NISTIR 8416

Machine Learning Based Wireless Interference Estimation in a Robotic Force-Seeking Application

Richard Candell
Mohamed Kashef
Karl Montgomery
Yongkang Liu
Sebti Foufou

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8416

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

AUTHENTICATED
U.S. GOVERNMENT
INFORMATION
GPO

NISTIR 8416

Machine Learning Based Wireless Interference Estimation in a Robotic Force-Seeking Application

Richard Candell
Mohamed Kashef
Karl Montgomery
Yongkang Liu
Smart Connected Systems Division
Communications Technology Laboratory

Sebti Foufou
Laboratoire d'Informatique
Université de Bourgogne

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8416

February 2022

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
James K. Olthoff, Performing the Non-Exclusive Functions and Duties of the Under Secretary of Commerce for Standards and Technology & Director, National Institute of Standards and Technology

Certain commercial entities, equipment, or materials may be identified in this document in order to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

National Institute of Standards and Technology Interagency or Internal Report 8416
Natl. Inst. Stand. Technol. Interag. Intern. Rep. 8416, 33 pages (February 2022)

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8416

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

Abstract

Wide deployment of wireless communications plays an essential role in the vision of future cyber-physical systems (CPSs), which includes massive transfer of automation information. Many practical considerations of industrial CPSs affect the success of the deployment of industrial wireless networks, and, thus, can inhibit their widespread adoption. These considerations include multi-path propagation, network congestion, and jamming interference. Jamming is of chief concern when wireless is used for mission critical or safety integrated systems. In this paper, an experimental platform consisting of a robot arm depressing a spring mechanism with a wireless force-feedback control algorithm is constructed. The robot applies downward pressure on a spring assembly until a predetermined force is detected and transmitted successfully to the controller under varying levels of sustained interference. Machine learning is used to learn and predict the signal-to-interference level of the communication link solely using position information from an independent vision tracking system. Various supervised learning algorithms are investigated and rated according to their performance.

Key words

industrial wireless, factory communications, cyber-physical systems, wireless networking, robotics, machine learning, IEEE 802.11.

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

Table of Contents

1 Introduction **5**
1.1 Motivation 6
1.2 Contributions 7
1.3 Organization 7
2 Related Work **7**
2.1 Deploying WLANs in Industrial Applications 7
2.2 Interference Analysis in CPSs 8
2.3 Impacts of Jamming on CPSs 8
2.4 Machine Learning for Jamming Detection 9
2.5 State Estimation for Industrial Control Systems 9
2.6 Conclusions of Related Work 10
3 Robot Arm Force-Seeking Application **10**
3.1 Testbed Components 11
3.2 Experimental Setup 13
3.3 RF Scenario 13
4 Data Analysis **13**
4.1 Collected Data 14
4.2 Feature Extraction 14
4.3 SIR Estimation 16
5 Results **17**
5.1 Machine Learning Algorithm Comparison 17
5.2 Tuning the Random Forest and Gradient Boosting Regressors 18
5.3 Impact of Data Segment Size M 19
5.4 Impact of Training Sequence Length T 21
5.5 Impact of Individual Features 22
5.6 Complexity of the Proposed Algorithm 24
5.7 Discussion and Future Direction 24
6 Conclusion **25**
References **25**

List of Tables

Table 1 MSE values of various regression algorithms for SIR_{AP} 17
Table 2 Variance score values of various regression algorithms for SIR_{AP} 18
Table 3 MSE of Random Forrest regression parameters for SIR_{AP} 18
Table 4 MSE of Gradient Boosting regression parameters for SIR_{AP} 19

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

List of Figures

Fig. 1 Robot force-seeking spring system with controlled wireless channel emulation and interference injection. 11
Fig. 2 Sample time series of probe position in mm 14
Fig. 3 Feature extraction model 15
Fig. 4 The mean squared error performance for SIR_{AP} of various algorithms against M. 20
Fig. 5 The variance score performance for SIR_{AP} of various algorithms against M. 20
Fig. 6 The mean squared error performance for SIR_{AD} of various algorithms against M. 21
Fig. 7 The mean squared error performance for SIR_{AP} of various algorithms against T. 22
Fig. 8 The mean squared error performance for SIR_{AP} of XGBM algorithm against M including individual features impact. 23
Fig. 9 The variance score performance for SIR_{AP} of XGBM algorithm against M including individual features impact. 23

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

1. Introduction

Industrial wireless systems (IWSs) are being deployed in various industrial environments due to the continued requirements pressure of cyber-physical systems (CPSs). Application domains for IWSs include flexible manufacturing, safety, process control, alerting and monitoring [1]. The advantages of deploying wireless communications in industrial applications include ease of scale, flexibility, and lower cost compared to the wired counterpart. Nevertheless, there are challenges in wireless deployments [2–4]. The main cause of these challenges is the unpredictable and random nature of wireless channels. In addition, the quality of wireless data communications is impacted by various wireless channel impairments such as path loss, fading, multi-path, and interference. As a result, a careful designing process of wireless communications and control networks is required to deal with these impairments [5, 6].

One of the major challenges in designing IWSs and the underlying industrial systems is interference detection and mitigation. Interference can result from various narrow-band or wide-band sources including coexisting wireless systems, intentional jamming sources, and non-communications devices such as industrial equipment and microwave ovens [7]. Interference can degrade the communication quality of service (QoS) significantly and hence IWS designers consider various interference management techniques.

In this article, we investigate the impact of various machine learning regression algorithms on indirect signal to interference ratio (SIR) estimation using robot position information, where the force feedback signal is transmitted over a wireless local area network (WLAN) deploying IEEE 802.11b/g/n [8]. We also endeavor to understand the impact of various robot arm movement features. Specifically, the position data from a vision-based tracking system is used to train a channel quality estimator which infers the SIR experienced by both the wireless access point and the wireless station used within the testbed. Five different robot arm movement features are extracted from the position data that is captured by the vision system.

In this work, we apply the proposed SIR estimation approach using machine learning in a force-seeking scenario as an example of a task that is commonly found in robotic industrial workcells. Generally, a force-seeking task is used when a robot needs continuous feedback to be aware of the physical environment where it is doing its job assignments such as placing a peg in a tight hole, or when a touch or collision should be detected for another action to start [9]. The force-seeking task is deployed by many higher-level applications such as finishing, object detection, grinding, part differentiation, and precision part fit-up [9, 10]. One of the main advantages of using the robot force-seeking capability in various applications is reducing the number of sensors necessary on a workcell for performing the corresponding task [10].

Furthermore, the proposed wireless interference level estimation is a general approach that can be applied in different industrial use cases. The approach follows the idea of using a data-driven anomaly detection and estimation approach in industrial control systems [11]. In this approach, a system model is trained for the industrial use case, while performing

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

under different conditions, using a machine learning technique. A prediction or estimation technique is then applied to estimate the system state while executing a certain task.

1.1 Motivation

The study of the impact, evaluation, and suppression of interference on industrial control systems has been investigated in [12, 13] and the references therein. Interference level in an industrial environment can be estimated either using the received signal strength at various nodes of the system or performing spectrum measurements through distributed nodes [14, 15]. Although both these approaches can achieve a good estimation result statistically, their real-time performance largely relies on the system's capabilities of transferring data between distributed nodes and processing of data from multiple sources. In [16], for instance, it was stated that implementing these conventional estimation approaches is often constrained by limited hardware capabilities, both from the sensing and the processing point of view. Generally, using received signal strength indicator (RSSI) or any similar energy-based interference estimation techniques require energy sampling with low complexity platforms which impairs signal detection accuracy due to several reasons. First, the operating frequency of the on-board microcontroller caps the energy sampling rate. More specifically, as the RSSI is typically a heavily-filtered real valued version of raw I/Q samples, it undergoes a significant loss of information on the envelope of the acquired signals [17].

In addition, the interference identification tasks can be rather time-consuming because of the harmonization of the scanning and processing time per the network schedule [16]. Also, interference estimation in the discussed use case, where industrial robots use their embedded wireless communications components, may require the collected data to be transmitted wirelessly, which leads to more congestion to the network and the transmission can be unreliable over wireless as well. Moreover, these conventional schemes measure the interference level without indicating its impact on the physical processes, where in many cases, interference may not lead to degradation in the physical process performance [18].

To the best of our knowledge, we are the first to introduce wireless interference estimation using physical process measurements. Specifically, our method introduces a localized level of interference estimation, which depends on the performance degradation at a single physical equipment of interest, namely in this work, a robot. Moreover, the proposed scheme can help in the process of interference identification and estimation, which improve interference characterization such as the scheme in [19], where estimating the wireless network time-frequency characteristics can enable dynamic spectrum access, interference-aware routing, or enhanced network coexistence. In this work, we introduce an approach of estimating the SIR through monitoring its impact on the physical process in an industrial wireless use case. In the results, we focus on comparing the proposed ML-based approach estimated SIR values with the ground truth SIR values in order to measure the performance gap of the approach. Other SIR estimation methods cannot obtain better estimations than the ground truth values, and hence, we compare the performance of the ML-based approach

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

to these ground truth values.

1.2 Contributions

We summarize the contributions of this paper as follows:

1. Our testbed approach demonstrates the effectiveness of ensemble-based machine learning for physics-based interference detection in CPSs such as the manufacturing workcell.

2. The superior performance of ensemble-based supervised learning algorithms for interference estimation in the chosen scenario is made clear.

1.3 Organization

Our paper is organized as follows: in Section 2, the related work is discussed. In Section 3, the use case and the testbed setup are briefly presented. In Section 4, the data processing phases are explained. Later, the results of the machine learning approach are presented in Section 5. In Section 6, we discuss the conclusions and future work.

2. Related Work

In the literature, two types of interference signals are considered, namely, intentional and unintentional interference. Reliable IWSs require dealing with interference through various approaches of assessment and mitigation. Machine learning has been widely used to detect and estimate interference information to enhance the performance of interference management algorithms.

2.1 Deploying WLANs in Industrial Applications

Industrial wireless networks are often utilized for the control of physical processes that demand strict performance requirements such as high reliability, very low latency with deterministic guarantees [20]. Current industrial wireless technologies, such as ISA100.11a and WirelessHART, are based on IEEE 802.15.4 and can achieve industrial system requirements for monitoring and supervisory applications [21]. However, the use of industrial wireless for closed-loop control and robotics applications requires much higher data rates and better coverage than the current IEEE 802.15.5-based wireless technologies. IEEE 802.11-based wireless networks can offer many benefits for industrial communications while, in its current available form, faces the limitations of the lack in determinism and insufficient reliability, due its randomness and contention-based nature [22]. In [22–24] and the references therein, multiple solutions are offered for reliable and deterministic communications using IEEE 802.11-based technologies through applying changes to the physical (PHY) and medium access control (MAC) layers. Moreover, next generation WLAN, such as IEEE 802.11ax, is based on orthogonal frequency division multiple access (OFDMA),

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

which offers a non-contention-based multiple access scheme and provides additional benefits for industrial communication [25]. In this work, we deploy IEEE 802.11-based technology to achieve the required data rate for the force feedback in the proposed robotic use case. In this case, we use the proposed machine learning approach to estimate the interference level, and hence, the wireless link quality and reliability are estimated.

2.2 Interference Analysis in CPSs

The interference analysis in cyber-physical systems has been considered in multiple works for various scenarios. In [26], in-network interference mitigation techniques are discussed for ultra-reliable low-latency wireless communications systems. The paper focused on mutual interference mitigation in an industrial automation setting, where multiple transmissions from controllers to actuators interfere with each other. In [27], an interference mitigating receiver architecture is proposed. The application scenarios are smart homes and modern factories where dense wireless communications devices exist. Moreover, in [28], interference cancellation of transmissions from neighboring cells in a 5G cellular network is presented. In [29], a dedicated node is used for link quality estimation (LQE), such that received data packets are used to identify interference without introducing additional traffic to the network. The use of LQE estimation, spectrum analysis, and traffic troubleshooting in IWSs to study interference impacts is demonstrated in various articles such as [30, 31].

2.3 Impacts of Jamming on CPSs

On the other hand, intended interference (i.e., jamming) can lead to service denial or poor performance in wireless networks. In [32], a literature review is presented that includes an overview of recent research efforts on networked control systems under denial-of-service attacks such as jamming attacks in wireless channels. One of the discussed challenges is how to achieve ultra-reliable low-latency signaling within industrial applications. In [33], a discussion is also provided on the recent developments concerning the design of attack-resilient control and communication protocols. Generally, a jamming attacker can block transmission of packets by emitting strong interference signals to a wireless channel [34, 35]. Jamming attacks can target various wireless technologies and can be a major concern for control systems, since they are easy to launch [35]. It is shown in [36] that off-the-shelf hardware can be used for generating jamming attacks on wireless networks. In cases of physical-layer attacks, the jammer targets a frequency band and is not required to follow the wireless protocol where it can cause a decrease in the SIR, thus preventing the receiver from successfully detecting transmitted packets [36]. In the case of medium access control (MAC)-layer attacks, both the packet sender and the jammer operate on the same channel; the jammer's goal is to cause packet collisions. In [37], the authors evaluate the CPS resilience to jamming attacks that disrupt wireless communications. They considered three jamming strategies, namely, constant, random, and protocol-aware jamming. They show, through experimental results, that various CPS control schemes can be impacted by constant and random jamming while only time-triggered control schemes can be impacted

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

by protocol-aware jamming. Moreover, the resilience of CPSs to periodic jamming signals is considered in [38]. Additionally, resilience to interference in general is considered in [39–41].

2.4 Machine Learning for Jamming Detection

Machine learning has been used for the detection and estimation of jamming attacks. In [42], an unsupervised machine learning algorithm based on a multi-layer auto-encoder is used to extract the interference source spectrum features. These features are then used to distinguish the interference sources' types and locations without labeling measured data. In [43], an unsupervised approach using a recurrent neural network was used to detect anomalies in the CPS performance and identify attacked sensors. In [44], a behavior-based machine learning intrusion detection approach is proposed to detect attacks at the physical process layer. The results are validated through the experimental study of a real, modern water treatment facility. In [45], the viability of machine learning methods in detecting the new threat scenarios of command and data injection is assessed. In that work, command and control communications in a critical infrastructure setting are monitored and vetted against examples of benign and malicious command traffic to identify potential attack events. In [46], the authors assessed discriminating types of power system disturbances through machine learning by detecting jamming attacks. They evaluated various machine learning methods as disturbance discriminators and discuss the practical implications for deploying machine learning systems as an enhancement to existing power system architectures.

2.5 State Estimation for Industrial Control Systems

The SIR estimation can be considered as an example of the wireless network state estimation, which in this case a subsystem of an industrial control system. Hence, the use of the state estimation techniques is generally possible, however, it is hard to be implemented due to the complexity of the system. As a result, a ML-based approach is proposed. Two general state estimation schemes can be used for industrial control systems, namely, model-driven and data-driven [47]. The model-driven state estimation methods require known system information. With the development of sensor technologies and data fusion technologies, the data-driven methods become more important for characterizing systems that cannot be described by models [48]. However, data-driven methods can lead to poor characterization accuracy when data is insufficient or of low quality and hence combining both the methods in hybrid-driven state estimation methods can improve the performance of state estimation [49]. Multiple publications discuss the state estimation of distributed dynamic systems using data-driven methods by deploying machine learning and artificial intelligence schemes such as [50, 51] and the references therein. Also, the use of a data-driven state estimation method for a time-varying uncertain dynamical network can be found in [52] where the effect of communication transmission delays is considered in the system state estimation between adjacent nodes in the network. However, the estimation

of the communications network state by utilizing measurement of the physical system has not been discussed in the literature. In this work, we propose a machine learning based approach for estimating the wireless link quality based on the robot position measurements where we validate our approach through an experimental study.

2.6 Conclusions of Related Work

In the literature, it was shown that impact of link quality on the physical performance of CPS cannot be fully studied by observing LQE. As a result, studying the performance of the physical components performance, while communicating over wireless links, complements the study of CPS performance over wireless. In this work, we deploy observations from the physical system to estimate the interference level, and hence, the quality of the wireless link through an experimental study.

This proposed algorithm and the obtained results can be beneficial to a wireless CPS in different levels. Typically, interference estimation is performed through spectrum monitoring systems where information can be transferred to control systems to react accordingly. The proposed wireless channel SIR estimation approach allows the robot arm control system to react to the estimated interference levels without the need of a probe to directly measure the interference. On the other hand, the obtained results show an example of the realistic wireless interference impact on the movement of a robotic system where theoretical and simulation-based studies cannot capture all aspects of practical industrial systems.

3. Robot Arm Force-Seeking Application

In this section, we discuss the testbed components, the experimental setup, and the radio frequency (RF) environment. The robot force-seeking apparatus is shown in Figure 1.

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

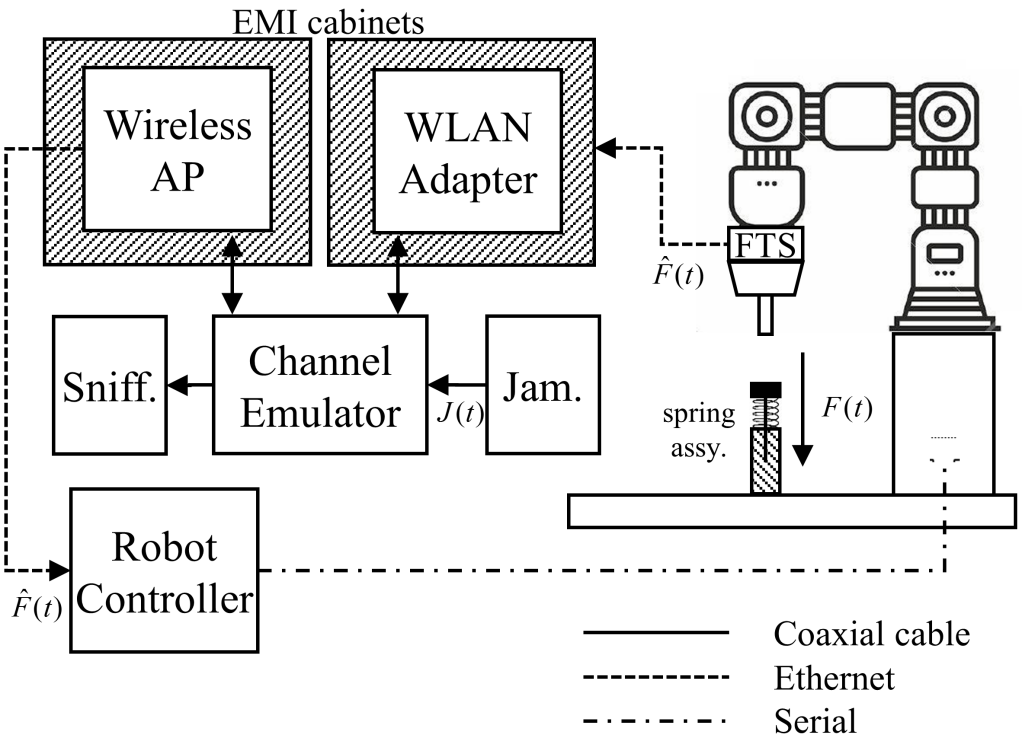

Fig. 1. Robot force-seeking spring system with controlled wireless channel emulation and interference injection.

3.1 Testbed Components

The system is composed of the following components:

- Physical Process Components: The physical process is composed of a robot arm and a spring assembly. The robot arm system has three main physical components, namely, the robot arm, the robot controller, and the force-torque sensor (FTS). The robot arm has 6 degrees-of-freedom and its joint motion is controlled by the robot controller with the control signal transferred over a fieldbus cable. The robot arm is equipped with a probe and a FTS where the probe is used to push the spring assembly vertically and the FTS provides the robot controller with force readings on the three Cartesian axes at a rate of 125 Hz. The FTS is connected to the controller over the wireless network.

- Wireless Networking Devices: The FTS readings to the robot controller are transferred over a wireless 802.11b/g/n network. The FTS compute box has an Ethernet interface which is connected to an Ethernet-to-WiFi adapter. On the other side, the wireless access point (AP) is connected directly to the Ethernet interface of the robot controller.

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

- Channel Emulator: In order to have precise control over the RF environment, a channel emulator is deployed between the wireless adapter and the AP. We deployed a radio frequency channel emulator capable of replicating the multipath and path loss environment for a mesh network of up to 8 physical nodes and 56 virtual links between those nodes. The channel emulator was RFnest D508, and the corresponding software was RFview [53]. The channel emulator supports an instantaneous bandwidth of 250 MHz (4 ns tap spacing) with an effective dynamic range of 73 dB, which includes all analog and digital realization impacts. The emulator is controlled by a nearby computer, which loads the path loss model and channel impulse response for each communications link. In this work, we used it to obtain various interference levels at different nodes, as discussed in the results section.

- Jammer: A vector signal generator is deployed to inject a white Gaussian noise (WGN) interference signal for the testbed. Interference level is controlled by the RF channel emulator to produce precise SIR levels.

- Vision Tracking System: The tracking system is deployed to obtain the accurate ground-truth positions of the robot arm during the experiments at a frame rate of 120 frames per second.

The used wireless protocol between the FTS to the robot controller is IEEE 802.11b/g/n. The traffic to the wireless node is generated by a proprietary application layer protocol over a transmission control protocol (TCP) connection. We found using Wireshark that the data stream includes packets that contains the sensing information of payload lengths ranging from 4 to 28 Bytes at a rate of 125 Hz at the application layer. The packets are transmitted with no acknowledgement (ACK) required to keep the real-time connection. We kept the remaining wired communication links between other components as shown in Figure 1. However, the proposed approach is not impacted by the used protocols since the ML model is being trained using various interference power levels while deploying the same exact communication protocols between different components. Specifically, the proposed approach can estimate the interference power level based on its impact on the physical process and hence the best performance of the proposed approach is achieved in the range of interference power where the physical performance is degraded. However, it cannot achieve the same performance while the interference power is low that it does not impact the physical process or high that it totally stops the process. These ranges for a specific use case are fully dependent on the communications protocols at various layers. Moreover, the approach performance also can be impacted by the communications protocols on different layers based on their interference resistance and error correction properties. As a result, in a specific use case, we train the model by the performance against various SIR values and hence the trained regression model is used for estimating the operational SIR values while the testbed is performing.

Moreover, interference estimation using data from industrial wireless robots, in general, may require the collected data to be transmitted wirelessly, which leads to more congestion

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

to the network and the transmission can be unreliable over wireless as well. Moreover, the use of independent remote monitoring system allows the wireless health information of distributed nodes to be evaluated, enabling improved applicability of the proposed approach in future. As a result, the idea of robot position tracking using an accurate vision tracking system is proposed. In order to collect accurate position information, we deployed a vision tracking system. We deployed the OptiTrack V120 Trio with three cameras, frame rate of 120 frames per second, and 640 x 480 resolution per camera [54]. The tracking system is connected to a computer for data acquisition and processing at which the real-time position information is tracked and stored. We used the supervisory computer for controlling the vision system in order to synchronize the beginning of tracking data acquisition to the force-seeking experiment start. We have found that the vision tracking error is less than or equal 0.2 mm.

3.2 Experimental Setup

The universal robot UR-3 is programmed to apply force on the spring assembly in a vertical direction as shown in Figure 1. The robot controller receives the periodic force readings from the FTS. Once the applied force reaches a predefined threshold, F_{th}, the robot arm moves back to the home position and starts a new force-seeking cycle.

The robot arm movement is impacted by the received FTS readings in two ways: i) it does not start the the force-seeking cycle until the robot controller zeros the FTS, and ii) the robot arm keeps descending until its force threshold is reached. The controller will continue to move the arm beyond the target force threshold if FTS reports are missed. Similarly, if no FTS reports are received during a period of the time, then the arm motion is interrupted.

3.3 RF Scenario

The channel emulator is used to set up the wireless environment among the three wireless nodes in the system, namely, the wireless adapter at the FTS, the wireless AP, and the jammer. In this work, we study the SIR at both the wireless adapter and the wireless AP. We denote the jammer received power at the AP and the adapter by $P_{\text{J-AP}}$ and $P_{\text{J-AD}}$, respectively. We also denote the adapter received power at the AP by $P_{\text{AD-AP}}$ and the received power at the wireless adapter of the AP transmission by $P_{\text{AP-AD}}$. These power values are defined in dBm. Hence, we can formulate the SIR values in dB at the AP and the wireless adapter, respectively, as follows:

$$SIR_{\text{AP}} = P_{\text{AD-AP}} - P_{\text{J-AP}}, \tag{1}$$

$$SIR_{\text{AD}} = P_{\text{AP-AD}} - P_{\text{J-AD}}. \tag{2}$$

4. Data Analysis

In this section, we present the collected data format and all the data processing steps performed in this experimental study. The data collection is performed using the vision track-

ing system. The data are processed using Matlab for feature extraction where these features are used in SIR estimation. Afterwards, the Sci-kit Learn library [55] over Python is used to train various types of regressors and estimate SIR levels given the extracted features.

4.1 Collected Data

The robot arm moves vertically to push the spring assembly. We denote the position of the robot arm probe by Z_{d} which is a function of time t. The time series is denoted by $Z_{\mathrm{d}}(t)$ where an example of one robot arm movement cycle is shown in Figure 2. The time series data are collected continuously using the vision tracking system for a period T. Each collected time series is labeled by the corresponding SIR values at the wireless AP and adapter, SIR_{AP} and SIR_{AD}, respectively.

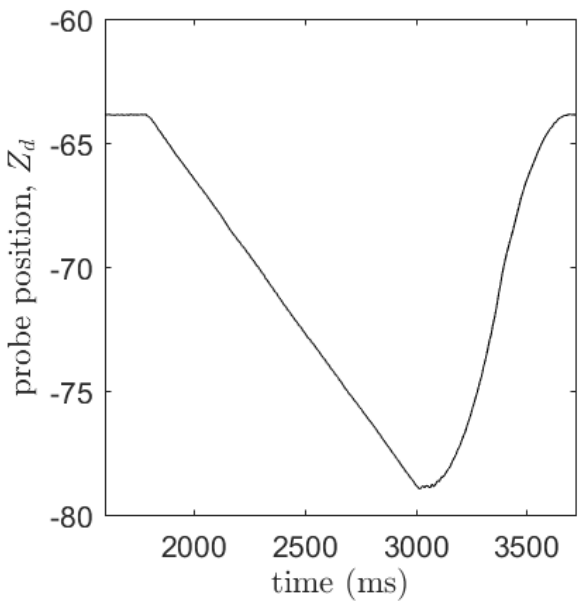

Fig. 2. Sample time series of probe position in mm

4.2 Feature Extraction

The use of time series features in machine learning is a more practical approach since it is less computationally expensive than using the time series samples. The robot arm repeats the force-seeking cycles and therefore we slice the time series data into cycles. Each cycle starts as the probe reaches the home position which is denoted by Z_{H}. The data of time series of length T are split into N cycles such that the nth cycle occurs between T_{n-1} and T_n, where T_0 equals 0. In each of these force-seeking cycles, the various time intervals of robot arm actions and the vertical movement range are selected to be studied.

The first obtained feature is the length of the robot arm movement range in mm, which is denoted by $\tilde{Z}_{\mathrm{d}}$. This feature is impacted by wireless communications such that the probe

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

can go further down if the force level reached the threshold while the FTS information is delayed or lost. In the nth cycle, the value of $\tilde{Z}_{\mathrm{d}}$ is defined as follows

$$\tilde{Z}_{\mathrm{d}} = Z_{\mathrm{H}} - \min_{t \in [T_{n-1}, T_n]} Z_{\mathrm{d}}(t). \tag{3}$$

As an example, shown in Figure2, let Z_{H} equal -64 mm which means that the probe starts at a point 6.4 cm below the reference point that is at z=0. The probe in the nth cycle between the times T_{n-1} and T_n should move downwards until the force reading in the z-axis is larger than F_{th}, and hence, the robot stops and starts moving in the upward direction. Let the value at which the robot stops equal -78 mm. The value of $Z_{\mathrm{d}}(t)$ through this specific cycle is the one shown in Figure 2 as a function of time. As a result, the value of min $Z_{\mathrm{d}}(t)$ in (3) equals -78 mm and the value of $\tilde{Z}_{\mathrm{d}}$ equals 14 mm. This value of $\tilde{Z}_{\mathrm{d}}$ changes depending on the communications delay, for which the robot receives the force value to stop its downward movement and pivot to the upward movement. The values of $\tilde{Z}_{\mathrm{d}}$ over time are collected to represent the robot arm movement ranges in various cycles, which depends on the communications quality within these cycles.

The rest of the extracted features are defined to be the various time intervals spent by the robot arm moving in a certain movement range. In order to define these features, we set a threshold Z_{th} at which the robot arm probe reaches a force level F_{th}, e.g., Z_{th} is the ideal deflection position with no communications impairments. We start by defining various points in the cycle where the robot arm changes its taken action. The points are shown in Figure 3. Point **a** is the cycle beginning point, point **b** is the point at which the probe begins to descend, point **c** is the point at which the probe reaches Z_{th} while descending, point **d** is the point at which the probe reaches Z_{th} while ascending after being below Z_{th}, and point **e** is the cycle end point which is exactly point **a** of the following cycle.

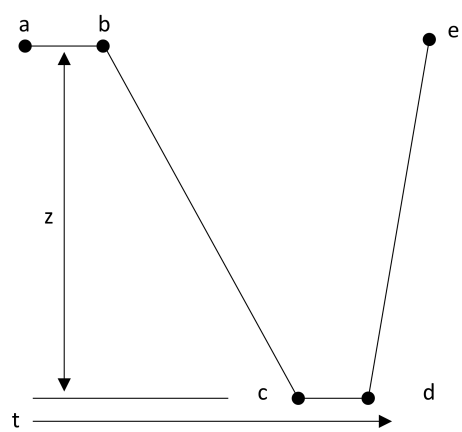

Fig. 3. Feature extraction model

The total cycle time, $\Delta t_{\mathbf{ae}}$, is a considered feature where a higher interference level is expected to increase the total cycle time. In the nth cycle, the value of $\Delta t_{\mathbf{ae}}$ is defined as

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

follows

$$\Delta t_{\mathbf{ae}} = T_n - T_{n-1}. \tag{4}$$

The time spent at the home position , $\Delta t_{\mathbf{ab}}$, is also considered where the robot arm controller communicates wirelessly with the FTS to zero it before the descent. In the nth cycle, the value of $\Delta t_{\mathbf{ab}}$ is defined as follows

$$\Delta t_{\mathbf{ab}} = \min_{t \in [T_{n-1}, T_n], Z_{\mathrm{d}}(t) < Z_{\mathrm{H}}} t - T_{n-1}. \tag{5}$$

The dwell time, while F_{th} is reached, is denoted by $\Delta t_{\mathbf{cd}}$. In order to evaluate the value of $\Delta t_{\mathbf{cd}}$ in the nth cycle, we start by calculating

$$t_{\mathbf{c}} = \min_{t \in [T_{n-1}, T_n], Z_{\mathrm{d}}(t) \leq Z_{\mathrm{th}}} t, \tag{6}$$

and then the value of $\Delta t_{\mathbf{cd}}$ is defined as

$$\Delta t_{\mathbf{cd}} = \min_{t \in [t_{\mathbf{c}}, T_n], Z_{\mathrm{d}}(t) > Z_{\mathrm{th}}} t - t_{\mathbf{c}}. \tag{7}$$

The descend time, $\Delta t_{\mathbf{bc}}$, is the last feature to include all the independent time intervals within a single cycle. In the nth cycle, the value of $\Delta t_{\mathbf{ab}}$ is defined as follows

$$\Delta t_{\mathbf{bc}} = t_{\mathbf{c}} - \min_{t \in [T_{n-1}, T_n], Z_{\mathrm{d}}(t) < Z_{\mathrm{H}}} t. \tag{8}$$

In conclusion, we consider five features in this study, namely, $\tilde{Z}_{\mathrm{d}}$, $\Delta t_{\mathbf{ae}}$, $\Delta t_{\mathbf{ab}}$, $\Delta t_{\mathbf{bc}}$, and $\Delta t_{\mathbf{cd}}$.

4.3 SIR Estimation

In this work, the SIR estimation problem using the extracted features is considered a machine-learning regression problem. It is a supervised learning problem where the output is an estimated SIR value, and the input is a features vector composed of the robot arm movement features of M cycles. Using a single cycle to estimate the SIR value cannot be accurate, and, hence, we allow the regression algorithm to consider M cycles, such that the input vector length is $5M$. Although increasing M should lead to a better SIR estimation result, it also leads to a higher computation cost for the regression algorithm.

The training for each regression model is performed by considering a fixed number of cycles at each SIR labeled data set. We denote the size of the training set for each SIR level by T. In the regression algorithm, a segment size of M cycles is used to collect the discussed features. In results, we compare various regression algorithms and the impact of their hyper parameters on the performance of the regression problem to obtain the SIR value directly from the physical robot arm movement tracking.

We compare various regression approaches using two performance criteria, namely, the mean squared error (MSE) and the R-squared variance score [56]. The MSE is corresponding to the expected value of the squared error in the SIR value. The R-squared variance

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

score indicates the percentage of the variance in the dependent variable that the independent variables explain collectively. The version of R-squared in the Sci-kit Learn library measures the strength of the relationship between the model and the dependent variable on a convenient -1 to 1 scale. The best possible outcome is 1 when predicted values capture the variance of the independent variable. It takes a value of 0 when the predicted value is constant and negative values when the regression model cannot follow the trend of the data.

5. Results

In this study, IEEE 802.11b/g/n was deployed for wireless data transfer between the FTS wireless adapter and the AP [8]. We evaluate the SIR values at both wireless nodes while varying the jamming interference power such that SIR_{AP} takes the values -10, -9, -8, -7 dB while SIR_{AD} takes the values 1, 2, 3 dB. Except otherwise mentioned, we set $M = 50$ and $T = 200$.

5.1 Machine Learning Algorithm Comparison

In this subsection, various regression approaches are compared using two performance criteria, namely, the MSE and the R-squared variance score [56]. We present the performance for different values of M. The compared algorithms are the random forest, gradient boosting, extreme gradient boosting (XGBM), decision tree, support vector machine (SVM), k-nearest neighbor (KNN), kernel ridge, and linear ridge [55, 57].

In Table 1, the ensemble-based algorithms, which are the random forest, gradient boosting, and XGBM, perform better than the rest of the algorithms. This happens because the ensemble-based algorithms learn from the training data without having an initial model to fit, thus allowing for capturing the randomness impacts on the collected data. Moreover, the XGBM gives slightly better performance than the random forest and gradient boosting algorithms, except at $M = 1$. The XGBM performs gradient boosting over random set of trees, and hence, has superior performance.

Table 1. MSE values of various regression algorithms for SIR_{AP}

	M = 1	M = 10	M = 30	M = 50	M = 100
Random Forest	0.61	0.54	0.49	0.48	0.46
Gradient Boosting	0.56	0.55	0.47	0.45	0.44
XGBM	0.62	0.53	0.45	0.43	0.41
Decission Tree	1.07	0.99	0.87	0.91	0.97
SVM	0.92	0.99	0.98	0.97	0.96
KNN	0.73	0.74	0.73	0.74	0.77
Kernel Ridge	0.97	0.73	0.71	0.71	0.71
Linear Ridge	0.7	0.7	0.7	0.71	0.71

The KNN and SVM algorithms performance does not improve against M. The main

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

reason is that both algorithms perform regression based on distance [58], and hence, are impacted by the ranges of features used. The higher values features are the ones that impact the performance of these algorithms the most. In our case, the features with the highest ranges are the total cycle time and the descent distance, which are found to be the ones with the least impact on the results, as shown later in Figure 9.

We also find that it has superior variance score in Table 2. Generally, Table 2 shows similar trends as Table 1 for various algorithms where ensemble-based algorithms have the best performance, and their performance is enhanced by increasing the segment size M of the measured data.

Table 2. Variance score values of various regression algorithms for SIR_{AP}

	M = 1	M = 10	M = 30	M = 50	M = 100
Random Forest	0.15	0.2	0.3	0.35	0.35
Gradient Boosting	0.02	0.2	0.34	0.37	0.42
XGBM	0.12	0.25	0.36	0.4	0.43
Decission Tree	-0.57	-0.44	-0.36	-0.3	-0.22
SVM	-0.34	-0.38	-0.37	-0.37	-0.32
KNN	-0.05	-0.02	-0.02	-0.08	-0.04
Kernel Ridge	-0.48	-0.05	0	0	0
Linear Ridge	0.01	0.01	0	-0.01	0

5.2 Tuning the Random Forest and Gradient Boosting Regressors

In this subsection, we study the impacts of the number of estimators and the depth for gradient boosting and random forest algorithms. The XGBM parameters are optimized automatically when the function is called to be executed. We show the MSE performance results for SIR_{AP} while the same trend holds for SIR_{AD} as well.

For the random forest algorithm, we show in Table 3 that the tree depth has more impact on the performance than the number of the estimators (# est.). However, increasing the depth improves the performance until the value of 5, where increasing the depth causes slight improvements.

Table 3. MSE of Random Forrest regression parameters for SIR_{AP}

Depth# est.	100	200	300	400	500	600
1	0.64	0.64	0.64	0.63	0.63	0.63
3	0.53	0.53	0.53	0.53	0.53	0.52
5	0.5	0.5	0.5	0.48	0.48	0.48
7	0.49	0.49	0.48	0.48	0.48	0.48
9	0.48	0.48	0.47	0.47	0.47	0.47

For gradient boosting, we show in Table 4 that a similar trend as a random forest exists where depth has a larger impact on the performance than the number of the estimators.

However, increasing the depth more than 7 causes the MSE performance to degrade significantly.

Table 4. MSE of Gradient Boosting regression parameters for SIR_{AP}

Depth# est.	100	200	300	400	500	600
1	0.48	0.47	0.46	0.46	0.45	0.46
3	0.47	0.47	0.47	0.47	0.46	0.47
5	0.48	0.47	0.47	0.47	0.47	0.47
7	0.48	0.48	0.48	0.48	0.46	0.49
9	0.58	0.51	0.5	0.5	0.57	0.58

5.3 Impact of Data Segment Size M

In this subsection, we study the performance of the ensemble-based algorithms against the segment size M. Generally, increasing the value of M increases the acquisition time of measurement data used for decision making and lowers the spread of predicted values around the correct value for various SIR values. In this paper, we study the impact of the three optimized algorithms employing the MSE and the variance score as performance metrics.

In Figure 4, we present the performance of the random forest, gradient boosting, XGBM, and the linear ridge regressors. The linear ridge is used as a simple model-based regressor, which practically cannot be used for prediction while it is used for comparison. In this figure, increasing the value of M enhances the performance of the ensemble-based algorithms significantly. Generally, the prediction accuracy increases when multiple measurement cycles are deployed in decision making with diminishing improvement as M increases.

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

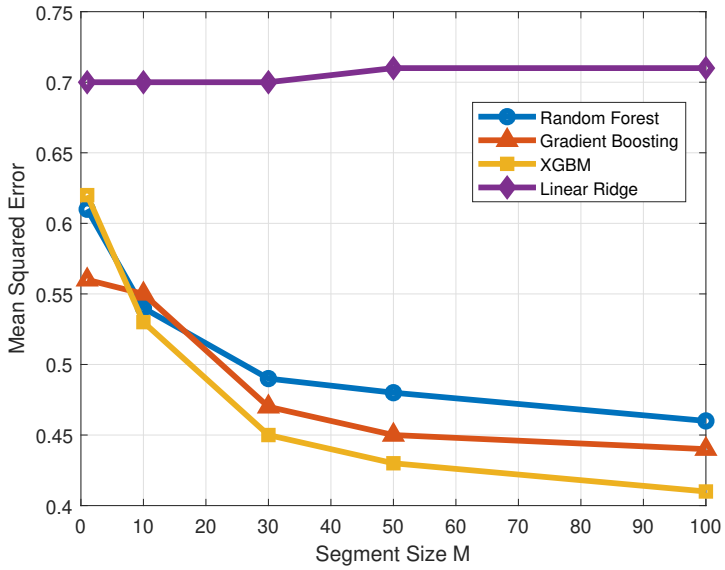

Fig. 4. The mean squared error performance for SIR_{AP} of various algorithms against M.

In Figure 5, we present the variance score for SIR_{AP} where a similar trend is found. The ensemble-based algorithms can achieve a variance score above 0.35 when M is larger than 50.

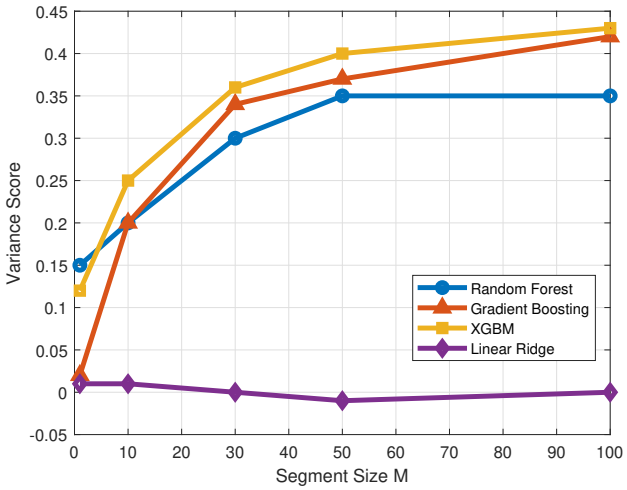

Fig. 5. The variance score performance for SIR_{AP} of various algorithms against M.

In Figure 6, we present the MSE performance of SIR_{AD} for the random forest, gradient boosting, XGBM, and the linear ridge regressors when the wireless adapter is impacted by the jamming signal. In this figure, the impact of the segment size is studied where the performance levels of the ensemble-based algorithms improve by increasing the value of M for

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

$M \leq 50$, after which the improvement diminishes. The prediction accuracy increases when multiple measurement cycles are deployed in decision making and no further improvement happens when correlation between samples is lost. Comparing Figure 6 to Figure 4, the MSE values for SIR_{AD} are lower than those for SIR_{AP} because the range of the predicted SIR values is smaller.

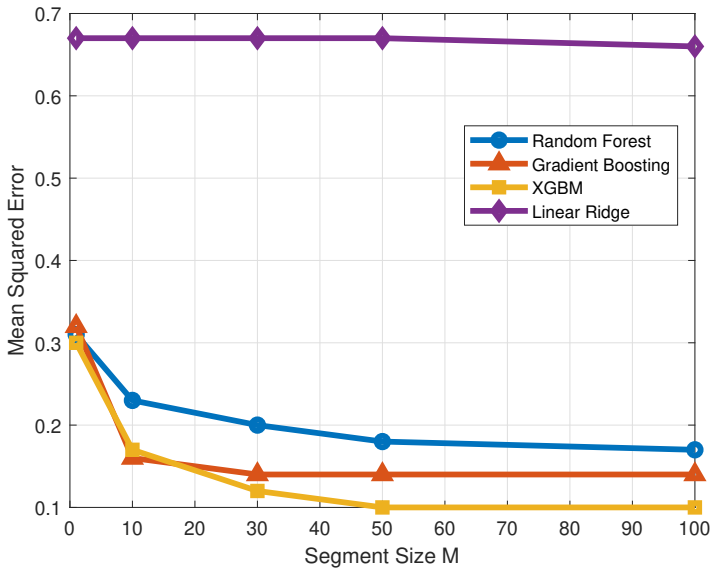

Fig. 6. The mean squared error performance for SIR_{AD} of various algorithms against M.

In this experimental study, we allowed the interference power to be fixed over a long period of time to allow us to study the impact of the segment size, M, on the performance. In real conditions, the impact time of a specific interference power level varies where the interference duration depends on the duty cycle of interference source. Hence, in this work, we have shown the results of the proposed approach over various window sizes starting by $M = 1$ to $M = 100$. Although the best performance of the approach is achieved for larger values of M, as expected, the results show how the approach performs in smaller values of M, which can be beneficial in some more realistic scenarios where interference power is fixed over a short period of time. Moreover, the study of the problem of estimating time-varying interference impact is another problem which includes detecting both the interference power level and its temporal characteristics which can benefit from the current approach but not a direct extension of it.

5.4 Impact of Training Sequence Length T

In this subsection, we briefly discuss the impact of the training length on the performance of various algorithms. The parameter T is the length of the training sequence where each element contains M of the force seeking cycles. In the following figures, we show the MSE performance curves against T for M = 30 and 100.

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

In Figure 7, increasing the training size T can improve the prediction performance significantly. Gradient boosting and XGBM have higher improvement rates with T than the random forest algorithm.

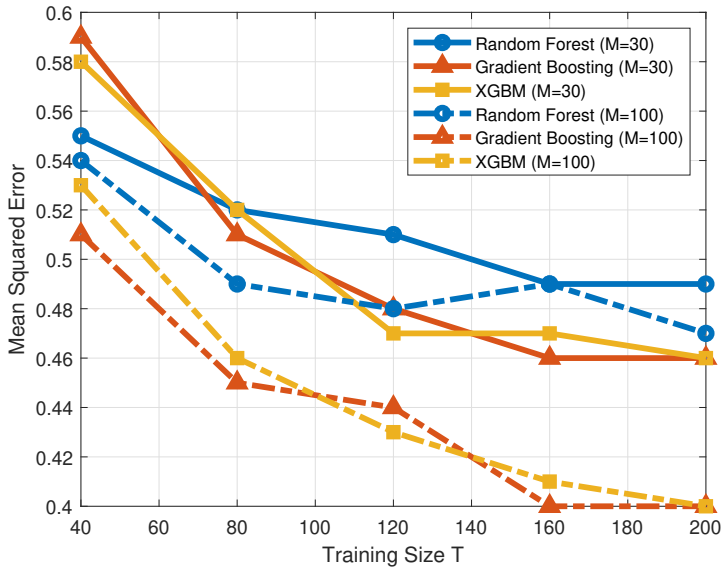

Fig. 7. The mean squared error performance for SIR_{AP} of various algorithms against T.

5.5 Impact of Individual Features

In this subsection, we study the impact of the individual features on the performance of the ensemble-based machine learning algorithms. Understanding the importance of each feature on the prediction algorithms is essential in selection of features and hence reducing the required processing power for an algorithm. We show the results for the XGBM algorithm for brevity and the similarities in behavior of various ensemble-based algorithms performance. In Figure 8 and 9, we refer to $\Delta t_{\mathbf{ab}}$, $\Delta t_{\mathbf{bc}}$, $\tilde{Z}_{\mathrm{d}}$, $\Delta t_{\mathbf{cd}}$, and $\Delta t_{\mathbf{ae}}$ by "t_high", "t_plunge", "Descent Distance", "t_bottom", and "Total Cycle", respectively.

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

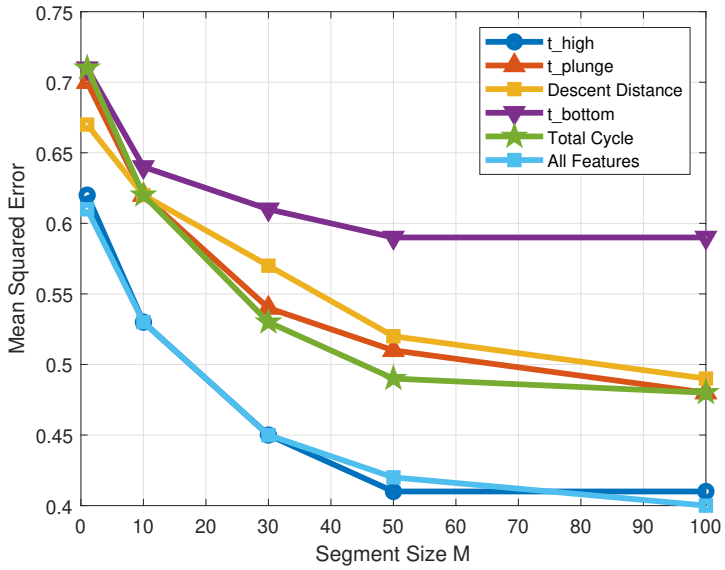

Fig. 8. The mean squared error performance for SIR_{AP} of XGBM algorithm against M including individual features impact.

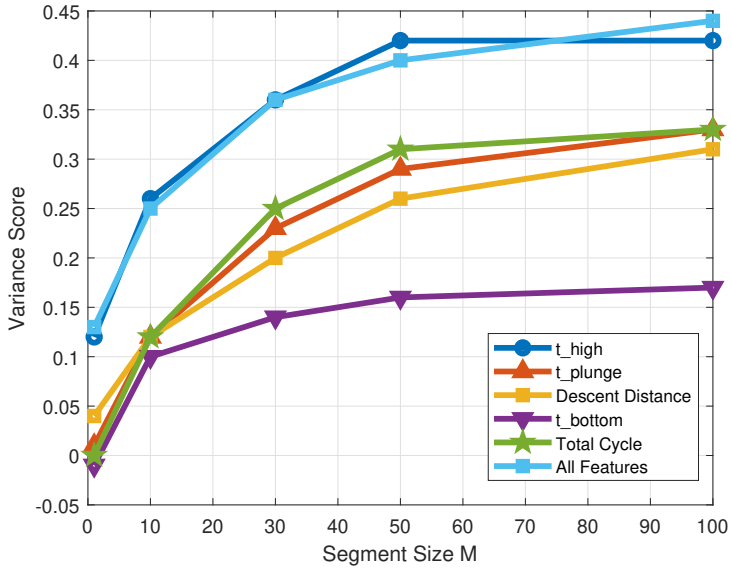

Fig. 9. The variance score performance for SIR_{AP} of XGBM algorithm against M including individual features impact.

The feature $\Delta t_{\mathbf{ab}}$ is the one that is most impacted by the SIR value where the performance using all features can be almost entirely predicted by the performance using only $\Delta t_{\mathbf{ab}}$. This result can be explained by knowing that in the robot arm force seeking program,

the FTS has to be zeroed before resuming the loop, and hence, there is a direct impact of the wireless transmissions on the value Δt_{ab}. On the other hand, the feature Δt_{bc} is the least impacted by the SIR value. This feature is defined as the reflection time of the robot arm to reverse its direction, and hence, it is minimally impacted by the SIR value. The rest of the features are impacted by the SIR to a certain level. Generally, adding an unnecessary feature can degrade the performance. As a result, in this work, we concluded that using $\Delta t_{\mathbf{ab}}$ can substitute for all the five features to get the same MSE and variance score performance with much less processing.

5.6 Complexity of the Proposed Algorithm

The results of this work show that ensemble-based approaches are suitable for estimating the SIR of the wireless link used by the force-seeking use case. The proposed ensemble-based techniques mainly depend on the use of decision and gradient trees for solving the machine learning regression problem. Each of the techniques deploys the training data by generating a large number of data trees. However, the construction process of the trees is different for different techniques. The average asymptotic training complexity of all of them is $O(Kdn\log(n))$, where K is the number of the trees (# est. in the results), d is the maximum depth of the trees, and n is the number of data points in the training data sets [59, 60]. Although, their training complexity is similar, the random forest complexity may be reduced when parallelization of the trees is utilized [60]. Also, XGBM can achieve better performance than Gradient boosting, because of the automatic optimization of the hyper parameters and the lower complexity achieved in the case of the missed data entries [59]. The average asymptotic complexity for the prediction using all the schemes is $O(Kd)$ [60]. Similarly, this can be lowered when using random forest penalization. Note that, tree boosting algorithms are not able to be computed in parallel as their algorithms run sequentially for each gradient tree to boost the performance of the subsequent tree.

5.7 Discussion and Future Direction

This work represents a building block for the use of physical process performance data to estimate the state of the wireless networks. In [61], CPSs require the monitoring of network health information in a real-time and continuous manner in order to maintain the appropriate performance. However, as CPSs become more complicated and faults are increasingly diverse, the traditional methods for CPS anomaly detection and network performance estimation become less effective [62]. This motivated the use of ML-based approaches that do not rely on domain-specific knowledge [62, 63]. These ML-based approaches have been widely used in CPS anomaly detection related to various types of attacks [62]. However, we propose to use a similar approach in wireless network health monitoring based on the measured physical performance quantities, such as the robot arm position in this case. This work proposed a prototype of the ML-based approach in a simple use case to better understand the process.

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

The main direction of future work is to deploy this idea to complement the existing wireless networks performance measurements system in order to standardize a framework for assessment of the impact of wireless network impairments on industrial IoT systems. Future directions include applying machine learning approaches, e.g., neural networks, to larger, more complex systems with higher degrees of freedom. These machine learning techniques can be used to model the industrial process such that wireless interference detection and estimation can be achieved through their impact on various equipment of the physical industrial process. Another research direction is developing the real-time deployment of the machine learning based algorithms and integrating them with current artificial intelligence capabilities in industrial IoT systems. The real-time version of the proposed approach can help the manufacturing community through integration of wireless link quality detection with automation controllers.

6. Conclusion

An experimental study was presented for a feedback control system where a wireless communications link was used for force feedback to a robot controller. It was demonstrated that the reliability of the wireless communication system directly impacted the performance of the physical system. A machine learning system was then trained to predict the SIR level by observing the probe position. Metrics of mean squared error and the R-squared variance score were used to ascertain the effectiveness of the predictor. Based on the experiment results presented in this paper, ensemble-based algorithms were demonstrated to have clear, superior performance over other regression algorithms.

References

[1] Candell R, Hany M, Lee KB, Liu Y, Quimby J, Remley K (2018) Guide to industrial wireless systems deployments (National Institute of Standards and Technology, Gaithersburg, MD), https://doi.org/10.6028/NIST.AMS.300-4. Available at http://nvlpubs.nist.gov/nistpubs/ams/NIST.AMS.300-4.pdf

[2] Sisinni E, Saifullah A, Han S, Jennehag U, Gidlund M (2018) Industrial Internet of Things: Challenges, Opportunities, and Directions. *IEEE Transactions on Industrial Informatics* https://doi.org/10.1109/TII.2018.2852491

[3] Bello LL, Åkerberg J, Gidlund M, Uhlemann E (2017) Guest Editorial Special Section on New Perspectives on Wireless Communications in Automation: From Industrial Monitoring and Control to Cyber-Physical Systems. *IEEE Transactions on Industrial Informatics* 13(3):1393–1397. https://doi.org/10.1109/TII.2017.2695585

[4] Pang Z, Luvisotto M, Dzung D (2017) Wireless High-Performance Communications: The Challenges and Opportunities of a New Target. *IEEE Industrial Electronics Magazine* 11(3):20–25. https://doi.org/10.1109/MIE.2017.2703603

[5] Lu C, Saifullah A, Li B, Sha M, Gonzalez H, Gunatilaka D, Wu C, Nie L, Chen Y (2016) Real-Time Wireless Sensor-Actuator Networks for Industrial Cyber-Physical

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

Systems. *Proceedings of the IEEE* 104(5):1013–1024. https://doi.org/10.1109/JPROC.2015.2497161. Available at http://ieeexplore.ieee.org/document/7348717/

[6] Kim D, Won Y, Eun Y, Park KJ (2017) W-Simplex: Resilient network and control co-design under wireless channel uncertainty in cyber-physical systems. *2017 IEEE Conference on Control Technology and Applications (CCTA)* (IEEE), , pp 49–54. https://doi.org/10.1109/CCTA.2017.8062439. Available at http://ieeexplore.ieee.org/document/8062439/

[7] Chiwewe TM, Mbuya CF, Hancke GP (2015) Using Cognitive Radio for Interference-Resistant Industrial Wireless Sensor Networks: An Overview. *IEEE Transactions on Industrial Informatics* 11(6):1466–1481. https://doi.org/10.1109/TII.2015.2491267. Available at http://ieeexplore.ieee.org/document/7299315/

[8] IEEE Standards Association (2013) *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications; Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz*, . https://doi.org/10.1109/IEEESTD.2013.6687187

[9] Owen-Hill A (2021) The 7 best robot applications for a force sensor, https://blog.robotiq.com/7-great-robot-applications-for-a-force-sensor. Accessed: 2021-8-4.

[10] Goodwin D (2021) Common uses of force control in a robotic cell, https://control.com/technical-articles/common-uses-of-force-control-in-a-robotic-cell/. Accessed: 2021-8-4.

[11] Li D, Ramanan P, Gebraeel N, Paynabar K (2020) Deep learning based covert attack identification for industrial control systems. 2009.12360.

[12] Promsuk N, Taparugssanagorn A (2020) Long short term memory network-based interference recognition for industrial internet of things. *Journal of Communications* :876–885https://doi.org/10.12720/jcm.15.12.876-885

[13] Cheffena M (2016) Propagation channel characteristics of industrial wireless sensor networks [wireless corner]. *IEEE Antennas and Propagation Magazine* 58(1):66–73. https://doi.org/10.1109/MAP.2015.2501227

[14] García-Dorado J, Ramos J, Gomez-Arribas FJ, Magaña E, Aracil J (2020) Instrumentation for measuring users' goodputs in dense wi-fi deployments and capacity-planning rules. *Wireless Networks* 26:2943–2955.

[15] Forenbacher I, Husnjak S, Jovović I, Bobić M (2021) Throughput of an ieee 802.11 wireless network in the presence of wireless audio transmission: A laboratory analysis. *Sensors* 21(8). https://doi.org/10.3390/s21082620. Available at https://www.mdpi.com/1424-8220/21/8/2620

[16] Grimaldi S, Mahmood A, Hassan SA, Gidlund M, Hancke GP (2021) Autonomous interference mapping for industrial internet of things networks over unlicensed bands: Identifying cross-technology interference. *IEEE Industrial Electronics Magazine* 15(1):67–78. https://doi.org/10.1109/MIE.2020.3007568

[17] Konings D, Faulkner N, Alam F, Noble F, Lai EMK (2017) The effects of interference on the rssi values of a zigbee based indoor localization system. *2017 24th Inter-*

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

national Conference on Mechatronics and Machine Vision in Practice (M2VIP), , pp 1–5. https://doi.org/10.1109/M2VIP.2017.8211460

[18] Zoppi S, Shantharam SP, Kellerer W (2020) Delay-reliability model of industrial wsn for networked control systems. *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, , pp 1–7. https://doi.org/10.1109/GLOBECOM42002.2020.9348072

[19] Grimaldi S, Mahmood A, Hassan SA, Gidlund M, Hancke GP (2021) Autonomous interference mapping for industrial internet of things networks over unlicensed bands: Identifying cross-technology interference. *IEEE Industrial Electronics Magazine* 15(1):67–78. https://doi.org/10.1109/MIE.2020.3007568

[20] Galloway B, Hancke GP (2013) Introduction to industrial control networks. *IEEE Communications Surveys Tutorials* 15(2):860–880. https://doi.org/10.1109/SURV.2012.071812.00124

[21] Montgomery K, Candell R, Liu Y, Hany M (2019) Wireless user requirements for the factory workcell. https://doi.org/https://doi.org/10.6028/NIST.AMS.300-8.

[22] Aijaz A (2020) High-performance industrial wireless: Achieving reliable and deterministic connectivity over ieee 802.11 wlans. https://doi.org/10.36227/techrxiv.12046230.v1. Available at https://www.techrxiv.org/articles/preprint/High-Performance_Industrial_Wireless_Achieving_Reliable_and_Deterministic_Connectivity_over_IEEE_802_11_WLANs/12046230/1

[23] (2021) IEEE 802.11 for industrial applications, https://www.bb-elec.com/Learning-Center/All-White-Papers/Wireless-Cellular/IEEE-802-11-for-Industrial-Applications.aspx. Accessed: 2021-8-10.

[24] Sudhakaran S, Montgomery K, Kashef M, Cavalcanti D, Candell R (2021) Wireless time sensitive networking for industrial collaborative robotic workcells. *2021 17th IEEE International Conference on Factory Communication Systems (WFCS)*, , pp 91–94. https://doi.org/10.1109/WFCS46889.2021.9483447

[25] (2021) IEEE 802.11ax high efficiency WLAN (HEW), http://www.ieee802.org/11/Reports/tgax_update.htm. Accessed: 2021-8-10.

[26] Ayoughi SA, Yu W, Khosravirad SR, Viswanathan H (2019) Interference mitigation for ultrareliable low-latency wireless communication. *IEEE Journal on Selected Areas in Communications* 37(4):869–880. https://doi.org/10.1109/JSAC.2019.2898786

[27] Kumar S, Kaltenberger F, Ramirez A, Kloiber B (2019) An SDR implementation of wifi receiver for mitigating multiple co-channel zigbee interferers. *EURASIP Journal on Wireless Communications and Networking* 2019(1):224. https://doi.org/10.1186/s13638-019-1512-3. Available at https://doi.org/10.1186/s13638-019-1512-3

[28] Bhushan N, Junyi Li, Malladi D, Gilmore R, Brenner D, Damnjanovic A, Sukhavasi R, Patel C, Geirhofer S (2014) Network densification: the dominant theme for wireless evolution into 5G. *IEEE Communications Magazine* 52(2):82–89. https://doi.org/10.1109/MCOM.2014.6736747. Available at http://ieeexplore.ieee.org/document/6736747/

[29] Gomes RD, Queiroz DV, Lima Filho AC, Fonseca IE, Alencar MS (2017) Real-

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

time link quality estimation for industrial wireless sensor networks using dedicated nodes. *Ad Hoc Networks* 59:116–133. https://doi.org/10.1016/j.adhoc.2017.02.007. Available at https://linkinghub.elsevier.com/retrieve/pii/S1570870517300434

[30] Baccour N, Koubâa A, Mottola L, Zúñiga MA, Youssef H, Boano CA, Alves M (2012) Radio link quality estimation in wireless sensor networks. *ACM Transactions on Sensor Networks* 8(4):1–33. https://doi.org/10.1145/2240116.2240123. Available at http://dl.acm.org/citation.cfm?doid=2240116.2240123

[31] Koepke GH, Young WF, Ladbury JM, Coder JB (2015) Interference and Coexistence of Wireless Systems in Critical Infrastructure (National Institute of Standards and Technology, Gaithersburg, MD), https://doi.org/10.6028/NIST.TN.1885. Available at https://nvlpubs.nist.gov/nistpubs/TechnicalNotes/NIST.TN.1885.pdf

[32] Ahmadi A, Moradi M, Cherifi C, Cheutet V, Ouzrout Y (2018) Wireless connectivity of cps for smart manufacturing: A survey. *2018 12th International Conference on Software, Knowledge, Information Management Applications (SKIMA)*, , pp 1–8. https://doi.org/10.1109/SKIMA.2018.8631535

[33] Cetinkaya A, Ishii H, Hayakawa T (2019) An overview on denial-of-service attacks in control systems: Attack models and security analyses. *Entropy* 21(2):210. https://doi.org/10.3390/e21020210. Available at http://dx.doi.org/10.3390/e21020210

[34] Wenyuan Xu, Ke Ma, Trappe W, Yanyong Zhang (2006) Jamming sensor networks: attack and defense strategies. *IEEE Network* 20(3):41–47. https://doi.org/10.1109/MNET.2006.1637931

[35] Pelechrinis K, Iliofotou M, Krishnamurthy SV (2011) Denial of service attacks in wireless networks: The case of jammers. *IEEE Communications Surveys Tutorials* 13(2):245–257. https://doi.org/10.1109/SURV.2011.041110.00022

[36] Fragkiadakis A, Askoxylakis I, Chatziadam P (2014) Denial-of-service attacks in wireless networks using off-the-shelf hardware. *Distributed, Ambient, and Pervasive Interactions*, eds Streitz N, Markopoulos P (Springer International Publishing, Cham), , pp 427–438.

[37] Tomić I, Breza MJ, Jackson G, Bhatia L, McCann JA (2018) Design and evaluation of jamming resilient cyber-physical systems. *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, , pp 687–694. https://doi.org/10.1109/Cybermatics_2018.2018.00138

[38] Shisheh Foroush H, Martínez S (2012) On event-triggered control of linear systems under periodic denial-of-service jamming attacks. *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, , pp 2551–2556. https://doi.org/10.1109/CDC.2012.6425868

[39] Persis CD, Tesi P (2014) Resilient control under denial-of-service. *IFAC Proceedings Volumes* 47(3):134 – 139. https://doi.org/https://doi.org/10.3182/20140824-6-ZA-1003.02184. 19th IFAC World Congress Available at http://www.sciencedirect.com/science/article/pii/S1474667016416058

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

[40] Cetinkaya A, Ishii H, Hayakawa T (2015) Event-triggered control over unreliable networks subject to jamming attacks. *2015 54th IEEE Conference on Decision and Control (CDC)*, , pp 4818–4823. https://doi.org/10.1109/CDC.2015.7402971

[41] Dolk VS, Tesi P, De Persis C, Heemels WPMH (2017) Event-triggered control systems under denial-of-service attacks. *IEEE Transactions on Control of Network Systems* 4(1):93–105. https://doi.org/10.1109/TCNS.2016.2613445

[42] Chen Y, Du T, Jiang C, Sun S (2019) Indoor location method of interference source based on deep learning of spectrum fingerprint features in smart cyber-physical systems. *EURASIP Journal on Wireless Communications and Networking* 2019(1):47. https://doi.org/10.1186/s13638-019-1363-y. Available at https://doi.org/10.1186/s13638-019-1363-y

[43] Goh J, Adepu S, Tan M, Lee ZS (2017) Anomaly detection in cyber physical systems using recurrent neural networks. *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)*, , pp 140–145. https://doi.org/10.1109/HASE.2017.36

[44] Junejo KN, Yau DKY (2016) Data driven physical modelling for intrusion detection in cyber physical systems. *SG-CRC*, , .

[45] Beaver JM, Borges-Hink RC, Buckner MA (2013) An evaluation of machine learning methods to detect malicious scada communications. *Proceedings of the 2013 12th International Conference on Machine Learning and Applications - Volume 02* ICMLA '13 (IEEE Computer Society, Washington, DC, USA), , pp 54–59. https://doi.org/10.1109/ICMLA.2013.105. Available at http://dx.doi.org/10.1109/ICMLA.2013.105

[46] Borges Hink RC, Beaver JM, Buckner MA, Morris T, Adhikari U, Pan S (2014) Machine learning for power system disturbance and cyber-attack discrimination. *2014 7th International Symposium on Resilient Control Systems (ISRCS)*, , pp 1–8. https://doi.org/10.1109/ISRCS.2014.6900095

[47] Jin XB, Robert Jeremiah RJ, Su TL, Bai YT, Kong JL (2021) The new trend of state estimation: From model-driven to hybrid-driven methods. *Sensors* 21(6). https://doi.org/10.3390/s21062085. Available at https://www.mdpi.com/1424-8220/21/6/2085

[48] Jin XB, Sun S, Wei H, Yang FB (2018) Advances in multi-sensor information fusion: Theory and applications 2017. *Sensors* 18(4). https://doi.org/10.3390/s18041162. Available at https://www.mdpi.com/1424-8220/18/4/1162

[49] Xie X, Yue D, Park JH (2020) Observer-based state estimation of discrete-time fuzzy systems based on a joint switching mechanism for adjacent instants. *IEEE Transactions on Cybernetics* 50(8):3545–3555. https://doi.org/10.1109/TCYB.2019.2917929

[50] Dehghanpour K, Wang Z, Wang J, Yuan Y, Bu F (2019) A survey on state estimation techniques and challenges in smart distribution systems. *IEEE Transactions on Smart Grid* 10(2):2312–2322. https://doi.org/10.1109/TSG.2018.2870600

[51] Jin X, Yin G, Chen N (2019) Advanced estimation techniques for vehicle system dynamic state: A survey. *Sensors* 19(19). https://doi.org/10.3390/s19194289. Available at https://www.mdpi.com/1424-8220/19/19/4289

This publication is available free of charge from: https://doi.org/10.6028/NIST.IR.8416

[52] Xu B, Hu J, Jia C, Cao Z, Huang J (2021) State estimation via prediction-based scheme for linear time-varying uncertain networks with communication transmission delays and stochastic coupling. *Systems Science & Control Engineering* 9(1):173–187. Available at https://doi.org/10.1080/21642583.2021.1888820.

[53] (2021) RFnest product specifications, http://www.i-a-i.com/wp-content/uploads/2017/07/RFnest-Specsheet-2017.pdf. Accessed: 2021-12-17.

[54] (2021) OptiTrack V120 Trio datasheet, https://d111srqycjesc9.cloudfront.net/V120-Duo%20and%20Trio%20Data%20Sheet.pdf. Accessed: 2021-12-17.

[55] (2020) scikit-learn: machine learning in Python. Available at http://scikit-learn.org/stable/.

[56] (2020) scikit-learn-Metrics and scoring: quantifying the quality of predictions. Available at https://scikit-learn.org/stable/modules/model_evaluation.html.

[57] Chen T, Guestrin C (2016) Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* KDD '16 (ACM, New York, NY, USA), , pp 785–794. https://doi.org/10.1145/2939672.2939785. Available at http://doi.acm.org/10.1145/2939672.2939785

[58] Thanh Noi P, Kappas M (2018) Comparison of random forest, k-nearest neighbor, and support vector machine classifiers for land cover classification using sentinel-2 imagery. *Sensors* 18(1). https://doi.org/10.3390/s18010018. Available at https://www.mdpi.com/1424-8220/18/1/18

[59] Louppe G (2015) Understanding random forests: From theory to practice. 1407.7502.

[60] Chen T, Guestrin C (2016) Xgboost. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* https://doi.org/10.1145/2939672.2939785. Available at http://dx.doi.org/10.1145/2939672.2939785

[61] Wang G, Nixon M, Boudreaux M (2019) Toward cloud-assisted industrial iot platform for large-scale continuous condition monitoring. *Proceedings of the IEEE* 107(6):1193–1205. https://doi.org/10.1109/JPROC.2019.2914021

[62] Luo Y, Xiao Y, Cheng L, Peng G, Yao DD (2021) Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities. *ACM Comput Surv* 54(5). https://doi.org/10.1145/3453155. Available at https://doi.org/10.1145/3453155

[63] Chandola V, Banerjee A, Kumar V (2009) Anomaly detection: A survey. *ACM Comput Surv* 41(3). https://doi.org/10.1145/1541880.1541882. Available at https://doi.org/10.1145/1541880.1541882