Patterns in the RSSI Traces from an Urban Environment

Nicholas M. Boers, Ioanis Nikolaidis, Pawel Gburzynski
Department of Computing Science
University of Alberta
Edmonton, Alberta T6G 2E8
Canada
Email: {nboers, nikolaidis, pawelg}@ualberta.ca

***Abstract*—Urban environments are notorious for high levels of noise and interference, particularly in their unlicensed radio bands. The interference patterns visible on a given channel depend on the particular devices competing for the spectrum. Meanwhile, the modern transceivers incorporated into wireless sensor network (WSN) nodes have the ability to measure the background noise/interference and change channels. This combination of capabilities suggests the need to better understand the noise and interference encountered in urban environments so that devices can make better decisions to avoid it.**

In this paper, we explore the noise and interference patterns found on 256 frequencies in an urban environment's 900 MHz ISM and non-ISM bands. We begin the process by using off-the-shelf WSN hardware to sample the environment at 5 kHz from 16 locations simultaneously. From these samples, we identify five prevalent patterns and then hand-classify the 4096 traces of noise and interference. Finally, we extract a variety of statistics from the traces and use them in a Bayesian network classifier.

I. Introduction

Dense and dynamic *urban* environments present both opportunities and challenges for wireless sensor network (WSN) deployment. The availability of electrical outlets and backbone wired networks allows for less dependence on batteries and fewer hops, respectively. On the other hand, the wireless devices typically operate within one of the industrial, scientific, and medical (ISM) radio bands. These bands are heavily used, e.g., by cordless phones, WLANs, building automation networks, and microwave ovens, so devices need to be resilient to interference.

When a wireless node receives a transmission, the ratio between its signal strength and any interference plus background noise (SINR) ultimately determines its fate [1]. In an environment without motion, received signal strengths tend to remain relatively stable over time. Background noise is often assumed to be additive white Gaussian noise (AWGN), and like received signal strengths, remains relatively stable. The final variable, interference, is a potentially large uncontrolled source for variation in the SINR, and yet, it has received relatively little attention in the literature. Strong interference can affect all of the nodes in an environment, and channels operating near their receive sensitivity are especially sensitive to even small changes in the SINR.

During experiments with a medium-scale indoor WSN [2], we found that nodes often experienced high packet losses even at short distances and periods of no congestion. These difficulties led us to this exploration of their behaviour at that frequency and many others. Our work adds to the already healthy scepticism related to the proper modelling in network simulations that has started in earnest with the paper by Kotz et al. [3]. However, we target specifically the problem of characterization based on evident interference patterns.

After sampling the interference and noise on 256 different frequencies, we aim to (a) highlight the importance of evaluating channels prior to performing real-world experiments and (b) show that while many channels may be suitable for low-powered communication, some clearly are not. Recognizing these poor channels is important because modern transceivers have the ability to change channels. When nodes detect that they are using a poor channel, they can make the change to a better frequency.

We have no illusions about the generality of our results: we sampled one specific environment, on a particular date, over a particular period of time. We realize that the same environment may exhibit different characteristics if sampled again, and other environments may be completely different. That said, it is important for researchers to realize that the combination of noise and interference is rarely straightforward AWGN.

Our work is relevant to the area of cognitive networking, and in particular it attempts a high-level characterization of channels that are potentially occupied by (licensed or unlicensed) users. In cognitive radio, the characterization is performed with the intent of determining the (near-term) *occupancy* as expressed through the observed noise+interference combination. Cognitive networking needs such a step before opportunistically using one or more of those channels. In fact, rather than producing a binary (occupied/unoccupied) classification of channels, we characterize channels into five different classes. Some of the classes, for example those exhibiting presence of spread spectrum or UWB interferers, could, depending on the cognitive network scheme, be used at the same time for narrowband transmissions by the cognitive network. However, in our study, we do not take a position as to *how* the channels will be used, but rather what is a good (and as exhaustive as possible) characterization of channels based on their noise+interference time series behaviour.

One should also keep in mind that the ultimate goal of any WSN performance study is to guide practical deployments involving tangible hardware with painfully idiosyncratic properties. The most extreme example of such a property in our case is the systemic suppression of certain channels separated by half-multiplies of the crystal frequency driving the RF chip (seen in Fig. 3). It is clearly impossible to account for all such "features" in any blanket theoretical model, as it is in any experimental study whose results one may be tempted to extrapolate onto unexplored hardware. Nonetheless, the actual behaviour of real devices is what ultimately matters the most; thus, we should try to bring as much order as possible into those necessarily unsystematic, but extremely valuable from a practical standpoint, observations made by real-life implementers.

In this paper, we make a number of contributions. We describe a setup for high-frequency RSSI sampling using off-the-shelf wireless sensor nodes and sufficient cabling. For a particular environment, we show the variety of channels that we encountered and classify them into a few groups. In many cases, these channels differ greatly from the commonly assumed AWGN model. In support of these classes, we compute statistics from each trace, use them to train a Bayesian classifier, and present the results of using that classifier.

II. Related Work

A. The Transitional/Gray Region

Many researchers have reported on a transitional (or gray) region in the SINR where, over time, a subset of nodes may fluctuate between successful and failed transmission. Patterns in the interference are most likely to affect these nodes first, and for that reason, we review related work on the transitional region.

Zhao and Govindan [4] quantified the size of the area in three different environments: an office building, a park, and a parking lot. They set up 60 motes operating in the 70-centimetre amateur radio band at 433 MHz in a line topology and had the node at one end transmit packets at 1 Hz. In the building and park environments, they noticed surprisingly large gray areas of almost one-third and one-fifth of the communication ranges, respectively.

Later experiments by Son, Krishnamachari, and Heidemann [5] considered one less variable: hardware variations. They discovered that for a particular node and level of signal strength, the gray region is actually quite narrow. Furthermore, the specific width and location of the gray region depends on both (a) the transmitter hardware and (b) the transmission power. The gray region only appeared relatively wide when many radios were used, and in that case it spanned roughly 6 dB.

More recently, Zamalloa and Krishnamachari [6] approached the problem from a different perspective – mathematically. Although they derived expressions for the location and extent of the transitional region, their model does not consider interference. They acknowledge that the noise floor varies over time, and mention large changes in temperature and interference as possible causes.

B. Sampling and Observations

From 2007 to 2010, researchers working on closest-fit pattern matching (CPM) sampled noise in various environments. In their experiments, they sampled noise at low rates (1 kHz or less), and they made only informal comments about interference patterns.

Lee, Cerpa, and Levis [7] used the CC2420 IEEE 802.15.4 transceiver to sample the chip's RSSI register at 1 kHz. Note that this transceiver updates its RSSI value at 62.5 kHz by averaging over the previous eight symbols periods (128 μs). They stored the retrieved measurements in the device's flash memory, which allowed them to record samples for 197 s. Fifteen of the sixteen IEEE 802.15.4 channels overlap with IEEE 802.11b, and they sampled noise on overlapping and non-overlapping channels. They sampled the noise in Wi-Fi enabled buildings, Wi-Fi enabled outdoor areas, outdoor quiet areas, and controlled areas. In their samples, they noticed three key characteristics: (a) the noise traces contained spikes, sometimes as strong as 40 dBm above the noise floor, (b) many of the spikes in the noise traces were periodic, and (c) the noise patterns changed over time. They did not encounter two of the other characteristics that we observed (Section IV).

Rusak and Levis [8] collected RSSI samples from packet transmissions in buildings at Cornell and Stanford University. They used TelosB motes with CC2420 transceivers and transmitted packets at both 4 and 100 Hz between a pair of nodes, concentrating on modelling the signal strength rather than just noise.

Most recently, Srinivasan, Dutta, Tavakoli, and Levis [9] expanded on much of their previous work. They explored the correlation of noise traces, where six synchronized nodes sampled RSSI values at 128 Hz. Furthermore, they observed 802.11b interference at 45 dBm impacting their 802.15.4 channel and, as a solution, suggest avoiding channels that coexist with 802.11b networks to minimize interference and loss.

III. Data Collection

To better understand the interference patterns present on channels in a WSN, we developed software to measure the conditions in our network. We set up a grid of nodes in the Smart Condo at the University of Alberta [2]. Within the 80 m^2 space, we deployed a four-by-four grid of 16 wireless nodes with 1.83 m spacing. We elevated each node 28 cm off of the floor. While running the experiments, we kept the room's doors closed and there was no movement within the room.

In the experiments, we used EMSPCC11 wireless nodes provided by Olsonet Communications [10]. These devices consist of a TI MSP430F1611 microcontroller and TI CC1100 transceiver. We configured the CC1100 for 38.4 kbit/s using 2-FSK modulation. In terms of software, they run a low-footprint operating system named PicOS [11] that supports multithreaded applications.

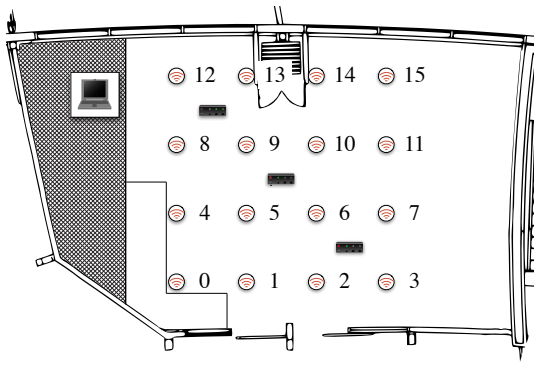

Fig. 1. The experiment setup within the Smart Condo. The circles represent the nodes. The small black boxes represent the 7-port USB hubs. The notebook computer in the top-left recorded the results.

A PicOS application collects noise measurements by reading an RSSI value from the CC1100's RSSI register and writing the 8-bit 2's complement value to the UART without converting it to dBm. It performs these actions in a tight loop that involves the OS scheduler at every iteration, but since no other threads are running, the scheduler latency is relatively constant. We verified the periodicity of our sampling by toggling an LED on every call to the output function and monitoring the pin with an oscilloscope. With this software, we can obtain the RSSI at over 5 kHz, while given our settings, the CC1100 updates the RSSI value at over 12 kHz.

Writing values to the UART necessitates connecting all the nodes to a computer by wire. The EMSPCC11 provides direct access to the MSP430's UART pins at TTL levels, so we opt to use a TTL RS232 to USB interface cable (FTDI TTL-232R-3V3). Now with a USB interface, it is easy to connect all 16 nodes to a single computer using a combination of USB extension cables and powered 7-port USB hubs (Digitus DA-70227). In making the connections, we do not exceed USB's maximum cable length of 16.4 feet. Fig. 1 shows the experiment setup on the blueprint for the space. We include the node numbers in this figure for future reference.

A single application on the PC opens all 16 serial ports when it starts. Samples arrive at 80 kHz, and the application stores each sample in a line of 25.6 bytes (average) in a CSV text file. The application is thus writing the text file at around 2 MB/s.

The whole application is very sensitive to latency. The TTL-232R-3V3 has a 256-byte receive buffer (about 20 ms of buffer space), and we initially experienced buffer overruns. To eliminate them, we take the following steps:

- introduce a large circular buffer of blocks of bytes (64-kb blocks),
- in the main thread, read the ports and write measurements to the circular buffer,
- assign this thread real-time priority (in Mac OS X 10.6, give it 2.5 ms of computation time every 5.0 ms and allow it to be preempted), and
- in a second thread, write completed buffer blocks to disk.

After taking these steps, the buffers never filled to more than 100 bytes.[1] In fact, in the early tests of this modified setup, the application would read samples from the buffer less than 10 bytes at a time.

The PC-based application produced timestamps for all measurements as they arrived. We adopted this approach to reduce the data sent over the serial link and to nullify the effect of node clock drift, which was surprisingly high even over short periods, When timestamping the data, it interpolates times using the last time we read a block and the current time as bounds. Since the readings are sent periodically, we can always rewrite the timestamps later if they must be perfectly periodic.

Before taking proper measurements, we verified whether the connection of our motes by wires to a single data collection point affected their RF communication. Note that there was no reason to suspect influence, because of the careful isolation of the RF tract on the EMSPCC11, which is a common practise in professionally designed RF equipment. By disconnecting individual nodes and (visually) inspecting their RSSI readings we were able to ascertain that neither the USB cables alone, nor their connections to the central hub had a perceptible impact on the RF channels.

With the described measurement framework in place, we proceeded to measure the RSSI on each of the node's 256 channels. Our nodes were configured with a base frequency of 904 MHz. The channels are spaced 199.9512 kHz apart, and each channel occupies a bandwidth of 101.5625 kHz. These settings allow our nodes to listen on frequencies from 904 MHz to 928 MHz (within the ISM band) and 929 MHz to 954 MHz (outside the ISM band). For each channel, we collected exactly 175 000 samples for each node for a duration of less than 35 s. For the whole collection process, the span was roughly 2.5 h and the final CSV data file consumed 18.38 GB.

IV. Channel Classification

After collecting the 4096 RSSI traces, we began exploring them by plotting each against time. Through visual inspection, we identified five general categories for the interference patterns.

1) The *quiet* channel is characterized by a low maximum.
2) The *quiet-with-spikes* channel is similar to (1), but it has short-duration spikes that give it a higher maximum.
3) The *quiet-with-rapid-spikes* channel has a higher frequency of spikes than (2).
4) The *high-and-level* channel exhibits a high and tight level and has a high minimum.
5) The *shifting-mean* channel has its RSSI samples distributed bimodally.

See Fig. 2 for an illustration of each class of channel.

Given the general classes of interference patterns, we hand-classified the trace for each channel/node combination. The task of classifying the samples was particularly difficult given that a single trace might contain overlapping patterns or a

[1] In the application, we added code to print out the buffered number of bytes if it ever exceeded 100. In our experiments, this code was never executed; the maximum buffered bytes may have been far less than 100.

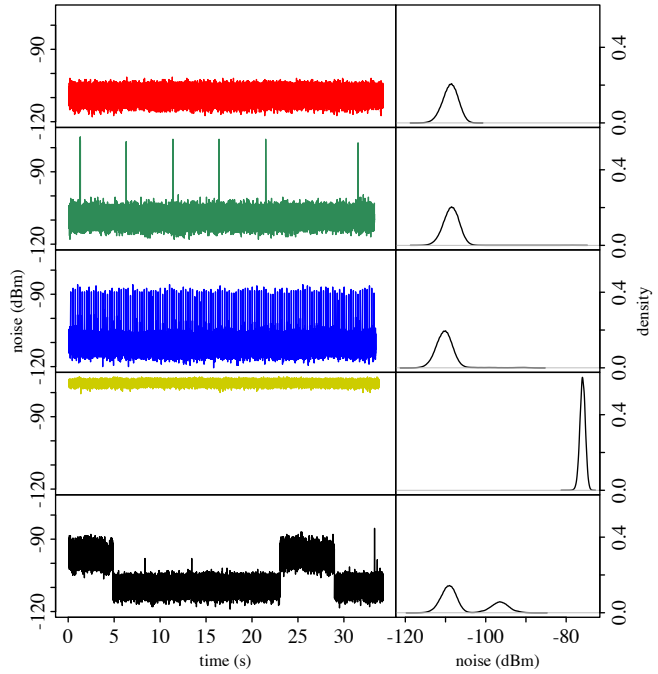

Fig. 2. The different primary classes of channels that we identified in our RSSI traces. From top to bottom, the figure shows the samples and density plot for the (a) quiet channel, (b) quiet-with-spikes channel, (c) quiet-with-rapid-spikes channel, (d) high-and-level channel, and (e) shifting-mean channel.

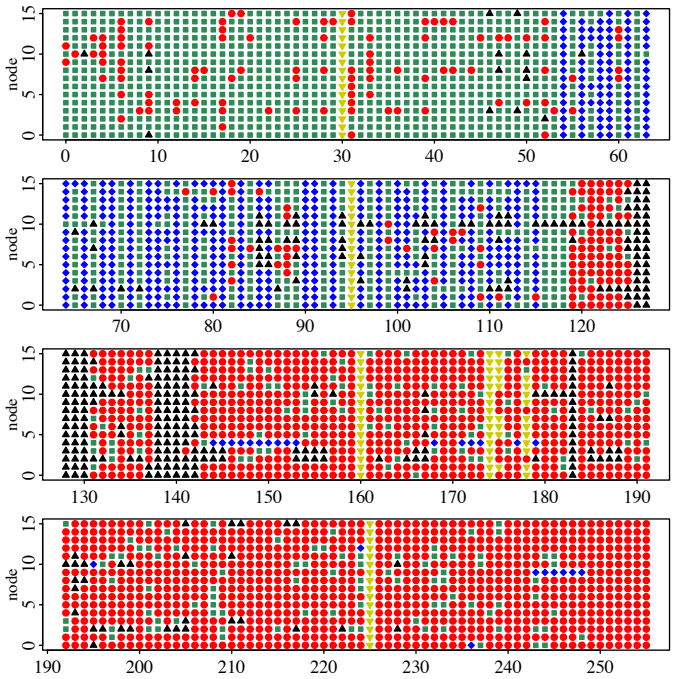

Fig. 3. The result of hand-classifying noise traces from 256 channels with 16 nodes per channel. The correspondence between colour and classification is as follows: (a) red ●: quiet, (b) green ■: quiet-with-spikes, (c) blue ◆: quiet-with-rapid-spikes, (d) yellow ▼: high-level, and (e) black ▲: shifting mean.

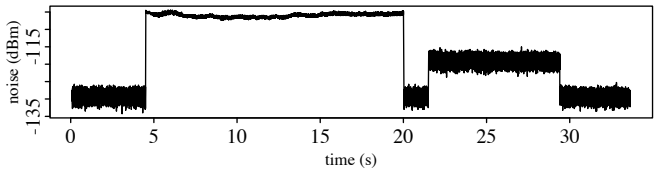

Fig. 4. A sample from ch. 139 (931.793 MHz) at node 6 that shows particularly strong interference. We attribute these non-ISM noisy samples to pagers.

pattern at only a weak strength. We did not test the intra-rater reliability of these hand-classifications, and we would expect some variance. When more than one characteristic was present in a trace, we tried to class it as the visually dominant one. For example, we tended to classify a trace as shifting-mean rather than quiet-with-spikes and quiet-with-rapid-spikes rather than shifting-mean. Fig. 3 shows the classifications for the 4096 combinations and Table I summarizes our counts for ISM and non-ISM bands.

TABLE I
THE NUMBER OF TRACES CLASSIFIED AS EACH TYPE FOR BOTH ISM AND NON-ISM BANDS.

Classification	ISM	Non-ISM	Total
quiet	155	1614	1769
quiet-with-spikes	1146	179	1325
quiet-with-rapid-spikes	523	25	548
high-level	32	69	101
shifting-mean	80	273	353

We encountered spikes predominantly within the ISM band. When inspecting some of the spikes, we calculated very short durations of around 6 ms; we speculate that they result from one or more frequency-hopping interferer.

Fig. 3 highlights a curious pattern in channels grouped in the class high-level. The four cases on channels 30, 95, 160, and 225 are located 65 channels apart. We set up the nodes in a new environment, added some shielding to them, and again measured these channels to find no decrease in the strength of the interference. That was suggestive of a hardware issue and, indeed, a closer inspection revealed the problem signalled in [12] consisting in a systemic attenuation of those channels whose frequencies fall at $806 + n \times 13$ MHz (13 MHz is $1/2$ of the chip's crystal frequency), which perfectly agrees with the observed "anomaly."

In the non-ISM band, we noticed that channels 126-130 and 138-141 had very powerful shifting-mean signals (e.g., see Fig. 4 for a node operating on channel 139 at 931.793 MHz). Using Spectrum Direct at Industry Canada, we searched the Assignment and Licensing System (ALS) database[2] and found that the closest registered frequency is 931.737500. This frequency is registered to Telus Communications Inc. for their 900 MHz paging service. For the lower channels, 126 to 130, we also found that they were registered for paging services.

We also found that the licensed spectrum is relatively quiet apart from the paging service (accounting for the explained anomalous behaviour of channels $30 + n \times 65$).

[2]See http://www.ic.gc.ca/eic/site/sd-sd.nsf/eng/h_00025.html.

A. BayesNet Classification

After manually classifying the traces, we explored the feasibility of automating such classification. From each trace, we extracted the following features: (a) mean, (b) standard deviation, (c) skew, (d) kurtosis, (e) minimum, (f) maximum, (g) dip, (h) 99.5 percentile, and (i) 99 percentile. The dip statistic is used in a test for unimodality [13]. We included the percentile metrics in an attempt to better identify the quiet-with-rapid-spikes traces.

Using Weka 3.6.2 [14], we explored using its BayesNet classifier on the features extracted from the traces. We loaded a data file prepared in the ARFF format containing the class of channel and the statistics. On these data, we ran 10-fold cross-validation with the BayesNet classifier. Overall, it correctly classified 81.03% of the instances. Table II provides accuracy statistics by class and Table III shows the confusion matrix.

TABLE II
ACCURACY STATISTICS FOR THE BAYESNET CLASSIFIER WITH 10-FOLD CROSS-VALIDATION.

	TP rate	FP rate	Precision	Recall
quiet	0.876	0.055	0.923	0.876
quiet-with-spikes	0.766	0.080	0.820	0.766
quiet-with-rapid-spikes	0.746	0.076	0.604	0.746
shifting-mean	0.703	0.041	0.620	0.703
high-level	0.960	0.001	0.951	0.960

TABLE III
CONFUSION MATRIX FOR THE BAYESNET CLASSIFICATION. ABBREVIATIONS AS FOLLOWS: Q: QUIET, QS: QUIET-WITH-SPIKES, QRS: QUIET-WITH-RAPID-SPIKES, SM: SHIFTING-MEAN, AND HL: HIGH-LEVEL.

		Classified as				
		q	qs	qrs	sm	hl
Actual	q	1550	98	54	62	5
	qs	113	1015	140	57	0
	qrs	6	101	409	32	0
	sm	7	24	74	248	0
	hl	3	0	0	1	97

The BayesNet classifier obtained the best performance with the quiet and high-level traces. The classifier had more trouble on the channels with spikes and the shifting means. All three of these types can look quite similar depending on the strength (dBm) of the pattern.

V. CONCLUSION

We deployed a WSN in an indoor urban environment and set out to sample its noise and interference on 256 channels ranging from 904 to 954 MHz. By visually inspecting the collected data, we observed five distinct patterns in the traces; we then hand-classified each trace as belonging to one of these types. Subsequently, we explored using a Bayesian network classifier on statistics extracted from the traces. The classifier performed reasonably well on all types of traces, exhibiting (not surprisingly) somewhat better accuracy for the quiet and high-level classes.

In the future, we will explore classifying channels based on fewer samples. Given that modern transceivers have the ability to change channels, it is important that they can assess channels quickly and accurately to select the best available. Once such classification can be carried out with a satisfying accuracy, one can think about organizing the low-level transmission techniques (physical and MAC layers) around the input from the classifier. To hint at the flavor of possibilities, consider the trade-off between multiple (redundant) transmissions, reduction of bit rate, various acknowledgement techniques, forward error correction, and so on. Many low-cost RF modules (including the CC1100, for example) make those options possible; however, to be helpful, the choice must be driven by the perceived characteristics of real-life channels, rather than some generic and blanket models whose ability to instigate constructive decisions in this area is seriously limited.

ACKNOWLEDGEMENTS

The authors thank Alberta Advanced Education and Technology and the Killam Trusts for financial support.

REFERENCES

[1] A. Räisänen and A. Lehto, *Radio engineering for wireless communication and sensor applications*. Artech House Publishers, 2003.

[2] N. M. Boers, D. Chodos, J. Huang, E. Stroulia, P. Gburzynski, and I. Nikolaidis, "The Smart Condo: Visualizing independent living environments in a virtual world," in *PervasiveHealth '09: Proceedings from the 3rd International Conference on Pervasive Computing Technologies for Healthcare*, London, UK, Apr. 2009.

[3] D. Kotz, C. Newport, R. Gray, J. Liu, Y. Yuan, and C. Elliott, "Experimental evaluation of wireless simulation assumptions," in *Proc. of the 7th ACM Intl. Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. ACM Press New York, NY, USA, 2004, pp. 78–82.

[4] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," in *SenSys '03: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*. New York, NY, USA: ACM Press, 2003, pp. 1–13.

[5] D. Son, B. Krishnamachari, and J. Heidemann, "Experimental study of concurrent transmission in wireless sensor networks," in *SenSys '06: Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*. New York, NY, USA: ACM, 2006, pp. 237–250.

[6] M. Z. Zamalloa and B. Krishnamachari, "An analysis of unreliability and asymmetry in low-power wireless links," *ACM Transactions on Sensor Networks*, vol. 3, no. 2, p. 7, 2007.

[7] H. Lee, A. Cerpa, and P. Levis, "Improving wireless simulation through noise modeling," in *IPSN '07: Proceedings of the 6th International Conference on Information Processing in Sensor Networks*. New York, NY, USA: ACM, 2007, pp. 21–30.

[8] T. Rusak and P. Levis, "Physically-based models of low-power wireless links using signal power simulation," *Computer Networks*, vol. 54, no. 4, pp. 658 – 673, 2010, advances in Wireless and Mobile Networks. [Online]. Available: http://www.sciencedirect.com/science/article/B6VRG-4X3DN2F-1/2/29828104e383e7d926e4bcbef9078240

[9] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, "An empirical study of low-power wireless," *ACM Trans. Sen. Netw.*, vol. 6, no. 2, pp. 1–49, 2010.

[10] Olsonet Communications Corp., "Platform for R&D in sensor networking," http://www.olsonet.com/Documents/emspcc11.pdf.

[11] E. Akhmetshina, P. Gburzynski, and F. Vizeacoumar, "PicOS: A tiny operating system for extremely small embedded platforms," in *Embedded Systems and Applications*, H. R. Arabnia and L. T. Yang, Eds. CSREA Press, 2003, pp. 116–122.

[12] Chipcon, Texas Instruments, "Note on CC1100 device discrepancies," Technical documentation, application note, Sep. 2005.

[13] J. A. Hartigan and P. M. Hartigan, "The dip test of unimodality," *The Annals of Statistics*, vol. 13, no. 1, pp. 70–84, 1985. [Online]. Available: http://www.jstor.org/stable/2241144

[14] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *SIGKDD Explorations*, vol. 11, no. 1, 2009.