CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Identifying distinct features based on received samples for interference detection in wireless sensor network edge devices
Authors	O'Mahony, George D.;Harris, Philip J.;Murphy, Colin C.
Publication date	2020-04-01
Original Citation	O'Mahony, G. D., Harris, P. J. and Murphy, C. C. (2020) 'Identifying Distinct Features based on Received Samples for Interference Detection in Wireless Sensor Network Edge Devices', 2020 Wireless Telecommunications Symposium (WTS), Washington, DC, USA, 22-24 April, (7 pp). doi: 10.1109/WTS48268.2020.9198724
Type of publication	Conference item
Link to publisher's version	https://ieeexplore.ieee.org/document/9198724 - 10.1109/WTS48268.2020.9198724
Rights	© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
Download date	2026-10-30 02:25:09
Item downloaded from	https://hdl.handle.net/10468/11186



University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

Identifying Distinct Features based on Received Samples for Interference Detection in Wireless Sensor Network Edge Devices

George D. O'Mahony
Dept. of Electrical and Electronic Engineering, University College Cork
Cork, Ireland
george.omahony@umail.ucc.ie

Philip J. Harris
United Technologies Research Center Ireland (UTRC-I)
Cork, Ireland
harrispj@utrc.utc.com

Colin C. Murphy
Dept. of Electrical and Electronic Engineering, University College Cork
Cork, Ireland
colinmurphy@ucc.ie

***Abstract*—Wireless Sensor Network (WSN) technologies have developed considerably over the past decade or so and, now, feasible solutions exist for various applications, both critical and otherwise. Often these solutions are achieved by using commercial off the shelf components combined with standardized open-access protocols. As deployments diverge into safety-critical areas, attack incentives intensify, leading to persistent malicious intrusion challenges, which are ever-changing as interference techniques evolve and dynamic hardware becomes increasingly accessible. Unique WSN security vulnerabilities, a fluctuating radio frequency (RF) spectrum and physical environment and spectrum co-existence escalate the problem. Thus, securing WSNs is a critical and demanding requirement, heightened by the burden of protecting sensitive transmitted information. This paper, by utilizing ZigBee and Monte Carlo simulations, aims to develop an initial framework for interference detection in WSNs. Initially, bit error location analysis motivates a feature-based detection strategy, relating to both subtle and crude forms of interference. The work expands to analyze Matlab simulated error-free and erroneous transmissions to investigate whether feature useful differences exist. A feature set, including the measured probability density function of, and statistics on, the in-phase and quadrature-phase samples is demonstrated and initially validated/feasibility tested using a designed support vector machine.**

***Keywords*-IEEE802.15.4, Detection, Interference, IoT, Machine Learning, Security, Support Vector Machine, WSN and ZigBee.**

I. Introduction

WSNs are continuing to become integrated into safety critical applications [1] and are, simultaneously, becoming an indispensable component of modern technology. Due to this use of WSNs in safety conscious applications and increasing congestion levels in the RF spectrum, new challenges concerning security, spectral coexistence and threat identification emerge. Utilizing WSNs is a direct result of over a decade of research and development, resulting in feasible solutions to various innovative application challenges. Consequently, this leads to strict operational and availability requirements being imposed on computationally constrained devices. Embracing WSNs will, likely, continue in the modern cost-centered age, as WSNs permit the benefits of easier design, installation and maintenance, while simultaneously providing new deployment options. The diverse range of applications include space-based WSNs [1], the Internet of Things (IoT), smart homes and cities, wireless networked control systems [2], aerospace [3] and using low earth orbit satellites as WSN components [4].

WSN protocols in use, typically, have very similar physical (PHY) and medium access control (MAC) layers [5] and are, generally, based on an open access standard, for example, IEEE 802.15.4. Typical device resources hinder the use of computationally intensive security protocols and publicly known standards can typically be reverse engineered by available tools. Both WSN operations and IoT devices operating in the RF spectrum continue to expand, leading to increased levels of congestion, especially in the 2.4 GHz industrial scientific and medical (ISM) band. Therefore, WSN security is becoming increasingly important and network compromise, whether malicious or unintentional, is achievable, can have significant consequences for privacy and safety and requires each communication link to be secure.

This paper presents Matlab simulation results revealing how different types of jammers influence the positions and probability of bit errors in 802.15.4 frames as a foundation for potential interference detection and classification strategies. The simulated samples are analyzed under additive noise, different jammer types and varying jammer-to-signal ratios (JSR) by employing the low rate wireless personal area network (LR-WPAN) protocol, ZigBee. Bit error locations are identified under varying JSR values to motivate feature analysis. Contributions focus on identifying features extracted from received in-phase (I) and quadrature-phase (Q) samples for varying JSR values. Error free and erroneous packets, containing one or more bit errors, are statistically compared to determine if any significant differences exist. Differentiation is achieved by neglecting network-wide operative analysis and information from higher up the protocol stack. Thus, packet delivery rates (PDRs), packet sending rates and received signal strength (RSSI) are neglected. Extracted features are experimentally validated and the approaches feasibility established by a support vector machine (SVM) classifier.

The remainder of this paper is organized as follows: Section

II outlines similar work in the area. Section III discusses WSNs, the chosen protocol, ZigBee, and WSN security in terms of requirements, vulnerabilities, attacks and defenses. Section IV outlines and discusses the executed experiments and how errors were indicated. Section V explains the main results, including the feature set and validation through the use of a SVM. Finally, section VI concludes this paper and provides the necessary future work.

II. Related Work

Detecting interference and intrusions in wireless communications is not an original concept, but an approach which, typically, requires enhancements to match current deployments and trends. Conventional WSN techniques and intrusion detection systems (IDSs), defined as software or hardware tools that monitor networks to detect internal or external attacks, typically use analysis of the RSSI and packet rates [6]. Machine learning techniques can also use network information to detect intrusions [7]. Detailed surveys on intrusion detection in WSNs, the main concepts, and the vital areas can be found in [8], [9]. Chip sequence error patterns are used in [10] to identify the channel and, as a result, the emitting interference. However, this technique requires edge devices to buffer known patterns and calculate a pattern recognition classifier. SonIC [11] samples received RSSI values to extract features for a decision tree classifier for edge device applications. The process requires a successful retransmission of an error packet for comparison and a buffer is required to store the most recent error packet. SVMs and RSSI samples are used in [12] to develop an accurate and fast interference detection process, which consists of four SVMs and a logic decision stage. The work in this paper differs from the above previous work as a methodology which can adapt to new channels and new environments is being developed. The solution focuses entirely on the effects of the wireless channel on received I/Q samples. In contrast to previous work, only the designed machine learning model is required to be on the device and both malicious and unintentional interference are being addressed.

III. WSN Discussion

Here, the IEEE 802.15.4 based LR-WPAN protocol, ZigBee, is the adopted protocol during simulations. ZigBee is the de-facto standard for WSNs as almost all available commercial and research sensor nodes are equipped with ZigBee transceivers [13]. ZigBee's PHY (Table I) and MAC are based on the IEEE 802.15.4 protocol, specifically the unlicensed 2.4 GHz ISM RF band. Operating centre frequencies correspond to (1), where the range is $2.405 - 2.4835\ GHz$, F_c is the centre frequency and i is the channel number. Direct sequence spread spectrum (DSSS) splits every byte into two 4-bit symbols, which are each spread to a predefined 32-bit pseudo-noise (PN) sequence. Offset quadrature-phase shift keying (O-QPSK) modulation is applied to ensure bit transmissions for the I and Q components occur at different time instants, as the components are mutually offset by half a symbol duration. Transmitted signals are pulse shaped using the raised cosine or half-sine method, which ideally achieves zero inter-symbol-interference at the maximum effect points. The carrier sense multiple access with collision avoidance protocol accesses the channel and uses a clear channel assessment before transmission to determine whether a channel is free or busy. A Tektronix RSA306B real time spectrum analyzer [14] and its associated digital phosphor technology (DPX) [15], which performs hardware digital signal processing and rasterizing of samples into pixel information, enables ZigBee signal visualization. Fig. 1 displays the structure of multiple ZigBee signals coexisting in the same geographical region.

TABLE I
IEEE 802.15.4 (ZigBee) PHY Parameters

Parameter:	2.4 GHz PHY Value:
Number of Channels	16
Channel Width / Spacing	2 MHz / 5 MHz
Data Rate	250 kb/s
Symbol Rate	62.5 ksymbols/s
Chip Rate	2 Mchips/s
Modulation Scheme	O-QPSK
Pulse Shaping	Half Sine/Normal Raised Cosine
Byte Spreading	DSSS
Maximum Packet Length	133 bytes

$$F_c = 2405 + 5(i-11)MHz, \ \ for\ i = 11, 12, ...26 \qquad (1)$$

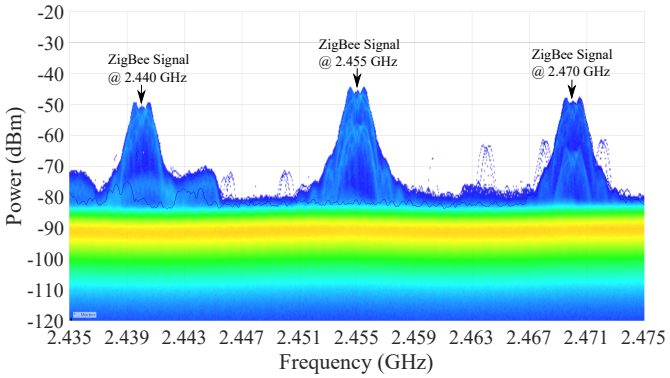


Fig. 1. DPX visualization of coexisting ZigBee signals on channels 18, 21 and 24 in the 2.4 GHz ISM band

Typically, WSNs consist of multiple low-cost lightweight devices used to sense the physical world and, generally, incorporate a radio transceiver, a micro-controller, sensors and a limited energy source. WSNs use a star, mesh or peer-to-peer topology and, in each case, are self-organizing, self-repairing and can exploit clustering techniques, where a cluster head aggregates and forwards data to a centralized sink or access point. The ZigBee protocol and devices are essential for the all-inclusive IoT communication architecture [16], shown in Fig. 2, as the sensing/actuating devices communicate with other sensing/actuating equipment and the gateway/coordinator using a LR-WPAN, like ZigBee. An internet connection is achieved by the sink, and so, for the

envisioned IoT deployment to be successful, WSN links must be secure and maintain uninterrupted, safe and non-malicious operation [5]. Additionally, WSN application areas are abundant, leading to a diverse range of deployment scenarios, environments and use cases. Examples can be classified into precision agriculture, environmental monitoring, vehicle tracking, health care, smart buildings, military and animal tracking [5]. Therefore, WSNs require security across a wide range of physical environments, deployment scenarios and structures, in which privacy and safety are pivotal. This, coupled with the sensitive data being transmitted and valuable application areas, incentivizes attackers to maliciously disrupt or compromise network operation.

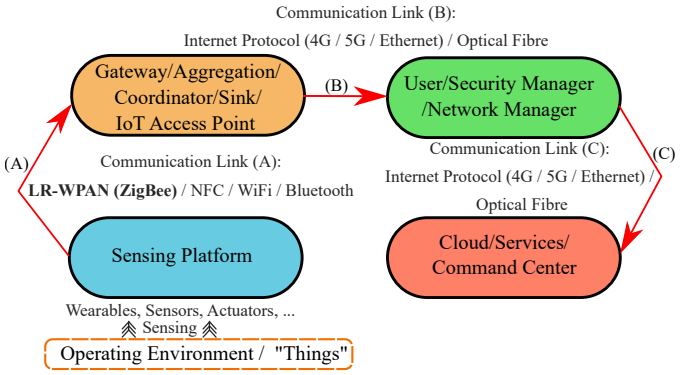


Fig. 2. IoT Communication Model, highlighting the potential use of WSN protocols as the communication link between sensing devices ("Things") and the IoT access point

Security in WSNs can, generally, be described in terms of four interlinked distinct components; requirements, vulnerabilities, attacks and defenses. Example requirements include confidentiality, which ensures the secrecy of important data being transmitted in the wireless channel, and authenticity, which asserts that received packets have not been modified in transit (data integrity) and originate from known locations (origin authenticity). However, guaranteeing requirements are met can be difficult as WSNs have known security vulnerabilities [5]. Examples include the open interface of the wireless channel, (unavoidably) publicly known WSN protocols and how nodes are frequently deployed and left unattended and physically available to potential attackers in hostile or remote environments, where continued surveillance is difficult to guarantee. Also, the low processing power, memory and speed of WSN devices, coupled with a finite energy source, impedes using conventional security protocols. WSN attack types are various and can occur across the entire communication protocol stack. The sensitive data and application scenario incentivizes attacks, which can vary from specific denial of service (DoS) attacks, which can corrupt all packets, to privacy attacks, which can seize sensitive data. However, techniques exist which are employed to protect important data and provide resilience against malicious attacks. For example, cryptography stops intruders from accessing data by simply listening to the wireless channel and DSSS adds resilience to interference as transmitted data resembles white noise and only receivers who know the spreading code can recover encoded information. Other security techniques exist and, in Sections IV and V, a detection technique is presented.

IV. Experimental Method

Here, designed and executed Matlab simulations describe bit error locations in ZigBee PHY frames (Fig. 3) under various jamming power levels to provide motivation for an interference detection strategy. The simulations investigated ZigBee transmissions, particularly node-to-node communications, and used the Monte Carlo method to obtain each numerical result. Every simulated transmission includes additive noise, which satisfies a Gaussian distribution, to support a simplified authentic transmission model. Different forms of interference are applied, with varying levels of power, to understand the effects of interference on wireless transmissions. A random phase offset is added to each interference signal to aid in resembling real world transceiver conditions. The selected intrusion types are continuous wave (CW) jamming, matched signal interference [17], which mimics the protocol in use (ZigBee), and WiFi (802.11b) coexistence. A CW jammer forms the baseline interference model as it corresponds to typical spurious jammers, including constant, random, deceptive and reactive approaches. The CW method does not need to know what protocol is in use and simply operates by emitting spurious RF signals into busy wireless channels. Matched signal interference operates by monitoring the network and identifying the operating protocol before injecting protocol specific interference, which is difficult to detect compared to conventional jamming techniques [17]. An example is emitting RF signals with the signal and frame structures as per Table I and Fig. 3, respectively. WiFi signals, at the three possible frequency offsets (2, 3 and $7MHz$), investigate the problem of system coexistence and whether it can lead to malicious interference when misused. Using these attacks, executed simulations develop a database of transmitted and received ZigBee samples across a range of JSR values, which are statistically analyzed to identify differences, if any, between received error free and erroneous transmissions (packets).

A maximum likelihood decoder (MLD) operates as the receiver in these simulations. In the MLD, each received 32-chip PN sequence P is compared with a lookup table of ZigBee's predefined sixteen DSSS PN codes $(PN_1, PN_2, ..., PN_{16})$. Here, the received samples are compared to an ideal set of samples for each PN code. In either case, the comparison produces a set of results, $(k_1, k_2, ..., k_{16})$, indicating the Hamming distances, h, of the received PN sequence and each sequence in the lookup table. The Hamming distance indicates the number of chips/samples which are mis-aligned between the two sequences being compared. Hence, minimizing k maximizes the correlation and, so, k is chosen as the minimum value in (2).

$$arg_k \ min \ h(P, PN_k), \ for \ k = (1, 2, ..., 16) \qquad (2)$$

Each of the PN codes are designed to have a sharp autocorrelation peak, low cross-correlation values and to be 2-leveled with an equal number of $1's$ and $0's$. This approach produces sequences resembling white noise, which increases resistance to both unintentional and intentional interference. However, during packet transmissions, chips can be corrupted due to spurious intentional and unintentional interference, coexistence, fading, path losses or obstacles. As long as the value of k (chip errors per PN code) is below a certain correlator error threshold (identified as 10 chip errors in [10]), the correct symbol will be selected. Next, the question of "What constitutes an error?" arises and is answered by a correlation failure, which is an incorrect symbol having the minimum Hamming distance. Here, a single bit error corresponds to a packet error, as this produces a failed correlation calculation.

By applying the described Matlab approaches, two distinct sets of experiments were performed, namely bit error location identification and feature analysis of error samples. The aim of the former procedure was to highlight where bit errors occur in the ZigBee PHY frame (Fig. 3), especially at lower JSR values. It was envisaged that this approach would provide sufficient support for the design of a detection scheme. Both error free and erroneous received samples were then explored to detect if any statistical differences (features) exist, which could identify interference signals. Particularly, erroneous packets at JSR values of 15 dB and lower were analyzed, as matched signal interference attains a packet error rate (PER) of approximately 0.18 at 0dB and 1 at 15dB [17]. JSR values above this point would be readily detectable due to high power levels and packet loss rates. Hence, both subtle and brute force attacks can have destructive results.

In practice, it is initially envisaged that a software defined radio (SDR) setup can be utilized for the collection of transmitted samples, which can be emitted using a ZigBee testbed as described in [18]. The collection of ZigBee samples can be achieved using available software programs, like Simulink or GNU Radio, and hardware, like the LimeSDR Mini or Analog Pluto SDR. If the feature approach proves to be useful based on real world data, which will be initially discussed in Section V, the availability of the received samples on a real WSN node, for example, TelosB, can be investigated. This is seen as achievable as the samples should be available in debug mode, at the very least. In this approach, the computational and energy costs and overhead of the IF sampling will need to be examined. However, notably, the simulated process can be reproduced using available low-cost hardware and software.

Synchronization Header (SHR)		PHY Header (PHY)	PHY Service Data Unit (PSDU)	
Preamble 4 Bytes	SFD 1 Byte	Length 1 Byte	Payload 0 - 125 Bytes	FCS 2 Bytes

Fig. 3. Simplified ZigBee physical layer frame

V. Results

Initially, bit error locations were produced for simulated ZigBee packet transmissions under specific interference signals. The transmissions were investigated using $\approx 18,000$ simulations and three different jamming conditions, as discussed in Section IV. The results are expanded as per the packet segments outlined in Fig. 3 and provided in Figures 4, 5 and 6. These figures provide an insight into how bit errors vary in the ZigBee PHY frame as the jamming power increases. This reveals the fact that the probability of bit errors occurring in the packet preambles decreases with jamming power. Thus, the probability of synchronizations to packets under the presence of a jammer increases at lower jamming powers. Therefore, an interference detection approach will need to analyze packets with bit errors and when no packets are being received, leading to approaches which can work using received packets and received channel samples. In this study, clearly, significantly more errors occur at low JSR values for matched signal interference compared to the other methods while, above $15dB$, high levels of packet corruption are evident in all but the 7 MHz WiFi interference, which requires a JSR of 22dB before errors begin to occur. The WiFi results suggest that at high JSR levels, the protocol can become malicious. In both the CW and matched signal cases, as the JSR value decreases, the probability of receiving an error free preamble increases, which is evident at $\leq 10dB$ for CW and $\leq -5dB$ for matched interference. Overall, the results demonstrate that, at high levels of jamming, bit errors and, consequently, packet errors, occur across the frame, which is as expected. However, as the interference signal becomes more subtle, the probability of receiving an error free preamble increases and bit errors are, likely, confined to the payload. Thus, nodes attempt to process erroneous packets, which eventually fail a frame check sequence and are rejected. This causes retransmissions and increased network and/or system latency which, potentially, has severe consequences for time-critical safety applications. The results illustrate that the cause of packet loss in the wireless channel becomes more challenging to identify, as power levels are as expected (JSR = 0dB). Retransmissions are also required at high levels of JSR, but, due to the high jamming power, it is typically easier to identify the presence of a jammer. Consequently, this error location work provided motivation to look at methods for identifying the presence of interference signals across the range of JSR values. For a distributed edge device investigative approach, it was decided to focus only on features based on the received I/Q samples and to neglect all network and packet rate information.

For the feature analysis, the results focused on matched signal interference, as it produced errors across the largest JSR range (Fig. 4 - 6) and, according to [17], it can achieve a PER of ≈ 0.18, even at a JSR of 0dB. Attention was aimed at determining features based on the statistical analysis of received I/Q samples. As the signals are all mathematically created in Matlab, each received signal can be equated to the appropriate transmitted signal to determine the bit errors present, even

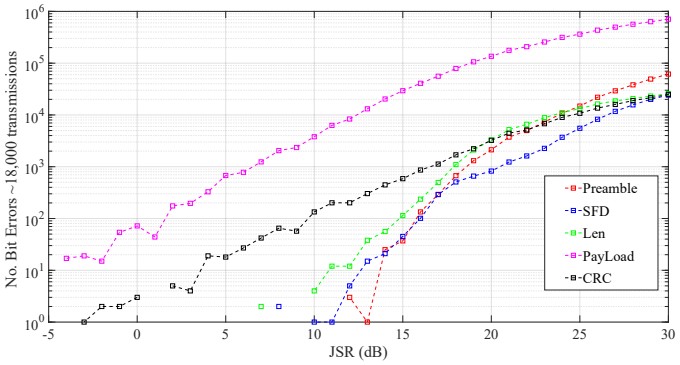


Fig. 4. ZigBee frame bit error locations under CW interference

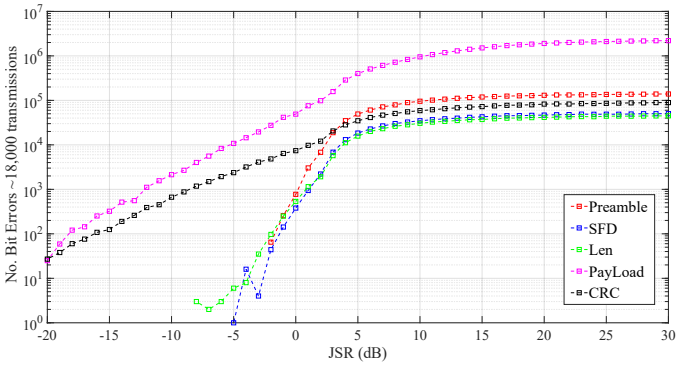


Fig. 5. ZigBee frame bit error locations under matched signal interference

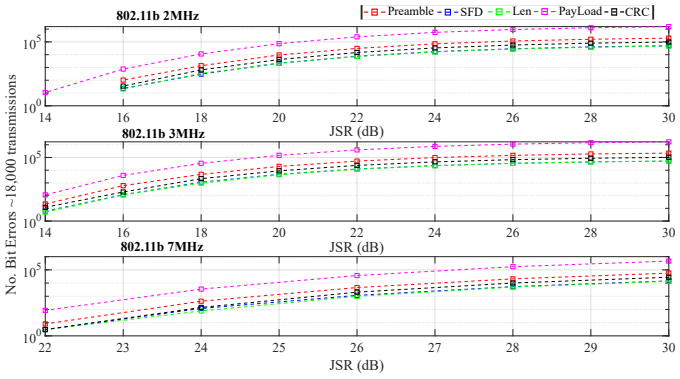


Fig. 6. ZigBee frame bit error locations under 802.11b coexistence

if the received packet was erroneous. As the probability of error (P_e) increases with JSR, the number of Monte Carlo simulations executed rises as the JSR decreases. For the matched interference method, simulations were executed in a logarithmic scale from 4,700 at 30dB JSR to 50,000 at -25dB JSR and for transmissions without interference present, $10,000$ simulations were completed.

At first, the statistical analysis focused on the measured probability density function (PDF) of the I/Q samples. For error free packets, a low-variance, unimodal sample distribution was expected while for erroneous packets a high variance, or bimodal, distribution was anticipated. Fig. 7 indicates that the compact distribution becomes a wide bimodal shape as JSR values increase. Notably, the error free PDF closely resembles what is seen in the spectrum, Fig. 1, as the zero bin is slightly smaller than its two nearest neighbors. This trend under increasing levels of JSR allows features to be extracted from the distribution by analyzing the area within certain regions, determining the maximum peak and the number of non-zero populations. These results are provided in Fig. 8, where JSR values of 5dB and above can be clearly identified. The I/Q samples, which are used to compute the PDF, can also be individually analyzed to produce features as per Fig. 9, where the variance (and standard deviation, which is provided as a consistency check), absolute maximum and mean of the samples, along with the entropy, all contain useful trends which can identify the presence of an interference signal. The entropy is described as "a statistical measure of randomness" and defined by (3), thus, implying that noise signals have a higher entropy value, when compared to high powered dominant signals, and this phenomenon is seen in Fig. 9 (d) as the JSR values increase. Essentially, the extracted features should, theoretically, allow an edge node to determine why erroneous packets are being received by analyzing received samples.

$$S = -\sum_{i} P_i \log_2 P_i \tag{3}$$

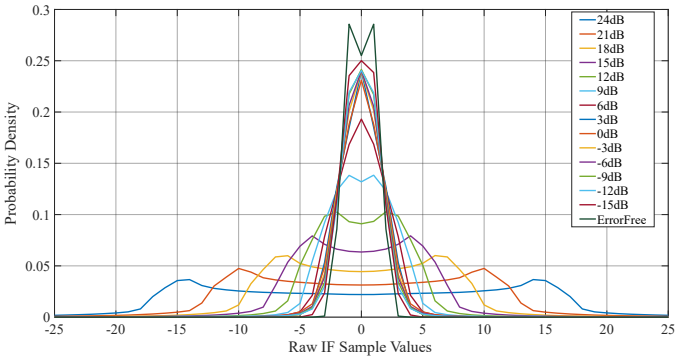


Fig. 7. Measured PDF of simulated I/Q samples under matched interference for various JSR values

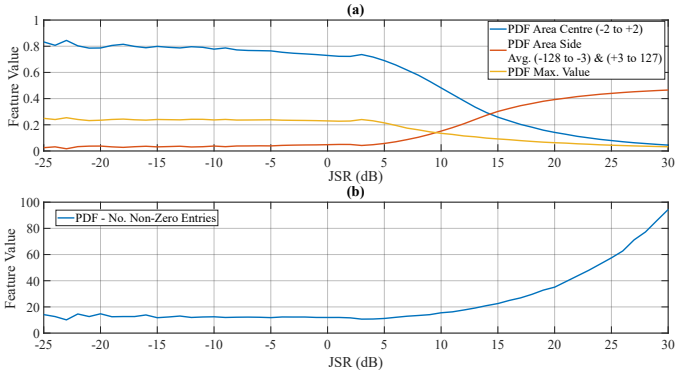


Fig. 8. Sample PDF features as a function of JSR

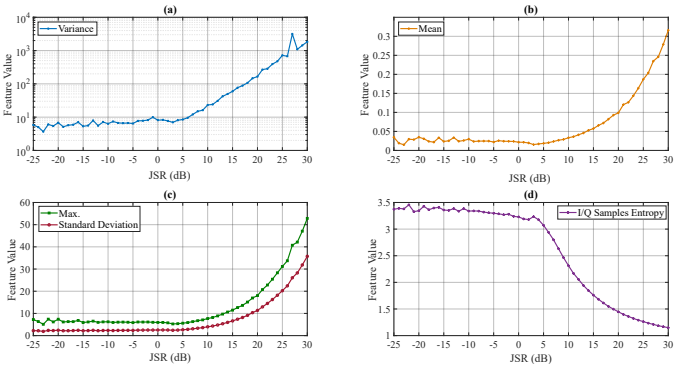


Fig. 9. Features based on I/Q samples

Separately, Table II evaluates the same features for error free packets using $10,000$ individual iterations, where the average, maximum and minimum values are provided to present value fluctuation. Table II demonstrates that the error free transmissions are likely to contain different values compared to erroneous samples. For example, the minimum area of the centre of the PDF for an error free transmission is 0.9065, while the highest error value in Fig. 8 is 0.8445. Additionally, the maximum variance in the error free I/Q samples is 2.4407, while the minimum value in Fig. 9 is 3.677. Hence, the simulated results have extracted features, which can be used to analyze received I/Q samples and, potentially, determine the presence of intentional interference. The initial results implied a threshold of 5dB JSR, but by exploiting a machine learning approach this threshold could be lowered based on the minimum and maximum values identified in Table II and the corresponding values in Figures 8 and 9.

TABLE II
ERROR FREE FEATURES BASED ON 10,000 PACKETS

Feature:	**Error Free Packet**		
	Average Value	Max. Value:	Min. Value
Area Centre	0.9122	0.9189	0.9065
Area Side	5.0044 e-04	9.3700 e-04	1.56 e-04
PDF Maximum	0.2874	0.2964	0.2786
Non Zero Entries	7	8	7
I/Q Samples - Variance	2.3925	2.4407	2.3464
I/Q Samples - Standard Deviation	1.5468	1.5623	1.5318
I/Q Samples - Abs. Max.	3.2069	4.0274	2.888
I/Q Samples Mean	0.005	0.0196	7.4 e-05
I/Q Samples Entropy	3.6486	3.7412	3.5510

To initially validate the usefulness of these features and to provide an initial indication of how effective they are in detecting interference, a SVM was formulated, evaluated and tested. A SVM is a supervised binary classification algorithm and was chosen as the introductory classification approach. Generally, the number of support vectors is much smaller than the total number of elements in the training dataset, hence, training a SVM can be resource intensive, but the actual classification algorithm can be lightweight. This is an important concept for the ability to implement this type of classification on an edge device. The data used to formulate the

TABLE III
SVM RESULTS (VALIDATION DATA): MULTIPLE DETECTION THRESHOLDS AND RADIAL BASIS FUNCTION KERNEL

JSR Detection Threshold:	Selection Reason	10-Fold Cross Validation Error	Test Data Error
$\geq$ 5 dB	Identified Initial Threshold from Feature Trends	7.1099%	4.5205%
$\geq$ 0 dB	Expected Spectral Power	3.9800%	2.6818%
$\geq$ -5 dB	Below -5dB: Likely Error-Free SHR (Preamble and SFD)	0.9736%	0.8390%
$\geq$ -10 dB	Lowest Training JSR Point	0.0016%	0.0020%

TABLE IV
SVM RESULTS (TRAINING DATA): MULTIPLE DETECTION THRESHOLDS AND RADIAL BASIS FUNCTION KERNEL

JSR Detection Threshold:	Selection Reason	10-Fold Cross Validation Error	Test Data Error
$\geq$ 5 dB	Identified Initial Threshold from Feature Trends	6.9619%	4.4508%
$\geq$ 0 dB	Expected Spectral Power	3.9628%	2.6515%
$\geq$ -5 dB	Below -5dB: Likely Error-Free SHR (Preamble and SFD)	0.9741%	0.8380%
$\geq$ -10 dB	Lowest Training JSR Point	2.35e-04%	0.0%

features were divided into separate training (70%), validation (20%) and testing (10%) datasets. For matched interference transmissions, data were split for erroneous results at each individual JSR value and simulation trials were increased logarithmically from 10,000 at 40dB to 60,000 iterations at -15dB, while error free data contained 220,000 trials. Training and validation were both implemented using the data from $-10dB \rightarrow 15dB$, which provided extra points for testing, and Matlab's "fitcsvm" function. Test points included the full JSR range from 40dB to -15dB and no test data was used in training. Initially, the validation data and built-in Matlab functions were used to determine the appropriate SVM kernel to use. The decision was based on the 10-fold cross-validation error and the model training time. This approach encompassed the linear, Gaussian, radial basis function (RBF) and third-order polynomial kernels. The RBF kernel was determined to be the optimal function and, so, was utilized in the SVM analysis. Results for both the cross-validation and testing data errors are supplied in Tables III and IV, which correspond to the validation and training data, respectively. Each table specifies classification errors for different JSR thresholds, which describe possible detection scenarios. For each dataset, the test data error was determined by using each model to predict the result for each data point in the testing dataset, where each prediction was compared with the corresponding annotation to determine the model generalization error. However, the results in Tables III and IV, which exhibit low levels of error, are artificially good as the data used to develop the SVM models were simulated and, so, could not model live wireless signals

exactly. Therefore, wireless channel variations, for example, fading levels, obstacles, path losses, spurious interference, etc., are inadequately modeled with this approach. Therefore, the designed models and methods need to be adapted for wirelessly received I/Q samples. However, the objective of this study was to provide an initial validation of the usefulness of the extracted features and the varying thresholds show that enough differences exist between the error free and erroneous samples, even before the suggested 5dB threshold in figures 8 and 9. Notably, as the threshold reduces, so too does the error, which suggests that features perform better when distinguishing between error free and erroneous samples only. These promising simulation results, which suggest that this framework is a feasible solution, bode well for a hardware approach that supplies real over the air live data signals.

VI. Conclusion

This paper contributed to interference detection in WSNs by focusing on both subtle, where the JSR values are causing PERs of 20% and below, and crude jamming attacks. Matlab, ZigBee and matched signal interference, which has an associated high bit error rate at low JSR levels, were utilized. Bit error location analysis motivated a detection approach for subtle and crude interference attacks. By focusing on received I/Q samples available on a single edge node, features were extracted from the PDF and individual samples. Enough differentiation between error free and erroneous samples existed to warrant an evaluation using a classifier. A SVM was designed and tested using the simulated results and was able to classify unknown signals. The extracted features and SVM demonstrated that this detection method can be suitable for subtle interference, when signals match expected spectrum usage and jamming situations. Notably, the approach neglects additional network information, as the analysis is solely based on received samples. However, this work requires expansion and needs to include real world wireless signals by utilizing the ZigBee testbeds outlined in [18] and software defined radios, for example, the Analog Pluto, which has Matlab/Simulink toolboxes and open source python and Linux libraries ("libiio"). The transition to live signal analysis is a direct result of the favorable SVM feature evaluation and feasibility test. Essentially, this paper developed an initial WSN framework for distributed external interference detection focused on received I/Q samples. The framework designed in this paper can now be adapted for live wireless signals and environments to attain genuine wireless results and characterize the use of the extracted features on received wireless I/Q samples.

Acknowledgment

This work has been jointly funded by the Irish Research Council (IRC) and United Technologies Research Center Ireland (UTRC-I) under the post-graduate Enterprise Partnership Scheme 2016, award number EPSPG/2016/66.

References

[1] T. Vladimirova, C. P. Bridges, J. R. Paul, S. A. Malik, and M. N. Sweeting, "Space-based wireless sensor networks: Design issues," *IEEE Aerosp. Conf.*, pp. 1–14, 2010.

[2] P. Park, S. C. Ergen, C. Fischione, C. Lu, and K. H. Johansson, "Wireless Network Design for Control Systems: A Survey," *IEEE Communiations Surv. Tutorials*, vol. 20, no. 2, pp. 978–1013, 2018.

[3] R. K. Yedavalli and R. K. Belapurkar, "Application of wireless sensor networks to aircraft control and health management systems," *J. Control Theory Appl.*, vol. 9, no. 1, pp. 28–33, 2011.

[4] A. Addaim, A. Kherras, and Z. Guennoun, "Design of WSN with Relay Nodes Connected Directly with a LEO Nanosatellite," *Int. J. Comput. Commun. Eng.*, vol. 3, no. 5, pp. 310–316, 2014.

[5] G. D. O Mahony, P. J. Harris, and C. C. Murphy, "Investigating the Prevalent Security Techniques in Wireless Sensor Network Protocols," in *30th Irish Signals Syst. Conf.*, 2019, pp. 1–6.

[6] M. Li, I. Koutsopoulos, and R. Poovendran, "Optimal jamming attacks and network defense policies in wireless sensor networks," in *INFOCOM 2007. 26th IEEE Int. Conf. Comput. Commun.* IEEE, 2007, pp. 1307–1315.

[7] Z. Yu and J. J. P. Tsai, "A Framework of Machine Learning Based Intrusion Detection for Wireless Sensor Networks," in *IEEE Int. Conf. Sens. Networks, Ubiquitous, Trust. Comput.*, 2008, pp. 272–279.

[8] O. Can and O. K. Sahingoz, "A Survey of Intrusion Detection Systems in Wireless Sensor Networks," in *6th Int. Conf. Model. Simul. Appl. Optim.*, 2015.

[9] A. Abduvaliyev, A.-S. K. Pathan, J. Zhou, R. Roman, and L. W.-C. Wong, "On the Vital Areas of Intrusion Detection Systems in Wireless Sensor Networks," *IEEE Communiations Surv. Tutorials*, vol. 15, no. 3, pp. 1223–1237, 2013.

[10] K. Wu, H. Tan, H. L. Ngan, Y. Liu, and L. M. Ni, "Chip error pattern analysis in IEEE 802.15.4," *IEEE Trans. Mob. Comput.*, vol. 11, no. 4, pp. 543–552, 2012.

[11] F. Hermans, O. Rensfelt, T. Voigt, E. Ngai, L.-Å. Nordén, and P. Gunningberg, "SoNIC : Classifying Interference in 802.15.4 Sensor Networks," *ACM/IEEE Int. Conf. Inf. Process. Sens. Networks*, pp. 55–66, 2013.

[12] S. Grimaldi, A. Mahmood, M. Gidlund, and M. Alves, "An SVM-based method for classification of external interference in industrial wireless sensor and actuator networks," *J. Sens. Actuator Networks*, vol. 6, no. 2, 2017.

[13] B. Stelte and G. D. Rodosek, "Thwarting attacks on ZigBee - Removal of the KillerBee stinger," in *Proc. 9th Int. Conf. Netw. Serv. Manag.*, 2013, pp. 219–226.

[14] Tektronix, "RSA306B USB Spectrum Analyser." [Online]. Available: https://www.tek.com/spectrum-analyzer/rsa306

[15] ——, "DPX Overview," 2019. [Online]. Available: https://www.tek.com/dpx-overview

[16] F. Behmann and K. Wu, *Collaborative Internet of Things (C-IOT)*, 2015.

[17] G. D. O Mahony, P. J. Harris, and C. C. Murphy, "Analyzing the Vulnerability of Wireless Sensor Networks to a Malicious Matched Protocol Attack," in *52nd IEEE Int. Carnahan Conf. Secur. Technol.*, 2018.

[18] ——, "Developing Low-Cost Testbeds for Enhancing Security Techniques in Wireless Sensor Network Protocols," in *30th Irish Signals Syst. Conf.*, 2019, pp. 1–6.