Adaptive Strategy to Improve the Quality of Communication for IoT Edge Devices

Bharath Sudharsan Confirm Smart Manufacturing Data Science Institute National University of Ireland Galway National University of Ireland Galway National University of Ireland Galway bharath.sudharsan@insight-centre.org

John G. Breslin

Confirm Smart Manufacturing Data Science Institute john.breslin@nuigalway.ie

Muhammad Intizar Ali Confirm Smart Manufacturing Data Science Institute ali.intizar@nuigalway.ie

Abstract-In an IoT system, the response time of edge devices is calculated during the design time. These edge devices continuously provide data streams to ensure the smooth execution of a real-time IoT system. However, edge devices are prone to errors, and very often suffer issues when trying to maintain a certain level of communication quality in the presence of external interference. Any loss of communication at the edge device level can lead to a failure of the entire system or to misleading information being provided. Due to there being a large number of heterogeneous devices within the IoT system, it is not a trivial matter to monitor all of these devices from a centralised location or to explore system logs to determine any loss of communication. Hence, in order to maintain the highest level of of communication quality in as close as possible to the best theoretical response time, there is a need for a lightweight intelligent layer on the edge devices which could adapt depending on changes in the context. In this work, we propose an adaptive algorithm, which can predict the quality of communication of WiFi and BLE with an accuracy of 94.14% and 92.25% respectively. The adaptive layer can recommend the next best alternative available wireless communication protocol in case the existing wireless protocol's quality degrades. Edge devices within IoT systems can be equipped with our proposed adaptive layer, which can help them to adapt according to dynamic context whilst ensuring the highest level of communication quality, thus, improving the overall resilience of the entire IoT system.

Index Terms-Adaptive Algorithms, Multiprotocol Switching, Received Signal Strength, RSS Prediction, WiFi, BLE, Long-Range RF, Embedded Systems, Edge Intelligence.

I. INTRODUCTION

In recent years, IoT devices have become an integral part of many real-time systems, and more and more IoT applications rely on real-time data streams collected from IoT devices. For example, in many Industry 4.0 scenarios, "edge" devices (i.e., hardware that controls the data flow at network boundaries) are fully integrated within collaborative manufacturing systems and have the ability to respond and react in real-time to dynamic factory floor conditions instead of existing in their own standalone ecosystems [1]. For such a smart manufacturing setup, many devices equipped with at least one IoT communication protocol have emerged

over the last decade, and the adoption of such devices is expected to grow in the future as part of a complete suite of connected solutions. In critical manufacturing setups, there is a strong need to ensure and maintain a high quality of communication among devices. Traditionally, edge devices are equipped with a single wireless protocol, and any failure in that protocol can lead to a failure of the entire system.

Multiple state-of-the-art sensor systems with experimental results have been proposed to effectively monitor and react to changes in environmental parameters [2] [3]. Also, some mission-critical systems notify/warn when any given sensor values go beyond the normal patterns and breach predefined threshold values. Whenever the notification/warning from these monitoring systems is not real-time, it is often due to a loss in communication quality for an IoT protocol. Additionally, if there are no alternative means for communicating the notification/warning, there are increased chances of compromising the health and safety of industrial workers.

With the latest advancements in technology, it is now easily possible to equip edge devices with multiple wireless communication protocols. To ensure complete resilience and have the ability to switch among the protocols, every edge device on the factory floor can be equipped with a light-weight intelligent layer that could facilitate switching between the protocols by adapting to the changing context. This layer should be run in the background of the main application on the edge device, to constantly monitor wireless communication quality and predict future communication quality. This layer should have low computational complexity without compromising accuracy, and should also be locationand device-independent. If the current communication quality is predicted to degrade in the near future, this layer should recommend the next best alternative wireless communication protocol available within the device to which it can switch.

There are a number of cases where industry has emphasized the need to have multiple communication protocols on a single factory floor to improve communication quality, and there have already been some scenarios implemented where multiple IoT communication protocols have been deployed on a single floor. Equipping these edge devices with algorithms to instruct the devices when to seamlessly switch

This publication has emanated from research supported by research grants from Science Foundation Ireland (SFI) under Grant Number SFI/16/RC/3918 (Confirm) and SFI/12/RC/2289_P2 (Insight), co-funded by the European Regional Development Fund.

will improve communication quality. Our main contributions in this paper are (i) introducing a resilience property in the IoT architecture at device level to realize reliable information distribution, beginning at the edge level; (ii) equipping embedded devices with multiple communication protocols, and providing interoperability at scale; (iii) enabling the edge devices to adapt to dynamically changing contexts and situations around them, and thereby continuously evolve; and (iv) providing run-time re-configuration of the embedded systems in accordance with the context.

II. ADAPTIVE STRATEGY BLOCK

We propose an Adaptive Strategy Block (ASB) for edge devices which is a light-weight intelligent layer capable of analyzing the context of IoT applications by understanding the strengths and weaknesses of wireless protocols, application requirements and predicting upcoming issues due to any potential degrade in the quality of communication. Based on the predictions, the ASB is capable of triggering a switch between wireless protocols to ensure the highest level of quality of communication. Fig. 1. gives an overall architecture of IoT applications on edge devices equipped with ASB. The three main components of ASB are:

- i Context Monitoring: This component continuously monitors the context of an IoT application by collecting the current state of the wireless communication protocols, and also requirements of the IoT application in terms of communication quality.
- ii RSS Prediction: This component uses our trained model to predict any future degrade in communication quality.
- iii Adaptation: This component triggers an adaptation based on the RSS prediction and ensures the application data flow is maintained using the alternate protocol.

As shown in Fig. 1, we are considering three wireless protocols, namely BLE, WiFi and RF, to adapt to dynamically changing situations while maintaining the highest quality of communication. We use Received Signal Strength (RSS) as a metric to assess wireless connection quality throughout this work. ASB can be integrated as a light-weight library inside an IoT application running on the edge device. We consider that devices are equipped with hardware to facilitate three types of wireless communication and their protocols are supported by the ASB.

III. WIRELESS COMMUNICATION PROTOCOLS COMPARISON

In this section, we will briefly discuss the strengths and weaknesses of the three selected wireless protocols.

A. Situation 1: BLE

The IEEE 802.15.1-based BLE is 30% more energy-efficient than WiFi, also the energy consumption for sending and receiving messages is almost the same [4]. According to [5], the benefits of BLE in terms of power consumption are impressive since it nearly doubles the run time of devices. The downsides of BLE are: obstacles and reflecting surfaces such as doors, walls, etc. affect the signal strength of BLE more than WiFi [5]; BLE has slow signal scanning speeds; and finally, BLE signals fade out faster than WiFi. Hence, BLE is best suited to continuous data streaming within a close range at lower power.

B. Situation 2: WiFi

The IEEE 802.11 WiFi is the second protocol we used. WiFi promises higher device-to-device radio signal range, much higher transfer speeds, and higher bandwidth when compared with BLE. In scenarios where the devices in use are battery-powered, BLE is used to transfer data with lower power costs when devices are close to each other, and when the device moves away, the data transfer still continues but using WiFi, resulting in slightly higher power costs. Therefore, although WiFi covers the entire operating range of BLE and also has some more benefits when compared to BLE, a more efficient method for transferring data over a range of distances is to use both BLE and WiFi.

C. Situation 3: Long-range RF

The final protocol is long-range RF. It consumes a relatively higher power level for data transmission than WiFi, providing long-range network availability (typically over 1 km), and has greater resistance to interference than WiFi and BLE. When a mobile device is predicted to move beyond the coverage range of WiFi, this third protocol is activated and data transfer is handed over to this protocol.

IV. RSS PREDICTION

The RSS prediction model of the ASB is fitted as a light-weight intelligence layer on resource-constrained edge devices as shown in Fig. 1. Hence, it should have low computational complexity without compromising accuracy, and also should be location and device-independent. In proposing such a model, datasets with increasing and decreasing patterns of RSS were selected and are outlined in Section IV-A. The techniques used to process the finalized datasets are detailed in Section IV-B. A series of raw data points representing increasing and decreasing RSS trends are extracted in Section IV-C. and are visualized to assist with choosing the best model for RSS prediction. The model built for RSS prediction is detailed in Section IV-D. and evaluated in Section IV-E.

A. Dataset selection

A dataset with increasing and decreasing patterns of signal strength is required for both BLE and WiFi. For WiFi, the kthrss dataset [6] is chosen over other relevant datasets such as buffalo/phonelab-wifi dataset [7], and cister/rssi dataset [8] because the kthrss dataset contains the RSS (Radio Signal Strength) data collected in both indoor and semi-outdoor environments using a KUKA youBot mobile robot. The indoor data collected in the KTH Royal Institute of Technology contains the RSS data from five wireless receivers in an indoor environment. Four of the wireless clients used directional antennas while the remaining one used an omnidirectional antenna. The wireless router to



Fig. 1. Architecture and components of the Adaptive Strategy Block (ASB).

which the client was connected used a directional antenna. This traceset contains 12 columns of data, from which only the timestamp, RSS1 (dBm), RSS2 (dBm), RSS3 (dBm), RSS4 (dBm) and RSS5 (dBm) were extracted for this work. The semi-outdoor data was collected in Dortmund. This traceset contains 8 columns of data, from which only the timestamp with its corresponding RSS(dBm) was extracted for this work.

For BLE, the BLE RSS measurements database and supporting materials [9] were chosen over the BLEBeacon Dataset [10] because the selected dataset contains over 4,700 fingerprints of measurements from BLE beacons and also the RSS measurements were taken in two zones from Universitat Jaume I in Spain. One zone is in an area with bookshelves that belongs to the university library, and the other zone is an office space area. From the dataset, two files that contain the RSS (dBm) measurements along with a timestamp from each zone were extracted for analysis.

B. Data processing

The real world is far from noise-free. The deterministic relationship between RSS and distance is negatively affected by noise that is due to environmental propagation conditions and also by peculiarities of the wireless equipment used to collect the measurements [11]. Also, the received power is typically affected by random noise terms due to signal propagation phenomena [11]. Due to these reasons, in both datasets, RSS & distance are not always linear, and temporal fluctuations are present in the RSS measurements. The noise and fluctuations from both datasets have to be removed without altering the pattern of the recorded signal. To perform this, LOESS (locally estimated/weighted scatterplot smoothing) [12] was chosen since it adapts well to bias problems at boundaries and in regions of high curvature. The datasets used to train the model to predict RSS for BLE and WiFi are described in Section IV-A. Segments of the original/raw data for both WiFi and BLE are shown in Fig. 2.a. and Fig. 3.a, with LOESS-smoothed versions in Fig. 2.b. and Fig. 3.b. respectively.

LOESS is carried out using local linear estimates. Initially, the x and y values for n raw data points are computed. For each value of x, the value of f(x) is estimated using its neighboring sampled (known) values. Two new variables x' and y' are introduced, where x' is the value of x for which y' has to be estimated. The LOESS algorithm is fed with x', and using the sampled x and y values, the estimate y' is obtained. Next, the x' values are fed to Euclidean distance formulae to find the k nearest neighbors for the estimate y'. The resulting ordered set (x', y' & k) is called D. The final step converts the set D of k distances into an ordered set W containing weights. These weights are calculated using Eqn. 1, which is a tri-cubic weight function. This function assigns an importance to each of the k neighbors of x according to its distance from x'. In the tri-cubic weight function, d(x, x') is the distance between x, one of the k nearest neighbors, and x'. The point at the extreme will have a weight of zero, whereas the point at zero distance will have the highest possible weight of one.

$$w(x) = \begin{cases} \left(1 - \left|\frac{d(x,x')}{max_i d(x_i,x')}\right|^3\right)^3 & |x| < 1, x_i \in D\\ 0 & |x| \ge 1 \end{cases}$$
(1)

C. Data extraction

The goal of data extraction in this work is to extract multiple series of raw data points representing signal strength with increasing and decreasing trends. The extracted trends are then visualized to assist in choosing the best regression model for predicting the RSS of the WiFi and BLE signals. To extract the trends, initially, the peak and trough points have to be found and plotted on the LOESS smoothed data. Here, the raw data cannot be directly used for finding peaks and troughs since some fluctuations in the raw data may have a higher amplitude than the peaks or troughs, and are distributed evenly throughout the time series. If peaks and troughs are plotted using raw data, there will be multiple peaks and troughs throughout the time series, and the data points between two plotted points will be less resulting in trends with short sizes, making the model choosing process laborious. Let us use maxima and minima as the plurals of maximum and minimum. In Fig. 2.c. and Fig. 3.c, the maxima are the peak points plotted in green and the minima are the trough points plotted in red. To find and plot the peaks (maxima) and trough (minima) points, initially, all data points from the time-series were iterated through. A Δ value is defined, which controls how



Fig. 2. WiFi dataset [6]: Raw data, Smoothed data, Peak and trough points, Increasing and decreasing RSS trend extraction, Predicted RSS.

much difference between values in the time series defines an extremum point. We iterate over the data points of the time series and consider a point to be a local maxima if it has the maximal value, and is preceded (to the left) by a value lower by Δ . Similar logic applies to find local minima.

The local minima and maxima were thus obtained and plotted on the LOESS smoothed curve as shown in Fig. 2.b. and Fig. 3.b. for WiFi and BLE respectively. The coordinates of the local minima and maxima points were used to extract the increasing and decreasing trends for signal strength from the raw dataset. For visualization purposes, three increasing and decreasing trends for signal strength were extracted and plotted as shown in Fig. 2.d. for WiFi and Fig. 3.d. for BLE.

D. Model building

Recurrent Neural Networks (RNN) are used in cases where patterns, combinations of trends, seasonality or cycles in data can be observed. Although abundant data is available in



Fig. 3. BLE dataset [9]: Raw data, Smoothed data, Peak and trough points, Increasing and decreasing RSS trend extraction, Predicted RSS.

the finalized WiFi and BLE datasets, it cannot be used to train RNNs for predicting future signal strength. Chaotic timeseries data such as tornadoes, stock markets, turbulence, and weather are ubiquitous in real-world signals. The most striking feature of such data is its unpredictability in terms of future values. Also, as mentioned previously, signal strength data will have fluctuations that vary due to changing propagation conditions in the surrounding environment.

Since the proposed model to predict RSS will be fitted as a light-weight intelligence layer on resource-constrained edge devices, it should have low computational complexity without compromising accuracy, and also should be locationand device-independent. Before proposing this method, a literature review was performed, and the review findings are outlined here. RSS prediction has been well-studied in the temporal [13] [14] and spatial domains [15]. Existing offline RSS prediction algorithms involve Kriging interpolation (geostatistics), a dedicated offline training phase with supervised learning methods, or a dedicated fingerprinting method. Existing online methods use Linear Regression or Gaussian Process Regression. Both online and offline methods produce excellent prediction performance but only for short-range applications since their primary focus is on radio signal mapping, localization applications, active motion control, and path planning. In this work, we are proposing RSS prediction for both BLE and WiFi, from short- to long-range applications (from tens to hundreds of meters), so as to be of practical use in IoT edge devices.

The goal of the model to be trained using the finalized dataset is to predict future RSSI levels for BLE and WiFi in dBm. As mentioned, to perform this, the traditional Neural Network based approach cannot be used. Increasing and decreasing trends for the target parameter RSS are extracted from datasets using the methods described in Sections IV-B. and IV-C. Time-trend RSS curves are obtained using matplotlib to visualize increasing and decreasing signal strength patterns. After visualizing the trends, it can be observed that the signal strength decreases, but not smoothly, and there are spikes (fluctuations), but the overall trend remains undisturbed. A trend similar to the extracted trends in Fig. 2.d. and Fig. 3.d. will often be followed when the signal strength of edge devices increases or decreases. As a result of this visualization, the best-suited approach to predict future RSS is a Radial Basis Function (RBF) kernel-based Support Vector Regression (SVR) [16]. The usage of kernel functions in SVR improves the flexibility of SVR because it implicitly maps the data to a higher dimensional feature space [16]. In our case, the extracted signal strength trends follow nonlinear patterns. This pattern is mapped to a higher dimensional feature space, hence the non-linearity in the original dimension corresponds to a linear solution in the higher dimensional feature space. RBF yields a more compressed and widely-supported kernel when compared with other kernels. This makes it suitable in terms of restricting the computational training process and improving the generalization efficiency of the SVR. Therefore, RBF is adopted in this study and defined as $K(x, x_i)$, with a kernel parameter σ as shown in Eqn. 2. and substituted into Eqn. 3. In Eqn. 3, b is the bias term, α contains the support values, and y contains the training labels [16].

$$K(x, x_i) = exp\left(-\frac{1}{\sigma^2}||x - x_i||^2\right)$$
(2)

$$SVR = f(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i) + b$$
(3)

When a series of data points underlying a trend or function is fed as an input to the RBF kernel-based SVR from Eqn. 3, the underlying trend is approximated. In Fig. 2.e. and Fig. 3.e, the predicted RSS is sketched in red. Here, the trend/slope of the predicted lines in red are very close to the extracted trends from the raw/original series shown in Fig. 2.d. and Fig. 3.d. So far, we have explained the method by which ASB performs predictions to classify whether the RSS will increase or decrease in the future. A simple equation to estimate the percentage future increase or decrease in RSS is given in Eqn. 4. The variables from Eqn. 4. are the RSS values in dBm from the y-axis of the predicted signal strength in Fig. 2.e. and Fig. 3.e. If the prediction is that the RSS will decrease in the future, then the value x_0 is the starting point of the signal, which is the first point on the top left. The value x_1 is the point at 70% of the smoothed curve for the extracted trend, and this is also the point from which the RSS prediction starts (sketched in red). The value x_2 is the final point which is on the bottom right of the RSS prediction graph, and this is also the ending point for RSS prediction. However, if the RSS is predicted to increase in the future, value x_0 is the starting point of the signal, which is the first point on the bottom left. x_1 is the point from which the RSS prediction starts (sketched in red). The endpoint x_2 is the final point which is on the top right of the RSS prediction graph.

$$RSSfuture(\%) = \frac{x_2 - x_1}{x_1 - x_0} * 100$$
(4)

To summarise, ASB's first step was to use SVR with an RBF kernel to predict the future RSS trend, and sketch it in red as shown in Fig. 2.e. and Fig. 3.e. The second step is to classify whether the RSS will increase or decrease in the future. The third step is to estimate the percentage future increase or decrease in the RSS based on the results from steps one and two.

E. Model Evaluation

To measure the prediction performance, Mean Absolute Error (MAE) is used as an evaluation metric. MAE is more robust for this use case since it is less sensitive to fluctuations (outliers). MAE was calculated using Eqn. 5. Here, z_{k+p} is the true measured percentage of increase or decrease in future RSS values, and \hat{z}^{μ}_{k+p} is the mean of the predicted percentage future increase or decrease in RSS. From MAE, the Mean Prediction Accuracy (MPA) is obtained using Eqn. 6. MPA normalizes the prediction error with respect to the true RSS values. The proposed SVR with an RBF kernel, used to predict the RSS of WiFi and BLE, was tested on unused columns of data from the finalized datasets and was also validated using new datasets (phonelab-wifi [7] for WiFi and BLEBeacon [10] for BLE). The proposed method was able to achieve an MAE of less than 3.91 dBm for WiFi and 5.33 dBm for BLE as shown in Table I.

$$MAE = AE^{\mu} = |\hat{z}^{\mu}_{k+p} - z_{k+p}|$$
(5)

$$MPA = 100 \left(1 - \sum \frac{|\hat{z}_{k+p}^{\mu} - z_{k+p}|}{z_{k+p}} \right)$$
(6)

V. RELATED WORK

The Semantic Gateway as a Service (SGS) proposed in [17] allows translation between messaging protocols such

TABLE I Evaluation of SVR with RBF kernel

Protocol	Dataset	MAE (dBm)	MPA (%)
WiFi	kthrss [6]	3.91	94.14
	phonelab-wifi [7]	3.41	94.95
BLE	BLE RSS [9]	4.12	93.83
	BLEBeacon [10]	5.33	92.25

as XMPP, CoAP and MQTT via a multi-protocol proxy architecture to provide seamless integration and interoperability between various protocols. This multi-protocol proxy-based protocol translation is performed at the gateway level, which is not a resource-constrained device and is not at a network's edge. Also, this proposed work does not involve switching or deciding when to switch between protocols, rather the gateway just translates the data so that the server and client (following different protocols) can understand.

A middleware framework to improve data delivery and resilience to failures & traffic spikes was proposed in [18]. This conceptual data exchange middleware is envisioned as an open communications hub for IoT, containing middleware abstractions for enabling hosts to connect via multiple protocols. This middleware can understand data from its connected hosts which communicate using different protocols such as XMPP, MQTT & HTTP. But this work has not mentioned the possibility of using the same host, communicating with the middleware using different IoT protocols. Again, it is implemented on a separate middleware and not on resource-constrained devices.

The benefits of wireless connectivity with multiple protocols on the same device are many and varied [19]. Nordic Semiconductor's hardware lineup supports multiple wireless connectivity protocols to add flexibility in product design. They have stressed the concept of "creat[ing] one device that can connect to multiple protocols", which means that the same device can be configured to use different IoT communication protocols. For example, in an industrial use case, when the network type is unknown during the hardware development phase, it could be Bluetooth Mesh, Thread or Zigbee. With Nordic Semiconductor's switched multiprotocol concept, it is possible to develop a generic application, and during deployment, the user can select the appropriate protocol to use which connects to a network that already exists. Hence, it is clear that adding multiple protocols on the same IoT hardware is a much-needed methodology to improve device interoperability. However, their multiprotocol feature requires the user too manually select the appropriate protocol for their application, and does not involve an algorithm to seamlessly switch and decide when to switch between protocols.

VI. CONCLUSION

The ASB proposed in this work uses SVR with an RBF kernel to predict the RSS of WiFi and BLE with an accuracy of 94.14% and 92.25% respectively and has the advantage

of being location and device-independent. Our proposed ASB can fit as a lightweight intelligence layer on IoT edge devices, with the aim of monitoring communication quality and recommending the next best alternative to switch to for maintaining a high level of communication quality, so that the system can keep communicating in real-time. In future work, our proposed ASB will be evaluated using more real-world experimental data to be gathered.

References

- D. Mourtzis, E. Vlachou, and N. Milas, "Industrial big data as a result of iot adoption in manufacturing," *Procedia CIRP*, vol. 55, pp. 290–295, 2016.
- [2] O. Blanco-Novoa, T. Fernández-Caramés, P. Fraga-Lamas, and L. Castedo, "A cost-effective iot system for monitoring indoor radon gas concentration," *Sensors*, vol. 18, p. 2198, 07 2018.
- [3] M. Benammar, A. Abdaoui, S. Ahmad, F. Touati, and A. Kadri, "A modular iot platform for real-time indoor air quality monitoring," *Sensors*, vol. 18, p. 581, 02 2018.
- [4] G. D. Putra, A. R. Pratama, A. Lazovik, and M. Aiello, "Comparison of energy consumption in wi-fi and bluetooth communication in a smart building," 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), 01 2017. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7868425
- [5] A. Lindemann, B. Schnor, J. Sohre, and P. Vogel, "Indoor positioning: A comparison of wifi and bluetooth low energy for region monitoring," *Proceedings of the 9th International Joint Conference on Biomedical Engineering Systems and Technologies*, 2016. [Online]. Available: https://scitepress.org/papers/2016/57046/57046.pdf
- [6] R. Parasuraman, S. Caccamo, F. Baberg, and P. Ogren, "Crawdad dataset kth/rss (v.2016-01-05)," CRAWDAD wireless network data archive, 01 2016. [Online]. Available: https://crawdad.org/kth/rss/20160105/
- [7] J. Shi, C. Qiao, D. Koutsonikolas, and G. Challen, "Crawdad dataset buffalo/phonelab-wifi (v.2016-03-09)," *CRAWDAD wireless network data archive*, 03 2016. [Online]. Available: https://crawdad.org/buffalo/ phonelab-wifi/20160309/wifi/
- [8] C. Noda, S. Prabh, M. Alves, T. Voigt, and C. A. Boano, "Crawdad dataset cister/rssi (v.2012-05-17)," CRAWDAD wireless network data archive, 05 2012. [Online]. Available: https://crawdad.org/cister/rssi/20120517/
- [9] G. Mendoza-Silva, M. Matey-Sanz, J. Torres-Sospedra, and J. Huerta, "Ble rss measurements dataset for research on accurate indoor positioning," *Data*, vol. 4, p. 12, 01 2019.
- [10] dimisik, "dimisik/blebeacon-dataset," 2018. [Online]. Available: https://github.com/dimisik/BLEBeacon-Dataset
- [11] A. Zanella, "Best practice in rss measurements and ranging," IEEE Communications Surveys Tutorials, vol. 18, pp. 2662–2686, 2016.
- [12] W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American Statistical Association*, vol. 74, pp. 829–836, 12 1979.
- [13] M. Cvjetkovic and V. Rakocevic, "Alternative topology construction for cooperative data distribution in mobile ad hoc networks," 2016 8th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 10 2016.
- [14] X. Long and B. Sikdar, "A real-time algorithm for long range signal strength prediction in wireless networks," 2008 IEEE Wireless Communications and Networking Conference, 03 2008. [Online]. Available: https://ieeexplore.ieee.org/document/4489233/
- [15] E. F. Flushing, M. Kudelski, L. M. Gambardella, and G. A. Di Caro, "Spatial prediction of wireless links and its application to the path control of mobile robots," *Proceedings of the 9th IEEE International Symposium on Industrial Embedded Systems (SIES 2014)*, 06 2014. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp= &arnumber=6871207
- [16] K. R. Müller, A. J. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik, "Predicting time series with support vector machines," *Lecture Notes in Computer Science*, pp. 999–1004, 1997.
- "Semantic gateway as a service architecture for iot interoperability

 ieee conference publication," 2015. [Online]. Available: https: //ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7226706
- [18] K. Benson, "Enabling resilience in the internet of things," 2015. [Online]. Available: https://www.ics.uci.edu/~dsm/papers/2015/ PERCOM-2015-Workshop.pdf
- [19] "Multiprotocol nordic semiconductor," 2019. [Online]. Available: https://www.nordicsemi.com/?sc_itemid=%7B2C7F9C4C-70FB-47DE-908B-479D059EA153%7D