectangle the low level optimization process necessarily tried to superimpose 2 parallel sides of the rectangle to falsely appear as one (fig.8,bottom right).

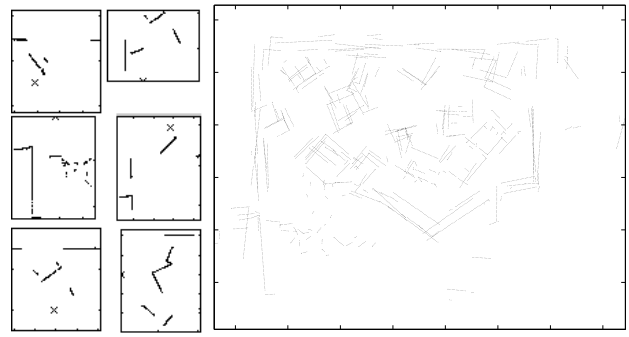


Figure 7: The NIST disaster area data set. Left: 6 example scans (from a total of 60). Crosses show each robot's position. Right: 60 scans superimposed using a rough pre-estimation, building the initial global map.

Comparison of fig.8, top left, and fig.6 shows the effect of feedback between the core FFS alignment process and the map analysis to create Virtual Scans. Figure 6 shows iteration 5 of the same experiment. Objects and object locations differ between the 2 Virtual Scans. Fig. 8 has discarded some hypotheses (objects) present in fig. 6, e.g. some of the rectangles. Other hypotheses are modified, e.g. the top wall is adjusted.

Note: The proposed system handles errors caused by premature belief in MLSC features by implementing the feedback principle, which evaluates a single MLSC-hypothesis. It is known that single hypothesis systems introducing higher knowledge tend to be not robust. Under certain circumstances this behavior could also be observed in experiments with our system. However, this single hypothesis system already significantly improves the alignment performance and therefore highlights the advantages of LLSC/MLSC connection. It can easily be embedded into a multiple hypotheses framework, e.g. particle filters, which will also be part of future work.

7. CONCLUSION AND OUTLOOK

The presented implementation of the alignment process FFS in feedback with MLSC recognition modules could significantly improve the results for the alignment task. The implementation proves the applicability of the presented concept for the combination of LLSC and MLSC processes. In this implementation, already the detection of simple elements (lines, rectangles) could improve the performance. These elements modeled assumptions of indoor disaster areas. Future implementations will work with different assumptions to adjust to outdoor disaster settings.

8. REFERENCES

[1] S. Bertel. Thomas barkowsky, dominik engel, christian freksa. computational modeling of reasoning with mental images: basic requirements. *D. Fum, F. del*

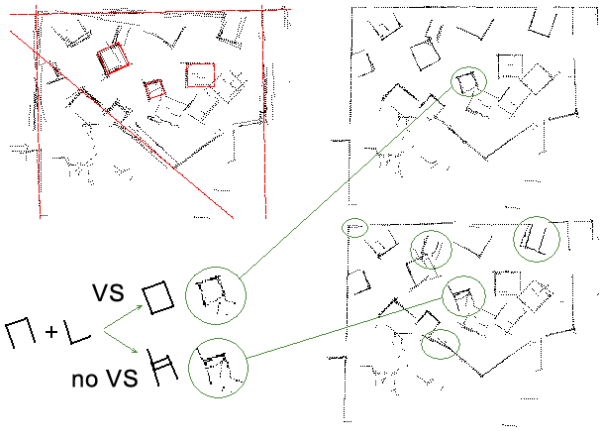


Figure 8: Alignment of NIST data set (initial alignment see fig. 6). Top left: after 10 iterations with detected line and rectangular objects plotted into the Virtual Scan (red). Top right: Final result using Virtual Scans, after 100 iterations. MLSC objects are not shown for clarity of display. Compare to Bottom Right: final result of alignment without Virtual Scans. Encircled areas show examples of improvement using Virtual Scans. Bottom left: The center rectangle could only be aligned correctly using MLSC information. Compare to motivational example fig. 2

Missier, A. Stocco (Eds.), Proceedings of the 7th International Conference on Cognitive Modeling ICCM06, 2006.

[2] P. Besl. N. mckay. a method for registration of 3.d shapes. *IEEE PAMI, 14(2)*, 1992.

[3] H. J. Chang. C.s.george lee and y.h. lu and y.c.hu. p-slam: Simultaneous localization and mapping with environmental structure prediction. *IEEE Transactions on Robotics, Vol.23, No.2*, April 2007.

[4] G. Dissanayake. H. durrant-whyte and t. bailey. a computationally efficient solution to the simultaneous localization and map building (slam) problem. *ICRA2000 Workshop on Mobile Robot Navigation and Mapping*, 2000.

[5] A. Doucet. N. defreitas, n. gordon. sequential monte carlo methods in practice. *Springer*, 2000.

[6] D. J. Field. A. hayes, r.f. hess. contour integration by the human visual system: evidence for a local association field. *Vision Research, 33*, pages 173–193, 1993.

[7] C. Freksa. M. knauff, b. krieg-brueckner, b. nebel, t. barkowsky. spatial cognition iv: Reasoning, action, interaction. *Springer*, 2004.

[8] T. Ghiselli-Crippa. Hirtle s.c., munro, p. connectionist models in spatial cognition. *The Construction of Cognitive Maps, Kluwer Academic Publishers. 1-51*, pages 87–104, 1996.

[9] W. Grimson. Object recognition by computer: The role of geometric constraints. *Boston, MA: MIT Press*, 1991.

[10] G. Grisetti. C. stachniss, burgard w. improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. *ICRA*, 2005.

[11] P. V. C. Hough. Methods and means for recognizing complex patterns. *US patent 3,069,654*, 1962.

[12] S. Huang. G. dissanayake. convergence analysis for extended kalman filter based slam. *IEEE International Conference on Robotics and Automation*, 2006.

[13] D. C. Knill. W. richards. perception as bayesian inference. *New York: Cambridge*, 1996.

[14] G. J. M. Kruijff. H. zender, p. jensfelt, h. i. christensen. situated dialogue and spatial organization: What, whereĚ and why? *Int. J. of Advanced Robotic Systems*, 2007.

[15] B. J. Kuipers. The cognitive map: 'could it have been any other way?'. *Spatial Orientation: Theory, Research, and Application. New York: Plenum Press*, pages 345–359, 1983.

[16] B. J. Kuipers. The spatial semantic hierarchy. *Artificial Intelligence 119*, pages 191–233, 2000.

[17] D. Lagunovsky. Ablameyko s. fast line and rectangle detection by clustering and grouping. *Proc. of CAIP'97, Kiel, Germany*, 1997.

[18] R. Lakaemper. L. j. latecki, d. wolter. geometric robot mapping. *Int. Conf. on Discrete Geometry for Computer Imagery (DGCI)*, 2005.

[19] R. Lakaemper. A.nuechter, n.adluru. performance of 6d lum and ffs slam. *Workshop on Performance Metrics and Intelligent Systems (PerMIS), Gaithersburg, MD*, August 2007.

[20] R. Lakaemper. N.adluru, l.j.latecki, r.madhavan. multi robot mapping using force field simulation. *Journal of Field Robotics, Special Issue on Quantitative Performance Evaluation of Robotic and Intelligent Systems*, 2007.

[21] L. J. Latecki. R. lakaemper. polygonal approximation of laser range data based on perceptual grouping and em. *IEEE Int. Conf. on Robotics and Automation (ICRA), Orlando, Florida*, May 2006.

[22] F. Lu. Milios, e. globally consistent range scan alignment for environment mapping. *Auton. Robots, 4(4)*, pages 333–349, 1997.

[23] A. Martinelli. A. tapus, k.o. arras, r. siegwart. multi-resolution slam for real world navigation. *Proceedings of the 11th International Symposium of Robotics Research, Siena, Italy*, 2003.

[24] A. Nuechter. K. lingemann, j. hertzberg, h. surmann, k. pervoelz, m. hennig, k. r. tiruchinapalli, r. worst, t. christaller. mapping of rescue environments with kurt3d. *Proceedings of the International Workshop on Safety, Security and Rescue Robotics (SSRR '05),Kobe, Japan*, June 2005.

[25] A. Pentland. Perceptual organization and the representation of natural form. *AI Journal, Vol. 28, No. 2*, pages 1–38, 1986.

[26] e. a. Thrun. Stanley: The robot that won the darpa grand challenge. *Journal of Field Robotics 23(9)*, pages 661–692, 2006.

[27] D. Uttal. Seeing the big picture: Map use and the development of spatial cognition. *Developmental Science, 3*, pages 247–286, 2000.

[28] S. Vasudevan. V. nguyen, r. siegwart. towards a cognitive probabilistic representation of space for mobile robots. *Proceedings of the IEEE International Conference on Information Acquisition (ICIA), Shandong, China*, pages 20–23, August 2006.

[29] W. K. Yeap. M.e. jefferies. on early cognitive mapping. *Spatial Cognition and Computation 2(2)*, pages 85–116, 2001.

The role of Bayesian bounds in comparing SLAM algorithms performance

Andrea Censi

Control & Dynamical Systems
California Institute of Technology

ABSTRACT

It is certainly hard to establish performance metrics for intelligent systems. Thankfully, no intelligence is needed to solve SLAM at all. Actually, when we cast the SLAM problem in the Bayesian framework, we already have a formula for the solution — SLAM research is essentially about finding good approximations to this computationally monstrous formula. Still, SLAM algorithms are difficult to analyze formally, partly because of such out-of-model ad hoc approximations. This paper explains the role of Bayesian bounds in the analysis of such algorithms, according to the principle that sometimes it is better to analyze the problem than the solutions. The theme is explored with particular regard to the problem of comparing algorithms using different representations and different prior information.

1. INTRODUCTION

In this paper I will often compare localization and SLAM. By "localization" I mean the problem of estimating the pose of the robot given a known, perfect map. Localization can be seen as a problem of the kind $\boldsymbol{y} = \mathbf{f}(\boldsymbol{x})$ where $\boldsymbol{x}$ is the pose (or the trajectory) of the robot, $\boldsymbol{y}$ are the sensor measurements, and a known function $\mathbf{f}$ encodes both the sensor model and the known map (which is "hidden" inside $\mathbf{f}$). For most sensors, noise can be modeled as an additive stochastic term. The other piece of information we have is some relation between successive poses, for example given by odometry, and usually described by a certain distribution $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$. SLAM, instead, is a problem of the kind $\boldsymbol{y} = \mathbf{f}(\boldsymbol{x}, \boldsymbol{m})$, where $\mathbf{f}$ is still thought as a known function, and $\boldsymbol{m}$, the map, is some parameterization for $\mathbf{f}$. This relation in the literature is usually written in the form $p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{m})$, but in this context it will be useful to highlight the functional relation between $\boldsymbol{y}$ and $\boldsymbol{x}, \boldsymbol{m}$.

There are many qualitative differences between localization and SLAM. On a superficial level, it is clear that "estimate $\boldsymbol{x}$ *and* $\boldsymbol{m}$" is much more difficult than "estimate $\boldsymbol{x}$". But there are subtler points. For one thing, while we know the physical meaning of $\boldsymbol{x}$, what exactly *is* the map $\boldsymbol{m}$?

One possible pragmatic answer is that the map is that which, together with the pose of the robot, allows to predict the observations, through a certain function $\mathbf{f}$. But this answer is not entirely satisfying: in this problem, it is literally true that "the map is not the territory"[1]; the design of a SLAM algorithm must deal up-front with the map-territory abstraction, which manifests itself as two main problems: the problem of representation, and the problem of the prior.

There is a "geometric" representation issue: unstructured maps are inherently infinite-dimensional, and yet the designer must chose a finite-dimensional approximated representation, such as occupancy grids [7], the Normal distribution transform [2], or other similar variations [9]. There is also a representation problem for probability distributions: in theory one should estimate the full distribution $p(\boldsymbol{x}, \boldsymbol{m}|\boldsymbol{y})$ which implies keeping track of the correlation between the pose and every part of the map. While this is relatively straightforward for landmark-based SLAM (a big covariance matrix), it is quite a problem keeping track of the correlation of the pose and, say, each cell of an occupancy grid (Rao-Blackwellized particle filters essentially work by not considering this correlation).

Another important problem is that of the prior: correct Bayesian inference needs the knowledge of the prior $p(\boldsymbol{m})$, and the posterior density $p(\boldsymbol{m}|\boldsymbol{y})$ can vary almost arbitrarily as a functional of $p(\boldsymbol{m})$. This distribution encodes all the information we know about the environment: What is the average size of the rooms? Is it structured (walls, corners, etc.), unstructured (cave-like), or is the robot outdoors? How many bathrooms are there in a typical suburban Pasadena house? These are all examples of prior knowledge, from measurement space to high-level semantic information, which should be encoded in the prior.

All modern SLAM research is well grounded in a probabilistic framework, but for SLAM algorithms it is not easy to prove the typical properties of estimators, like unbiasedness, efficiency, and so on: there are no SLAM algorithms working with dense sensors and maps for which it is possible to guarantee a given accuracy in a given environment. The main reason is that, for efficiency reasons, SLAM algorithms are a carnival of reasonable approximations, and this makes their analysis difficult.

I argue that in this context it is fruitful to study Bayesian bounds for SLAM: sometimes it is easier to study the problem than a particular solution to it. This theme has not

[1]Remark by Alfred Korzybski, 1879-1950, linguist and philosopher.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08 August 19–21, 2008, Gaithersburg, MD, USA
Copyright 2008 ACM 978-1-60558-293-1 ...$5.00.

been approached in the literature; this paper gives a first exploratory look at it, with particular regard to the problem of comparing SLAM methods which use different representations and those which use different prior information on the map.

The remainder of this paper is organized as follows. Section II recalls some basic notions of estimation theory. Section III touches on some of the reasons it is hard to analyze SLAM algorithms formally. Section IV illustrates the use of Bayesian bounds in SLAM. Section V discusses a practical example.

2. PROPERTIES OF ESTIMATORS

In the Bayesian formalization, a SLAM algorithm is nothing but an estimator of $\boldsymbol{x}$ and $\boldsymbol{m}$: "performance metrics" are just the usual properties one looks for in an estimator.

Basic properties of an estimator are its **bias**, its **variance**, and the mean squared error (**MSE**) which takes into account both bias and variance. In some fields, an estimator is called **robust** if it tolerates out-of-model errors. For example, the vanilla linear-least-squares estimator is not robust to outliers, as any one of them can lead the estimate astray.

There are two different meanings associated to "consistency" in the literature. A **consistent(1)** estimator is one whose estimate converges in probability to the true value, as the number of observations grows to infinity. A **consistent(2)** filter is one that it is not overconfident; that is, it is not too optimistic about the quality of its estimate.

Intuitively, an estimator is statistically **efficient**[2] if it makes use of all the information contained in the data; formally, it is efficient if it reaches the Cramér–Rao Bound (CRB).

2.1 The Cramér-Rao bound

There are many results about theoretical bounds for the performance of estimators [13]. The Cramér-Rao inequality is a classic one, and the proof is surprisingly easy and enjoyable [3].

Cramér-Rao inequality. *If the density $p(\boldsymbol{y}, \boldsymbol{x})$ satisfies some regularity conditions* [3]*, then for any unbiased estimator $\hat{\boldsymbol{x}}$,*

$$cov[\hat{\boldsymbol{x}}] \geq (\boldsymbol{\mathcal{I}}[\boldsymbol{x}])^{-1} \tag{1}$$

The $n \times n$ symmetric matrix $\boldsymbol{\mathcal{I}}[\boldsymbol{x}]$, called Fisher's information matrix *(FIM), is defined as*[3]

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}] = E\left\{ \frac{\partial \log p(\boldsymbol{y}, \boldsymbol{x})}{\partial \boldsymbol{x}}^{\mathrm{T}} \frac{\partial \log p(\boldsymbol{y}, \boldsymbol{x})}{\partial \boldsymbol{x}} \right\} \tag{2}$$

In the case of a Gaussian measurement model: $\boldsymbol{y} \sim \mathcal{N}(\mathbf{f}(\boldsymbol{x}), \Sigma)$, the FIM is

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}] = \frac{\partial \mathbf{f}}{\partial \boldsymbol{x}}^{T} \Sigma^{-1} \frac{\partial \mathbf{f}}{\partial \boldsymbol{x}} \tag{3}$$

There is an extension to the case of biased estimators, for which it is possible to prove a bound that can be lower or higher than (1):

$$\mathrm{cov}[\hat{\boldsymbol{x}}] \geq \left[\boldsymbol{I} + \frac{d}{d\boldsymbol{x}} \mathrm{bias}_{\hat{\boldsymbol{x}}}(\boldsymbol{x}) \right] (\boldsymbol{\mathcal{I}}[\boldsymbol{x}])^{-1} \left[\boldsymbol{I} + \frac{d}{d\boldsymbol{x}} \mathrm{bias}_{\hat{\boldsymbol{x}}}(\boldsymbol{x}) \right]^{\mathrm{T}}$$

This result is however not practically useful because one needs to know the function $\mathrm{bias}_{\hat{\boldsymbol{x}}}(\boldsymbol{x})$ which in general is not known.

The bound depends on the particular "true" value of $\boldsymbol{x}$ which is considered fixed but unknown. In general, there is no guarantee that this bound is tight, except in special cases, for example when the model is linear with Gaussian noise ($\boldsymbol{y} = \mathbf{f}(\boldsymbol{x}) = L\boldsymbol{x} + \epsilon$).

There are tighter bounds that consider higher-order derivatives of the density; unfortunately the computations quickly become convolute: see [13] for details.

3. WHY IS IT HARD TO ANALYZE SLAM ALGORITHMS?

In estimation theory there are a number of ready-made results for the "canonical" estimators: for example, we know that the maximum-likelihood estimator is consistent(1) while it is not efficient. This kind of results are not usable in SLAM, because SLAM algorithms is that they are often crude ad-hoc approximations to one of the canonical estimators, and while it might be intuitive that "almost" a ML estimator would be "almost" consistent, it is difficult to assess the impact of the approximations on the quality of the estimate.

It is interesting to give an example of where exactly these approximations are introduced. In the Bayesian framework, one already knows the solution, and the problem is just of matter of making its computation efficient: many papers start with the formula for the Bayesian filter, and then start to demolish it using the dreaded approximation symbol "$\simeq$" until there is something which is actually computable. As a typical example of this practice — to not smear the work of others — I will take my paper [5]: it is about global localization and its use inside SLAM for loop closing. We start with the usual formula for the Bayesian filter, and then we make the assumptions that for small distances the scan matching estimate can be considered overly precise (the distribution is equivalent to a Dirac delta) with respect to, say, the finite resolution of the grid. This completely reasonable assumption makes it hard to analyze the algorithm: it seems to work splendidly, but we cannot say how this approximation makes its performance differ from the exact algorithm.

A canonical example of theoretically sound, but hard to analyze, estimators are particle filters. Pretty much the only results that one can get is that everything is fine when the number of particles goes to infinity, but nothing is guaranteed for a finite computation [6]. The answer to the question of how many particles does one actually need is usually very fuzzy [8]. Additionally, there is a whole line of research involved with something that is, essentially, "cheating": using more relaxed observation models to deal with the problem of having just so many particles (see for example [11] and references therein); although this makes the filter more robust and computationally efficient, it is yet another out-of-model approximation that makes the analysis difficult.

It should be noted that not even for landmark-based SLAM it is possible to do correct inference, even though the state space is finite-dimensional. In fact, methods like the Ex-

[2] which has nothing to do as with *computationally* efficient.

[3] A more correct but cumbersome expression for the FIM would be $\boldsymbol{\mathcal{I}}[\boldsymbol{x}](\boldsymbol{x})$, in the sense that it is the information for $[\boldsymbol{x}]$, but it also depends on the particular point $(\boldsymbol{x})$ being considered. We will drop the functional dependence throughout the paper.

tended Kalman Filter (or the Information filter [12]) work by doing some kind of linearization, hence they are very far from correct inference. Linearization is a necessary evil, but it completely destroys the structure of the problem: in a certain sense, a linearized filter is not solving the original problem anymore. As a consequence, it can be shown that the EKF is not consistent(2) with respect to the original problem (see for example [10, 1]).

4. BAYESIAN BOUNDS FOR SLAM

In the PerMIS context, Bayesian bounds and the CRB in particular have many uses:

- The CRB provides a lower bound for accuracy that is a baseline for comparing the actual experimental results.
- It allows to verify the realism of accuracy claims: an algorithm claiming an accuracy below the CRB would be clearly wrong.
- In some cases, when it is tight, it provides the actual, or asymptotic, accuracy of canonical estimators like the ML.

4.1 The "easy" case: CRB for localization

The paper [4] studied the CRB for the case of 2D localization with range-finders. In that case, the observation are a set of ranges $\{\tilde{\rho}_i\}$, each corresponding to a direction φ_i. The equation $\boldsymbol{y} = \mathbf{f}(\boldsymbol{x})$ becomes in this case:

$$\tilde{\rho}_i = r(x, y, \theta + \varphi_i) + \epsilon_i \qquad i = 1 \dots n \tag{4}$$

where ϵ_i is a Gaussian noise with variance σ_i^2, and r is the "ray-tracing" function: $r(x, y, \psi)$ is the distance from point (x, y) to the first obstacle in direction ψ. The FIM depends on the geometry of the environment: in the following α_i is the surface orientation at the point intercepted by the i-th ray, and β_i is the angle of incidence, defined as $\beta_i \triangleq \alpha_i - (\theta + \varphi_i)$:

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}] = \sum_i^n \frac{1}{\sigma_i^2 \cos^2 \beta_i} \begin{bmatrix} \boldsymbol{v}(\alpha_i)\boldsymbol{v}(\alpha_i)^{\mathrm{T}} & r_i \sin \beta_i \boldsymbol{v}(\alpha_i) \\ * & r_i^2 \sin^2 \beta_i \end{bmatrix} \tag{5}$$

4.2 The CRB for SLAM

Assume we have a Gaussian independent measurements according to the model $\boldsymbol{y} = \mathbf{f}(\boldsymbol{x}, \boldsymbol{m}) + \boldsymbol{\epsilon}$. The FIM for SLAM can be written as

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}, \boldsymbol{m}] = \begin{bmatrix} \boldsymbol{\mathcal{I}}[\boldsymbol{x}] & \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}] \\ * & \boldsymbol{\mathcal{I}}[\boldsymbol{m}] \end{bmatrix} \tag{6}$$

The upper left block $\boldsymbol{\mathcal{I}}(\boldsymbol{x})$ is exactly the same as the FIM for localization, so it is (5) in the case of range finders. Here $\boldsymbol{x}$ should properly be a parameterization of the whole trajectory; however, for simplicity, we let $\boldsymbol{x}$ be a single pose (this models a sort of "one-shot" SLAM).

The term $\boldsymbol{\mathcal{I}}[\boldsymbol{x}]$ represents the achievable accuracy for localization; that is, assuming $\boldsymbol{m}$ known. Analogously, there is an interesting interpretation of the block $\boldsymbol{\mathcal{I}}[\boldsymbol{m}]$: it represents the achievable accuracy for *mapping*, that is how precise can the reconstruction of $\boldsymbol{m}$ be if we assume $\boldsymbol{x}$ known. The term $\boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}]$ represents the mutual information that $\boldsymbol{x}$ gives about $\boldsymbol{m}$ and vice versa. The upper-right block $\boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}]$ represents the "simultaneous" in Simultaneous Localization and Mapping.

The CRB is the inverse of the FIM:

$$\mathrm{cov}[\hat{\boldsymbol{x}}, \hat{\boldsymbol{m}}] \geq (\boldsymbol{\mathcal{I}}[\boldsymbol{x}, \boldsymbol{m}])^{-1} \tag{7}$$

and must be interpreted as a bound on the *joint* covariance of $\boldsymbol{x}$ and $\boldsymbol{m}$, given one sensor reading. The bound (7) depends on the particular representation used for $\boldsymbol{m}$. This is bad, because one would want to compare the performance of SLAM algorithms that use different representations.

However, it can be shown that the upper left block of $[\boldsymbol{\mathcal{I}}(\boldsymbol{x}, \boldsymbol{m})]^{-1}$ is independent of the representation of $\boldsymbol{m}$, in the sense that it is invariant to a change of variable $\boldsymbol{m}' = g(\boldsymbol{m})$. The upper-left block is given by $(\boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{m}])^{-1}$ where $\boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{m}]$ is a non-standard but useful notation:

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{m}] \triangleq \boldsymbol{\mathcal{I}}[\boldsymbol{x}] - \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}] \left(\boldsymbol{\mathcal{I}}[\boldsymbol{m}]\right)^{-1} \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}]^{\mathrm{T}}$$

The correct interpretation of $\boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{m}]$ is as the information in the data about $\boldsymbol{x}$, after considering that the nuisance parameter $\boldsymbol{m}$ is unknown. Note that if $\boldsymbol{x}$ and $\boldsymbol{m}$ were "orthogonal" ($\boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}] = 0$), then one would have $\boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{m}] = \boldsymbol{\mathcal{I}}[\boldsymbol{x}]$.

Formally, we prove the following proposition:

Proposition 1: Assume a model of the kind $\boldsymbol{y} = \mathbf{f}(\boldsymbol{x}, \boldsymbol{w})$. Then the CRB for $\boldsymbol{x}$ is independent of a 1-to-1 reparametrization $\boldsymbol{w} \mapsto \boldsymbol{m} = p(\boldsymbol{w})$.

Proof: Suppose we do a reparametrization $\boldsymbol{w} \mapsto \boldsymbol{m}$. The new sensor model is a function $\boldsymbol{g}$ of $\boldsymbol{m}$:

$$\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{m}) = \mathbf{f}(\boldsymbol{x}, p^{-1}(\boldsymbol{m}))$$

The FIM for $\boldsymbol{x}$ is

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{m}] = \boldsymbol{\mathcal{I}}[\boldsymbol{x}] - \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}] \left(\boldsymbol{\mathcal{I}}[\boldsymbol{m}]\right)^{-1} \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}]^{\mathrm{T}} \tag{8}$$

Assuming a normalized covariance for the measurements ($\sigma^2 = 1$), the elements in (8) are given by

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}] = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}^{\mathrm{T}} \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{m}} \qquad \boldsymbol{\mathcal{I}}[\boldsymbol{m}] = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{m}}^{\mathrm{T}} \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{m}}$$

Now note that the chain rule says that

$$\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}} = \frac{\partial \mathbf{f}}{\partial \boldsymbol{x}}$$

$$\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{m}} = \frac{\partial \mathbf{f}}{\partial \boldsymbol{w}} \frac{\partial}{\partial \boldsymbol{m}} p^{-1}(\boldsymbol{m}) = \frac{\partial \mathbf{f}}{\partial \boldsymbol{w}} \left[\frac{\partial p(\boldsymbol{w})}{\partial \boldsymbol{w}}\right]^{-1}$$

Let $\boldsymbol{J} \triangleq \frac{\partial p(\boldsymbol{w})}{\partial \boldsymbol{w}}$ to obtain

$$\boldsymbol{\mathcal{I}}[\boldsymbol{m}] \quad = \quad \boldsymbol{J}^{-\mathrm{T}} \frac{\partial \mathbf{f}}{\partial \boldsymbol{w}}^{\mathrm{T}} \frac{\partial \mathbf{f}}{\partial \boldsymbol{w}} \boldsymbol{J}^{-1} = \boldsymbol{J}^{-\mathrm{T}} \boldsymbol{\mathcal{I}}[\boldsymbol{w}] \boldsymbol{J}^{-1}$$

and

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{m}] = \frac{\partial \boldsymbol{g}}{\partial \boldsymbol{x}}^{\mathrm{T}} \frac{\partial \mathbf{f}}{\partial \boldsymbol{w}} \boldsymbol{J}^{-1} = \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{w}] \boldsymbol{J}^{-1}$$

By substituting these in (8), one obtains:

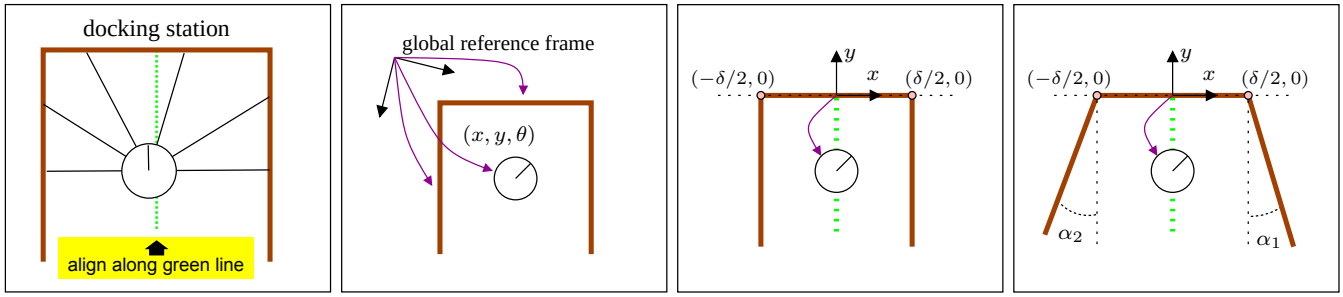
$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{m}] = \boldsymbol{\mathcal{I}}[\boldsymbol{x}] -$$

$$\left[\boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{w}]\boldsymbol{J}^{-1}\right] \left[\boldsymbol{J}^{-\mathrm{T}} \boldsymbol{\mathcal{I}}[\boldsymbol{w}] \boldsymbol{J}^{-1}\right]^{-1} \left[\boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{w}]\boldsymbol{J}^{-1}\right]^{\mathrm{T}}$$

$$= \boldsymbol{\mathcal{I}}[\boldsymbol{x}] - \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{w}] \left(\boldsymbol{\mathcal{I}}[\boldsymbol{w}]\right)^{-1} \boldsymbol{\mathcal{I}}[\boldsymbol{x}|\boldsymbol{w}]^{\mathrm{T}} = \boldsymbol{\mathcal{I}}[\boldsymbol{x}/\boldsymbol{w}]$$

□

This result makes sense: no matter what exotic representation one uses for estimating the map, still there is a well-defined bound for the accuracy of $\boldsymbol{x}$.



(a) Docking task (b) Unobservable representation (c) Rectangular assumption (d) Polygonal assumption

Figure 1: (a): The robot must execute a simple docking task. (b): Expressing the pose of the robot and of the walls in an absolute frame of reference makes the representation unobservable. (c) shows a possible representation, with the only unknown being the width of the docking station. (d) augments this representation by introducing the possibility of not perpendicular walls.

In the context of PerMIS, this suggests that a correct way of evaluating a SLAM algorithm is by comparing its accuracy in estimating the posterior of $\boldsymbol{x}$ with the well-defined CRB: implicitly, this will also assess the accuracy of the map. This has some practical consequences: one only needs the ground truth for $\boldsymbol{x}$, and it is possible to compare the accuracy of algorithms using different map representations.

Unfortunately, this result is not useful to treat the case of different prior information. That would be the case in which the two representations $\boldsymbol{w}$ and $\boldsymbol{m}$ belong to two sets that cannot be put in a 1-to-1 correspondence. The example in the next section illustrate such a case and shows that the prior information on the map does influence the achievable accuracy for the pose.

5. AN EXAMPLE

The purpose of this example is to show in detail a practical computation of the CRB in a SLAM setting, and to illustrate the impact of different prior information for the map on the achievable accuracy for the pose.

5.1 Scenario

Suppose we are designing a mobile robot which must execute a docking task, perhaps to attach to its battery charger. The docking station geometry is not completely known: for example, it might be known that it is rectangular, but the precise dimensions are not known. This is a practical case in which the robot must do SLAM: estimate both the geometry of the docking station *and* its position inside it.

We are interested in how accurately such task can be accomplished given the robot's range-sensor sensor accuracy, the geometry of the environment, and the a-priori information.

We consider three cases:

1. (Fig. 1(a)) The robot knows the docking station is rectangular, with a known geometry. The robot must dock at the middle of the side, properly orientated.
2. (Fig. 1(c)) The robot knows the docking station is rectangular, but it does not know the width δ.
3. (Fig. 1(d)) The robot knows the docking station is polygonal.

The first case corresponds to localization; the other two are SLAM with different prior hypotheses. Notice that the three cases are progressive generalizations: if we denote by M_i the sets of possible worlds in the i-th case, we have that $M_1 \subset M_2 \subset M_3$.

5.2 Steps for the analysis

The CRB analysis follows these steps:

1. Find a parametrization of the map $\boldsymbol{m}$ and of the poses which is observable.
2. Write all measurements as a function $\boldsymbol{y} = \boldsymbol{r}(\boldsymbol{x}, \boldsymbol{m})$ and compute the FIM.
3. Decide which variables are known, completely unknown, or unknown with some a priori information, and obtain the CRB for a chosen subset of the state which is of interest.

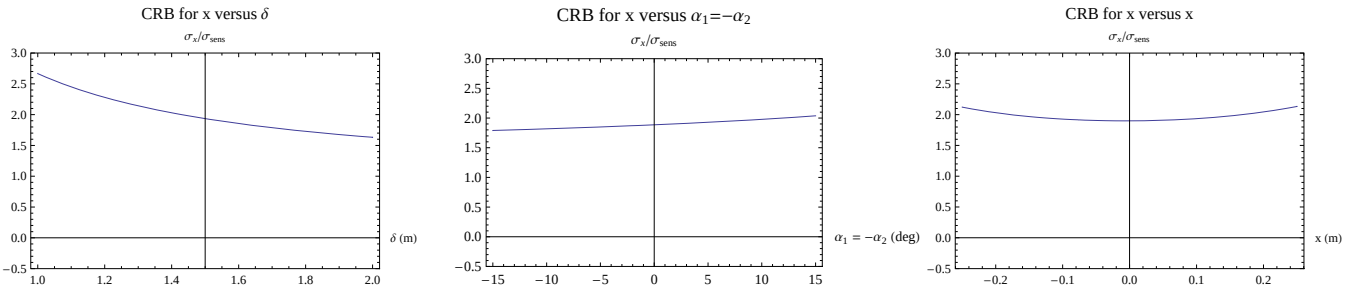
The computations are not difficult, but pretty tedious: this is an excellent case for which one should use a computer aided algebra system; Appendix I shows the Mathematica notebook used.

5.3 Parameterizing the world

Even if not strictly necessary, it is better to work with a representation of the world which is fully observable. To achieve this, it suffices to choose the reference frames correctly: SLAM with relative measurements is not observable, if global coordinates are chosen. For example, Fig. 1(b) shows a unobservable way to parametrize the environment: a relative distance sensor can only observe the distance between itself and the walls, but not their absolute position.

It is convenient to choose the variable of interest among the state variables. Because we assume the task is to align along the middle of the front edge, we put the reference frame at its midpoint, and we call δ the total length (Fig. 1(c)).

In 2D it is a good choice to use the polar representation of lines: $\{(x, y) | \cos(\alpha)x + \sin(\alpha)y = \rho\}$. In this environment, the lines on the sides pass through the points $(-\delta/2, 0)$ and $(\delta/2, 0)$: this gives another constraint between α and ρ. Assuming α as the independent variable, we obtain for the first line $\rho_1 = \cos(\alpha_1)\delta/2$, and similarly for the other. (Thanks to



(a) CRB for x as a function of δ (b) CRB for x as a function of α_1 (c) CRB for x as a function of x

Figure 2: Changes in the CRB for x as a function of small perturbations of the other variables.

Proposition 1, the representation does not matter, hence we could have chosen ρ_1 as the independent variable). Fig. 1(d) shows the resulting parametrization $\boldsymbol{m} = (\delta, \alpha_1, \alpha_2)$ for the full SLAM case.

5.4 Writing the measurements equation

Suppose the i-th ray is colliding with the j-th line, which has coordinates (α_j, ρ_j). We should find the expression for the "ray-tracing function" $r(x, y, \phi)$ for a line expressed in polar coordinates (α, ρ). A bit of geometry shows that the distance from point (x, y) to the line (α, ρ) along direction ϕ is

$$r_{(\alpha,\rho)}(x, y, \phi) = \frac{\cos(\alpha)x + \sin(\alpha)y - \rho}{\cos(\alpha - \phi)}$$

The measurement equation is written for each ray i:

$$y_i = r_{(\alpha_i,\rho_i)}(x, y, \theta + \varphi_i) + \epsilon_i$$

where (α_i, ρ_i) is the line intercepted by the ray, (x, y, θ) is the robot pose, and φ_i is the orientation of the ray in the robot frame.

Group all the measurements in an equation

$$\boldsymbol{y} = \boldsymbol{r}(x, y, \theta, \delta, \alpha_1, \alpha_2)$$

and compute the FIM using (3).

5.5 Computing the CRB

To compute the CRB given the global FIM, one should divide the variables into three groups:

1. The variables of interest $\boldsymbol{x}^i$ for which one wants to compute the CRB.
2. The variables considered known $\boldsymbol{x}^k$.
3. The other unknown variables $\boldsymbol{x}^n$, called *nuisance variables*, which are not of interest but the fact of them being unknown affects the achievable accuracy for the variables of interest.

In the three scenarios, the division is as follows:

Case	interest ($\boldsymbol{x}^i$)	known ($\boldsymbol{x}^k$)	nuisance ($\boldsymbol{x}^n$)
1	x, θ	$\delta,\ \alpha_1,\ \alpha_2$	y
2	x, θ	$\alpha_1,\ \alpha_2$	y, δ
3	x, θ		$y, \delta, \alpha_1, \alpha_2$

By doing permutations if necessary, rewrite the FIM as:

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}^i, \boldsymbol{x}^n, \boldsymbol{x}^k] = \begin{bmatrix} \boldsymbol{\mathcal{I}}[\boldsymbol{x}^i] & \boldsymbol{\mathcal{I}}[\boldsymbol{x}^i|\boldsymbol{x}^n] & \boldsymbol{\mathcal{I}}[\boldsymbol{x}^i|\boldsymbol{x}^k] \\ * & \boldsymbol{\mathcal{I}}[\boldsymbol{x}^n] & \boldsymbol{\mathcal{I}}[\boldsymbol{x}^n|\boldsymbol{x}^k] \\ * & * & \boldsymbol{\mathcal{I}}[\boldsymbol{x}^k] \end{bmatrix}$$

Variables that are considered known are effectively ignored and their rows/columns can be cancelled from the matrix. We are left with

$$\boldsymbol{\mathcal{I}}[\boldsymbol{x}^i, \boldsymbol{x}^n] = \begin{bmatrix} \boldsymbol{\mathcal{I}}[\boldsymbol{x}^i] & \boldsymbol{\mathcal{I}}[\boldsymbol{x}^i|\boldsymbol{x}^n] \\ * & \boldsymbol{\mathcal{I}}[\boldsymbol{x}^n] \end{bmatrix}$$

To compute the CRB for the variables of interest, do not invert the whole matrix, just compute the upper left block of $\left(\boldsymbol{\mathcal{I}}[\boldsymbol{x}^i, \boldsymbol{x}^n]\right)^{-1}$, which in Section IV was shown to be computed as $\left(\boldsymbol{\mathcal{I}}[\boldsymbol{x}^i/\boldsymbol{x}^n]\right)^{-1}$.

5.6 Results

The full details for the computation are provided in the Mathematica notebook in Appendix I. In the following, σ_{sens} is the accuracy of the sensor and serves as a scale parameter.

We obtain for the first case (localization):

$$\sigma[x] \geq 0.49\,\sigma_{\text{sens}} \qquad \sigma[\theta] \geq 0.90\,\sigma_{\text{sens}}$$

and for the case of unknown δ:

$$\sigma[x] \geq 0.49\,\sigma_{\text{sens}} \qquad \sigma[\theta] \geq 0.91\,\sigma_{\text{sens}}$$

It appears that not knowing δ does not have much impact in the accuracy for estimating x. It is usually not immediate to interpret these results. In this case, one intuitive explanation might be that the robot needs to align halfway through the front edge by observing the distances to the left and right walls; even if there is an error in the estimate for δ, this error is evenly distributed on the left and right side, and it does not impact the accuracy of the task (the small variation is due to second-order couplings between the variables).

The results for the full SLAM case are:

$$\sigma[\boldsymbol{x}] \geq 1.94\,\sigma_{\text{sens}} \qquad \sigma[\theta] \geq 0.98\,\sigma_{\text{sens}}$$

In this case, not knowing α_1, α_2 has a serious repercussion on the accuracy on x.

Further insights on the process can be gained by considering a variation of the parameters. Suppose we start with a rectangular docking station ($\alpha_1 = \alpha_2 = 0$). If we want to improve the accuracy on the task, would it be better to increase δ? Fig. 2(a) shows that it is: if the sides are further away, they are a better localizing reference. Fig. 2(b)

shows that, if possible, it is better to design a docking station which is slightly triangular ($\alpha_1 = -\alpha_2 < 0$). Fig. 2(b) shows that the accuracy for x is maximum at the middle of the environment ($x = 0$) so it would not pay to change the docking position.

6. CONCLUSIONS AND OUTLOOK

The study of the FIM provides good insight into the SLAM process. The CRB is a necessary component in every experimentalist's toolkit. In the PerMIS context, the CRB for SLAM suggests that the right way of comparing SLAM algorithms is by comparing the accuracy in the posterior estimate of $\boldsymbol{x}$. This is also the most practical way and it bypasses the problem of comparing algorithms that use a different representation for the map.

However, it should be understood that, in general, the CRB is not a tight bound, in the sense that it can be optimistic. As a rule of thumb, the more the problem is nonlinear, the more the CRB is optimistic. In the case of localization with range-finders, in [4] it has been shown that the CRB is quite tight. A similar experimental investigation should be completed for SLAM, and other sensors should be considered as well.

7. REFERENCES

[1] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot. Consistency of the EKF-SLAM algorithm. In *IEEE/RSJ Intern. Conf. on Intelligent Robots and Systems (IROS)*, 2006.

[2] P. Biber and W. Strasser. The Normal distributions transform: a new approach to laser scan matching. In *IEEE/RSJ Intern. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[3] G. Casella and R. L. Berger. *Statistical Inference*. Duxbury Press, 1990.

[4] A. Censi. On achievable accuracy for range-finder localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4170–4175, Rome, Italy, Apr. 2007.

[5] A. Censi and G. D. Tipaldi. Lazy localization using the Frozen-Time Smoother. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Pasadena, CA, May 2008.

[6] D. Crisan and A. Doucet. A survey of convergence results on particle filtering for practitioners. *IEEE Trans. Signal Processing*, 50(3):736–746, 2002.

[7] A. Elfes. Using occupancy grids for mobile robot perception and navigation. *IEEE Computer*, 22(6), 1989.

[8] D. Fox. Adapting the sample size in particle filters through KLD-sampling. *International Journal of Robotics Research*, 22, 2003.

[9] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, Feb. 2007.

[10] S. Huang and G. Dissanayake. Convergence and consistency analysis for Extended Kalman Filter based SLAM. *IEEE Transactions on Robotics*, 23(5):1036–1049, Oct. 2007.

[11] P. Pfaff, C. Plagemann, and W. Burgard. Gaussian mixture models for probabilistic localization. In *Intern. Conf. on Robotics and Automation (ICRA)*, Pasadena, CA, 2008.

[12] S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte, and A. Ng. Simultaneous localization and mapping with sparse extended information filters. *International Journal of Robotics Research,*, 23(7-8), 2004.

[13] H. L. V. Trees and K. L. Bell. *Bayesian Bounds for Parameter Estimation and Nonlinear Filtering/Tracking*. Wiley-IEEE Press, 2007.

APPENDIX
A. MATHEMATICA CODE

This is the model of a range finder: distance to the line (α,ρ) from point (x,y) in direction ϕ.

```
In[1]:= range[α_, ρ_, x_, y_, ϕ_] := (Cos[α] x + Sin[α] y - ρ) / Cos[α - ϕ];
```

These are the three lines (R=Right, L=Left, U=Up):

```
In[2]:= rangeR[x_, y_, ϕ_] := range[α1, Cos[α1] δ / 2, x, y, ϕ];
        rangeL[x_, y_, ϕ_] := range[α2, -Cos[α2] δ / 2, x, y, ϕ];
        rangeU[x_, y_, ϕ_] := range[π / 2, 0, x, y, ϕ];
```

Because we don't want to write here a full-fledged ray-tracer, we make the hypothesis that rays 1,2 intersect line on the right, rays 3,4 intersect the line in front, rays 5,6 intersect the line on the left. The analysis is valid only in such interval.

```
In[5]:= Y = { rangeR[x, y, θ + ϕ1], rangeR[x, y, θ + ϕ2],
           rangeU[x, y, θ + ϕ3], rangeU[x, y, θ + ϕ4], rangeL[x, y, θ + ϕ5], rangeL[x, y, θ + ϕ6]};
```

Computing the FIM is a easy but tedious task, we are happy in letting *Mathematica* do the derivations for us. The vector vars chooses the order of variables in the matrix.

```
In[6]:= vars = {x, y, θ, δ, α1, α2};
        FisherInformation[Y_, vars_] := Transpose[D[Y, {vars}]].D[Y, {vars}];
        fim = FisherInformation[Y, vars];
```

To compute the Cramér-Rao Bound correctly, we need to take into account nuisance parameters.

```
In[9]:= iCRB[fim_, varOfInterest_, varNuisance_] := (
           VarIndexes[v_] := Flatten[Map[Flatten[Position[vars, #]] &, v]];
           inte = VarIndexes[varOfInterest]; nuis = VarIndexes[varNuisance];
           If[ Length[varNuisance] > 0, fim[[inte, inte]] - fim[[inte, nuis]].Inverse[fim[[nuis, nuis]]].fim[[nuis, inte]],
            fim[[inte, inte]] ]
          );
        CRB[fim_, vars_, nuisances_] := Inverse[iCRB[fim, vars, nuisances]];
        CRBdesc[fim_, vars_, nuisances_] := ( mat = CRB[fim, vars, nuisances];
           Map[σ[vars[[#]]] ≥ (mat[[#, #]]) ^ (1 / 2) σsens &, {1, 2}]);
```

To evaluate the FIM and CRB, we must decide a true state:

```
In[12]:= trueState =
           {α1 → 0.1, α2 → -0.1, δ → 1.5, x → 0.1, y → 0.5, θ → π / 2 + 0.1, ϕ1 → -π / 2, ϕ2 → -π / 3, ϕ3 → -π / 4, ϕ4 → π / 4, ϕ5 → π / 3, ϕ6 → +π / 2};
```

In the case of localization, we assume the variables $\alpha_1, \alpha_2, \delta$ are known:

```
In[13]:= CRBdesc[fim /. trueState, {x, θ}, {y}]

Out[13]= {σ[x] ≥ 0.489147 σsens, σ[θ] ≥ 0.897132 σsens}
```

In the first example of SLAM, we assume δ is not known; that is, it is a nuisance parameter.

```
In[14]:= CRBdesc[fim /. trueState, {x, θ}, {y, δ}]

Out[14]= {σ[x] ≥ 0.489148 σsens, σ[θ] ≥ 0.910006 σsens}
```

In the last example, $\alpha_1, \alpha_2, \delta$ are not known:

```
In[15]:= CRBdesc[fim /. trueState, {x, θ}, {y, δ, α1, α2}]

Out[15]= {σ[x] ≥ 1.93523 σsens, σ[θ] ≥ 0.980067 σsens}
```

This is the code to create the graphs in the last figure:

```
In[16]:= p1 = Plot[(CRB[fim /. {α1 → β (π / 180), α2 → -β (π / 180)} /. trueState, {x}, {δ, α1, α2, y, θ}]) ^ (1 / 2), {β, -15, 15}
           , Frame → True, PlotRange → {-0.5, 3},
           AxesLabel → {"α1 = -α2 (deg)", "σx/σsens"}, AxesOrigin → {0, 0}, PlotLabel → "CRB for x versus α1=-α2"];

In[17]:= p2 = Plot[(CRB[fim /. {δ → d} /. trueState, {x}, {δ, α1, α2, y, θ}]) ^ (1 / 2), {d, 1, 2},
           Frame → True, PlotRange → {-0.5, 3},
           AxesLabel → {"δ (m)", "σx/σsens"}, AxesOrigin → {1.5, 0}, PlotLabel → "CRB for x versus δ"];

In[18]:= p3 = Plot[(CRB[fim /. {x → d} /. trueState, {x}, {δ, α1, α2, y, θ}]) ^ (1 / 2), {d, -0.25, 0.25},
           Frame → True, AxesLabel → {"x (m)", "σx/σsens"},
           PlotLabel → "CRB for x versus x", AxesOrigin → {0, 0}, PlotRange → {-0.5, 3}];

In[19]:= Export["crb1.pdf", p1]; Export["crb2.pdf", p2]; Export["crb3.pdf", p3];
```

Map Quality Assessment

Asim Imdad Wagan[1], Afzal Godil[1], Xiaolan Li[1,2]
[1]National Institute of Standards and Technology
[2]Zhejiang Gongshang University
{wagan, godil, lixlan}@nist.gov

ABSTRACT
The maps generated by robots in real environment are usually incomplete, distorted, and noisy. The map quality is a quantitative performance measure of a robot's understanding of its environment. Map quality also helps researcher study the effects of different mapping algorithms and hardware components used. In this paper we present an algorithm to assess the quality of the map generated by the robot in terms of a ground truth map. To do that, First, localized features are calculated on the pre-evaluated map. Second, nearest neighbor of each valid local feature is searched between the map and the ground truth map. The quality of the map is defined according to the number of the features having the correspondence in the ground truth map. Three feature detectors are tested in terms of their effectiveness, these are the Harris corner detector, Hough Transform and Scale Invariant Feature Transform.

Categories and Subject Descriptors
I.2.9 [**Robotics**]: Robot Map Quality.

General Terms
Algorithms

Keywords
Harris corner detector, Scale Invariant Feature Transform, Hough Transform

1. INTRODUCTION
Assessing the map quality generated by the robots is one of the useful ways to assess the capability of a robot's understanding of its surrounding environment. Robot map quality is one of the quantitative measures which can be helpful in determining which robots will perform better in the field. When a robot moves, it generates the map which helps in its localization and planning. Normal image noise measures are not suitable to assess the map quality because the differences in maps are structural in nature. To assess these kinds of maps we had assumed that the quality is defined not in terms of overall image but on the structural detail contained inside the image. This can be observed in the ground truth image as shown in figure 1 while one test map is shown in figure 2 which was generated by the robot. So we consider a map as accurate even if there is noise and distortions present but it contains all the salient details of the ground truth.

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

There is not much work done in this field. Most of the work done is either in the field of image quality measure or in the range data quality measure for moving robots. The initial work was done by Chandran et al in [12] and [13]. Their method measures the quality of the map from 3D point cloud generated from the robot. This point cloud is further classified into plausible and suspicious patches using the conditional random fields. The number of suspicious patches is used to calculate the quality. Although this method seems promising but it is dependent upon the mapping where the data generated is in 3D point cloud format. The testing was done with data generated from 3D laser scanners.

In [14] the authors have used the polylines to model the shapes from the robot generated maps and utilization of these models for the solution of the SLAM problem. Although this paper is not directly related with the map quality problem but it provides an interesting insight into map generation which can be used to identify different parts of the map. As identification of different regions inside the map can be helpful in assessing the map quality.

Recently a manual map evaluation toolkit," Jacob's Map Analysis Toolkit" has been suggested by [16]. This application provides the manual map viewer which is designed in a way to help the quality assessor to judge the quality of the map by overlapping the maps over each other. This toolkit also provides a simple measure to assess the map quality based on evolutionary algorithms. However, using the evolutionary algorithms does not guarantee that the results for the map quality will be similar when used each time.

Another recent development is the usage of localized features for map quality assessment [17]. Authors have used the rooms as the localized features. They propose an algorithm to detect the rooms and then find the map quality using these localized features. But it is not necessary that a map will always contain features such as rooms and room like structures. Most of the time the robot generated maps do not contain lines but the collection of point clouds generated from the sensors. In that case, it will be difficult to identify rooms.

Our proposed algorithm is based on techniques which try to cover most of the short comings found in other algorithms. There are still many cases where our algorithm can fail. The reason to use multiple features is that if one kind of algorithm fails in some specific type of map, there are still two other options to judge the map quality correctly.

Some of the limitations are because of inherent nature of the algorithms used which will be discussed later in this paper.

Rest of our paper discusses our algorithm which consists of three separate sub parts. Our algorithm can be described in following steps:

1. Generation of the localized features on the map.
2. Similar features are identified in Target image and ground truth image.
3. Quality is calculated from the final number of matched features.

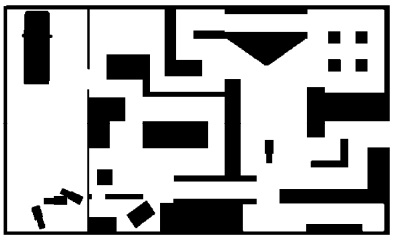


Figure 1. Ground Truth Map.

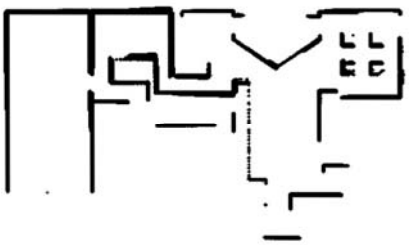


Figure 2. Robot generated Map

2. MAP QUALITY

2.1 Introduction

Measuring the map quality is a very difficult task because it is difficult to define quality in terms of the image. There can be different criteria to define the map quality. Some of them can be based on the noise generated in the maps and on the other side some can be based on the rotation and translation observed in the generated map. There can be many ways to define the map quality. But one important factor which we want to measure in the map quality for robot is the structural details in the map, so although there might be some other noise in the map it is assumed that any map is accurate if it thoroughly represents all the important structure features when compared to the ground truth. So to assess this measure we cannot use the noise to signal ratio or some other measures. We are proposing a novel method to assess the map quality based on three separate algorithms each corresponding to different type of features found in the map. These are Harris Corner Detector, Hough Transforms and Scale Invariant Feature transform. These measures will gives us three values which can be used to assess the quality of the map in three different terms.

2.2 Harris based algorithm

Our first algorithm is defined on the principals of closest point matching. Let us assume we are given two images to compare named X and Y. To compare these images we need to find interest points in these images. These images are binary so we have limited choice in selecting the interest point algorithms. Most of the interest point detectors work on gray scale or color images. The interest points should be useful with enough detail so that they can be compared with points in other image.

Corner detectors are effective in case we have binary images so we have chosen Harris corner detector [8] [9]. This algorithm is very effective in capturing corners and is effectively invariant to rotation, scale, illumination variation and image noise. This is a desirable metric which will enable us to deal with minor noise, rotation and scale problems in the map, see figure 3.

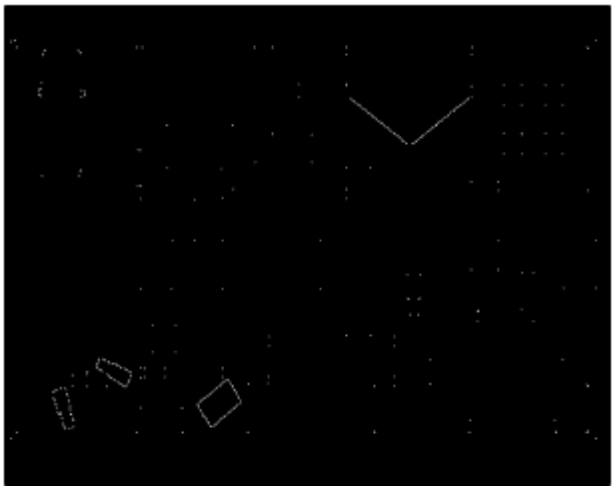


Figure 3. Harris corner detector

After calculating the interest point using the Harris corner detector, we use the closest point matching process to generate the vector maps which are later used for calculating the quality metric. To generate the vector map we find the corners which are closest to the point under consideration and then use that point in map and find its closest point in the ground truth and eliminate those points from both maps with increase in the value for true points matched counter for the map quality.

2.3 Hough based algorithm

To account for the structural detail we have used Hough transform [5] [6] to transfer the map from Euclidean space to Hough space. This has the benefit of identifying lines in the image. These lines are compared according to the position of lines as points in the Hough space. Hough space is created by exchanging the Euclidean coordinates with the parameterized values form the parametric form of the equation of the line.

$$r(\theta) = x\cos\theta + y\sin\theta \tag{1}$$

This helps in identifying lines easily as in the Hough space the points with large values will be highly likely to represent the lines. This same process can be repeated to generate the space for circle and other geometrical objects detection. A variation of the Hough transform which is known as the generalized Hough transform, can be used to detect different type of arbitrary shapes in the

image. This can be used to detect lines, squares (rooms etc), circle (roundabouts etc) in the map which will be a more generalized way to calculate the map quality.
After detection of these features the matching features can be located in the ground truth map and compared for the map quality as described in the last section.

2.4 Scale Invariant Feature Transform

Scale Invariant Feature Transform (SIFT) was introduced by the David Lowe in [15]. Since then SIFT based localized feature have gained prominence among researchers due to there invariability to rotation scale and even dynamic changes. To assess the map quality we have proposed an algorithm based the SIFT. SIFT feature are calculated from extrema detection by finding the extrema points from difference of Gaussian images as shown in equation 2, where the G_m and G_n represent the Gaussian filters at multiple scales and I is the original image. These points are further processed to find out the stable point under various conditions like edge response and low contrast point elimination.

$$DOG(I) = (G_m * I) - (G_n * I) \tag{2}$$

SIFT points detection is the first part of the process, after detection usually a descriptor is calculated and stored for each point so that it can be used to compare point from different images. The length of the SIFT detector is equal to 128 elements, which is basically the directional histogram of the local region.

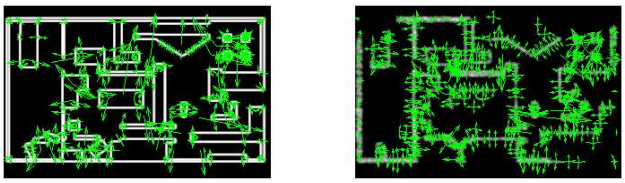


Figure 4. SIFT features on ground truth and robot generated maps.

For our algorithm we have used the following procedure:

1- First the entropy [18] of the image is calculated so that important regions with high entropy are identified. As our maps are binary images it is necessary to convert them into multiple scales with more information so that useful features are calculated.

2- This image is passed on to the SIFT for feature detection and descriptor calculation, see figure 4 for an example of SIFT features.

2.5 Closest Point Matching

Closest point matching is performed by finding the closest point to the corresponding interest points in one image to another. Each point in the ground truth is mapped in a one to one fashion between the ground truth image and the target image. To keep points from matching to a point which is extremely far, the matching is performed only for the points which exist below a specified threshold. So it generates a displacement map for each point from one image to another image. The obvious benefit is the localized identification of the object interest points.

The closest point match can be described by equation 3.

$$\text{Match} = \text{Dis}\,(\text{FV}(\text{P}(\text{x},\text{y})) - \text{FV}(\text{P}_\theta(\text{x},\text{y}))) \tag{3}$$

Where equation 3 describes that the match is the point which is equivalent to the point in one map to the corresponding region in another map under an specified threshold, where FV is the feature vector of the P(x,y) and Dis is the distance between two corresponding feature vectors. Only in the case of the SIFT features the comparing criteria is based on the calculated descriptors.

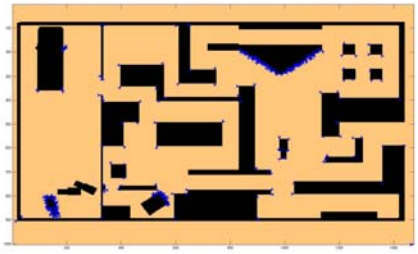


Figure 5. Map showing displacement of interest points.

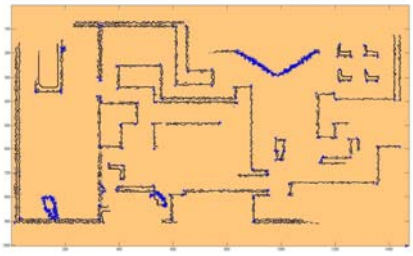


Figure 6. Displacement in test image.

2.6 Vectorial Space

The displacement or vector map calculated in the last step provides much more information regarding the kind of distortion which appeared in the image. This way this vector map is a localized distortion map in the image. This can be done in both directions to identify the missing features which were not captured and extra features which don't really exist. The figure 5 shows the displacement of closest points in ground truth while the vectorial space is shown in figure 6 for the test image.

2.7 Quality Measure

The map quality measure is calculated using the ratio between the set of features. The map quality can be defined mathematically as

$$q = RMF / GTF \tag{4}$$

where RMF are the number of valid feature points found in the robot generated map while GTF are the number of feature points in the ground truth map.

The map quality obtained from the set of test images (as show in Figure 8) is shown in Table 1 and figure 7.

Image quality assessment is difficult [7] [10] because for each case there can be different criteria to define the quality. For these robot generated maps the most important quality measure is the amount of features or landmarks (points, lines, etc) which are contained in the generated map. That is why we have based our quality measure on the feature having same shapes. We have not

used the texture and color information because the maps are only binary images.

Table 1. Quality values obtained with different algorithms.

Map	Hough	SIFT	Harris
1	0.61818	0.52966	0.68981
2	0.74545	0.55085	0.81481
3	0.41818	0.47175	0.35648
4	0.58182	0.50000	0.67593
5	0.47273	0.41102	0.84722
6	0.50909	0.40395	0.71759
7	0.49091	0.39407	0.40741
8	0.30909	0.45763	0.24074

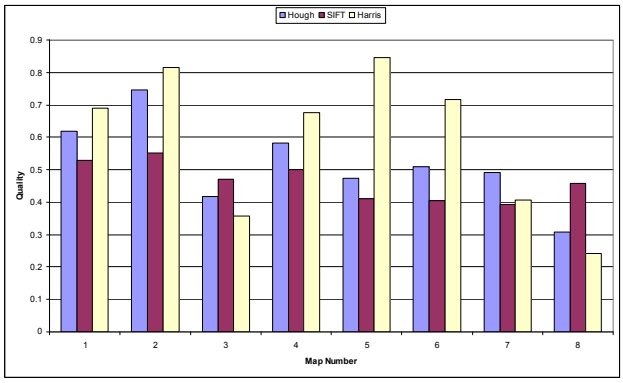


Figure 7. Comparative view of different algorithms.

A very subtle issue is with the finding of the quality of the maps when they are the subset of a larger map. The ground truth is assumed to be the superset of all the maps so it contains all the features and information. So to assess the quality of the map which is smaller than the ground truth, we have to identify the subset from ground truth for which the map was generated. This remains an issue with this algorithm although for maps which are equivalent to the ground truth the algorithm gives fairly accurate results.

Only other remaining issue is the utilization of the threshold. Utilization of threshold can be a problem because we will not be able to match features if the maps are not aligned as in the case of Harris and Hough transform but this is not the case for SIFT based detector because it can detect matches even if they are far away, independent of scale, rotation and dislocation. Although for the Harris and Hough alignment of the map remains an important point. Alignment can be achieved by a startup marker that identifies a stable point between the robot generated map and the ground truth. A map can be considered more accurate if it consistently shows good performance in all three measures.

3. LIMITATIONS

This system is only suitable for offline-measurement for the quality of the maps. As per definition the measure of quality is very difficult to define because requirements on which the map quality is based can be changed according to the need.

This algorithm measures the quality only on the basis of the information content of the image. These maps only contain bi-level images without any additional information. Map distortions and noise are not considered because the information is intact even with the added noise.

Some of the limitations which are observed are due to the type of maps used for processing. If the map has signal noise, such as, a jagged line or map with distortions, most likely the Harris corner detector will find lots of corners which could give erroneous results. Also Hough transform will fail for the case when point cloud data is separated quit far apart. Similarly for the SIFT case, if there is too much noise in the maps, this will introduce additional features which can cause problems during comparison of the features, because closely related features will give similar results.

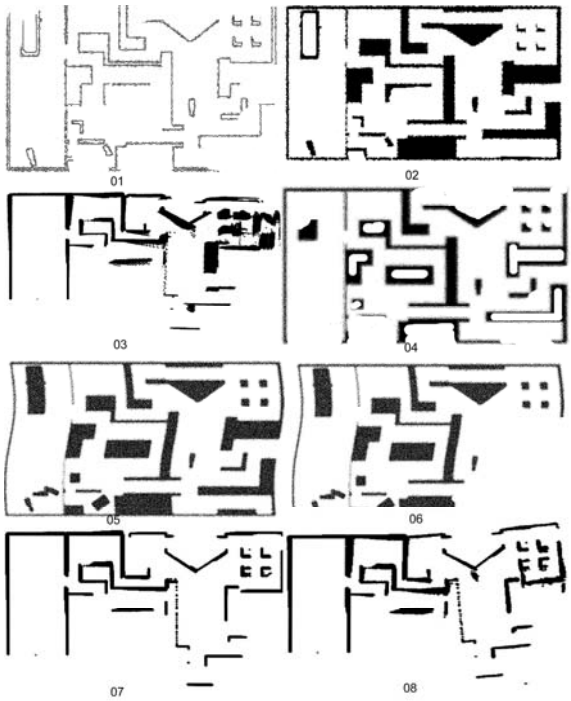


Figure 8. Maps used for the comparison.

4. CONCLUSION

We have tested our algorithm on the test map images generated by robots. The map images are also augmented with additional set of artificially created images to check the quality.

In conclusion, we have devised an automated method to calculate the quality of the maps generated by the robot. We have used the Harris corner detector to detect the interest points while we have used Hough transforms to detect lines, another important localized feature which we have used is scale invariant feature transform (SIFT). In the end we propose the three measures that can define the map quality in three separate terms. We also provide a vectorial map that basically tells us the local distortions found in the image.

ACKNOWLEDGMENTS

We would like to acknowledge Chris Scrapper and Raj Madhavan for their help and the map data. We would also like to thank the SIMA program and the IDUS program for supporting this work.

REFERENCES

[1] B. Girod, “What's wrong with mean-squared error,” Digital Images and Human Vision, A. B. Watson, Ed. Cambridge, MA: MIT Press, 1993, pp. 207-220.

[2] P. C. Teo and D. J. Heeger, “Perceptual image distortion,” in Proc. SPIE, vol. 2179, 1994, pp. 127-141.

[3] A. M. Eskicioglu and P. S. Fisher, “Image quality measures and their performance,” IEEE Trans. Commun., vol. 43, pp. 2959-2965, D

[4] M. P. Eckert and A. P. Bradley, “Perceptual quality metrics applied to still image compression,” Signal Processing, vol. 70, pp. 177-200, Nov. 1998.

[5] J. Illingworth and J. Kittler, “A survey of the Hough transform,” *Computer Vision, Graphics, and Image Processing* 44, no. 1 (1988): 87-116.

[6] D. H. Ballard, “Generalizing the Hough transform to detect arbitrary shapes,” *Pattern Recognition* 13, no. 2 (1981): 111-122.

[7] Z. Wang and A. C. Bovik, “A universal image quality index,” IEEE Signal Processing Letters, vol. 9, pp. 81-84, Mar. 2

[8] C. Harris and M. Stephens, “A combined corner and edge detector,” *Alvey Vision Conference* 15 (1988): 50.

[9] M. Trajkovic and M. Hedley, “Fast corner detection,” *Image and Vision Computing* 16, no. 2 (1998): 75-87.

[10] Z. Wang, A. C. Bovik, and L. Lu, “Why is image quality assessment so difficult,” in Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing Orlando, FL, vol. 4, May 2002, pp. 3313-3316.

[11] J. L. Mannos and D. J. Sakrison, “The effects of a visual fidelity criterion on the encoding of images,” IEEE Trans. Inform. Theory, vol. IT-4, pp. 525-536, 1974.

[12] M. Chandran and P. Newman, “Motion Estimation from Map Quality with Millimeter Wave Radar,” Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on (2006): 808-813.

[13] Manjari Chandran-Ramesh and Paul Newman, Assessing Map Quality using Conditional Random Fields, International Conference on Field and Service Robotics (FSR), Chamonix, July 2007.

[14] D. Wolter and L. J. Latecki, “Shape matching for robot mapping.” Proceedings of 8th Pacific Rim International Conference on Artificial Intelligence, Auckland, New Zealand, August (2004).

[15] D. G. Lowe, "Distinctive image features from scale-invariant key points", IJCV, vol. 2, no. 60, pp. 91-110, 2004.

[16] Ioana Varsadan, Andreas Birk, and Max Pfingsthorn, "Determining Map Quality through an Image Similarity Metric", RoboCup 2008: Robot WorldCup XII, L. Iocchi, H. Matsubara, A. Weitzenfeld, C. Zhou (Eds.) Lecture Notes in Artificial Intelligence (LNAI), Springer.

[17] Johannes Pellenz,Dietrich Paulus, “Mapping and Map Scoring at the RoboCupRescue Competition “ , Robotics Science and Systems Workshop, 2008.

[18] Gonzalez, R.C., R.E. Woods, S.L. Eddins, *Digital Image Processing Using MATLAB*, New Jersey, Prentice Hall, 2003, Chapter 11.

[19] C. J. van Rijsbergen (1979) Information Retrieval (London: Butterworths)

[20] Kondrak, G., Marcu, D. and Knight, K. (2003) "Cognates Can Improve Statistical Translation Models" in Proceedings of HLT-NAACL 2003: Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, pp. 46--48

Introduction to Biological Inspiration for Intelligent Systems

Gary Berg-Cross

Engineering, Management and Integration

Potomac, MD 20854

301-424-9256

Gary.berg-cross@em-i.com and gbergcross@gmail.com

Abstract

This paper serves as a short introduction for the special PerMIS session on Biological Inspiration (BI) for Intelligent Systems. The paper is organized into 4 parts. Part 1 provides a brief introduction to the idea and history of bio-inspiration for computation. Part 2 discusses the increased relevance of bio-understanding and presents a few examples of increased bio-understanding in cognitive areas. Part 3 provides examples of alternate focuses and levels of bio-reality that BI can take. Finally Part 4 notes some potential challenges in BI for intelligent systems.

Categories and Subject Descriptors

F.1 COMPUTATION BY ABSTRACT DEVICES, I.2.9 Robotics, K.2 HISTORY OF COMPUTING

General Terms

Performance, Design, Experimentation

Keywords

Biological Inspiration, Developmental Robotics, Intelligent Systems, Dual Cognitive Systems

1. INTRODUCTION

Biology has long "inspired" computation and cognitive models through examples and principles. A representative sampling from the origins of digital computation includes classic discussions in various disciplines involved in the study of cognitive systems, namely biology/neuroscience, developmental sciences, psychology, philosophy and artificial intelligence. Notable examples might start with Alan Turing, whose role in algorithmic computation and symbolic processing is well known. Turing is less known for using interaction as way of naturally educating "intelligent machines" (Turing, 1950). He pointed out that one might want to develop an intelligent machine by "simulating the child's mind" the way it develops interactively in nature. This is similar to Piaget's approach to understanding cognitive growth as multi-cellular organisms grow in complexity from relatively simple initial states. In this model a child's mind provides a relatively simple state that we might understand as a step to understanding the full range of natural computing. Another bio-model for computation is to relate it directly to the function of the nervous system and its hierarchy of sub-components. Thus, the way neurons assemble into computational networks has been a common source of designs starting with the famous example from research on how neural networks might support logical calculation (McCulloch & Pitts, 1943). This work develops the idea of a brain-computer analogy, something that "inspired" John von Neumann's subsequent work on the logical design of digital computers. Such brain analogies to automata were Also part of the Dartmouth Summer Research Project on Artificial Intelligence in 1956. However the bulk of work continuing into the 60s was less bio-inspired and viewed thinking as a form of symbolic computation. This moved computer science away from natural computing to a human engineering one where computation processes are "optimized" by an architect rather than selected from natural, evolutionary processes. This is not necessarily a fatally flawed approach since it is reasonable to believe that significant progress in symbolic computing can occur without the application of principles derived from the study of biology. Computation may follow some general principles like gases and organic tissue and need not always imitate biological processes or our idea of mental operations. On the other hand, it is not obvious that human engineers have an adequate understanding of problems and computational approaches to build-in solutions before a system is operating and can be gauged against and adapt to problems. Indeed, the human brain is currently the only known entity capable of exhibiting the type of general aptitudes & capabilities we target for many intelligent systems. These include some integrated form of capabilities such as adaptive/creative problem solving, language/communications processing, integrated planning, creative, and multi-level, goal directed learning. Nevertheless the computational path has followed a more abstract approach with computational class of machines technically referred to as 'automata'. Underlying the approach is that computation is effectively equivalent to the products of thinking and can be modeled within the domain of the mechanical rather than what we think we understand from the biological. Some doubts about this simple view have crept into the discussion recently.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

PerMIS'08 August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08...$5.00.

2. THE INCREASED RELEVANCE OF BIO-UNDERSTANDING

While there is an early history of inspiration from biology to computational system, the nature of this inspiration lessened as traditional algorithmic computational systems became the dominant example of computation. However this trend has slowed recently and biological inspiration seems increasingly relevant for two reasons. The first is due to broad advances in biological understanding and the second is the accumulation of problems with the traditional automata approach. There is much to say on the first point where we have recently experienced notable advances across a broad range of biological sciences. Some argue that we have reason to believe that "we are on the advent of being able to model or simulate biological systems multiple levels (Coates, 2007)." The way it works is shown in Figure 1. An increased understanding of bio-reality leads to better hypotheses to guide experimentation which turn can validate models of underlying mechanisms. Robotics researchers increasingly agree that models from biology and self-organization provide valuable insights for the

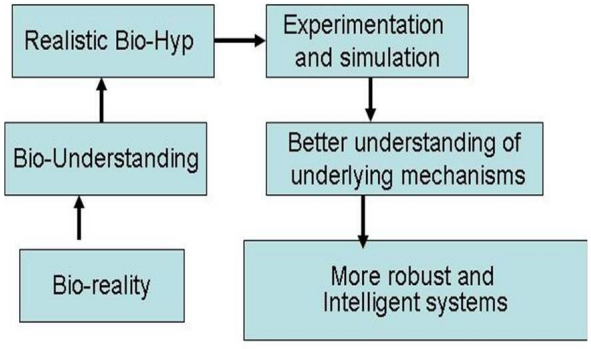


Figure 1: Bio-Reality to More Intelligent Systems

engineering of autonomous robots. These models may help to understand complicated and non-obvious underlying neural dynamics, which in turn can serve as basis for simulation, robotic experimentation etc. Taken as whole all of this can then be used to design and build more robust and intelligent systems. For example, improved bio-inspiration might lead to significant progress in computing by applying adaptive principles that are derived from the enhanced study of biology. For example, more abstractly biological models (e.g. ethological, neuro-physiological, functional & anatomical organization etc.), may suggest what data is critical to gather to validate both traditional computational models and bio-inspired cognitive models. Scientists studying artificial intelligence have traditionally separated physical behavior and sensory input, but recent work stresses a more biological hybrid approach (Berg-Cross, 2004). These can provide some understanding of deep relations between cognition, environment and embodiment. For example, experiments involving real and simulated robots suggest that the relationship between physical movement and sensory input can be important to developing intelligent systems (IS). Tests involving both real and simulated robots have shown that feedback between sensory input and body movement is crucial to successfully navigating the surrounding world. Understanding this relationship may help engineers build more adaptive and intelligent robotic systems. Among the more rigorous summaries discussing useful issues are Wooley and Lin (2005) and de Castro and Von Zuben (2004). Wooley and Lin (2005) discuss how a biological system may operate according to general principles that have applicability to our now traditional non-biological computing problems such as security and autonomy. By studying an appropriate biological system and its characteristics, research may develop an understanding of the relevant principles that can be used to handle a traditional computer problem. Among the notable characteristics biological systems exhibit and whose principles may be relevant are: Metabolism, Growth/Development, Reproduction, Adaptability, Autonomy/Self-maintenance, Self-repair, Reactivity, Evolution and Choice. Wooley and Lin (2005) note, for example, one value of considering biological models is that all biological organisms have some mechanisms for things like self-repair, a characteristic related to autonomy and of value to approaches to computer security:

> "in the sense that it classifies and eliminates pathogens and repairs itself by replacing damaged cells without the benefit of any centralized control mechanism. Given the growing security burden placed on today's computer systems and networks, it will be increasingly desirable for these system and networks to manage security problems with minimal human intervention." Wooley and Lin (2005)

This illustrates how biology can be relevant to computing as principles emerge directly from the study of biological phenomena. The second reason to look to the biological realm for inspiration is that traditional systems have proven fragile, difficult to maintain and subject to catastrophic failure. A bio-development hypothesis suggests why the engineering of Intelligent Systems sometimes fail. In biological systems the adaptive nature of behavior derives a very indirect relationship between the properties of the interacting elements and the emergent results of the interactions. Thus, as Nolfi et al (2008) says, "behavioral systems can hardly be designed while they can be effectively developed through self-organizing methods in which properties emerging from interactions can be discovered and retained through an adaptive process based on exploration and selection." In contrast to engineered systems, biological ones represent powerful cognitive architectures for managing complexity in a robust and sometimes even an elegant manner. An example of this is computer vision, where some visual tasks (e.g, segmentation and motion) aren't well handled perhaps due to the over simplified way classical models have been developed. To handle this gap, some current research lines are going back to more biologically "plausible" models of visual perception based on an understanding, modeling and simulation of the mechanisms observed in neural processes within the brain.

3. DIFFERENT DIRECTIONS & LEVELS FOR BIO-INSPIRATION

Intelligent robotics, like animals are behaving systems with sensors and effectors that easily be embedded in the physical world. Both have sensors and actuators and require an autonomous control system that enables them to successfully carry out various tasks in a complex, dynamic world. It is pretty natural to see the study of autonomous robots as analogous to the study of animal behavior (Dean 1998). But there are many variations on how to proceed from this broad analogy. For example in "Biorobotics" autonomous agents have been used as empirical models of simple behavioral patterns to study the influence of morphology on adaptive behavior. Alternatively the bio-characteristic of interaction with the environment can be combined with the idea of development to study how intelligence emerges. A panel at last year's PerMIS called "**Can the Development of Intelligent Robots be Benchmarked? Concepts and Issues from Developmental/Epigenetic Robotics"** (Berg-Cross et al 2007) made a start on describing emerging this subfield of AI. As described there developmental robotics (DR) studies how autonomous robots can learn to acquire behavior and knowledge on their own, based in their interactions with an "environment". A major benefit of this view of intelligent systems to that it removes engineer bias (Blank, Kumar, & Meeden 2002) and opens up the possibility of adaptive intelligence – a major topic at prior PerMIS workshops. Among other things the DR panel session discussed what primitive capabilities should be built into an intelligent system. Blank, Marshall and Meeden (Berg-Cross et al 2007) described their developmental algorithmic design using core ingredients of abstractions, anticipations, and self-motivations. This design is proposed to allow a mobile robot to incrementally progress from sensory-motor activities through levels of increasingly sophisticated behavior. This design attempts to answer the question of how to learn and represent increasingly complex behavior in a self-motivated, open-ended way.

Another direction BI takes is a more faithful biological realism of the central nervous system and in particular the brain. Recent work such DARPA's BICA program is illustrative of recent efforts to emphasize "Bio-neural Realism" and "develop integrated psychologically-based and neurobiology-based cognitive architectures that can simulate human cognition in a variety of situations." Among the Phase I projects perused in DARPA's BICA program was the development of cognitive theories that could map cognitive functions to neurological functions and provide an inspiration for the emerging field of developmental robotics. Figure 2 illustrates the two parts of such neuroscience related efforts to fully understand neural processes and their relations to mental processes. On the left are models of brain/CNS system reality and on the right are corresponding level of cognitive activities, from simple to complex that might map to the operations of some levels. As shown in the Figure CNS components span sizes at the molecular (such as DNA)/ synapse level to structures 10^5 larger. Corresponding time for functional activity can span the order of nanoseconds to seconds or longer. A neuroscience theory to simultaneously handel all these level would be a single bio-physical/chemical theory and obviously such a "theory of everything" does not currently exist. Nor can one expected one that integrates most of these levels in the near future. At this stage of computational modeling we can understand animal/human behavior by studying computer models of a subset of levels. For example, one current model of the brain suggests how neurons are organized into basic functional units and operate much like microcircuits in a computer. The model proposes that in the cortex, neurons are organized into these functional units are roughly cylindrical volumes 0.5 mm wide by 2 mm high, each containing about 10,000 neurons that are connected in an intricate but consistent way. Each microcircuit, known as the neocortical column is repeated millions of times across the cortex and the model can be tested by the larger computer systems currently available (IBM Blue Brain Project, 2005). It remains to be seen if this neural circuit-cortical model is correct and if our current hardware is up to simulating such large models. There are obviously many other biological issues to consider in modeling the CNS, but there are also many related issues to consider on the cognitive-behavioral side too. Some of these are summarized in the concluding section.

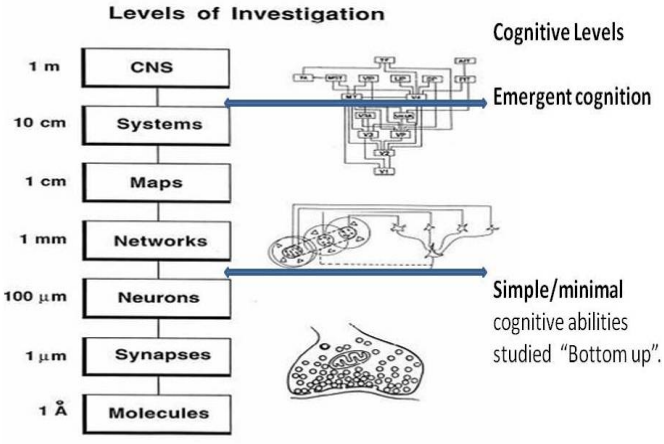


Figure 2: Biological and Cognitive Levels of Investigation (after Churchland and Sejnowski, 1992)

4. SOME CHALLENGES IN BIO-INSPIRATION OF INTELLIGENT SYSTEMS

There is considerably evidence that humans reasoning and cognition is based on two distinct processing systems with different evolutionary histories. Essentially the dual cognitive processes theory proposes that higher animal brains have two mental systems that compete in humans for control of reasoned actions. There is a range of experimental psychological evidence supporting this view, as summarized in Stanovich (2004), although this dichotomy may oversimplify the evolutionary nature of the divide (Evans 2008). For example, there is evidence for the presence of higher-level cognition in many species which Toates

(2006) argues may have developed into human consciousness. In the standard theory there are two cognitive systems. System 1[1] is the evolutionarily older one which is common to many animals and is characterized by stimulus-bound, heuristic processes that are fast, automatic, effortless and unconscious (Stanovich 1999). System 1 is thought to be made up of a set of relatively autonomous/modular subsystems that include both innate input modules and domain-specific knowledge that is constructed by one or more general learning mechanism. System 2[2] is evolutionarily recent with higher, more controlled processes that are slow, deliberative, and conscious – making them more characteristic of humans and other primates. System 2 permits abstract reasoning and hypothetical thinking, but is constrained by modular working memory capacity. Dual processing challenges classical ideas of monolithic control which seems important for characterizing the plasticity and robustness of human reasoning as a broad interplay between an automatic and belief-based system (Berg-Cross et al. 2007). But we have scant research on the interplay of System 1 and System 2 adaptive mechanisms "designed" by natural selection to solve the kinds of problems faced by human. Thus, we do not yet have adequate models of cross-linkage between specialized intelligences and those with wider reaching domain-generality. This is in part due to the fragmentary nature of the evolutionary data on which to base our theories. The harmonization of dual cognitive processes may have evolved through a still unknown, complex selection path. Developmental and epigenetic robotics may be one way to investigating a path that characterizes the robust interplay between an easily believed perception-based system and a more cognitively demanding logic-based reasoning system of human reasoning. This views intelligence as a developed phenomena that balances multiple reasoning mechanisms, together with scruffy modules of knowledge which learn to deal with situations that are only partial predictability, due to dynamics, and the absence of precisely defined states. Berg-Cross (2004, 2006) has suggested that a multi-level, hybrid architecture, based on a cognitively realistic foundation, could approximate human performance for this class of problems. To be practical such an architecture would build on the existing agent models, semantic web technology and standards, as well as a reasonably adequate knowledge and domain models.

In concluding it may be worth suggesting that designing intelligent systems based on slavish adherence to bio-cognitive "realism" at this level of understanding is likely to yield systems with strengths and weaknesses in an attempt to harmonize the two processes. Thus it may a dual cognitive process inspired approach to developing an IS could yield some greater ability to integrate information from many sources and effective coupling of "higher-level" cognition with perception. However, it may yield integration this with a relatively high error rate and slow responses compared to what is possible with digital processing time scales. Effectively large memory may be achieved using "experience" and a constructionist approach to memory that integrates "relevant" past experience with the present "situation". But we lack adequate theories to handle such distributed and quite fallible memory representations. For example we don't know how to handle interferences from broadly distributed memories. And human reasoning is often biased by stereotypical judgments (built in pre-judgments we call biased) that is influenced by a limited set of experiences. And we don't know how to balance the fast, modular a-logical/un-analytic processes that seem to be built in for typical situation prediction and understanding. In this regard we have a interesting road ahead which is well summarized by Evans (2006):

> "If the conscious, analytic system is at best only partially in control and in competition with not one but several implicit systems, how come everything works so well? Understanding how generally adaptive behavior can result from such an apparently chaotic cognitive architecture is one of the great challenges for cognitive science."

[1] System 1 is a neutral term. Other labels are experiential, heuristic, implicit and associative.

[2] Non neutral labels include: rational, analytic, systematic, explicit and rule-based.

5. REFERENCES

[1]. Barto, A; Singh, S.; and Chentanez, N. (2004). "Intrinsically motivated learning of hierarchical collections of skills. "In 3rd International Conference on Development and Learning.

[2]. Berg-Cross, G. (2004). "Developing Rational-Empirical Views of Intelligent Adaptive Behavior " Performance Metrics for Intelligent Systems. PerMIS Proceedings 2004.

[3]. Berg-Cross, G.; Fu W.; and Kwon A. (2004). "A Cognitive-based Agent Architecture for Autonomous Situation Analysis" Performance Metrics for Intelligent Systems. PerMIS Proceedings 2007.

[4]. Blank, D.; Kumar, D.; and Meeden, L. (2002). *Bringing up robot: Fundamental mechanisms for creating a self motivating, self-organizing architecture.* In Proc. of the Growing Up Artifacts that Live workshop at Simulation of Adaptive Behavior 2002.

[5]. Coates, Christopher; (2007); " The Air Force 'In Silico' -- Computational Biology in 2025, DTEC Report No.(s): AD-A474845; Nov 2007; 49 pp.de Castro, L. N. and Von Zuben, F. J. (2004) *Recent Developments in Biologically Inspired Computing*, , Idea Group Publishing. Dean J. (1998) "Animats and what they can tell us. " *Trends in Cognitive Sciences* 2(2):60–67.

[6]. Gunning D. (2005) "Biologically-Inspired Cognitive Architectures" (BICA),

[7]. http://www.darpa.mil/IPTO/programs/bica/bica_phase1.asp

[8]. IBM Blue Brain Project (2005), http://bluebrainproject.epfl.ch/

[9]. McCulloch, W. S. and Pitts, W. H. (1943). "A logical calculus of the ideas immanent in nervous activity. " *Bulletin of Mathematical Biophysics*, 5:115-133.

[10]. Nolfi, N., Ikegami,T. and Tani, J., "Behavior as a Complex Adaptive System: On the role of Self-Organization in the Development of Individual and Collective Behavior", *Adaptive Behavior*, Vol. 16, No. 2-3, 101-103 (2008)

[11]. Prince, C., Helder, N. A., and Hollich, G. J. (2005). "Ongoing emergence: A core concept in epigenetic robotics." In *Proceedings of the fifth international workshop Epigenetic Robotics*, pages 63–70.

[12]. Schlesinger, M., Amso, D. & Johnson S. (2007) *Simulating Infants' Gaze Patterns during the Development of Perceptual Completion.*, Proceedings of the Seventh International Conference on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems. *Lund University Cognitive Studies, 135*.

[13]. Stanovich, K. E. (1999). *Who is Rational? Studies of Individual Differences in Reasoning*. Mahway, NJ: Lawrence Elrbaum Associates.

[14]. Stanovich, K. E. (2004). *The robot's rebellion: Finding meaning the age of Darwin*. Chicago: Chicago University Press.

[15]. Toates, F., (2006) "A model of the hierarchy of behaviour, cognition, and consciousness.", *Consciousness and Cognition*, Mar;15(1):75-118.Torres-Huitzil, C., Girau B. and Castellanos-Sanchez, D., (2005) "Digital Implementation of a Bio-inspired Neural Model for Motion Estimation", International Joint conference on Neural Networks, IJCNN05.

[16]. Turing, A. (1950), 'Computing Machinery and Intelligence.', *Mind* 59(236), pp. 433–460.

[17]. Wooley, J. C., and Lin, H. S. (2005) "Biological Inspiration for Computing." In: Catalyzing Inquiry at the Interface of Computing and Biology, Computer Science and Telecommunications Board, The National Academies Press, John C. Wooley and Herbert S. Lin, editors.

Applying Developmental-Inspired Principles to the Field of Developmental Robotics

Gary Berg-Cross
Engineering, Management and Integration
Potomac, MD 20854
301-424-9256
Gary.berg-cross@em-i.com and gbergcross@gmail.com

ABSTRACT

Bio-inspired principles of development and evolution are a special part of the bio-models and principles that can be used to improve intelligent systems (IS). Such principles are central to cognitive developmental robotics (DR) and its effort to understand cognition by imitating development. DR approach takes inspiration from nature process so that engineered intelligent systems may create solutions to problems in way similar to what is believed to occur with biologics in their natural environment. This paper uses a three-level, bio-inspired framework to illustrate methodological issues in DR research. I stress the importance of using bio-realistic developmental principles to guide research keeping models and implementation separate to avoid the possible of falling into a Ptolemaic paradigm of endless tweaking of models. Several of Lungarella's design principles for developmental robotics are discussed as constraints on intelligence as it emerges from a ecologically balanced, three-way interactions between an agents' control systems, physical embodiment, and the external environment.

Categories and Subject Descriptors

F.1 COMPUTATION BY ABSTRACT DEVICES, I.2.9 Robotics, K.2 HISTORY OF COMPUTING.

General Terms

Performance, Design, Experimentation.

Keywords

Bio-inspiration, developmental robotics, bio-realism, Ptolemaic paradigm, embodiment.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

PerMIS'08 August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08...$5.00.

1. INTRODUCTION

As noted in Berg-Cross (2008) efforts to make synthetic/artificial systems more intelligent increasingly borrow from biologically-inspired models in part due to advanced understanding in the relevant domains. Indeed the bio-adaptive nature of intelligence in realistic environments has been discussed previously at PerMIS meetings and is well summarized by Freeman (2003):

> "Why do brains work this way? Animals and humans survive and flourish in an infinitely complex world despite having finite brains. Their mode of coping is to construct hypotheses in the form of neural activity patterns and test them by movements into the environment. All that they can know is the hypotheses they have constructed, tested, and either accepted or rejected. The same limitation is currently encountered in the failure of machines to function in environments that are not circumscribed and drastically reduced in complexity from the real world. Truly flexible and adaptive intelligence operating in realistic environments and cannot flourish without meaning."

Over the last few years embodied, development thinking has emerged in the form of a "new robotics" partly in response to lack of progress with the information processing paradigm which currently seems ill-suited to come to grips with natural, adaptive forms of intelligence. But part of the question on bio-inspiration for intelligent systems (IS) involves epistemological and methodological questions of the risks of such inspiration to improve our understanding and implementations. Following an introduction the development as an inspiration some of these general methodological issues are broached and some bio-developmental principles from the literature discussed as important constraints on unbridled influence.

2. DEVELOPMENTAL ROBOTICS METHODS

One sub-set of this work is developmental robotics (DR), with various names depending on the emphasis[1]. Developmental

[1] Terms besides Developmental Robotics include the more general BioRobotics, the more specific ones of Cognitive Developmental Robotics, Epigenetic Robots and Evolutionary Robotics

Robotics (DR) is an emerging experimental science inspired by increased understanding of the importance of a developmental stance . That is, DR uses "synthetic" experimental studies of development as a core idea e.g. Piagetian stage-theory processes involving prolonged epigenetic development to elucidate general mechanisms of intelligence development, starting with proposed cognitive development mechanisms. While the idea may not be new, using present day robotic technology embedding neural systems as building blocks we now have the unique opportunity to test challenging hypotheses representing complex principles of cognitive development in realistic environments. Such embedded, bio-adaptive nature of intelligence, responding to environmental challenges was recently discussed at a PerMIS Special Session on Epigenetic/ Developmental Robotics (Berg-Cross, 2007). The session discussed how research leverages our increased understanding of the mechanisms underlying development using computational models of development, cognitive architecture and neural models. The session included, as an example, the work of Blank, Marshall and Meeden (2002) who describe a developmental, algorithmic design using core ingredients of abstractions, anticipations, and self-motivations. This design is designed to allow a mobile robot to incrementally progress from sensory-motor activities through levels of increasingly sophisticated behavior and it also attempts to answer the question of how to learn and represent increasingly complex behavior in a self-motivated, open-ended way. Self motivation is hypothesized as an important link between "low level" cognitive phenomena, such as pattern formation and higher complexities (such as concept formation, plans, the emergence of simple syntactic categories, e.g. action names) are research topics. DR often takes such constructivist perspective to study how higher processes emerge via self-organize for general-purposes as skill and knowledge is learned from environmental interactions. The overall DR research program is synthetic and iterative: initially, robot technology is used to instantiate and investigate models originating from developmental sciences, and results eventually feeds back to new hypotheses about the nature of development and intelligence. The resulting improved models can then be used to construct better robotic systems by exploiting insights gained from deeper understanding of developmental mechanism. A major attraction of DR is this combination of development-inspired principles combined with progressive, empirical methods for validating derived hypotheses. A simple four part version of the methodology is summarized in Figure 1 starting with a cognitive developmental model based on "developmental sources" such as prior research and/or theory. To mimic development we also start with a realistic approach to some innate substrates (e.g.

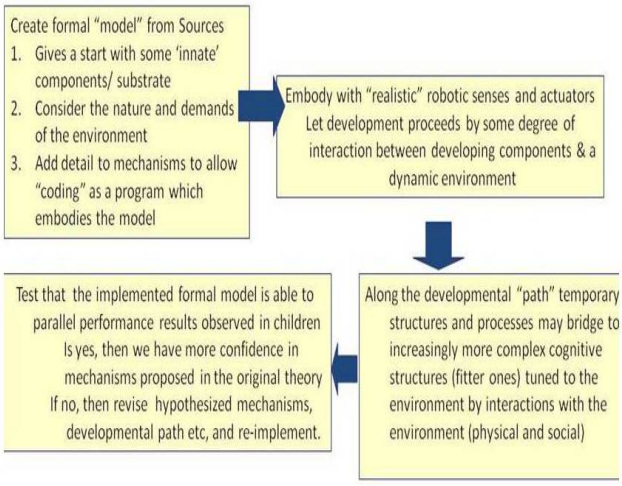


Figure 1: DR Approach Described in Four Steps

motivation, attention), a cognitive design that may afford emergence through interactions with some realistic environment. Model-based hypotheses can be empirically tested in a 2nd step by building a robot (embodiment) that is situated and can develop certain capabilities though interaction with the environment. One example is Piaget's hypothesis on the importance of sensory-motor interaction, staged competence learning and the sequential lifting of constraints over a developmental path. Such hypotheses suggest robotic "development" studies that can checked against behavioral stages (periods of growth and consolidation) followed by transitions (phases where new behavior patterns emerge) that we observe in children. The next section discusses some of the issues that influence design, modeling and experimentation within DR in an attempt to improve the overall process.

3. A SCIENTIFIC FRAMEWORK FOR DR

Generally, DR methods follow a basic hypothetical-deductive framework of a scientific method and its characteristic dual sources of inspirations from observations and prior theory and a central use of models. Webb (2001) provides a useful three-level, general frame (Figure 2) to discuss some of the bio-inspiration issues. We can approximately call the levels running from bottom to top bio-reality, model-based experimentation, and embodied implementation. Taken together they integrate hypotheses, models and robots as a scientific device to learn about & understand the "bio-reality" of developmental phenomena such as adaptive learning. It is worth noting in passing that the right side of the figure that compares target behavior, predicted behavior and the embodied behavior achieved as part of the actual experiment. This is often the topic of PerMIS papers and provides the general model-experimental context for these measurements.

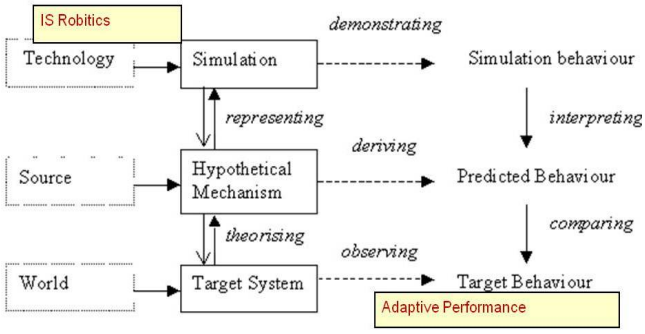


Figure 2: Method to Study "bio-reality" Phenomena such as Adaptive Control via Robots (after Ward, 2002)

We can understand the bio-modeling in many types of DR studies using this framework, starting at the lower level, which grounds itself in the reality of biological "target systems" that exist in the world. In DR studies are targets are typically human infants/children whose behavior is known for certain circumstances at certain developmental "stages". For example, there is the sequence seem from inaccurate reaching reflex to accurate visual target fixation and learning reaches that correspond to visual targets. It is commonly observed that visually guided reaching is the earliest "accurate" reaching behavior to occur. Infants spend notable time observing their hands around 12 weeks and by 15 to 20 weeks are performing "visually guided" reaching. This bottom level is experimentally related to the second (middle) level which is essentially part of the first step shown in Figure 1 where "sources" are used to set a hypothesized mechanism that can be tested via the robot mechanism experiments of steps 2 and the developmental structures of step 3. An example model at this level is Scassellati's (2001) shared attention model whose source was the more the general developmental theories of children's cognitive abilities (Baron-Cohen, 1995). In Scassellati's hypothesis there are modular processes such as "Eye Direction Detector" which can determine the angle of gaze of an individual, usually the parent, and extrapolate to find an object of gaze interest. Complementing this is an "Intentionality Detector" observes other's motion patterns and attributes the simple intentional states of desire and fear to animate objects. Finally a "Shared Attention Mechanism" produces representations of attentional states and allows a child to observe highly salient objects that are under consideration by the observing child. All of this can be implementation as shown in the upper level of Figure 3 by a robotic system, whose behavior now simulates the gaze behavior of young children.

Webb's (2001) framework nicely illustrates some of the problems that may arise in such DR efforts. One of them is that they sometimes attempt to test and validate a bio-mechanistic hypotheses based on particular models (e.g. models of neural circuits for particular regions of the brain) but then use related neural networks as the implementation method to test such models. That is Source models at level 2 and Implementation at level 3 are effectively the same and we have circular validation. Or they may "simulate" virtual implement an embodied robot rather than physically implement it and tweak parameters in the simulation to make it fit observation. This is hardly a real validation as the model at level 2 and the test at level 3 are both mathematical models and connection to the real world has been skipped. That is, there are problems with some circular aspects of how researchers may define source, mechanisms, implementations and how close performance must be and to what level of detail it is tested in some of the work. Of particular interest are questions that can be generated from Webb's (2001) framework such as:

- whether all "biases" are eliminated simply by letting a system develop from general capabilities since one still has to pick some innate substrate, relevant biological levels, and relevant environments.
    - There are many competing view of what is "real" vs. hypothesized concepts & principles e.g. sensori-motor schemas are the source of more abstract schemas.
- Related to this is the issue of relevance & realism: do models test and generate hypotheses that are bio-developmentally applicable?
- When collections of distributed, specialized neural models are the source of inspiration multiple "hierarchical units" need to be modeled - neuron to nets to circuits to brains.
- Generality: what range of bio-developmental systems & phenomena can the model represent?
    - If adaptive mechanisms are included in the model, there may be long chains of environmental interactions that may be needed to carry through a realistic experiment.

Some of these issues interact with the issue of performance, for example, the issue of levels. We have trouble comparing target, expected and actual performance since the amount of detail to do this starting at the lowest "level" exceeds not only our ability to build such distributed specialized networks but even to simulate them. And the physical levels have correspondingly different time scales on which they work, so integrating these scales within a robot implementation is explicitly handled Lungarella (2004).

Performance definition is a central issue as Webb (2001) notes, arguing that we need to consider more than a naive match of behavior. When a direct comparison of the development of higher processes is attempted, the possible variability going into the match between various behaviors is considerable. Researchers have to consider whether the behaviors need to be identical (indistinguishable by some interpretive criteria) or merely similar (again by some interpretive criteria). Indeed Deakin (1990) argues that a behavioral match is <u>never</u> sufficient evidence for drawing conclusions about the accuracy or relevance of a model.

This seems to be particularly true when the target is adaptive behavior which may respond to chains of small changes in the environment where time is important. There are an enormous range and number of variables to consider. An overall problem is that researches may "finesse" these problems in various ways such simplifying performance accuracy assumptions, tweaking simulations etc. (Webb, 2001). It may make the models and implementations take on a Ptolemaic character (Gary Berg-Cross, 2003). By that I mean that the current state of work (in both IS and related, if simpler, problem fields) seems a bit like a Ptolemaic paradigm. As noted above mathematical models can be used as sources in DR. Such models are a bit Ptolemaic in that they break complex intelligent behavior into known neat, related components that are bit analogous to Ptolemy's "perfect circular motions". It is a system of step by step complexity build by cycles/epicycles with certain perfect cognitive

process/mechanisms as the functional/circular primitives. Bio-inspired to begin with we accept heuristic devices for their practical computational usefulness and ease of implementation. Such core functions are added to in an ad hoc (rather than bio-development constrained way) as required to obtain any desired degree of performance and accuracy. Using large combinations of constructions we are able to measure performance in some small problem domain for some of the agreed upon intelligent behaviors within the standards of observational accuracy. The ultimate concern is that, like a Ptolemaic system, when we encounter anomalies, parameters can all-to -easily be modified to account for them. Thus the model becomes more an after-the-event description then a deep capture of the underlying problem. Such descriptions may be very useful, making it possible to economically summarize a great amount of brute observational data, and to produce empirical predictions, but they often prove to be brittle, do not scale and might not be fruitful to further predictive IS research. The field seems to understand this potential downside of some of these issues and has begun to develop some principles to constrain and guide the work. In the next section we consider some of Lungarella's principles (2004) that can help, behind the scenes, adding realistic constraints as part of DR experimentation.

4. DEVELOPMENTAL PRINCIPLES WITHIN AN EMBODIED, INTERACTIVE MODEL

Lungarella (2004) has outlined a broad set of bio-development principles that DR can employ to guide research. He proposes that we embed these within a developmental framework the interactive coupling of three things - control, body, and environment as shown in Figure 4. This coupling reflects principles of ecological balance between the three factors. Such balancing of embodied intelligent systems includes adaptation to natural constraints from gravity, friction, energy limitations, living with damage, etc.

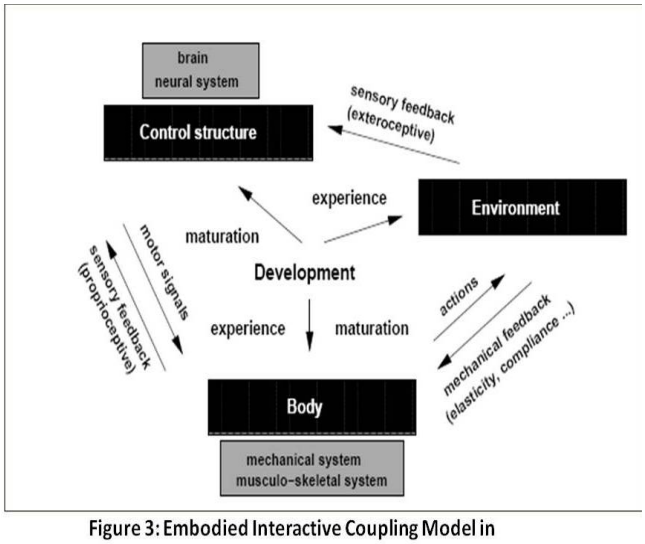


Figure 3: Embodied Interactive Coupling Model in a Developmental Framework (Lungarella 2004)

This reflects what Lungarella (2004) calls the principle of "cheap design" which means that the design of a developmental agent must be parsimonious, and must opportunistically exploit the physics of the system-environment interaction, as well as the constraints of an agent's ecological niche. Building on this is the principle of developmental "design for emergence" of higher cognition. This principle says that when engineering agents one should not design them completely for target behavior, but instead design should endowed an agent with more general ability that will let higher abilities emerge. For example, as in Blank et al (2002) they should self-direct the exploration of their own sensory-motor capabilities, and have some realistic capabilities to enhance their limited initial behavioral repertoire, and acquire a cognitive-experiential history. A related principle is that of "ecological balance", which can be understood in terms the 3 part coupling model. For balance an agent's complexity or behavioral diversity has to match the complexity of the environment as measured against the agent's sensory apparatus. Thus when designing or developing an agent for a particular task environment, a balance is required between the complexity of the sensor, motor, and control system. This happens naturally in natural environment through selection, but has to be through in design - a difficult problem since the coupling of the three elements is not obvious. This happens more readily in nature as understood by Lungarella's "value principle", which provides a substrate repertoire of motivated learning & adaptive behavior. A motivation substrate is needed for a developmental process to take place and for an autonomous agent to behave adaptively in the real world, along with a set of mechanisms for self-supervised learning. This substrate provides values that shape the development of the agent's control and bodily structure. Principles for the embodiment add important constraints since embodiment implies far than just some limiting physical constraints to an organism. embodiment is a challenge that actively supports and promotes intelligent information processing. It encourages agents to exploit the dynamics of the interaction between an embodied system and their environment. It thus directly supports the selection and processing of information by coupling with the agent control systems and interaction within the environment (Pfeifer and Bongard, 2006). They argue that " artificial evolution together with morphogenesis is not only 'nice to have' but is in fact a necessary tool for designing embodied agents." This coupling is different from non-biological views of computation in that it favors the developmental emergence of stable computational patterns, including adaptivity and robustness against changing environments and indeed the corporal changes of development or damage. Lungarella (2004) has formalized how embodiment affects agent processing as the principle of "information self-structuring" (active perception). As part of sensory activity an agent is not passively exposed to information from its surrounding environment. Due to its particular morphology, and through its actions on the environment; it actively structures, selects, and exploits such information. This reflects what Lungarella (2004) calls the "starting simple" principle for development. We don't start with a high level model but an unfolding hybrid with a gradual and well-balanced increase of both the agent's internal complexity (perceptual and motor) and its external complexity (regulated by the task environment or an instructor). This speeds up the learning of tasks and the acquisition of new skills, compared to an agent that is complex from the onset. From this perspective, behavior informs and shapes cognition as it is the outcome of the dynamic interplay of physical and information theoretic processes, and not the end result of a control process (computation) that can be understood at any single level of analysis. Instead there emerges an emphasis on a general understanding of cognitive aspects.

5. SUMMARY

I have discussed some of the bio-developmental thinking that inspires work to understand and develop more intelligent systems. A major benefit argued for a DR view of intelligent systems (IS) to that it removes engineer bias (Blank, Kumar, & Meeden 2002) inherent in artificial design. There is a good deal to this criticism, based on problems we see with engineered systems. On the other hand bio-inspiration might be considered to have some biases of its own based on chance events along the evolutionary path. In addition DR research is still influences by interpretations of reality and selection among the complexity of bio-reality. For example researchers may select some levels of the neural system to emulate and some design selections are driven by the sheer practicality of the time scales involved in phenomena as outlined in Berg-Cross (2008). I have noted a number of research issues in such inspiration including the, challenge of integrating levels of bio-reality and cognitive levels; the mixing of models with implementation especially when mathematical simulations are used; and problems with handling the degree of behavioral similarity that constitutes validation of hypotheses. Indeed there remains a serious difference of opinion on how to proceed based on models. One group which I'll call "neat formalists" believes in centering on models as expressed by Barto (1991) that "the complexity of animal behavior demands the application of powerful theoretical frameworks" (and "nervous systems are simply too complex to be understood without the quantitative approach that modeling provides"). Another group, that I'll call scruffy empiricists, are represented by Croon and van de Vijver (1994) who are uncomfortable with "developing formalized models for phenomena which are not even understood on an elementary level is a risky venture: what can be gained by casting some quite gratuitous assumptions about particular phenomena in a mathematical form?"

Advice for the current stage of work DR researchers might be follow such principles rather than trying to tweak particular results against target behavior. That is, they should be concerned with: "building a complete, but possibly rough or inaccurate model, than with strict accuracy per se" (Webb, 2002). This would be a complete system that connects action and sensing to achieve a task in an environment, even if this limits the individual accuracy of particular parts of the model because of necessary substitutions, interpolations etc. . The direction proposed herein is not to slavishly follow bio-inspiration. Rather we should proceed with a developmental approach using developmental principles based on our understanding of how intelligence develops. A final note is an observation that I add as Gary's principle. It's based on the observation of the large, interdisciplinary nature of the teams now working on bio-inspired robots which stand in some contrast to the simple application development teams of the past. Due to the inherent complexity of cognitive systems, interdisciplinary teams are required where biological, theoretic and robotic systems are studied at different levels of granularity. This suggestion is simply that successful developments will require a considerable, explicit expertise in the development team to understand what is implicit in the agent to be developed.

6. REFERENCES

[1]. Baron-Cohen, S.. Mindblindness MIT Press, 1995.

[2]. Berg-Cross, G. (2003), "A Pragmatic Approach to Discussing Intelligence in Systems", PerMIS Proceedings 2003.

[3]. Berg-Cross, G. (2007), Panel discussion "Can the Development of Intelligent Robots be Benchmarked? Concepts and Issues from Epigenetic Robotics", PerMIS Proceedings 2007.

[4]. Berg-Cross, G. (2008) "Introduction to biological inspirations for intelligent systems", PerMIS Proceedings 2008.

[5]. Croon, M. A. & van de Vijver, F. J. R. (1994) Introduction. In: Viability of mathematical models in the social and behavioural science, ed. M. A. Croon & F. J. R. van de Vijver. Swets and Zeitlinger.

[6]. Deakin, M. (2000) "Modelling biological systems." In: Dynamics of complex interconnected biological systems", ed. T. L. Vincent, A. I. Mees & L. S. Jennings. Birkhauser.

[7]. Freeman W. (2002), "On Communicating with Semantic Machines", PerMIS Proceedings, 2002.

[8]. Lee, M., Meng Q. and Chaom F. "Developmental Robotics from Developmental Psychology"

[9]. Lungarella., M. (2004) "Exploring Principles Toward a Developmental Theory of Embodied Artificial Intelligence", Ph.D. Disertation, Zurich. http://www.ifi.uzh.ch/ailab/people/lunga/Download/PhDThesis/thesis040504_complete.pdf

[10]. Pfeifer R, and Bongard, J. C. (2006) How the body shapes the way we think—A new view of intelligence. Cambridge (Massachusetts): MIT Press.

[11]. Scassellati (2001) "Foundations for a Theory of Mind for a Humanoid Robot" Ph.D. Dissertaion, MIT, http://www.cs.yale.edu/homes/scaz/papers/scassellati-phd.pdf.

[12]. Webb, B. (2001) "Can robots make good models of biological behaviour?" BEHAVIORAL AND BRAIN SCIENCES (2001) 24, 1033–1050.

Overcoming Barriers to Wider Adoption of Mobile Telerobotic Surgery: Engineering, Clinical and Business Challenges

Gerald R. Moses, PhD
MASTRI Center, University of Maryland Medical Center
22 S. Greene Street
Baltimore, MD 21201
410-328-8431
gmoses@smail.umaryland.edu

Charles R. Doarn, MBA
University of Cincinnati Advanced Center for Telemedicine and Surgical Innovation
231 Albert Sabin Way
Cincinnati, Ohio 45267
513 558 6148
charles.doarn@uc.edu

Blake Hannaford, PhD & Jacob Rosen, PhD
Biorobotics Laboratory, University of Washington
Dept. of Electrical Engineering
Seattle, Washington
206 543 2197
http://brl.ee.washington.edu

ABSTRACT

A portable robotic telesurgery network could remove the geographic disparity of surgical care and provide expert surgical support for first responders to traumatic injury. This is particularly relevant to battlefield medicine where surgical intervention is currently not available to the most perilous fighting circumstances. Similar utility applies to the peacetime healthcare mission. The authors identify the potential advantage to healthcare from a mobile robotic telesurgery system and specify barriers to the employability and acceptance of such a system. This presentation will describe a collaborative effort to design and develop one or more portable robotic systems for telesurgery and develop those systems through successful animal trials. Recent advances in engineering, computer science and clinical technologies have enabled prototypes of portable robotic surgical platforms. Specific challenges remain before a working platform is suitable for animal trials, such as the inclusion of image-guidance and automated tasks

Other barriers to the development of mobile robotic surgical platform will be described. These include technical challenges of refinement of robotic surgical platforms, reduction of weight, cube, complexity and cost, and expansion of applications of technology to several procedures. Clinical challenges involve the protection of patient rights and safety, selection of surgical procedures appropriate for the system, the application of surgical skill to evaluate hardware and the application of surgical lore to software programs. Finally, business challenges include resolution of intellectual property considerations, legal liability aspects of telesurgery, patient safety and HIPPA, reimbursement and insurance issues, FDA approval of the final product and development of a commercialization plan.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
Copyright 2008 ACM 978-1-60558-293-1…$5.00.

Keywords

Surgery, robotic telesurgery, access to surgical care.

1. INTRODUCTION

The past decade has witnessed the growth of both interest in and capability of telesurgery systems. These developments are based in great measure upon enabling of technologies related to engineering, computer science, robotics, telecommunications, medical informatics, and surgery [1]. Barriers to the conduct of telesurgery have been identified and to some extent ameliorated [2]. The physical barriers of latency, visual discrepancy, round-trip delay, jitter and limited bandwidth have been studied and measured as quality of service indicators [3,4]. Additional challenges include strong business cases for wider adoption in clinical practice.

Various demonstrations have shown the possibility and potential of telerobotic systems; including the conduct of the first transcontinental telesurgery in the United States, the conduct of collaborative experiments with NASA within the NASA Extreme Environments Mission Operations (NEEMO) program, the refinement of prototype microsurgery equipment as a model for portable surgery systems, robotic laser tissue welding, robotic replacement for surgical scrub technicians, control of time-delayed telesurgery, and the use of high altitude platforms for transmission of telesurgery signals [5,6,7,8].

2. THE CHALLENGE OF FUNDING

Various working groups have identified the lack of multi/interdisciplinary collaboration as a barrier that could be overcome by funding specifically targeted to improve interdisciplinary research, design and commercialization. Failure to resolve intellectual property issues could impair and potentially stop development of robotic surgery. The lengthy and costly IP battle waged between Computer Motion and Intuitive Surgical exemplifies how intellectual property litigation can consume resources that could be used for further system development.

It was estimated by a Defense Department working group that funding in excess of $380M would be required to advance robotic surgical assistants to the point of "crossing the chasm" into early acceptance, from the perspective that we are now at the stage of "early adopters". This places the effort in the realm of "Grand Challenges" on par with the National Nanotechnology Initiative, where many believe it rightly belongs. Early reports do demonstrate that these surgical robots can allow the performance of safer, faster surgery, but the technology is tightly bound to economies of scale as long as the current designs and poor business practices are utilized.

The primary hurdles that need to be overcome in order to even begin addressing the roadmap include funding; the resistance of funding agencies to fund, and academia to support, large-scale, distributed, multidisciplinary teams; the culture and communication barriers between the disparate groups that would need to collaborate; industry's resistance to open architectures and large-scale collaboration. The "Grand Challenge" of developing surgical robotics should begin with a "grand" meeting where the relevant technologies, their current state, and the roadmap are described. Any future effort to plan a strategic roadmap for telesurgery research funding must engage all of the stockholders. Relevant federal agencies such as NIBIB, NIH, NSF, FDA, NIST, stakeholders from industry, academia and professional and standards organizations would need to join the community of clinicians and scientists who seek to advance telesurgery as a vital force in the provision of health care.

3. CURRENT STATUS OF TELESURGERY

For nearly a decade, the Department of Defense medical research agencies have explored the development of a portable robotic telesurgery system. Both the Defense Advanced Research Projects Agency (DARPA) and the Telemedicine and Advanced Technology Research Center (TATRC) have invested considerable resources in this investigation. The DARPA Trauma Pod project demonstrated that a single surgeon located outside of the operating environment could control multiple surgical assistant robotic functions. TATRC's support to several projects demonstrated that live animal telesurgery could be performed safely and effectively over the Internet, that telesurgical manipulation could be accomplished across great distances and into extreme environments and that alternative modes of transmitting signals could support robotic telesurgical tasks and movements.

The development of the current M7 "Army Arm" for robotic telesurgery demonstrated the possibility of a portable robotic surgery system. This system was constructed by SRI International from an earlier ophthalmic telepresence microsurgery model.

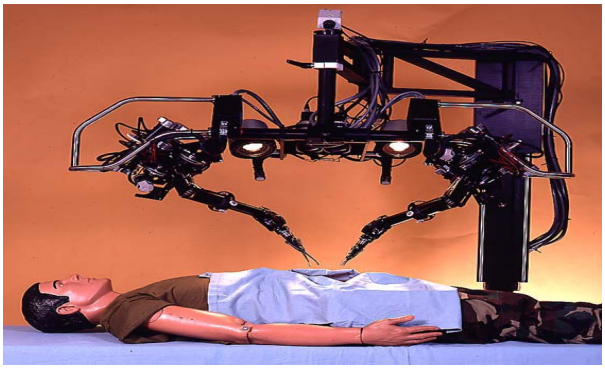


Figure 1. The M7 Surgical Robotic System

Similarly, progress was made in a robotic C-Arm system, named RAVEN, by the University of Washington HIT Lab. The development of a surgical robotic arm by MD Robotics in Toronto, Canada rounded out a significant North American effort to expand capabilities in the realm of medical robotics for telesurgery. MD had proven success as a company that develops robotic arms for space exploration, and had applied space-based robotic lore to produce a prototype single-arm system.

Figure 2. The RAVEN robotic surgery system.

In addition to the engineering and computer science related challenges, there are non-technical challenges that impede the implementation of a mobile telesurgery system. There is a need for industry and political arena representatives to establish a nation-wide communications network capable of supporting telesurgery. Also, licensure, privacy, liability and third-party payer issues must be addressed to enable wider adoption of telecollaboration and its effective use. Robust business models will offer greater adoption across a broader spectrum.

Given the current nature of support to mobile robotics and the reduction in DOD funding there has been slowdown in development of mobile robotic surgery systems. What is needed now is clarification of DIRECTION of research efforts and restoration of momentum for research and development.

4. POSSIBILITIES AND DIRECTION

Our presentation will address the identification and description of challenges to the development of portable telerobotic surgery, then articulate specific actions to overcome those challenges. The long-term goal of this and related research is the development of a portable telerobotic surgery system, capable of permitting surgical intervention across distances and into extreme and remote environments. That goal will likely be attained by the engagement of a multidisciplinary team of engineers, surgeons, computer scientists and informaticists to build on developing technologies to create the portable telerobotic surgery system. The multidisciplinary team would then need to address the technical challenges of quality of service over a telecommunication system and other systems to reduce latency and jitter of the signal. It is anticipated that animal trials would be conducted, leading up to a human surgical event within five years. The ultimate goal of the project is to enhance patient safety and quality of care by way of distributed surgical intervention.

Because the problems related to the development of a portable telesurgical system are many and varied, the contributions of collaborators can take many forms. There is the singular challenge of developing a robotic surgery system that is considerably smaller and lighter than current, commercial hospital-based systems. The reduction in weight and cube must be accomplished without degradation of efficiency and accuracy of robotically-controlled aspects. Once a master-slave system has been developed, the two parts must be connected over a secure, efficient and unfailing transmission link, which guarantees robustness, reliability, redundancy, minimal or zero latency, and quality of service. Various transmission modes have been explored, including the use of the Internet with enhanced connectivity to ensure quality of signal and an airborne platform capable of "bouncing" signals across distances.

Before this endeavor of developing a portable robotic telesurgery system can be attempted, a roadmap for its attainment must be drawn. That roadmap should identify the currently known obstacles to success and predict some likely obstacles to be encountered. The purpose of this presentation is to identify the barriers to success and chart alternative routes to the long-term goal.

5. A ROADMAP FOR RESEARCH

We propose to develop a comprehensive strategic roadmap for investment of research dollars for the development of a portable robotic telesurgery system, one that leads to animal trials within one year of completion. Our plan of work is to convene a conference of recognized experts in robotic telesurgery with the objective of crafting specific guidance for development, testing and refinement of a robotic telesurgery system. We predict that several physical barriers to telesurgery will be identified; their characteristics will need to be defined. These include control latency, visual discrepancy, round-trip delay, jitter, bandwidth and quality of service. Our plan is to understand the nature of these barriers and develop solutions to them. We anticipate also that non-physical barriers will be identified and hypothesize that they will include the need for a refined nation-wide communications network, resolution of privacy and confidentiality issues, acceptance of a broad license-to-practice, clarification of legal and regulatory matters and creation of an electronic medical record

The methodology of the study will begin with an extensive review of related research and background literature related to robotic telesurgery. Findings from recent experiments throughout the world will be considered in framing the research and development challenge. We will conduct discussions with scientists, surgeons, and engineers who understand robotics, telecommunications, and surgery, then expand our investigation to include related factors of intellectual property, device development as approved by the FDA, medical reimbursement, security and confidentiality of data, and patient safety.

We will obtain information and guidance from experts in two ways; first by means of convening a strategic planning conference; second by direct interview. The conference will address developments in robotic form, factor and function; effector structure and function requirements; controller ergonomics and haptics; integration with imaging; quality of service in signal transmission; and preparation for testing a prototype robotic telesurgery system by way of animal trials. Individual direct interviews will expand on the topics of establishing and maintaining intellectual property, resolution of business practices and patent registration arising from collaborative product development, informing the FDA of plans for both product development and animal trials, conducting discussions with CMMS and third-party payers regarding reimbursement of eventual health care services delivered by way of telesurgery.

The deliverable from the research effort will be a comprehensive report with specific targets and timelines for research and development leading to the establishment of a functional and safe robotic telesurgery system.

6. REFERENCES

[1] Cleary K and Mun SK, "OR 2020, The Operating Room of the Future", Workshop Report, March 2004.

[2] Curley K and Moses G, "Innovation by Committee" Reigniting Surgical Robotics Research and Development", Surgical Innovation, December, 2005, 12(4), 291-95.

[3] Curley K, Broderick T , Marchessault R, Moses G, Taylor R, Grundfest W, Hanly E, Miller B, Gallagher A, Marohn M., "An Integrated Research Team (IRT) meeting for Surgical Robotics – The Next Steps, TATRC Report, No. 04-03, 2005, January.

[4] Doarn C, Broderick T, Harnett B, Marchessault, Moses G and Anvari M, "Telesurgery and Telerobotics: Where we Need to Focus Research" Meeting Summary, 2004, May.

[5] Hanly E, Marohn M, Schenkman N, Miller B, Moses G, Marchessault R and Broderick, T, "Dynamics and

Organizations of Telesurgery", European Surgery, Vol 37, No. 5, Oct, 2005.

[6] Hanly RJ, Miller BE, Herman BC, Marohn MR, Broderick TJ, Shih SP, Sterbis, JR, Doarn CR, Harnett B, Hasser CJ, Talamini MA, Moses GR, Marchessault R. Schenkman NS. Remote Robotic Surgery Using the Public Internet: Cheap Transcontinental Telesurgery. 11th Annual Meeting and Exposition of the American Telemedicine Association, San Diego, CA. May 2006. *Telemed and E Health* 2006; 12(2):223.

[7] Hoffman L, Doarn CR, Harnett B, Moses G, Broderick TJ. "Recent Advances in Robotic Telesurgery and Applications to Battlefield Trauma", 2nd Annual Meeting of the Association of Academic Surgeons. Phoenix, AZ. February 2007.

[8] Harnett BM, Doarn CR, Rosen J, Hannaford B, Broderick TJ. Evaluation of Unmanned Airborne Vehicles and Mobile Robotic Telesurgery in an Extreme Environment. Telemed J E Health 2007; 14(6):537-42.

Calibration of a Computer Assisted Orthopedic Hip Surgery Phantom

Daniel Sawyer
NIST
100 Bureau Drive
Gaithersburg, MD20899
(301) 975-5863
dssawyer@nist.gov

Nick Dagalakis
NIST
100 Bureau Drive
Gaithersburg, MD20899
(301) 975-5845
dagalaki@nist.gov

Craig Shakarji
NIST
100 Bureau Drive
Gaithersburg, MD20899
(301) 975-3545
shakarji@nist.gov

Yong Sik Kim
NIST
100 Bureau Drive
Gaithersburg, MD20899
(301) 975-8081
yong@nist.gov

ABSTRACT
Orthopedic surgeons have identified a need for calibration artifacts (phantoms) to establish the traceability (to the SI unit of length) of measurements performed with Computer Assisted Orthopedic Surgery (CAOS) systems. These phantoms must be lightweight, easy to transport and simple to use. In collaboration with medical professionals, National Institute of Standards and Technology (NIST) researchers have developed a family of novel CAOS phantoms designed to meet the metrology needs of this critical segment of the US healthcare industry. The phantoms function as a surrogate hip joint and pelvis and can be measured with CAOS systems using the same technique employed to measure a patient's hip joint and pelvis for replacement surgery. The phantoms contain a mechanical ball joint, which functions as a substitute for a patient's hip joint and small holes, referred to as target holes, for receiving the measuring probe of CAOS systems. The location of the mechanical ball joint and the relative positions of the target holes are measured using the CAOS system and the results compared to the known values for these quantities. The results of this comparison are then used to verify the CAOS system performance specifications. In order to determine the known values for the critical dimensions of the phantom, the mechanical ball joint location and target hole positions are measured with a Coordinate Measuring Machine [1] (CMM), which is more accurate than CAOS Systems. This paper will report on the calibration of the NIST prototype phantom using a CMM and simulation tools at NIST.

This paper is authored by employees of the United States Government and is in the public domain.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

Keywords
computer assisted surgery, computer assisted orthopedic surgery, hip arthroplasty, phantom, calibration.

1. INTRODUCTION
The NIST operating room artifact (known as the phantom)[2][3] is designed to evaluate the measurement performance of three-dimensional coordinate measuring instruments that are used for Computer Assisted Orthopedic Surgery (CAOS). The intent of this report is not to describe the development and use of the phantom. (See [2] and [3].) Instead, this report provides a description of the critical features of the NIST phantom with only enough details about its construction to clearly describe the calibrated dimensions and their general intent. Further, this document will not provide the values of the results of the measurements. Such information is not part of the description of the measurement procedures. Finally, a description of a novel tool for calculating the task specific measurement uncertainty for all of the measurements performed is provided [4].

The phantom is comprised of two independent parts. The base is the primary support structure and is the "L" shaped part depicted on the right-hand side of figure 1 and shown in the photograph of figure 2. The femur bar is removable and connected to the base via a magnetic ball socket; it has kinematic constraints similar as to the human femur bone relative to the pelvis, which is depicted on the left-hand side of figure 1.

The base and femur bar are constructed of INVAR with a coefficient of thermal expansion of 1.30 $\mu m \cdot m^{-1} \cdot {}^{\circ}C^{-1}$ and include three types of calibrated features. First, there are small holes (approximately 1 mm in diameter and 1.4 mm in depth) on the base of the phantom, referred to as target holes. Collectively, these target holes function as a three-dimensional point coordinate artifact. The base has target holes aligned along two nominally orthogonal axes and additional target holes aligned along a semicircular path about the intersection of these axes.

(See figure 3.) The three-dimensional coordinates of these target holes are provided as part of the phantom calibration and when referenced to the appropriate coordinate frame (see the next section) these known values can be used to evaluate the performance of CAOS systems in point coordinate measurements.

Second, a stainless steel magnetic kinematic ball nest is bolted to the base; it is designed to firmly hold a highly spherical 38.1 mm diameter stainless steel sphere. The coordinates of the center location of the sphere placed in the kinematic nest, along with its measured diameter, are included in the calibration report. This kinematic ball nest functions as a surrogate hip socket. When fitted with the sphere on the femur bar, CAOS systems can measure the ball center location in a manner consistent with operating room procedures. The calibrated value for the ball center can then be used to evaluate the performance of the CAOS system in determining the location of the center of rotation. This is similar in practice to determining the center of rotation of a patient's hip joint.

Finally, the angles between three sets of planes are also provided. CAOS systems are often configured with a supplemental cutting blade and spatula tools. When these tools

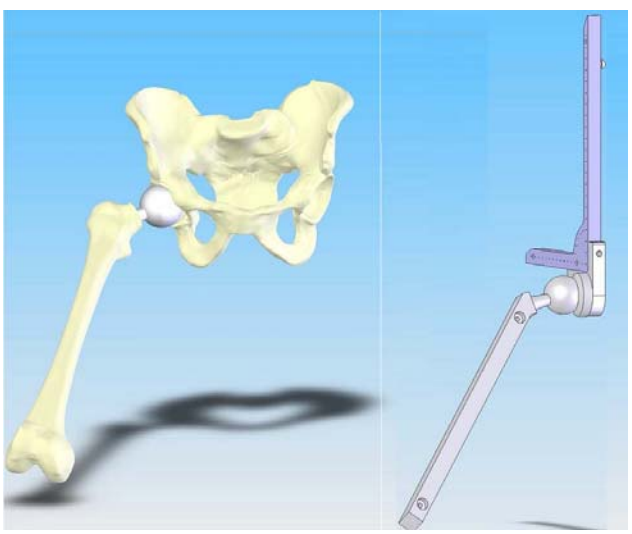


Figure 1. The image on the left is a depiction of a patient pelvis along with a femur bone and an artificial ball joint. Comparing this to the NIST phantom and femur bar on the right side of the figure, it is easy to recognize the motivation for the construction of the measurement artifact.

are present, the sides of the tools will be brought into contact with these surfaces and their angular orientation recorded by CAOS systems. The calibrated value for the angles between each set of planes can be compared to the CAOS measured angles to determine the tool angular positioning performance.
In all cases, the measured coordinates provided as part of the calibration report are relative to a common coordinate frame. The details for establishing this common or part-coordinate frame and the method employed to inspect each of the calibrated features are included in the applicable section of this paper.

Calibration of the phantom features was performed using a Coordinate Measuring Machine (CMM) with active part temperature compensation and the task specific measurement uncertainty was evaluated using Pundit/CMM software[1] [3]—which incorporates a measurement uncertainty tool developed at NIST [4]—and following NIST Technical Note 1297, *Guidelines for Evaluating and Expressing the Uncertainty of NIST Measurement Results.*

2. PART COORDINATE FRAME

Referring to figure 2, the part-coordinate frame origin is located at the center of the bottom plane of the target hole labeled *Origin*. The X-axis is defined as the line that passes through the origin and the bottom center of the target hole labeled *X20*. The X-Y plane contains the X-axis and the bottom center of target hole Y15. And finally, the Y-axis lies nominally along the direction of the line from the origin through the bottom center of target hole Y15. It's important to note that the coordinate frame as drawn in figure 2 is misleading as the part-coordinate frame's X- and Y-axes lie along the bottom of the target holes and, consequently, the part coordinate frame's X- and Y-axes are inside the part.

3. TARGET HOLE MEASUREMENTS

During calibration, the phantom is positioned with the X- and Y-axes in the X-Y plane of the NIST Resource Engineering Incorporated[1] (REI) CMM. This class of CMM is a standard commercial grade CMM. (Volumetric performance number [6] is approximately 10 µm.) The center of each target hole is designated by its X and Y coordinates. These centers are determined by measuring 4 points distributed along a circular path around the interior of the cylindrical bore that forms the top two thirds of the target hole. The Z coordinates are then obtained by probing vertically, towards the bottom of the target hole, with the probe positioned in the center of the target hole. (Uncertainties associated with the non-perpendicularity of the target hole axis and the X-Y plane have been assessed and are included in the uncertainty statement that is part of the phantom calibration report.)

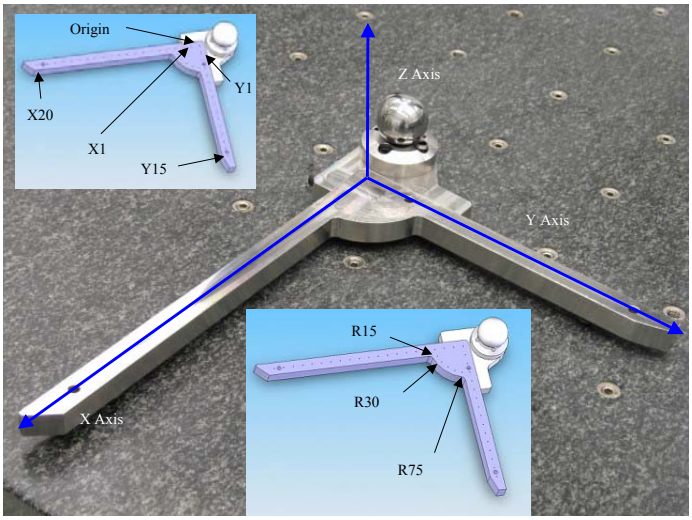
Unlike the CAOS systems, which use pinpoint probes, CMMs use spherical probes with finite diameters. Consequently, the measured coordinate obtained using a CMM represent the center of the probe. As a consequence, the Z coordinates had to be corrected for the effect of the probe tip radius to be consistent with the coordinates measured by the CAOS systems.

The target holes with names that begin with *R* are labeled from R15 through R75 in figure 3. These target holes are labeled using integer increments of 15. The number following *R* in the name corresponds to the approximate angle in degrees that a vector through the origin and the target hole makes with the X-axis, as shown in figure 3.

[1] Certain commercial products and processes are identified in this paper to foster understanding. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products and processes identified are necessarily the best available for the purpose.

4. SPHERE MEASUREMENT (CENTER OF ROTATION)

A sphere with 38.1 mm nominal diameter was measured while positioned in the kinematic ball nest. (See figure 2 below.) The sphere was measured with five points; one on the pole and 4 points distributed approximately 15° above the equator. This measured quantity represents the center of rotation when the femur bar is attached to the phantom base. There is approximately a 3 µm difference in diameter between the size of the sphere attached to the femur bar and the sphere used during calibration. The effects of the difference in size of the two



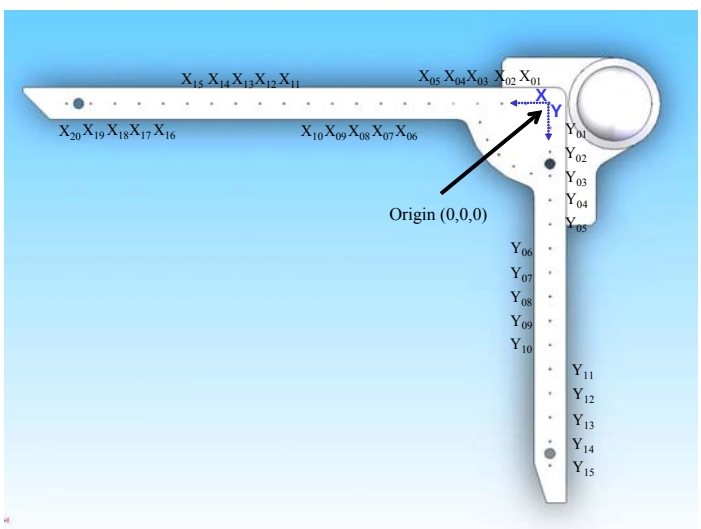


Figure 2. The coordinate of each target hole is defined as the point in its bottom center. The names for the target holes are labeled X1 through X20. They are labeled sequentially from the origin through point X20. The numbering is the same for the Y-Axis, although there are fewer target holes along this axis.

spheres are included in the uncertainty analysis, and the expanded uncertainty provided in the calibration report reflects the effect of this difference.

5. ANGLE MEASUREMENT

Three sets of planes were measured and the angle between the calculated surface normals is reported for each set of planes. The planes were measured using 5 points: 4 points in each of the

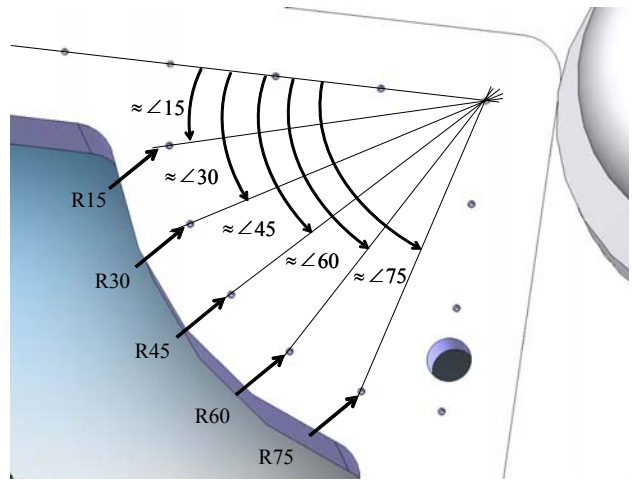


Figure 3. The target holes with names that begin with *R* are labeled from R15 through R75. These target holes are labeled using integer increments of 15. The number following *R* in the name corresponds to the approximate angle in degrees that a vector through the origin and the target hole makes with the X-axis, as shown in figure 2b.

corners of the plane and one point in the approximate center. These planes are shown in figures 4 and 5.

6. MEASUREMENT UNCERTAINTY

Because of the versatility of CMMs, the calculation of measurement uncertainty for the broad array of measurements they can perform is a challenging task. This challenge is a result of the large number of factors that affect the accuracy of CMM measurements. Specifically, CMM hardware errors, environmental effects, form errors in the feature under inspection and the number and distribution of points used to inspect the feature all affect the accuracy of the measurements performed. In response to this challenge, industry has developed some tools to simulate the measurement task while varying the parameters that characterize these factors. To calculate these critical components of uncertainties for the measurements described in this paper, NIST used one such tool, PUNDIT/CMM [4]. This software tool implements a technique called Simulation by Constraints [5] to estimate the components of the measurement uncertainty mentioned above. In the software, all of these effects are modeled and varied and the measurement process simulated.

In particular, the determination of the uncertainty in the X and Y target hole coordinates is complicated by the fact that the cylindrical portion of the target hole is neither perfectly cylindrical nor orthogonal to the X-Y plane. In fact, the angle and shape of each target hole varies slightly due to manufacturing inaccuracy. Clearly, the number and the distribution of points along the surface of the cylindrical portion of the target hole affect the value of calculated coordinates. In order to include the variation of the axis direction and the form error, 4 of the target holes were measured using a large number of points (approximately 26 points total). The data was then

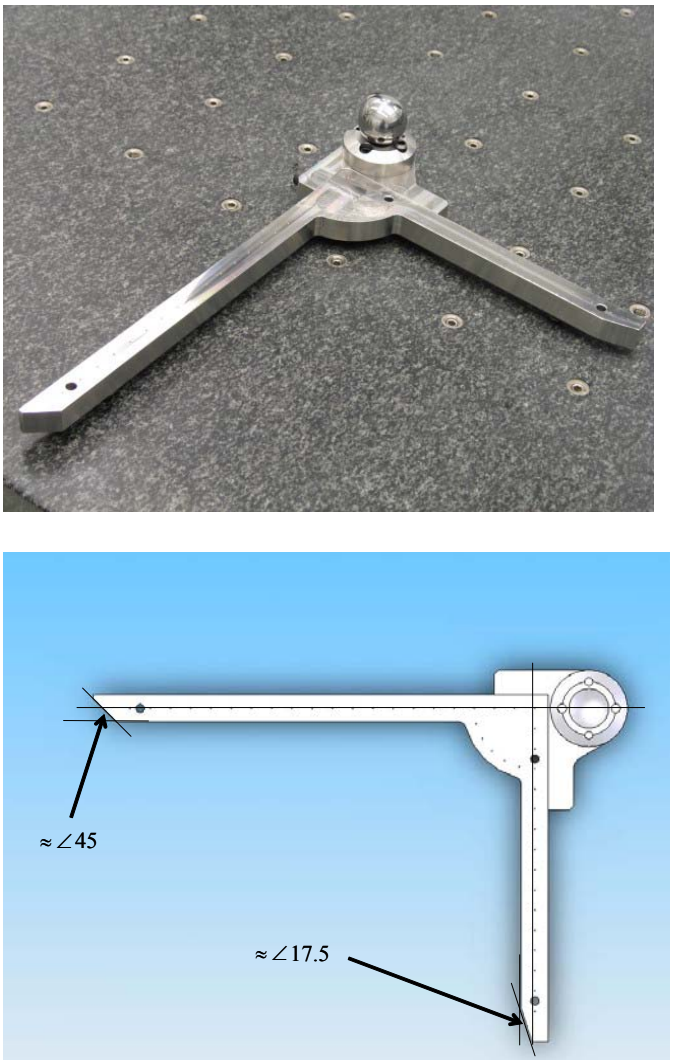


Figure 4. The angle between the adjacent planes labeled in the figure is provided to evaluate the angular measurement performance of CAOS systems when fitted with a supplementary cutting tool.

evaluated and information about the characteristic form error in the cylindrical surface and the variation of the target hole axis direction was entered into the uncertainty software. Further, the bottoms of the target holes are not perfectly flat and smooth. To obtain information for input into the simulation software, a large number of points were measured along the bottom surface of the target hole. The software then uses this information to vary the form of the surface for each measurement simulation. The

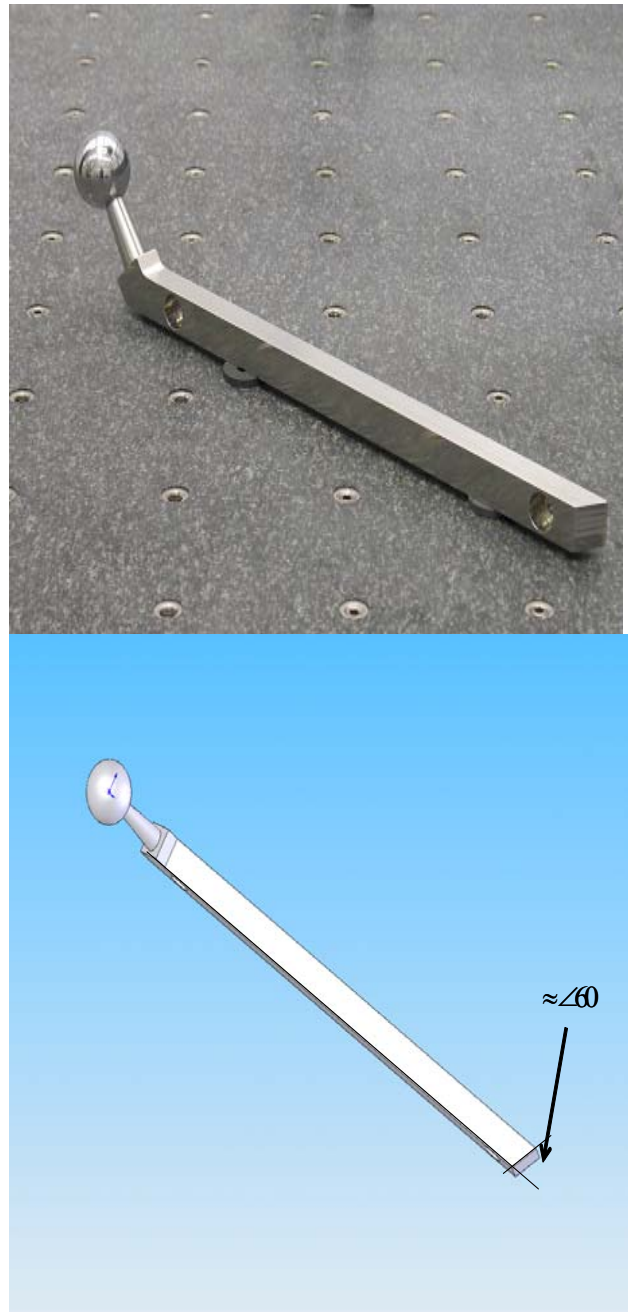


Figure 5. The femur bar shown with the sphere attached. The sphere shown in Figure 3 is removed and replaced with the sphere shown here allowing the femur to be rotated much like a hip joint.

variation in the simulation results characterizes these important components of the measurement uncertainty.

The simulation tool is also used to determine which points should be measured. That is, potential users of the phantom specified the required uncertainties for the calibration before

hand. As typical CAOS point coordinate measurement uncertainty is on the order of 1 mm, it is desirable for the calibration uncertainty to be on the order of 0.1 mm. Consequently, the number and distribution of the points used to calibrate each of the features of the phantom were determined using the simulation software. Specifically, the uncertainty software was used to perform experiments to find more efficient procedures to calibrate parts given a required uncertainty. That is, it would not be an efficient use of measurement and programming time to measure a large number of points on each surface if the required uncertainties do not warrant such effort. By using the simulation tool, sampling points can be increased or decreased and the measurement simulation performed to determine if the required level of accuracy is obtained. In fact, to obtain the required level of accuracy, only measurements of a circular feature on the cylindrical surface of the target hole are required. This greatly simplified the measurement programming and significantly reduces the measurement time.

7. CONCLUSION

We have described the methods, procedures and tool used to calibrate the critical dimensions of the NIST phantom artifact. Field-testing of the prototype phantom artifact has led to proposed refinements in the phantom artifact design. However, the methods and procedures described here may be used to develop a measurement strategy to calibrate future designs of this important artifact.

8. REFERENCES

[1] John A. Bosch, *Coordinate Measuring Machines and Systems*, Marcel Dekker, Inc., 1995

[2] http://www.orthosupplier.com/bonezone/online/2007/fall/editorial_dagalakis.pdf

[3] http://www.isd.mel.nist.gov/medical_devices/PerMIS_OR_CAOHS_Artfct_Vs32.pdf

[4] http://www.metrosage.com/pundit.htm

[5] S.D. Phillips, B. Borchardt, D. Sawyer, et al., The Calculation of CMM Measurement Uncertainty via the Simulation by Constraints, Proceedings of the ASPE, 1997, pages 443-446

[6] ANSI/ASME B89.4.1-1997 Methods for Performance Evaluation of Coordinate Measuring Machines

HLPR Chair – A Novel Patient Transfer Device

Roger Bostelman
National Institute of Standards and Technology
100 Bureau Drive MS 8230
Gaithersburg, MD 20899
301-975-3426
roger.bostelman@nist.gov

James Albus *(retired)*
National Institute of Standards and Technology
100 Bureau Drive MS 8230
Gaithersburg, MD 20899
301-975-3420
james.albus@nist.gov

Joshua Johnson
Florida Gulf Coast University
10501 FGCU Blvd., South
Fort Myers, FL 33965-6565
(239) 590-1000
joshua.johnson@nist.gov

ABSTRACT
In this paper, we briefly describe the design of the Home Lift, Position and Rehabilitation (HLPR) Chair, invented at the National Institute of Standards and Technology (NIST), Manufacturing Engineering Laboratory (MEL) under the Healthcare Mobility Project. The HLPR Chair was designed to provide independent patient mobility for indoor tasks, such as moving to and placing a person on a toilet or bed, and lift assistance for tasks, such as accessing kitchen or other tall shelves. These functionalities are currently out of reach of most wheelchair users. One of the design motivations of the HLPR Chair is to reduce back injury, typically an important issue in the care of this group. Static and dynamic stability tests of the HLPR Chair prototype were also recently completed and are also described here. The tests followed the appropriate and current wheelchair standards and provide suggestions for improvement to these standards. While the Healthcare Mobility Project has recently ended, stability test methods of a lift wheelchair, such as the HLPR Chair, potentially overlap into forklift standard stability test methods and could be useful to the manufacturing industry.

General Terms
Design, Performance, Experimentation, Standardization.

Keywords
HLPR Chair, wheelchair standards, stability, forklift, static, dynamic.

1. INTRODUCTION
Reference [1] says "today, approximately 10 percent of the world's population is over 60; by 2050 this proportion will have more than doubled" and "the greatest rate of increase is amongst the oldest old, people aged 85 and older." She follows by adding that this group is subject to both physical and cognitive impairments more than younger people. These facts have a profound impact on how the world will maintain the elderly independent as long as possible from caregivers. Both physical and cognitive diminishing abilities address the body and the mental process of knowing, including aspects such as awareness, perception, reasoning, intuition and judgment. Assistive technology for the mobility impaired includes the wheelchair, lift aids and other devices, all of which have been around for decades. However, the patient typically or eventually requires assistance to use the device; whether it's someone to push them in a wheelchair, to lift them from the bed to a chair or to the toilet or for guiding them through cluttered areas. With fewer caregivers and more elderly, there is a need for improving these devices to provide them independent assistance.

There has been an increasing need for wheelchairs over time. L.H.V. van der Woude [2] states that mobility is fundamental to health, social integration and individual well-being of humans. Henceforth, mobility must be viewed as being essential to the outcome of the rehabilitation process of wheelchair dependent persons and to their successful (re-)integration into society and to a productive and active life. Thrun [3] said that, if possible, rehabilitation to relieve the dependence on the wheelchair is ideal for this type of patient to live a longer, healthier life. Van der Woude continues stating that many lower limb disabled subjects depend upon a wheelchair for their mobility. Estimated numbers for Europe and USA are 2.5 million and 1.25 million, respectively. The quality of the wheelchair, the individual work capacity, the functionality of the wheelchair/user combination, and the effectiveness of the rehabilitation program do indeed determine the freedom of mobility.

Just as important as wheelchairs are the lift devices and people who lift patients into wheelchairs and other seats, beds, automobiles, etc. The need for patient lift devices will also increase as generations get older. When considering if there is a need for patient lift devices, several references state the positive, for example:

- "The question is, what does it cost not to buy this equipment? A back injury can cost as much as $50,000, and that's not even including all the indirect costs. If a nursing home can buy these lifting devices for $1,000 to $2,000, and eliminate a back injury that costs tens of thousands of dollars, that's a good deal," [4]
- 1 in every 3 nurses becomes injured from the physical exertion put forth while moving non-ambulatory patients; costing their employers $35,000 per injured nurse. [5]
- 1 in 2 non-ambulatory patients falls to the floor and becomes injured when being transferred from a bed to a wheelchair. [6]
- "Nursing and personal care facilities are a growing industry where hazards are known and effective controls are available," said OSHA Administrator John Henshaw. "The industry also ranks among the highest in terms of injuries and illnesses, with rates about 2 1⁄2 times that of all other general industries..." [7]

(c) 2008 Association for Computing Machinery. ACM acknowledges that this contribution was authored or co-authored by a contractor or affiliate of the U.S. Government. As such, the Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.
PerMIS'08, August 19-21, 2008, Gaithersburg, MD, USA
ACM ISBN 978-1-60558-293-1/08/08.

- "Already today there are over 400,000 unfilled nursing positions causing healthcare providers across the country to close wings or risk negative outcomes. Over the coming years, the declining ratio of working age adults to elderly will further exacerbate the shortage. In 1950 there were 8 adults available to support each elder 65+, today the ratio is 5:1 and by 2020 the ratio will drop to 3 working age adults per elder person." [8]

Wheelchairs and patient lift devices have been built and are commercially available today. What has not been built, prior to the HLPR Chair, is a combined: intelligent, powered, lift-wheelchair, geared towards home use that can provide independent or dependent patient transfer to beds, chairs, and/or toilets while also providing a support structure for rehabilitation. Similarly, there are no standards for such devices. [9] Therefore, NIST MEL developed the HLPR Chair to investigate this type of patient transfer device while advancing standards in this area. Of particular concern for safe operation of a device like HLPR Chair is its stability. The question here is: can the device meet current wheelchair standards and are there new standards that can be suggested for these devices as they become commercialized?

This paper includes the HLPR Chair: design, specifications and recent static stability test descriptions and results. Conclusions and references close the paper. References [10, 11, 12, and 13] provide in-depth discussion of the HLPR Chair design and capabilities.

2. HLPR CHAIR DESIGN

The HLPR Chair [10, 11, 12, 13] prototype, shown in Figure 1, is based on a manual, steel, inexpensive, off-the-shelf, and sturdy forklift. The forklift includes a U-frame base with casters in the front and rear and a rectangular vertical frame. The lift and chair frame measures 58 cm (23 in) wide by 109 cm (43 in) long by 193 cm (76 in) high (when not in the lift position) making it small enough to pass through even the smallest, typically 61 cm (24 in) wide x 203 cm (80 in) high, residential bathroom doors. The HLPR Chair frame could be made lighter with aluminum instead of steel.

The patient seat structure is a double, nested and inverted L-shape where the outer L is a seat base frame that provides a lift and rotation point for the inner L seat frame. The L frames are made of square aluminum tubing welded as shown in the photograph. The outer L is bolted to the lift device while the inner L rotates with respect to the seat base frame at the end of the L as shown in Figure 1. The frame's rotation point is above the casters at the very front of the HLPR Chair frame to allow for outside wheelbase access when the seat is rotated 180º and is the main reason access to other seats is available. Drive and steering motors, batteries and control electronics along with their aluminum support frame provide counterweight for the patient to rotate beyond the wheelbase. When not rotated, the center of gravity remains near the middle of the HLPR Chair. When rotated to 180º with a 136 kg (300 lb) patient on board, the center of gravity remains within the wheelbase for safe seat access. Heavier patients would require additional counterweight.

The HLPR Chair is powered similarly to typical powered chairs on the market. Powered chairs include battery powered, drive and steer motors. However, the HLPR Chair has a tricycle design to simplify the need to provide steering and drive linkages and provide for a more vertical and compact drive system design. The drive motor is mounted perpendicular to the floor and above the drive wheel with chain drive to it with maximum speed set to 0.7 m/s (27 in/s). The steering motor is coupled to an end cap on the drive motor and provides approximately 180º rotation of the drive wheel to steer the HLPR Chair. The front of the robot has two casters mounted to a U-shaped frame.

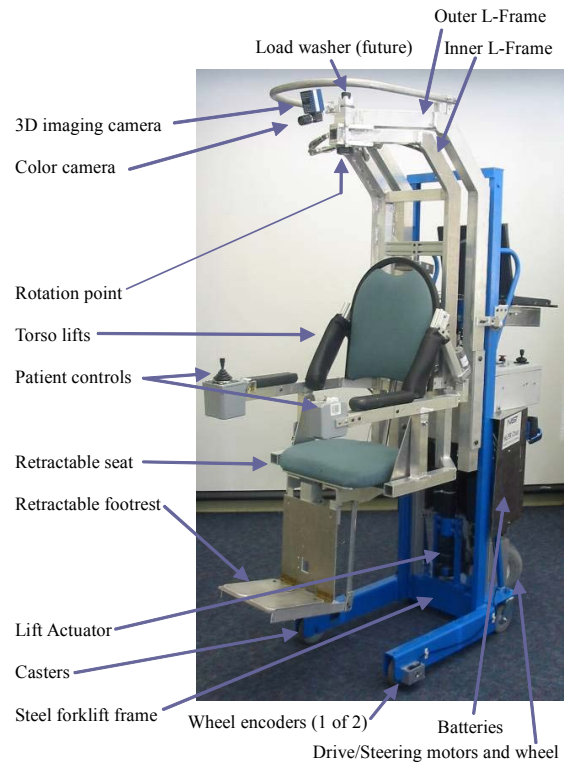


Figure 1: HLPR Chair Prototype

Steering is a novel single wheel design hard stopping the wheel at just beyond 180º for safety of the steering system. Steering is reverse Ackerman controlled as joystick left rotates the drive wheel counterclockwise and joystick right rotates the drive wheel clockwise. The steering rotation amount can be limited by the amount of drive speed so as not to roll the frame during excessive speed with large steering rotation.

Access to and from the HLPR Chair, lift, and rehabilitation configurations, as well as autonomous control efforts and designs are described in detail in [10, 11, 12, 13]. Two prototypes of the HLPR Chair have been built where the first version is used to study stability and autonomous control. The second was built to study ergonomics and manufacturability of the seat and sling designs.

3. HLPR CHAIR SPECIFICATIONS

Current specifications for the HLPR Chair are listed in Table 1. Weight was measured unloaded using a spring scale. Maximum payload was designed into the seat and frame structures. The manufacturer's specifications of the manual forklift and lift actuator lists 227 kg and 454 kg (500 lb and 1000 lb), respectively. Maximum speed can be adjusted via the drive amplifier but has been set at 0.7 m/s (28 in/s). A conservative tilt estimate is shown in the table as 0.06 rad (10º) as determined using a CAD model of the HLPR Chair as previously shown in

[13]. However, Stability tests, explained in Section 4, note a larger static tilt in some orientations. A future test of the battery per-charge range is expected.

Size:	
Mobility Configuration	145 cm long x 58 cm wide x 178 cm high
	(57 in long x 23 in wide x 70 in high)
	with 57 cm (22 ½ in) seat ht. above floor
Full Lifted Configuration	145 cm long x 58 cm wide x 241 cm high
	(57 in long x 23 in wide x 95 in high)
	with 125 cm (49 in) seat ht. above floor – currently
	can be adjusted to lift 91 cm (36 in)
Weight (unloaded)	136 kg (300 lbs)
Payload:	136 kg (300 lbs) (designed)
	91 kg (200 lbs) (tested to date)
Tilt	0.06 rad (10°)
Max. Speed	0.7 m/s (28 ips)
Turning Radius	86 cm (34 in) centered about the rider
Chair Rotate Angle	0.5 rad (90°) CCW to 1 rad (180°) CW
Wheels:	
Rear Drive/Steer	25.4 cm (10 in) diameter pneumatic
Front Caster	12.7 cm (5 in) diameter solid
Ground Clearance	4.4 cm (1 ¾ in)
Battery	2-12Vdc dry cells (series 24V)
Per-Charge Range	unknown to date
Battery weight	11.6 kg (26 lbs) each
Drive Train	1 motor chain drive, 1 gearmotor direct steer
Battery Charger	off-board

Table 1: HLPR Chair Specifications.

4. STABILITY TESTS

A test platform measuring 2.4 m x 1.2 m (8 ft x 4 ft) was recently designed and built, as shown in Figure 2, to perform static stability tests on the HLPR Chair. The platform, made of extruded aluminum framing and plywood base, was lifted with a hoist on one end. Safety straps attached to the facility structure were attached to the HLPR Chair during all tests. Slip prevention bars were also attached to the platform to prevent HLPR Chair from slipping down the ramp as the platform was tilted.

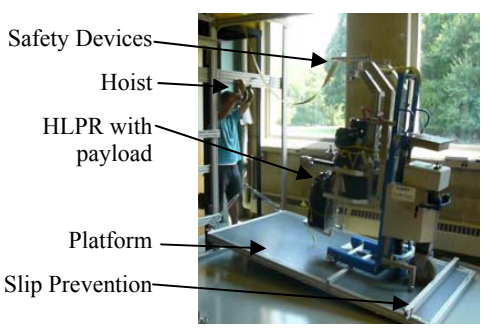


Figure 2: Static Stability Test Set-up.

For stability tests, we needed to verify safe operating parameters of HLPR Chair, including: maximum safe loading capabilities and angle of operation, braking/retardation capabilities, and lift height in relation to the payload. We followed existing wheelchair (patient mobility and transfer application) and forklift (manufacturing application) standards as a basis for the development of the stability tests. For wheelchair standards, we studied various ANSI/RESNA (American National Standards Institute)/(Rehabilitation Engineering and Assistive Technology Society of North America) and ISO (International Organization for Standardization) standards. Those standards were:

- ANSI/RESNA WC/Vol. 1-1998 Wheelchairs- Volume 1: Requirements and test methods for wheelchairs (including scooters)
- ISO 7176-1 Wheelchairs- Part 1: Determination of static stability
- ISO 7176-2 "Wheelchairs- Part 2: Determination of dynamic stability of electric wheelchairs.

For manufacturing, we studied the ISO 1074 Counterbalanced fork-lift trucks - Stability Tests standard.

In the static stability tests the discrete tip angle was measured by placing a piece of paper under the tipping wheel as suggested by the above standards. When the paper could easily be removed from beneath the wheel, the angle was recorded. The test was not designed to measure mechanical failure or device durability using a payload of 114 kg (250 lbs). Instead, the discrete tip angle in the most and least stable configuration was tested. Factors included: load/lift height, load orientation, and HLPR orientation on the test platform. Load height was chosen to be a medium height of 1.3 m (4.2 ft) and a high height of 1.8 m (5.9 ft), the highest the HLPR Chair can lift. Figure 3 shows a series of example test configurations including the forward, lateral and rear configurations. Figure 4 shows a series of load or seat orientations including forward, side and rear orientations. Results of the static stability test are shown in Table 2.

Figure 3: Photos of example test configurations, including: (left) forward, (middle) lateral, and (right) rear configurations.

Figure 4: Load/Seat Orientations of the HLPR Chair with respect to the frame including: (left) forward, (middle) side, and (right) rear orientations.

Test Configuration	Seat Orientation to Frame	Ramp Angle	
		Medium Load Ht. 1.3 m (4.2 ft)	High Load Ht. 1.8 m (5.9 ft)
Forward	Forward	18.4º (0.32 rad)	12.9º (0.23 rad)
	Side (90º)	17.8º (0.31 rad)	12.7º (0.22 rad)
	Rear	17.3º (0.30 rad)	12.2º (0.21 rad)
Lateral	Forward	8.0º (0.14 rad)	4.9º (0.09 rad)
	Side	8.1º (0.14 rad)	4.8º (0.09 rad)
	Rear	8.7º (0.15 rad)	5.1º (0.09 rad)
Rear	Forward	>25º (>0.44 rad)	>25º (>0.44 rad)
	Side	>25º (>0.44 rad)	>25º (>0.44 rad)
	Rear	>25º (>0.44 rad)	>25º (>0.44 rad)

Table 2: Static Stability Test Results.

In future dynamic stability tests, we will be looking for loss of contact of the load-bearing wheels when the platform angle relative to horizontal is at 0, 0.05, 0.1, and 0.15 rad (0º, 3º, 6º, and 9º, respectively). Results will be based on the severity of the lost wheel (drive, caster or stabilizer) contact with the ground as follows:

- 3 – No Tip
- 2- Transient Tip (lifting wheels lose contact then drop back onto the test plane
- 1- Stuck on Anti-tip device
- 0- Full Tip (device is $\pi/2$ (90°) or more from original orientation)

Dynamic tests will include:

- Rearward dynamic stability on a ramp
- Forward dynamic stability on a ramp
- Lateral dynamic stability on a ramp
- Lateral dynamic stability while turning in circles
- Lateral dynamic stability while turning suddenly
- Dynamic stability while traversing a step

Suggestions for changes or additions to current standards, regarding static stability tests, include:

- "Loose" tie-down supports for the vehicle base to the ramp allowing wheel lift from the ramp without catastrophic tip,
- Safety straps to support structures near or at the top of the lift-wheelchair,
- Duplicate tests for aligned-seat-with-base-frame and various misaligned-seat-with-base-frame configurations.
- The standard reads that the operator can currently use a sheet of paper beneath the lifted wheel caused by the tilted platform. Instead, suggested text stating to "use a paper-retractor device (e.g., spring or rubber-band) attached to the paper under the wheel" will allow a safer, single operator for the static tests.

Once dynamic tests are completed, further suggestions to wheelchair standards can be made.

5. CONCLUSION

The HLPR Chair has been prototyped in two versions. Static stability tests were completed on the first HLPR Chair prototype. The results proved higher than expected tilt angles. The second prototype has been loaned to the Florida Gulf Coast University as an example device for their Bio-Engineering Product Design Course. Improvements to the HLPR Chair design are the subject of the course. Future plans are to perform dynamic stability tests and to transfer the HLPR Chair design to the healthcare industry and to the manufacturing industry for semi-autonomous forklift control research.

6. REFERENCES

[1] Pollack, Martha, *Intelligent Technology for Adaptive Aging,* Presentation, AAAI-04 American Association for SArtificial Intelligence Conference Keynote Address, 2004.

[2] L.H.V. van der Woude, M.T.E. Hopman and C.H. van Kemenade, *Biomedical Aspects of Manual Wheelchair Propulsion: The State of the Art II*, Volume 5, Assistive Technology Research Series, 1999, 392 pp., hardcover.

[3] Thrun, Sebastian, Visit to Stanford University to discuss healthcare mobility devices, August 2006.

[4] Marras, William, *Lifting Patients Poses High Risk for Back Injuries*, Ohio State University, http://researchnews.osu.edu/archive/resthome.htm, 1999.

[5] Blevins, Healthcare Statistics: Blevins Medical, Inc., http://www.patientlift.net/282164.html, 2006.

[6] U.S. Bureau of Labor Statistics, from Blevins website: http://www.patientlift.net/282164.html, 1994.

[7] John Henshaw, http://www.osha.gov/SLTC/nursinghome/solutions.html, Occupational Safety and Health Administration, 2005

[8] Wasatch Digital iQ, *InTouch Health's Remote Presence Robot Used by Healthcare Experts*, http://www.wasatch digitaliq.com/parser.php?nav=article&article_id=43, Santa Barbara, CA & Salt Lake City --(Business Wire)--June 16, 2003.

[9] Bostelman, Roger; Albus, James, *Survey of Patient Mobility and Lift Technologies Toward Advancements and Standards,* NISTIR #7384, 2006.

[10] Bostelman, R., Albus, J., *HLPR Chair – A Service Robot for the Healthcare Industry*, 3rd International Workshop on Advances in Service Robotics, Vienna, Austria, July 7, 2006.

[11] Roger Bostelman, James Albus, Tommy Chang, Tsai Hong, Sunil K. Agrawal, Ji-Chul Ryu, *HLPR Chair: A Novel Indoor Mobility-Assist and Lift System*, Proceedings of IDETC/CIE 2007, ASME 2007 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, Las Vegas, Nevada, September 4-7, 2007.

[12] Bostelman, R., Albus, J., *Sensor Experiments to Facilitate Robot Use in Assistive Environments,* Proc. of the 1st International Conference on Pervasive Technologies Related to Assistive Environments (PETRA), Athens, Greece, July 15-19, 2008.

[13] Bostelman, R., Albus, J., Chang, T., *Recent Developments of the HLPR Chair,* Proc. of the 10th edition of the International Conference on Rehabilitation Robotics (ICORR 2007), Noordwijk aan Zee, Netherlands, June 13-15, 2007.

Robotic Navigation in Crowded Environments: Key Challenges for Autonomous Navigation Systems

James Ballantyne Salman Valibeik Ara Darzi Guang-Zhong Yang
Department of Biosurgery and Surgical Technology
Institute of Biomedical Engineering
Imperial College London
London, United Kingdom

ABSTRACT
Crowded environments provide significant challenges for autonomous navigation systems. The robot must be fully aware of its surroundings and incorporate this knowledge into its decision-making and planning processes. The purpose of this paper is to outline major challenges that an autonomous navigation system needs to overcome to enable effective navigation in crowded environments such as hospital wards. We will discuss several key components of autonomous navigation systems to include localization and mapping, human-robot interaction, and dynamic object detection.

Categories and Subject Descriptors
I.2.9 [**Artificial Intelligence**]: Robotics – *Autonomous Vehicles, Sensors*

General Terms
Navigation Algorithms, Performance, Reliability, Verification.

Keywords
Robotic navigation in crowded environments.

1. INTRODUCTION
Autonomous navigation systems aim to provide seamless integration of robots into everyday life. The robot must fully understand the environment and sense alterations and potential problems. Furthermore, interaction between humans and robots provides an important link that allows humans to help in difficult navigation situations. This paper outlines the current challenges that autonomous navigation systems must overcome before full immersion is possible.

Developing accurate environment maps is one of the most difficult challenges for autonomous navigation systems. An accurate representation of the environment improves the capabilities of the robot in navigation and dealing with dynamic objects. The pre-requisite of the task is correct map creation while maintaining an accurate estimate of the pose of the moving robot. This is commonly referred to as the Simultaneous Localization and Mapping (SLAM) problem. The inclusion of dynamic objects can cause difficulties to SLAM, and therefore may require additional information to simplify the problem.

One such source of information is Human-Robot Interaction (HRI), which can play a valuable role in preventing the robot from getting lost in crowded and highly congested areas. However, there are many difficulties involved in achieving successful human guided navigation including robust identification of human and human gestures. An effective means of establishing such communication can come from the identification of facial expression and eye-gaze. To this end, localizing the person with attention is necessary. This requires the detection and tracking of objects such as the face, eyes, hands, as well as the posture of the whole body. In a crowded environment, however, this is difficult due to scene complexity, constant occlusions of objects, and changes in lighting conditions due to inter-reflection and shadows.

In SLAM, dynamic objects are the primary reason for failures if the majority of features tracked belong to the dynamic objects. Without the effective use of temporal information, the system can have difficulties in distinguishing dynamic and static parts of the environment. Therefore, dynamic object detection is essential. In a crowded environment, this also entails the segmentation of different moving objects as each of them can have different motion characteristics. In the following sections, we will review some of our approaches in dealing with the problems mentioned above. In particular, we will discuss issues related to motion planning and its dependency on SLAM, HRI, and dynamic object detection.

2. SYSTEM HARDWARE SETUP
To demonstrate our methods for navigation in crowded environments, a Mesa-Imaging SR-3000 [1] *Time-of-Flight* camera and a Point Grey DragonFly2 [2] digital camera mounted on an ActivMedia PeopleBot [3] were used. The system set-up is illustrated in Figure 1.

The time-of-flight camera provides 4D information at frame rates up to 30fps. The sensor provides a pixel array of size 176×144, where each pixel provides x, y, z, and intensity information. The field of view of the camera is about 48 by 40 degrees of visual angle. We used a time-of-flight camera over the traditional laser range finder due to its compact size and high frame rate. The digital camera provides 648×488 RGB color images at 30 fps. For the HRI experiment, we used a lens with adjustable focal length

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
PerMIS'08, August 19–21, 2008, Gaithersburg, MD, USA.
ACM ISBN 978-1-60558-293-1…$5.00.

between 3.5–8mm, while for SLAM, we used a fisheye lens with a 1.78 mm focal length.

3. LOCALIZATION AND MAPPING

In an ideal situation, robots need to create accurate maps of the environment without the use of prior knowledge. SLAM has received significant attention [4, 5, 6, 7, 8, 9, 10, 11] in recent years and existing techniques can efficiently map large environments with high accuracy if the environment is static.

Crowded environments present many difficulties for SLAM due to its reliance on static landmarks. The following example demonstrates the limitations of the existing solutions in a crowded and congested environment. Range-based SLAM systems are capable of mapping large-scale environments, both indoors and outdoors. However, these environments typically only have limited moving objects. Figure 2 shows both the scan line (top row) and intensity image (bottom row) taken from the time-of-flight camera for a short video sequence, which gives insight into the fundamental challenge for range-based SLAM systems. The majority of the scan corresponds to dynamic objects (circled in red). This prevents scan matching algorithms from functioning correctly, thus forcing the system to fall back to odometry readings. In the initial map-building phase, scan-matching failures prevent the understanding of the environment due to a lack of knowledge of its location. Furthermore, even when the static objects exist in one frame, there is no guarantee that they will be visible in the next.

Vision-based systems provide an alternative to range-based methods. They aim to identity landmarks in the environment for reliable matching to help determine the location of the robot in the environment. The key to feature-based methods is to identify high quality features that are easily recognizable from varying orientations and illumination conditions. For our investigation, we used an implementation of MonoSLAM [6], which has shown favorable results for indoor environments based only on

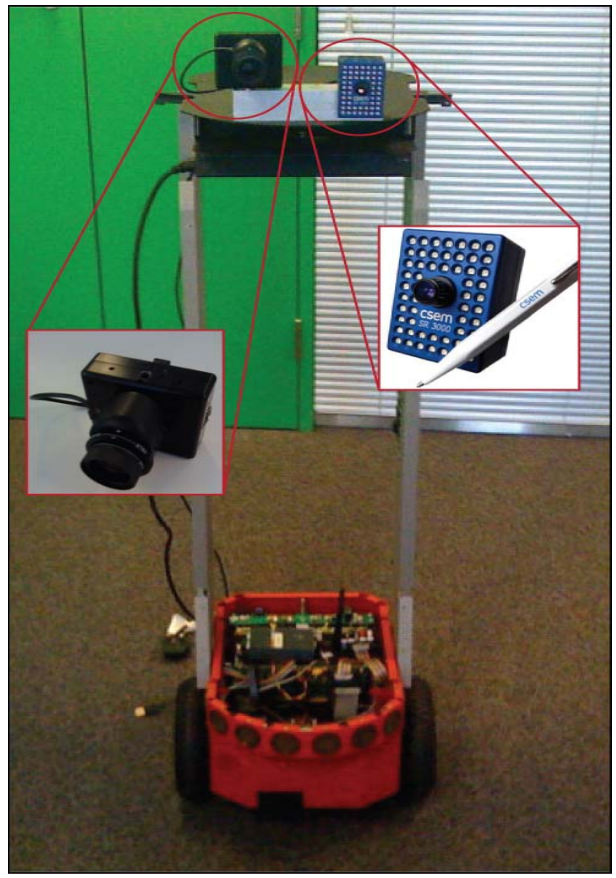


Figure 1. Experimental Setup; an Activmedia PeopleBot equipped with a Mesa-Imaging SR-3000 time-of-flight camera (right) and a Point Grey DragonFly2 camera (left) are used.

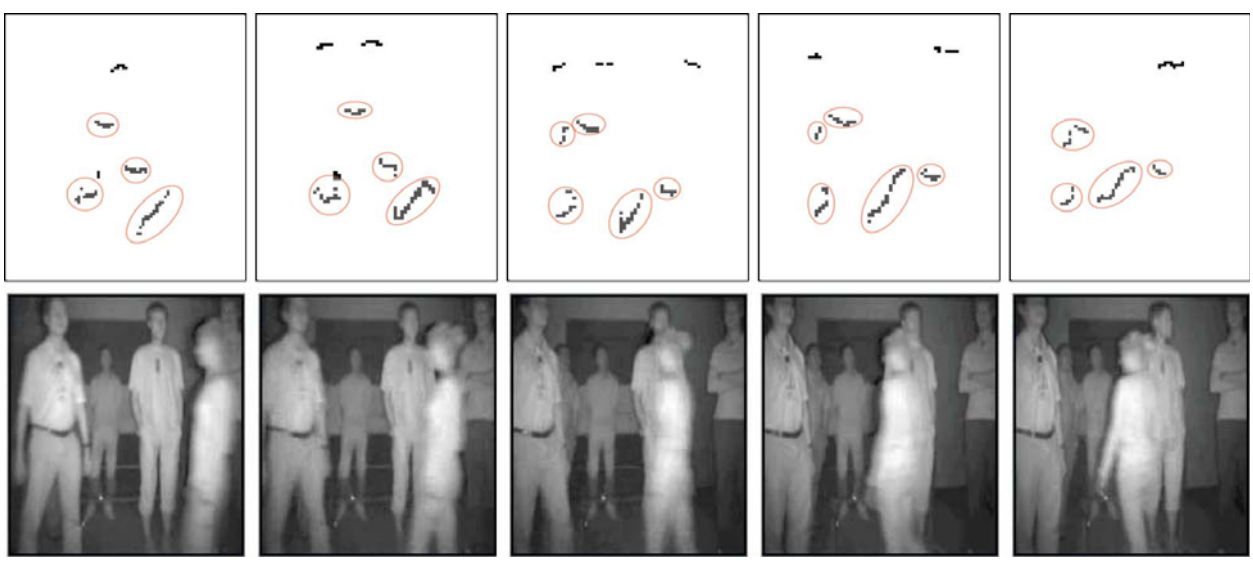


Figure 2. The top row provides scan lines taken from the time-of-flight camera. A red circle identifies the dynamic areas of the environment, which were manually segmented. This short sequence illustrates the problem for range-based systems, where the lack of static information prevents scan matching.

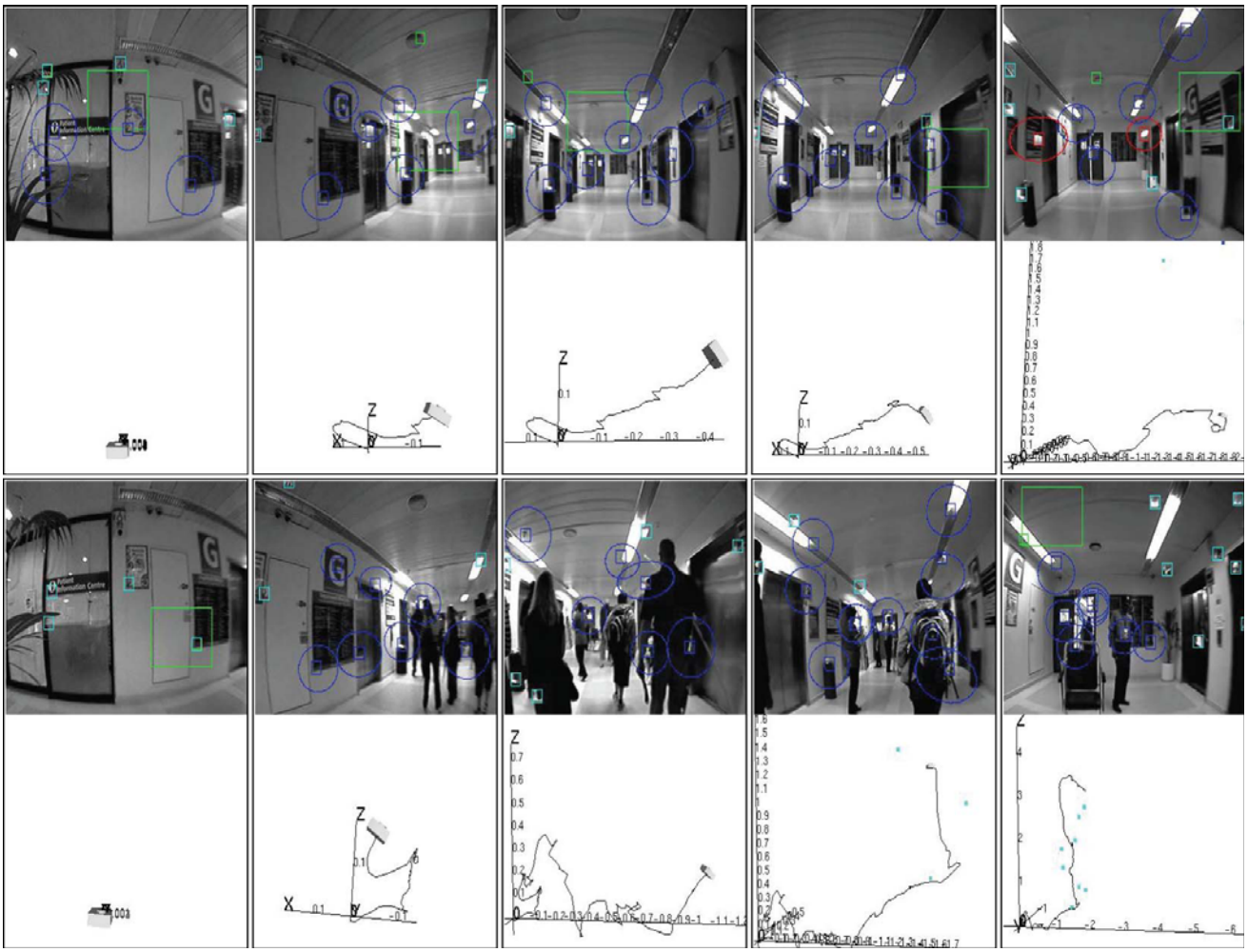


Figure 3. Top: a short sequence taken from a moving robot in a static scene. The SLAM algorithm correctly tracks the forward movement of the robot down the corridor. Bottom: a similar sequence but with a group of people in the corridor. In this case, the majority of tracked features belong to the moving people and for this reason the SLAM algorithm is unable to function during the initial map-building phase.

monocular vision. The system uses Shi and Tomasi feature detection [12] to identify salient regions. The acquired sequences were captured from a fisheye lens because features can remain visible for longer periods of time.

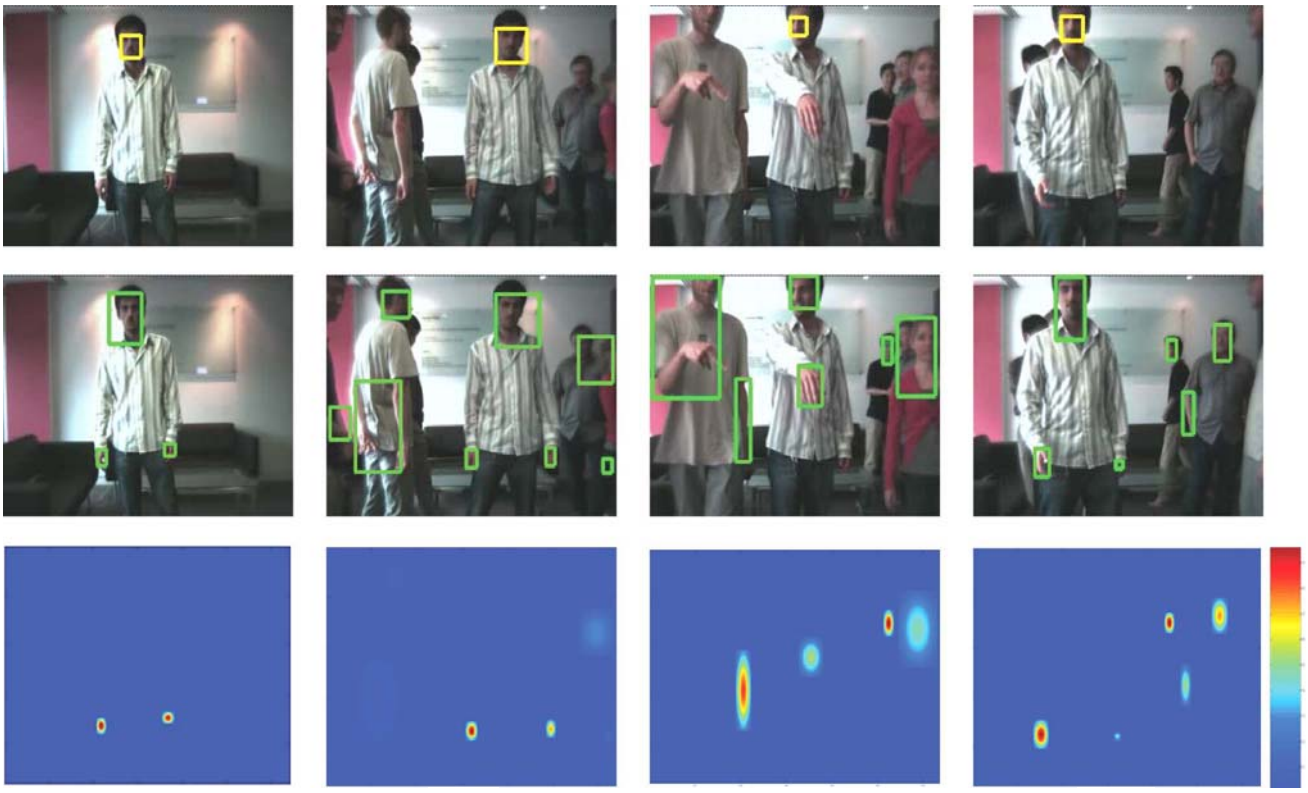
Figure 3 illustrates some select image frames from two different sequences. The top row corresponds to the current image taken from the camera, while the bottom row illustrates the estimated movement as determined by the SLAM algorithm. The first sequence involved the robot travelling down a corridor with no dynamic objects in the field-of-view, while the second sequence shows the corresponding scene with a crowd of people.

It is evident that the SLAM approach maintained an accurate estimation of the robot's motion in the static scene. However, the system failed to track the movement of the robot accurately in the crowded environment. This occurred because the majority of features corresponded to the dynamic objects. As the objects moved, the system was not able to maintain enough features for accurate tracking. Even with a large field of view based on the fisheye lens, the static features remained scarce and many of which corresponded to the ceiling and lights, which could cause further problems if lighting condition changed.

The above results illustrate the main difficulties of the existing SLAM implementations in crowded environments. The main difficulties are involved in the initial map-building phase where the amount of usable sensor information is limited. In such situations, we propose the use of HRI and dynamic object detection to help with this initial phase in SLAM. HRI would allow humans to guide the robot through congested areas of the environment where it would be able to track its estimated position based on the movement of the person. Dynamic object detection could further improve the SLAM algorithm by segmenting out the dynamic information, leaving only static data for environment mapping.

4. HUMAN-ROBOT INTERACTION

Human guided navigation is used to allow the robot to learn from humans about a specific environment such that the domain specific knowledge acquired can be used for subsequent



(a) (b) (c) (d)

Figure 4. A probabilistic framework for face and gesture association. Top: the result of face detection (labeled with yellow rectangles). Middle: potential hand gestures identified with green rectangles. Bottom: data association based on normalized likelihood. In (a) and (b), the likelihood is correctly computed, whereas in (c) and (d) the correlation fails and further visual cues are required for correct data association.

navigation. Currently, there are a number of difficulties involved in achieving successful human guided navigation. The main difficulty is in the robust identification of humans and establishing effective communication via visual cues involving facial expression, hand gestures and posture of the body. Face detection is a common method for identifying humans based on its distinctive geometrical and color features and limited pose variability with respect to other parts of the body [13, 14, 15]. However, existing methods are sensitive to illumination, size and orientation changes. Furthermore, the presence of multiple people in the environment can significantly increase the complexity of the algorithm.

For commanding the robot, hand gesture detection is desirable as it best represents human-machine interaction [16, 17, 18]. This allows the robot to identify different gestures suitable for HRI. Most dynamic and static hand gesture recognition techniques rely on finding large hand silhouettes [18, 19], which is difficult in a crowded scene. To facilitate the association of hand silhouettes with the right person and his/her face, we have used a statistical method based on a fast and accurate face detection system as described by Kienzle *et al* [20]. Initially, we trained the system by segmenting the correlated faces and hands based on an observation factor O defined as: $O = [F_{width}, F_{height}, H_{width}, H_{height}, D]$. Where F_{width} and H_{width} represent the face and hand width, F_{height} and H_{height} indicate the face and hand height, and D specifies the distance between the face and hand. The training data involves different gestures and movements. When a face was detected, we stored feature information for the hand silhouettes according to the observation factor. This then allows the creation of a Gaussian Mixture Model (GMM) to describe the statistical description of the gesture space given a detected face.

For real-time deployment, the mean and covariance matrices can be used to identify the likely position of the hands after face detection. Figure 4 illustrates some example results from the GMM based on a real-time footage. Figures 4 (c) and (d) illustrate potential problems arisen in a crowded and dynamic environment. The correlation between hands and face was wrongly identified due to a large number of potential hand silhouettes. In contrast, Figures 4 (a) and (b) provide the correct correlation between the face and hand due to a less complicated environment. This highlights the difficulty of HRI in a crowded environment.

5. DYNAMIC OBJECT DETECTION

Another key component for autonomous navigation systems in crowded environments is dynamic object detection. This requires the system to segment the scene into static and dynamic objects. Segmentation can potentially help alleviate some of the drawbacks of SLAM. In our work, we have attempted two methods for dynamic object detection. Since our main sensor is

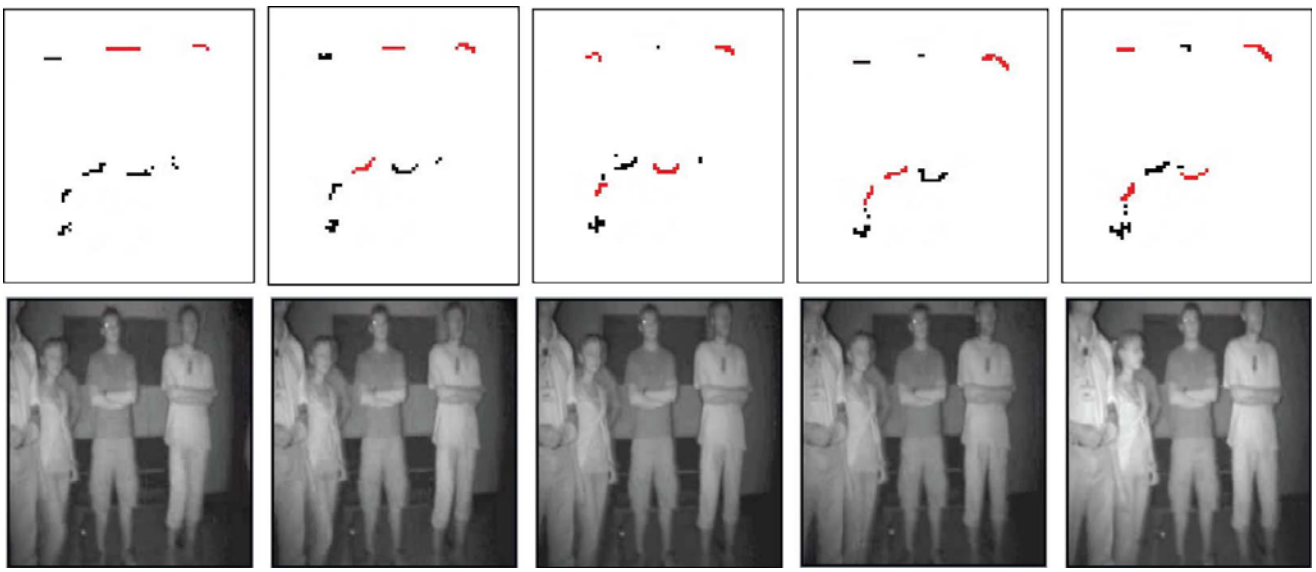


Figure 5. Top: the local occupancy grid from the time-of-flight camera. Bottom: the full scene as seen through the camera's viewpoint. Cells colored in red correspond to detected dynamic objects in the environment. This short sequence illustrates the key challenge of handling both local and global motion. The figure shows incorrect labeling of both static and dynamic parts of the environment.

range based, we have mainly focused on range-based frame differencing techniques.

The goal of frame differencing is to distinguish static and dynamic objects by comparing the current frame with the previous one. The underlying principle regards the scene as static, therefore specifying objects as dynamic requires finding current occupied space that was previously unoccupied. We chose frame differencing due to its speed and low computation complexity.

The focus of our investigation of dynamic object detection is how to enhance the power of SLAM. It should be noted that our intent is to detect the dynamic aspects of the scene on a frame-to-frame basis, although long-term tracking should further enhance the robustness of the results. As an example, we implemented a two dimensional system as described by Prassler *et al* [21]. The method used Time-Stamping Maps (TSM) to compare cells occupied in the current map, TSM_t, against the previous map, TSM_{t-1}. Any cells with no correspondence were considered dynamic. To prevent false moving object detection, groups of cells were created by linking neighboring cells. Then, if the majority of a group was moving, the whole group was marked as moving.

To identify potential problems in crowded environments, we gathered sequences of images with a group of people in front of the robot while it performed simple rotations and translations. Figure 5 provides an example of five frames where the robot performed a small rotation to the right. The top row shows the scan line from the time-of-flight camera, while the bottom row shows the intensity image of the scene. This short sequence demonstrates the challenge of the detection systems in a crowded environment, where global and local motion separation is necessary. During rotation, static objects were highlighted as moving for two reasons. First, new information entered the field of view that was not visible in the previous image. Second, previously occluded objects became visible as other objects moved.

The main problem stems from the lack of understanding of the robot motion. Since, the majority of the scene was moving, it was difficult to determine the robot's motion. This prevented the correct projection of the previous frame into the current frame's reference point. Thus, there are both false and missing detections.

As a second example, we implemented a 3D system where we performed a straight comparison of the current and previous images using 3D information taken from the time-of-flight camera. The algorithm worked as follows:

1) Grab the current image.
2) Project the previous image into the current field of reference.
3) Mark pixels as moving if there is a difference in depth between current and previous.
4) Perform a flood-filling algorithm to highlight the full object.

To project the previous image into the current frame of reference, we first converted the depth map to the world coordinate system using the estimated pose from odometry. Then, we computed a reverse projection onto the current coordinate system using the method described by Stipes *et al* [22]. Finally, we compared each pixel in the projected image against the current image. Any pixels that did not match within a certain threshold (0.1m for this study) were marked as dynamic. The final step highlighted the full object using a flood-filling algorithm based on depth.

The top row of Figure 6 presents three consecutive frames to highlight one of the problems for 3D frame differencing. As the robot rotates to the right with several moving objects, the system incorrectly labeled objects due to inaccuracies in the estimated pose, which causes errors in the reverse projection.

The bottom row of Figure 6 highlights the second difficulty for the frame differencing algorithms and dynamic object detection, which is the cancellation of global and local motion causing

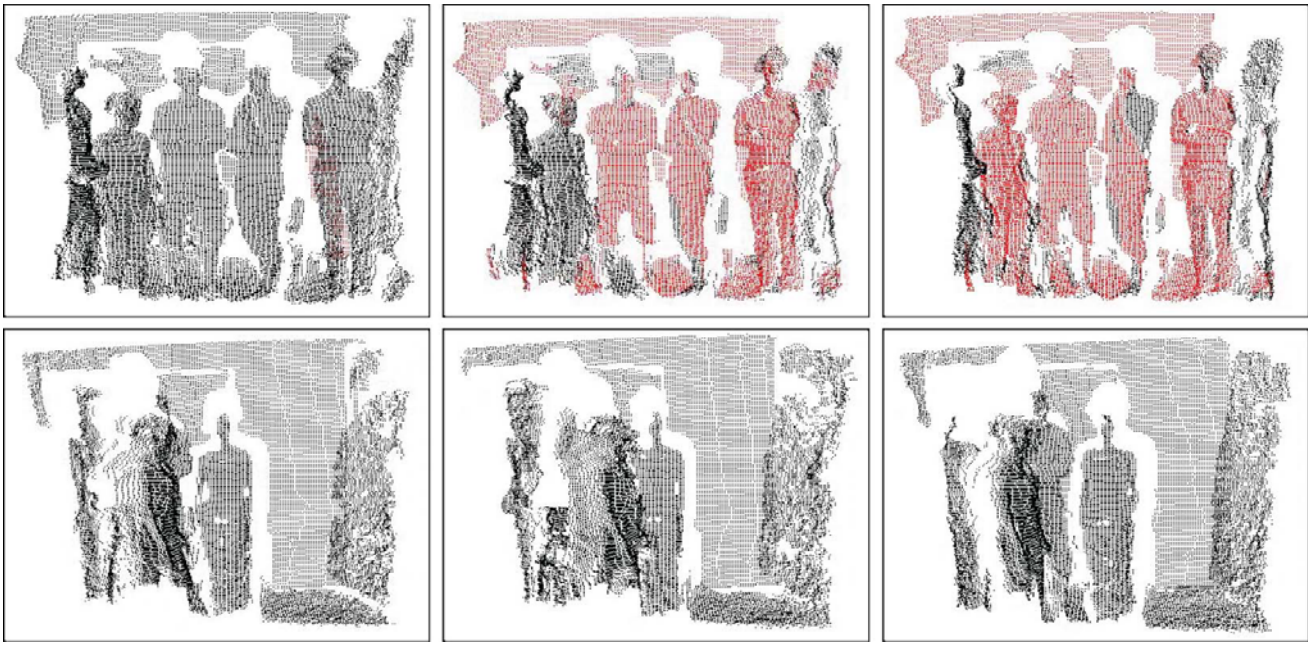


Figure 6. Top: an illustration of the errors introduced due to global motion in a crowded environment based on 3D frame differencing. Bottom: errors introduced due to global and local motion, where the two cancel out each other, thus preventing the correct identification of moving objects.

missing dynamic objects. The above examples outline the importance of global and local motion separation.

6. MOTION PLANNING

The final component for an autonomous navigation system is the perceived intelligence of the robot. The goal of any path planning system is to devise efficient and effective plans to complete any given task. The system must do this in such a way to avoid running into objects whether static or dynamic. The path planning system mainly consists of local and global navigation. Local navigation, often called reactive control, learns or plans local paths using only the current sensory input and no prior knowledge of the environment. Global navigation, often termed as deliberate control, learns or plans global paths using the system's current knowledge of the environment.

Local navigation or reactive control systems mainly devise plans to navigate locally. The system uses only the current sensory input without any prior knowledge of the environment. The system performs tasks such as obstacle avoidance and point-to-point moving. Some examples of reactive control approaches are potential fields [23] and virtual force fields [24].

These basic reactive control systems can be extended to full behavioral systems [25, 26, 27, 28, 29] where the goal is to design a list of acceptable behaviors for the robot. Examples of behavioral control architectures are fuzzy logic, neural networks, evolutionary computation, and reinforcement learning. The major hurdle for reactive systems is the lack of prior knowledge.

The effectiveness of local navigation relies on the understanding of dynamic objects. It is not feasible for the system to determine safe paths through the environment without fully understanding the motion of other objects involved. If accurate dynamic detection methods are available, intelligent local avoidance systems can be created, such as the system by Berg *et al* [30].

HRI provides alternative ways to improve local navigation. With increasing complexity of the environment, even advanced dynamic object detection system can fail. Furthermore, the unpredictability of humans can further disrupt local navigation. Therefore, a reliable HRI system can safely help a robot navigate through even the most complex environments.

Global navigation or deliberate control systems attempt to generate global paths based on the current knowledge of the environment. These systems are concerned with generating paths or plans for the robot to complete any given task. The success of the global planner relies on the assumption that the environment map is accurate. Without an accurate map, the robot is unable to complete any given task due to lack of knowledge of its surroundings. With a full representation of the environment, there are many available planning algorithms. Both Latombe [31] and LaValle [32] give comprehensive reviews of potential solutions. Therefore, without an accurate SLAM system, global navigation is difficult to achieve.

7. FUTURE WORK

This paper illustrates some of the key challenges for robotic navigation in crowded and congested environments. We have concentrated our discussion on the initial phase of autonomous navigation for understanding the environment. It has been shown that the SLAM approach is challenging for both the range and vision based systems due to the lack of sufficiently accurate static information in the scene at any given time. Dynamic objects (e.g., people within the scene) can further confuse the system. To avoid these difficulties, it is useful to incorporate explicit HRI to rule out some of the intrinsic ambiguities in navigation. To this end, the reliable of tracking human gesture, expression is necessary.

8. REFERENCES

[1] Mesa-Imaging, "http://www.mesa-imaging.ch."

[2] "Point Grey Research," *http://www.ptgrey.com/products/dragonfly/index.asp*.

[3] "Mobile Robots: ActivMedia Robotics," *http://www.activrobots.com/*.

[4] Hähnel., D., Burgard, W., Fox, D. and Thrun, S., "An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, USA, 2003, pp. 206-211.

[5] Thrun, S., Liu, Y., Koller, D., Ng, A., Ghahramani, Z. and Durrant-Whyte, H., "Simultaneous Localization and Mapping with sparse extended information filters," *International Journal of Robotics Research,* vol. 23, pp. 693-716, 2004.

[6] Davison, A. J., Reid, I. D., Molton, N. D. and Stasse, O., "MonoSLAM: Real-Time Single Camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 29, pp. 1-16, 2007.

[7] Montemerlo, M., Koller, S. T. D. and Wegbreit, B., "FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *International Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003, pp. 1151-1156.

[8] Murphy, K., "Bayesian map learning in dynamic environments," in *Conference on Neural Information Processing Systems (NIPS)*, Denver, CO, USA, 1999, pp. 1015-1021.

[9] Montemerlo, M., Thrun, S., Koller, D. and Wegbreit, B., "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem," in *National Conference on Artificial Intelligence*, Edmonton, Alberta, Canada, 2002, pp. 593-598.

[10] Grisetti, G., Tipaldi, G. D., Stachniss, C., Burgard, W. and Nardi, D., "Fast and Accurate SLAM with Rao-Blackwellized Particle Filters," *Robotics and Autonomous Systems,* vol. 55, pp. 30-38, 2007.

[11] Eliazar, A. and Parr, R., "DP-SLAM: Fast, robust simultaneous localization and mapping with predetermined landmarks," in *International Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003, pp. 1135-1142.

[12] Shi, J. and Tomasi, C., "Good Features to Track," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 1994.

[13] Li, S. Z. and Zhenqiu, Z., "FloatBoost learning and statistical face detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 26, pp. 1112-1123, 2004.

[14] Chengjun, L., "A Bayesian discriminating features method for face detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 25, pp. 725-740, 2003.

[15] Ming-Hsuan, Y., Kriegman, D. J. and Ahuja, N., "Detecting faces in images: a survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 24, pp. 34-58, 2002.

[16] Xia, L. and Fujimura, K., "Hand gesture recognition using depth data," in *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, 2004, pp. 529-534.

[17] Hasanuzzaman, M., Ampornaramveth, V., Tao, Z., Bhuiyan, M. A., Shirai, Y. and Ueno, H., "Real-time Vision-based Gesture Recognition for Human Robot Interaction," in *Robotics and Biomimetics, 2004. ROBIO 2004. IEEE International Conference on*, 2004, pp. 413-418.

[18] Ying, W. and Huang, T. S., "Hand modeling, analysis and recognition," *Signal Processing Magazine, IEEE,* vol. 18, pp. 51-60, 2001.

[19] Yuanxin, Z., Haibing, R., Guangyou, X. and Xueyin, L., "Toward real-time human-computer interaction with continuous dynamic hand gestures," in *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, 2000, pp. 544-549.

[20] Kienzle, W., Bakir, G., Franz, M. and Schölkopf, B., "Face Detection - Efficient and Rank Deficient," 2005.

[21] Erwin Prassler, J. S., Alberto Elfes, "Tracking Multiple Moving Objects for Real-time Robot Navigation," *Autonomous Robots,* vol. 8, pp. 105-116, April 2000.

[22] Stipes, J. A., Cole, J. G. P. and Humphreys, J., "4D Scan Registration with the SR-3000 LIDAR," in *IEEE International Conference on Robotics and Automation (ICRA)*, Pasadena, CA, USA, 2008.

[23] Khatib, O., "Real-time obstacle avoidance for manipulators and mobile robots," *International Journal of Robotics Research,* vol. 5, pp. 90-98, 1986.

[24] Borenstein, J. and Koren, Y., "Real-time Obstacle avoidance for fast mobile robot," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 19, pp. 1179-1187, 1989.

[25] Zhu, A. M., Yang, S. X., Wang, F. J. and Mittal, G. S., "A neuro-fuzzy controller for reactive navigation of a behavior-based mobile robot," *Lecture Notes Computer Science,* vol. 3498, pp. 259-264, 2005.

[26] Thrun, S., "Probabilistic algorithms for robotics," in *AI Magazine*. vol. 21, 2000, pp. 93-109.

[27] Kondo, T. and Ito, K., "A Reinforcement learning with evolutionary state recruitment strategy for autonomous mobile robots control," *Robotic Autonomous Systems,* vol. 46, pp. 111-124, 2004.

[28] Chen, C., Li, H.-X. and Dong, D., "Hybrid Control for Robot Navigation," in *IEEE Robotics & Automation Magazine*, 2008.

[29] Chen, C. L. and Chen, Z. H., "Reinforcement learning for mobile robot: From reaction to deliberation," *Journal of System Engineering Electronics,* vol. 16, pp. 611-617, 2005.

[30] Berg, J. V. D., Patil, S., Sewall, J., Manocha, D. and Lin, M., "Interactive navigation of multiple agents in crowded environments," in *Proceedings of the 2008 symposium on Interactive 3D graphics and games*, Redwood City, CA USA, 2008, pp. 139-147.

[31] Latombe, J.-C., *Robot Motion Planning*. Norwell, MA, USA: Kluwer, 1991.

[32] LaValle, S. M., "Planning Algorithms [Online]," in *http://planning.cs.uiuc.edu/*, 2006.