Calibrating RSSI Measurements for 802.15.4 Radios

Yin Chen yinchen@cs.jhu.edu
Andreas Terzis terzis@cs.jhu.edu

Abstract

Wireless sensor network protocols and applications, including those used for localization, topology control, link scheduling, and link quality estimation, make extensive use of Received Signal Strength Indication (RSSI) measurements. In this paper we show that inaccuracies in the RSSI values reported by widely used 802.15.4 radios, such as the CC2420 and the AT86RF230, have profound impact on these protocols and applications. Furthermore, we experimentally derive the response curves which translate actual RSSI values to the raw RSSI readings that the radios report and show that they contain non-linear and even non-injective regions. Fortunately, these curves are consistent across radios of the same model, making RSSI calibration practical. We present a calibration mechanism that removes the artifacts in the raw RSSI measurements, including ambiguities created by the non-injective regions in the response curves, and generates calibrated RSSI readings that are linear. This calibration removes many of the outliers generated when raw RSSI readings are used to estimate Signal to Noise (and Interference) ratios, estimate radio model parameters, and perform RF-based localization.

1 Introduction

The IEEE 802.15.4 standard specifies that a radio's PHY layer must provide an 8-bit integer value as an estimate of the received signal power [9]. This value is commonly known as the Received Signal Strength Indication (RSSI) in the wireless sensor networks (WSN) community. Numerous WSN protocols use RSSI measurements extensively, including those for localization [8, 12, 22, 24], link quality estimation [13, 19], packet reception ratio modeling [23] and transmission power control [11, 17, 18].

While many protocols directly use the RSSI measurements that the radios provide, the standard only requires that the reported RSSI values should

be linear and within ±6 dB of the actual RSSI values. However, ±6 dB is a wide error margin. For example, Packet Reception Ratio (PRR) can decrease from 100% to 0% with a 2 or 3 dB difference in the received signal strength [13]. The consequence of this observation is that possible inaccuracies in the reported RSSI values can profoundly impact applications that rely on RSSI measurements.

In this paper we examine two 802.15.4 compliant radios, the widely used Chipcon/TI CC2420 [20] and Atmel AT86RF230 [2], and show that they do indeed introduce systematic errors in the RSSI measurements they provide. As a matter of fact, the coarse *RSSI value vs. input power* graph included in the CC2420 datasheet hints at the existence of non-linearities. Nevertheless, the manufacturer states that the RSSI response curve is very linear [20]. We independently derive high resolution RSSI response curves using a variable signal generator and verify the existence of the non-linearities hinted by the CC2420 datasheet. We also note that the AT86RF230 datasheet does not provide an equivalent graph. Fortunately, these response curves are radio-specific but device independent. In other words, different physical devices that use the same model of radio have identical response curves. Consequently, mitigating these nonlinearities does not require calibrating each device individually.

This result allows us to develop a generic calibration scheme to compensate for the radio's inaccuracies. Specifically, we derive a reference RSSI response curve which determines the calibrated RSSI value for the raw RSSI value that the radio reports. However, due to the existence of the non-injective regions in which a raw RSSI value maps to multiple actual RSSI values, the reference RSSI response curve is not able to always provide the necessary mapping. To resolve this problem, we leverage the ability of 802.15.4 radios to transmit at multiple power levels and dynamically fit a receiver's raw RSSI measurements to the radio's RSSI response curve. This approach provides an excellent fit and is able to accurately resolve the ambiguities that non-injective regions generate. Finally, we present the profound impact of the RSSI nonlinearities and quantify the benefits of the proposed calibration scheme on a wide variety of applications.

The paper has five additional sections. The section that follows reviews background material on low-power radios and RSSI measurements. Section 3 presents the results of an experiment that investigates the influence of packet size on the Packet Reception Ratio (PRR). Section 3 also presents the mechanism we developed to derive the high resolution RSSI response

curves exhibiting the nonlinearities mentioned previously. Section 4 details the proposed RSSI calibration scheme, while Section 5 presents the adverse impact of the RSSI nonlinearities on various protocols and applications, and the improvements that the proposed calibration scheme achieves. We close in Section 6 with a brief discussion.

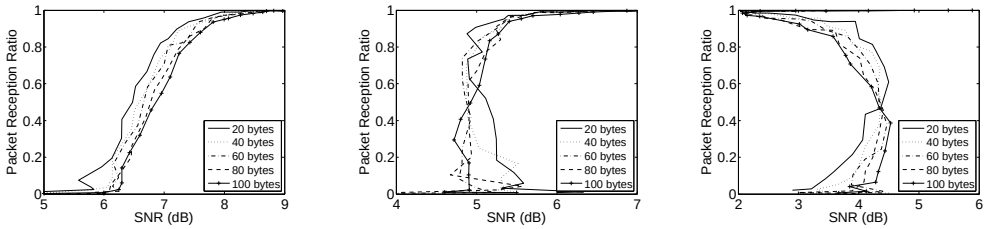
2 Background

Many of the popular hardware platforms in wireless sensor networks today use radios complying to the IEEE 802.15.4 standard [9]. This standard was developed specifically for low-power and low-cost embedded devices and implementations from multiple vendors are available today.

One such implementation, the TI/Chipcon CC2420 [20] is used in multiple platforms [5, 15]. It allows the user to select one of eight output power levels, ranging from -25 dBm to 0 dBm. The Atmel AT86RF230 [2] is another 802.15.4 radio, used in the Iris mote [6]. In addition to higher receiver sensitivity this radio transmits at one of 16 power levels, from -17.2 dBm to 3 dBm.

Both radios provide an 8-bit register which indicates the strength of the received radio signal (RSSI). The 8-bit RSSI value is averaged over 8 radio symbol periods, i.e., 128 μs. Reported RSSI values are measured in dBm, in one dBm increments. There are two categories of RSSI measurements. The first category measures the strength of the radio signal corresponding to a received packet, while the second measures the power of the ambient channel noise. Using these two RSSI values, one can compute the Signal-to-Noise ratio (SNR) for a received packet. We will refer to these two types of RSSI values as **signal RSSI** and **noise RSSI** respectively throughout the rest of this paper. Furthermore, we name the RSSI values provided by the radio chips as **raw RSSI** or **reported RSSI** interchangeably. We will show that reported RSSI values are nonlinear with respect to actual received signal power, defined as **actual RSSI**. The calibration scheme introduced in Section 4 can eliminate the nonlinearity and we term the resultant RSSI values as **calibrated RSSI**.

As part of our effort to improve the fidelity of the TOSSIM simulator [10], we performed an experiment, detailed in Section 3.1, to generate a model for the relationship between Packet Reception Ratio (PRR) and SNR. Specifically, TOSSIM does not consider the packet's size when determining whether



(a) Mote 1. (b) Mote 2. (c) Mote 3.

Figure 1: Packet Reception Ratios (PRR) as a function of Signal to Noise ratio (SNR). The curves were experimentally derived from three different motes using CC2420 radios. PRR curves for multiple payload sizes are generated for each case. While the results from cases (a) and (b) follow the expected pattern, the pattern in case (c) is counter-intuitive, with PRR improving as SNR decreases.

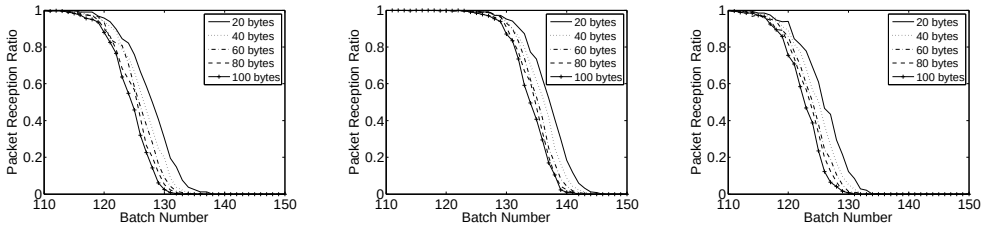
it will be successfully received. This simplification can underestimate or overestimate the packet loss that an application will experience in practice because packet sizes can vary from a few bytes (e.g., ACKs) to above one hundred bytes.

Figure 1 presents the PRR versus SNR curves we experimentally derived using three Tmote Sky motes and various payload sizes. SNR values are determined using the reported RSSI values. It is evident from this figure that there is no consistent correlation between packet size and PRR. More alarmingly, Figure 1(c) suggests that PRR improves as SNR decreases! We will show in the next section that this counter-intuitive (and incorrect) result is the consequence of the RSSI nonlinearity.

3 Accuracy of RSSI Measurements

3.1 Influence of Packet Size on Packet Reception Ratio

We conducted the packet size experiment in an indoor testbed comprising 13 Tmote Sky motes equipped with CC2420 radios [20]. The motes were placed at fixed locations in a quiet office and were powered through their USB ports to eliminate variations due to different battery power levels. A sole transmitter periodically broadcasted packets to the other motes. Furthermore, to minimize interference from co-located WiFi networks, we used



(a) Mote 1. (b) Mote 2. (c) Mote 3.

Figure 2: PRR versus batch number for three motes with various payload sizes. All mote curves are consistent, shifted only by X-offsets corresponding to location differences.

802.15.4 channel 26 that does not overlap with any 802.11 b/g channels [14].

Considering that the mote locations are fixed and radios can transmit at only eight power levels[1], we generate a wide range of SNR values by varying the ambient noise level N. We do so by generating noise signals of variable power levels using a Universal Software Radio Peripheral (USRP) [7]. The noise signal the USRP generates has an almost flat power spectral density within the frequency range of 802.15.4 channel 26.

In this experiment, we increase the noise strength linearly (in dBm) using a constant step size. The linearity was validated using the Anritsu MS2721B spectrum analyzer [1]. At each noise strength level, the transmitter broadcasts a batch of 2,500 packets of five different payload sizes. To minimize the impact of temporal variations in the radio channel, the transmitter broadcasts packets with different sizes at an inter-packet interval of 25 ms. For each batch of received packets we calculate the PRR and average SNR using reported RSSI at each receiver mote. Figure 1 presents the results of these calculations for three receiver motes. It is clear from the mote-specific patterns that different motes report different results. The results in Figure 1(c) are especially puzzling, suggesting that hardware variations or even faults may be at play.

We use Figure 2, which plots the PRR versus batch number curves generated from the same data, to verify that the radios function correctly. Note that noise strength increases with each successive batch, while the signal

[1]While the CC2420 datasheet mentions a total of 31 possible transmission levels, it specifies the output power levels for only eight of them.

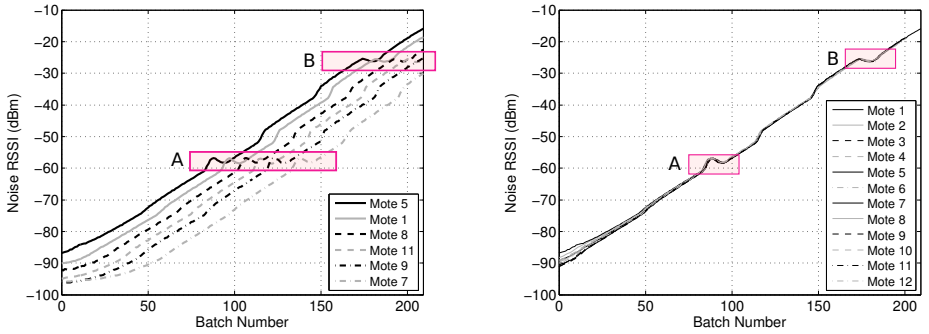
strength remains constant. The SNR therefore decays as batch number increases and thus one expects that PRR will decrease accordingly. Indeed, Figure 2 confirms this trend. Furthermore, unlike Figure 1, the results from the three motes are consistent. The X-axis offsets are due to the different locations of the motes, leading to different received signal strengths and noise levels. This result indicates that the underlying cause of the discrepancies shown in Figure 1 is not device variability or failure. Instead we posit that they are due to inaccuracies in the RSSI values that the motes report, leading to inaccurate SNR calculations.

3.2 RSSI Response Curves

Next, we design an experiment to derive high resolution RSSI response curves and verify the hypothesis in the previous section that the inaccurate reported RSSI values lead to the results presented in Figure 1.

We conducted this experiment in the same indoor testbed used for the previous experiment. However, unlike the previous experiment, there is no mote transmitter. Instead, twelve Tmote Sky motes periodically sense the noise signal that the USRP generates. The benefit of this approach is that it allows us to generate signals with a much wider range of transmit powers, compared to the eight levels available from the CC2420. Like the previous experiment, the strength of the USRP noise increases linearly (in dBm) with each successive batch, therefore the actual RSSI at the motes should also be linear with respect to batch number. We note that although the radios report integer RSSI values, sub-dBm accuracy can be achieved by averaging a series of RSSI measurements. We also note that the noise strength increment per batch is different from the previous experiment.

Figure 3(a) illustrates the RSSI measurements recorded by six of the twelve motes. We omit the results from the remaining motes in the interest of clarity because they show similar patterns. It is evident that the noise RSSI curve for every mote can be divided into several major linear segments. Within each segment, the mapping between the noise RSSI and the batch number is linear. Moreover, the slopes for all the linear segments are almost equal. In the transitional regions that connect these linear segments, however, the mapping between the noise RSSI and the batch number is linear with different slopes or even nonlinear. Furthermore, some of these transitional regions are not monotonically increasing. This violates the most important assumption about RSSI: RSSI readings should be higher for stronger



(a) 6 motes, unaligned.

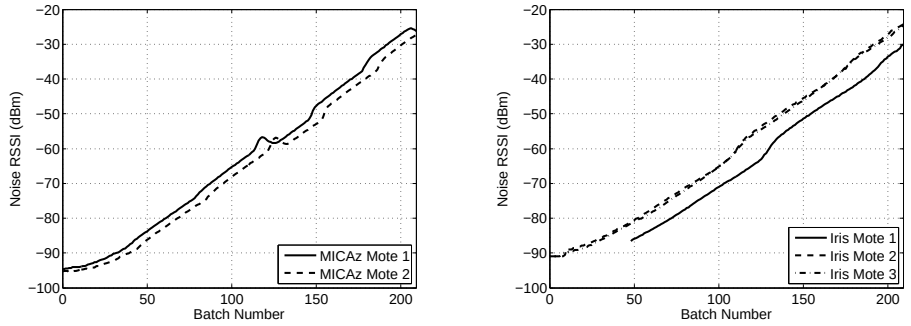
(b) 12 motes, aligned.

Figure 3: (a) RSSI measurements reported by six Tmote Sky motes as the noise strength increases linearly in dBm. While the response is mostly linear, it includes multiple nonlinear regions. Similar results from six additional motes are omitted in the interest of clarity. (b) Aligned RSSI response curves for all twelve Tmote Sky motes. Device-specific variations are minimal. Boxes A and B indicate the non-injective regions.

signals. In fact, this assumption is the basis of range-free localization mechanisms [8]. Also, due to the existence of these non-monotonic regions, the mapping from actual signal strength to the RSSI readings that the radios report is non-injective. Considering that the nonlinearities exist for all the motes tested, we categorize them as systematic errors in the RSSI measurements by the CC2420 radio.

Another important observation from Figure 3(a) is that the mote-specific RSSI curves are considerably similar. In fact, the major difference among the curves is the offset on the X-axis. This is mainly due to the different signal strength attenuations resulting from the varying distances between individual motes and the USRP. Given this similarity, we select one RSSI curve as the reference and align the other curves to it. Figure 3(b) shows the result of this process. It is clear that overall the RSSI curves for different motes match very well. The mismatches at the lower end of the graph are likely due to the fact that RSSI readings in this region are approaching the ambient noise level.

We note that the results in Figure 3(b) were achieved by shifting the RSSI curves only along the X-axis. This is desirable because it suggests



(a) MICAz motes. (b) IRIS motes.

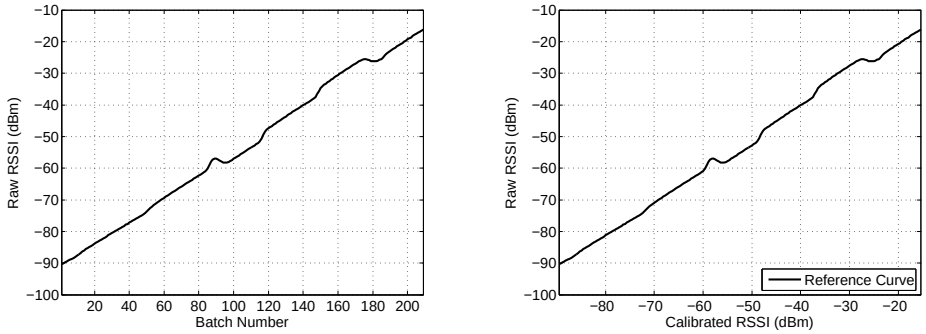
Figure 4: (a) RSSI response curves from two MICAz motes using the same CC2420 radio as the Tmote Sky. Response curves are consistent across platforms that use the same radio. (b) RSSI response curves for three IRIS motes using the AT86RF230 radio. While the radio responds differently from the CC2420 radio, it also has nonlinear regions.

that even though nonlinear and non-injective regions exist, they occur at the same reported RSSI values for different motes. In other words, the device-specific variations regarding the nonlinearity and non-injectiveness are minimal. Consequently, mitigating these errors does not require calibrating each device individually.

3.3 Platform and Radio Variability

In order to investigate the influence of the hardware platform on RSSI measurements, we performed the same experiment using two MICAz motes. MICAz motes use the same CC2420 radio but are otherwise different from the Tmote Sky motes used thus far. Figure 4(a) presents the RSSI response curves for two MICAz motes. It is clear that the curves in Figure 4(a) are very similar to the ones in Figure 3. This similarity indicates that the RSSI measurement errors are caused by the CC2420 radio chip itself and are platform independent.

Finally, to investigate whether the observed nonlinearities are specific to the CC2420 radio, we performed the same experiment using three Crossbow Iris motes which use the AT86RF230 radio. Figure 4(b) presents the results from this experiment. While different from those in Figures 3 and 4(a), the



(a) (b)

Figure 5: (a) Combination of the 12 curves in Figure 3(b). (b) The reference RSSI curve for CC2420 radios, derived by linearly transforming the X-axis from (a).

RSSI response graphs of the IRIS motes exhibit consistent nonlinearities. On the other hand, the RSSI response graphs do not exhibit non-injective regions. Finally, we observe consistent non-linearities across all three motes, indicating that the systematic errors in AT86RF230 raw RSSI measurements are also device independent.

4 RSSI Calibration

The results from the previous section show that radios have a non-linear, yet consistent response curve that maps the actual received signal strength to reported RSSI measurements. Figure 5(a), derived from combining the individual curves in Figure 3(b), shows such a response curve for the CC2420 radio. The issue with this curve is that the X-axis is in units of batch number instead of actual RSSI values. The noise strength increases linearly with respect to the batch number and therefore the relationship between batch number (n) and actual RSSI (r) should be $r = \alpha \times n + \beta$. The noise strength increment α can be measured experimentally, using the Anritsu MS2721B. On the other hand, measuring β accurately would require measuring the power of the signal that comes out of the receiving mote's antenna with a pre-calibrated receiver. Fortunately, as we explain next, we do not need to estimate β accurately.

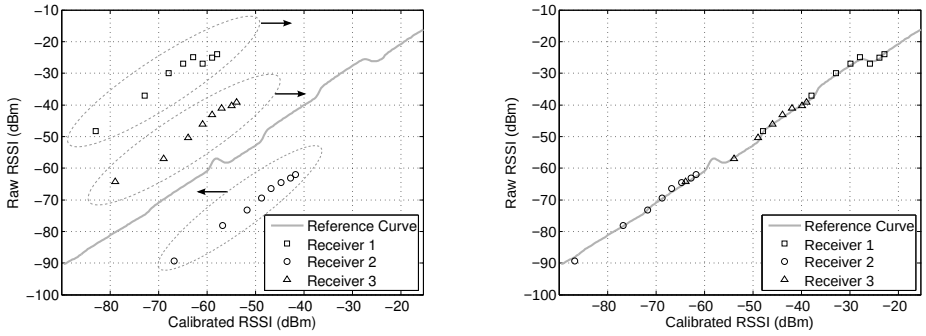


Figure 6: (a) Aligning a set of eight pairs $(P_i - L, R_i)$ to the reference curve for three motes, with $x = P_i - L$ and $y = R_i$. The relative positions among the eight points for each mote are fixed. (b) All mote measurements align well to the reference curve.

Any errors in estimating β will lead to a constant offset between the actual and *calibrated* RSSI. However, this offset is not important because it does not affect the SNR and SINR calculations. Furthermore, because the offset is consistent across different devices that are using the same model of radio, directly comparing calibrated RSSI values is equivalent to comparing actual RSSI values. Considering these arguments, we settle for estimating calibrated RSSI $r' = \alpha \times n + \beta' = r + \epsilon$ and we select β' such that batch number $n = 140$ corresponds to calibrated RSSI $r' = -40$ dBm. We selected the (140,-40) pair because it makes the reported RSSI values almost equal to the calibrated RSSI values in most of the linear regions of the curve. Figure 5(b) presents the result of this translation.

Figure 5(b) can then be used to translate raw RSSI readings to calibrated RSSI values. This figure however cannot resolve the ambiguities in the non-injective regions, in which a raw RSSI value maps to multiple calibrated RSSI values. Fortunately, we can leverage the ability to control the transmitter's power to resolve these ambiguities as we describe next.

Consider the case in which the raw RSSI value R_1 for a received packet lies within one of the non-injective regions of Figure 5(b). The receiver then requests the transmitter to reveal the power level P_1 used to transmit that packet and to transmit additional packets using different power levels

$P_2, \ldots, P_m$[2]. The receiver records the raw RSSI values $R_2, \ldots, R_m$ for each of the additional packets. If at least one of the R_i's falls within the radio's injective response region, it is possible to translate it to the calibrated RSSI value R'_i via Figure 5(b). Note that $R'_i = P_i - L$, where L is the link attenuation in dB. Knowing the values for both P_i and R'_i we can solve for L. Then $P_1 - L$ can be assigned to be the calibrated RSSI value corresponding to R_1, because L is consistent across different transmit powers. The computational cost is trivial, because only one raw RSSI value (R_i) needs to be translated into the calibrated RSSI (R'_i), via a lookup table corresponding to the reference curve.

To be more robust against measurement errors and noise, we can also select the value of L that minimizes the mean square difference between the m points $(P_1 - L, R_1), \ldots, (P_m - L, R_m)$ and the reference curve. The computational cost would be increased in this case because multiple table lookups are necessary.

Figures 6(a) and 6(b) present an example of the m-point calibration process for three receivers, with $m = 8$, equal to the number of power levels available in CC2420. One can see that the eight points for each mote fit well to the reference curve. Note that generally m can be arbitrarily chosen between 2 and the number of available power levels.

5 Applications

In what follows we explore the impact of RSSI calibration in modeling, protocol behavior, application performance, and simulation veracity.

5.1 PRR-SNR Model

First, we investigate the benefits of applying the RSSI calibration mechanism described in Section 4 to the problem of understanding the relationship between PRR and SNR. In turn, this understanding can be used in a variety of applications ranging from online link estimation to link modeling and simulation.

We conducted this experiment in the same indoor testbed used for the packet size experiment. One Tmote Sky mote was chosen as the transmitter

[2]The number m is upper-bounded by the number of available transmit power levels from the radio, and the actual P_i values are listed on the radio's datasheet.

while the other twelve motes acted as receivers. However, unlike the packet size experiment, all packets had the same size. Moreover, the transmitter varied the output power levels to produce a larger range of SNR values.

The signal to noise ratio (SNR) is computed as $SNR = \frac{S}{N}$, where S is the power of the received packet and N is the power of the ambient noise. Let both S and N be measured in milliwatts (mW). In logarithmic scale the above equation becomes $SNR_{dB} = S_{dBm} - N_{dBm}$ where S_{dBm} and N_{dBm} are the logarithmic scale powers of the received signal and ambient noise respectively.

In order to measure S and N, the receivers record both packet RSSI (S_{RSSI}) and noise RSSI (N_{RSSI}). Then, $S_{RSSI} = 10\log_{10}(S + N)$ and $N_{RSSI} = 10\log_{10} N$. Therefore, S_{RSSI} is essentially the sum of the power of the radio signal and the power of the noise. Nevertheless, when $S_{RSSI} \gg N_{RSSI}$ one can approximate SNR as $SNR_{dB} \approx S_{RSSI} - N_{RSSI}$. On the other hand, when S_{RSSI} is comparable to N_{RSSI} we need to compute SNR through

$$SNR_{dB} = 10\log_{10}(10^{S_{RSSI}/10} - 10^{N_{RSSI}/10}) - N_{RSSI} \quad (1)$$

because $S = 10^{S_{RSSI}/10} - 10^{N_{RSSI}/10}$.

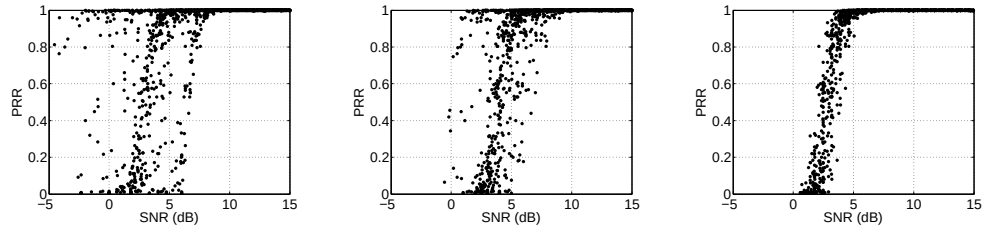
We use Equation 1 to calculate the SNR values used in the PRR vs. SNR scatter plot shown in Figure 7(a). One can see from this figure that there is a large transitional region, through which the relationship between PRR and SNR is noisy and unpredictable. The existence of this transitional region has been widely reported in the wireless sensor networks literature [21, 23, 25].

At the same time, given the nonlinearity presented in Figure 3(b), using raw RSSI values to calculate SNR can be problematic. For instance, if S and N are both within the non-injective regions, the reported RSSI value for their sum might be smaller than the reported RSSI value for S or N alone.

To eliminate this issue, we configured the transmitter to broadcast one additional batch of packets at each of the eight output power levels, while keeping the USRP turned off. This allows us to use the packet RSSI measurements directly, without having to calculate S from S_{RSSI} and N_{RSSI}. In this case, we denote the reported packet RSSI value as $\hat{S}_{RSSI}$, and calculate SNR as

$$SNR_{dB} = \hat{S}_{RSSI} - N_{RSSI} \quad (2)$$

Doing so assumes that the channel conditions do not change dramatically throughout the course of the experiment. This is however reasonable, as the



(a) SNR calculated using Equation 1.

(b) SNR calculated using Equation 2.

(c) SNR calculated using the RSSI calibration scheme.

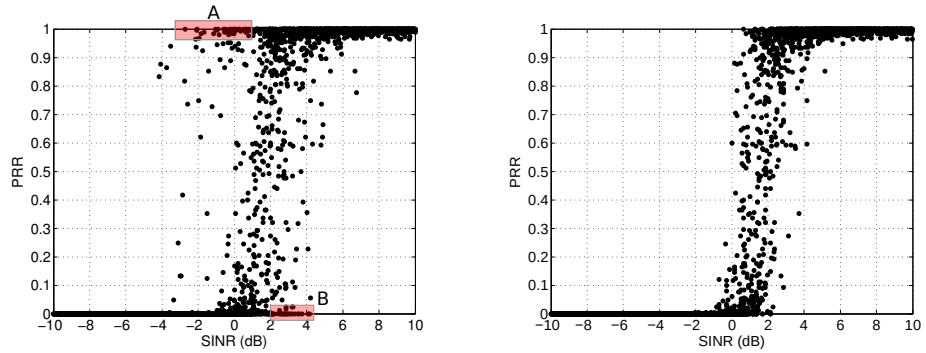
Figure 7: Experimentally derived PRR vs. SNR curves, calculated using three increasingly accurate schemes. Using calibrated instead of raw RSSI measurements can remove most of the outliers in the PRR vs. SNR relationship.

measurements were collected at night when the environment at our indoor testbed was static. We use Equation 2 to calculate the SNR values used in Figure 7(b). One can see that the extent of the transitional region is considerably smaller compared to Figure 7(a). This observation validates our intuition that Equation 1 is polluted by the nonlinearities in the measurement of S_{RSSI}. At the same time, the SNR in Equation 2 is computed using the raw values for $\hat{S}_{RSSI}$ and N_{RSSI} and therefore it is also susceptible to the nonlinearities' adverse effects.

Finally, Figure 7(c) shows the equivalent scatter plot when the SNR in Equation 2 was calculated using the calibrated RSSI values for $\hat{S}_{RSSI}$ and N_{RSSI}. It is evident from Figure 7(c) that the transitional region becomes significantly smaller compared to the previous two graphs. This result indicates that the RSSI nonlinearity can account for a large portion of the noise and outliers in the PRR vs. SNR model.

5.2 SINR Modeling and Concurrent Transmission

The previous section investigated the relationship between PRR and SNR. When multiple transmitters are active at the same time, they start to interfere with each other and PRR is determined by another metric, the SINR (Signal to Interference and Noise Ratio). Maheshwari et al. conducted an extensive study on the relationship between SINR and PRR for the CC2420 radio [13]. However, the SINR-PRR graphs in [13] have a remarkable volume



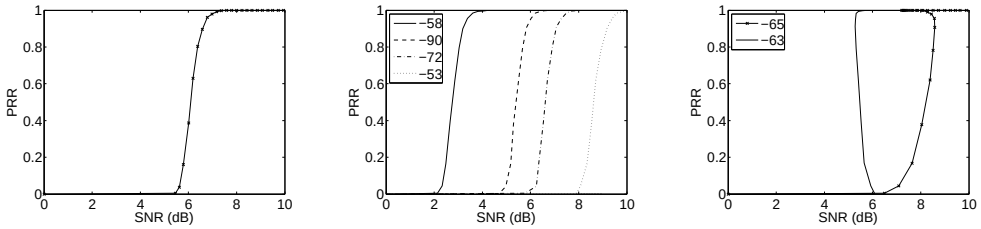
(a) Raw SINR. (b) Calibrated SINR.

Figure 8: PRR vs. SINR results for 2 concurrent transmitters and 14 receivers. The calibrated SINR shown in (b) eliminates most of the outliers present at the raw SINR graph shown in (a). Links in boxes A and B are the extreme outliers that complicate the SINR-PRR modeling.

of outliers for which high SINR links have low PRR, while links with negative SINR exhibit high PRR. Maheshwari et al. thus concluded that the SINR-PRR model is still far from perfect to be employed in TDMA scheduling [13].

We conjecture that the CC2420 RSSI nonlinearity accounts for some of the outliers seen in [13]. In order to validate this conjecture, we performed an experiment with two Tmote Sky motes configured to broadcast simultaneously to 14 Tmote Sky motes. Figure 8(a) and 8(b) present the derived uncalibrated and calibrated SINR-PRR scatter plots. One can clearly see that in our experiment, most of the outliers were indeed introduced by the CC2420 RSSI nonlinearity. Approximately 25% of the links in this experiment experience more than 2 dB change in their SINR values when applying the calibration scheme. For the data points located within the $[-4, 5]$ dB region in Figure 8(a), 8% are outliers. After calibration, 94% of these outliers are corrected in Figure 8(b).

Accurate SINR models are important to protocols, such as CMAC [17], that attempt to schedule multiple, non-interfering transmissions. Specifically, CMAC utilizes an SINR-PRR model to set the nodes' transmission powers such that multiple interfering links can be used concurrently. Doing so can significantly increase system throughput. Nevertheless, the outliers exposed in Figure 8(a) can lead CMAC to suboptimal transmission schedules.



(a) (b) (c)

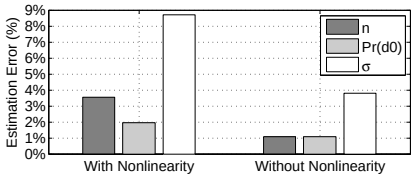
Figure 9: PRR vs. SNR curves generated by a TOSSIM simulation. (a) shows the curve from the original TOSSIM. Curves in (b) and (c) are derived from a modified version of TOSSIM that simulates the CC2420 RSSI nonlinearity. Each curve in (b) and (c) was derived by keeping the noise power constant and varying signal strength to create a dynamic SNR range. The noise power listed in the legends is in dBm units.

For example, we observed in the experiment that one of the two senders (mote 0) could deliver $> 98\%$ of its packets to receiver 11 when transmitting at -7 dBm, while the other sender (mote 1) could deliver at the same time $> 98\%$ of its packets to receiver 14 using transmit power of -15 dBm. However, the raw SINR value calculated at mote 14 is -0.128 dB which translates to a very low PRR according to the SINR-PRR model. For this reason, a power scheduling protocol based on the SINR model, such as CMAC [17], would not schedule mote 1 to transmit at power -15 dBm. On the other hand, if CMAC used the calibrated SINR value at mote 14 (= 2.2056 dB) it would correctly schedule the concurrent transmission. We note that link $1 \rightarrow 14$ is one of the links in box A shown in Figure 8(a).

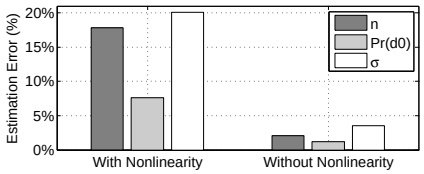
5.3 WSN Simulation

Existing wireless sensor network simulators such as TOSSIM [10] do not simulate the radio-specific RSSI measurement nonlinearities. Nevertheless, it is straightforward to integrate RSSI response curves, such as the one in Figure 5(b), to these simulators. Doing so requires constructing a lookup table and using linear interpolation to convert actual RSSI values (i.e., X-axis in Figure 5(b)) into reported RSSI values (i.e., Y-axis in Figure 5(b)).

We implemented such a mechanism for TOSSIM and Figure 9 presents a few sample PRR-SNR curves. Specifically, Figure 9(a) shows the PRR



(a) $n = 3.5$, $P_r(d_0) = -30$, $\sigma = 3$



(b) $n = 2$, $P_r(d_0) = -30$, $\sigma = 3$

Figure 10: Errors in estimating log-normal path loss model parameters.

versus reported SNR curve in the current version of TOSSIM. Without the integration of the RSSI response curve, the shape of this PRR-SNR curve does not change as RSSI varies. In contrast, Figures 9(b) and 9(c) show that different curves emerge as we vary the power of the ambient noise, due to the nonlinearity in the reported RSSI values. In particular, the curves in Figure 9(c) resemble the experimentally derived curves in Figures 1(b) and 1(c).

5.4 Estimating Radio Propagation Model Parameters

A variety of WSN applications and protocols rely on radio propagation models. The first step in using such a model is to estimate the corresponding model parameters. This step is usually accomplished by deploying motes to record the radio signal strength (i.e., RSSI), at various locations within the area of interest. Therefore, the non-linearities of RSSI measurements can directly pollute the estimation of the model's parameters and thus the performance of the protocols that rely on the model's accuracy.

A commonly used radio propagation model is the log-distance path loss model with log-normal shadowing [16]. According to this model, the received signal strength $P_r(d)$ (in dBm) at a given distance d from the transmitter is given by:

$$P_r(d)[dBm] = P_r(d_0)[dBm] - 10n \log(\frac{d}{d_0}) - X_\sigma \tag{3}$$

where $P_r(d_0)$ is the expected signal strength at reference distance d_0, n is the path-loss exponent, and $X_\sigma \sim N(0, \sigma)$ is a normal random variable (in dB).

In order to investigate the impact of CC2420 RSSI nonlinearity on parameter estimation, we simulate the procedure of deploying motes at various

distances from a transmitter to derive the log-normal parameters $P_r(d_0)$, n and σ. Specifically, we generate the $P_r(d)$ samples using a set of log-normal parameters and use the RSSI measurements to estimate those parameters. We note that doing so assumes that the log-normal model perfectly characterizes the RF propagation, a premise which might be violated in reality. Nevertheless, this treatment isolates the sources of errors in model parameter estimation and therefore allow us to focus on the errors that the RSSI nonlinearity introduces. A total of 240 samples were generated, corresponding to measurements collected at locations uniformly spaced at distances between 1 and 30 meters from the transmitter. Two samples were generated for each distance. Figure 10 presents the estimation errors with and without the presence of the CC2420 RSSI nonlinearity for two sets of model parameters. It is clear that the nonlinearity can cause significant errors. Errors in estimating these parameters can directly impact the applications that rely on them, such as RF based localization [24] and network coverage prediction [4].

5.5 RF Based Localization

Localization techniques based on RF signal strength use RSSI measurements to estimate the distances of a mobile device to several reference servers whose locations are known. *Trilateration* can then be used to estimate the device's location [24]. The previous section demonstrated that the nonlinearities in the CC2420 RSSI measurements impact the estimation of the radio model parameters. In turn, these errors can directly diminish the accuracy of such localization algorithms. On the other hand, localization schemes that employ RSSI signatures should intuitively be less affected by such nonlinearities. For example, the RADAR system collects a database of RSSI signatures by having a mobile node broadcast packets to three reference servers from a set of known locations [3]. The resulting RSSI measurements collected at the three servers, along with the mobile device's location, form the 5-tuples $[RSSI_1, RSSI_2, RSSI_3, X, Y]$ that constitute the localization database.

Once this training phase is complete, a device that needs to estimate its location broadcasts a series of packets to the reference servers. The system then finds the entry in the localization database with the minimum mean square difference from the RSSI measurements and uses the entry's $[X, Y]$ coordinates as the estimate of the node's current location. The MoteTrack system extends this simple approach and makes it highly robust and decentralized [12].

Transmission Power (dBm)	Average Localization Error (cm)	Percentage
-15	138.97	7%
-10	133.19	2%
-7	148.80	14%
-5	146.79	13%
-3	137.39	5%
-1	134.10	3%
0	140.26	8%
Calibrated	130.35	-

Table 1: Localization errors for the RSSI-signature-based localization technique as a function of transmission power. The rightmost column represents the localization error as a percentage on top of the error achieved using the calibrated RSSI measurements.

We performed an experiment similar to the one performed for RADAR in a 20 m^2 room using four Tmote Sky motes. Three of the motes were setup as reference servers while the fourth played the role of the mobile device. A total of 70 locations were tested and the mobile device was configured to broadcast at the seven transmission powers at 25 ms intervals.[3] Thus seven databases were constructed corresponding to the seven power levels. Each database was then used to evaluate localization errors for the corresponding transmission power. The method we used to estimate localization accuracy is the same with the one used by the original RADAR mechanism [3]. Namely, we select one of the database entries and try to localize it using only the other database entries. The localization error is then equal to the Euclidean distance between the entry's actual location and the location of the closest database entry. We iterate through all the database entries in this way and calculate the average localization error.

Table 1 lists the resulting localization errors. It is evident from the table that different transmission powers lead to different errors. This should not happen if the RSSI readings were linear, because a linear constant does not change the mean square difference, the metric used to select the most similar record from the database[4]. The table's last row indicates that calibrating the raw RSSI measurements reduces the localization error.

[3]The lowest transmission power (-25dBm) was not sufficient to ensure packet reception at the reference servers from all the tested locations.

[4]Assuming that the signal strength is significantly higher than the ambient noise, which was true during the course of this experiment.

6 Conclusion

This paper verifies the existence of the oft-ignored RSSI non-linearities for the popular Chipcon/TI CC2420 802.15.4 radio and shows that similar non-linearities exist in the Atmel AT86RF230 radio. Furthermore, the paper experimentally derives the non-linear RSSI response curves for the two radios, shows that they are consistent across devices that use the same model of radio, and proposes a scheme to calibrate raw RSSI measurements including those that fall within a curve's non-injective regions. Last but not least, we evaluate the impact of non-linearities in RSSI measurements on PRR modeling, WSN simulation, as well as protocols for concurrent link scheduling and RF-based localization.

The implications of our results to future designs are twofold. First, protocol and application designers need to be mindful that RSSI response curves may be non-linear or even non-injective and include techniques to compensate for such non-linearities. Second, considering the dependence of multiple protocols on RSSI measurements, future radio designs should strive to produce linear or at least injective RSSI response curves.

The calibration data used in this paper is available online at `http://www.cs.jhu.edu/~yinchen/calibrate`.

Acknowledgments

We extend our gratitude to Neal Patwari, Phil Levis, and Prabal Dutta for their insightful comments and suggestions. We would also like to thank the anonymous reviewers that for helping us improve the paper's presentation. This research was supported in part by NSF grants CNS-0834470 and CNS-0546648. Any opinions, finding, conclusions or recommendations expressed in this publication are those of the authors and do not represent the policy or position of the NSF.

References

[1] Anritsu Company. Spectrum Master MS2721B.

[2] Atmel Corporation. AT86RF230: Low Power 2.4 GHz Transceiver for ZigBee, IEEE 802.15.4, 6LoWPAN, RF4CE and ISM applications.

[3] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF-based User Location and Tracking System. In *Proceedings of INFOCOM*, 2000.

[4] O. Chipara, G. Hackmann, C. Lu, W. D. Smart, and G.-C. Roman. Radio mapping for indoor environments. Technical report, Washington University in St. Louis, 2007.

[5] Crossbow Corporation. MICAz Specifications, 2004.

[6] Crossbow Corporation. Iris Specifications, 2007.

[7] Ettus Research LLC. Universal Software Radio Peripheral, 2007.

[8] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. In *MobiCom '03*, pages 81–95, New York, NY, USA, 2003. ACM.

[9] IEEE Standard 802.15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), May 2003.

[10] P. Levis, N. Lee, A. Woo, M. Welsh, and D. Culler. TOSSIM: Accurate and scalable simulation of entire TinyOS Applications. In *Proceedings of Sensys 2003*, Nov. 2003.

[11] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He. ATPC Adaptive Transmission Power Control for Wireless Sensor Networks. In *Proceedings of the 4^{th} ACM Sensys Conference*, 2006.

[12] K. Lorincz and M. Welsh. Motetrack: a robust, decentralized approach to rf-based location tracking. *Personal Ubiquitous Comput.*, 11(6):489–503, 2007.

[13] R. Maheshwari, S. Jain, and S. R. Das. A measurement study of interference modeling and scheduling in low-power wireless networks. In *Proceedings of Sensys 2008*, pages 141–154, New York, NY, USA, 2008. ACM.

[14] R. Musaloiu-E. and A. Terzis. Minimising the effect of wifi interference in 802.15.4 wireless sensor networks. *Int. J. Sen. Netw.*, 3(1):43–54, 2007.

[15] J. Polastre, R. Szewczyk, and D. Culler. Telos: Enabling Ultra-Low Power Wireless Research. In *IPSN/SPOTS 05*, Apr. 2005.

[16] T. S. Rappaport. *Wireless Communications: Principles & Practices*. Prentice Hall, 1996.

[17] M. Sha, G. Xing, G. Zhou, S. Liu, and X. Wang. C-MAC: Model-driven Concurrent Medium Access Control for Wireless Sensor Networks. In *Proceedings of IEEE Infocom*, 2009.

[18] D. Son, B. Krishnamachari, and J. Heidemann. Experimental study of concurrent transmission in wireless sensor networks. In *Proceedings of ACM Sensys*, 2006.

[19] K. Srinivasan and P. Levis. RSSI is Under Appreciated. In *Proceedings of the 3rd Workshop on Embedded Networked Sensors (EmNets)*, May 2006.

[20] Texas Instruments. CC2420: 2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver, 2006.

[21] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges in reliable multihop wireless sensor networks. In *Proceedings of ACM Sensys*, 2003.

[22] K. Yedavalli, B. Krishnamachari, S. Ravula, and B. Srinivasan. Ecolocation: a sequence based technique for rf localization in wireless sensor networks. In *Proceedings of IPSN 2005*, page 38, Piscataway, NJ, USA, 2005. IEEE Press.

[23] M. Z. Zamalloa and B. Krishnamachari. An analysis of unreliability and asymmetry in low-power wireless links. *ACM Transactions on Sensor Networks*, June 2007.

[24] G. Zanca, F. Zorzi, A. Zanella, and M. Zorzi. Experimental comparison of rssi-based localization algorithms for indoor wireless sensor networks. In *REALWSN '08*, pages 1–5, New York, NY, USA, 2008. ACM.

[25] J. Zhao and R. Govindan. Understanding Packet Delivery Performance In Dense Wireless Sensor Networks. In *Proceedings of the ACM Sensys*, Nov. 2003.