RSSI is Under Appreciated

Kannan Srinivasan[†] and Philip Levis[*†]

[†]Department of Electrical Engineering and [*]Department of Computer Science, Stanford University, Stanford, CA

Abstract

There is a general belief in the Wireless Sensor Network (WSN) community that the received signal strength indicator (RSSI) is a bad estimator of the link quality. This belief is due to the existence of many asymmetry links in older radios such as CC1000 and TR1000. Newer radios that are based on IEEE 802.15.4 standard such as CC2420 implement another parameter called link quality indicator (LQI) which is believed to be a better indicator than RSSI. There is so far no extensive evaluation of CC2420 to validate this claim. We have conducted such an evaluation and our preliminary results indicate that RSSI for a given link has very small variation over time for a link. Our results also indicate that when the RSSI is above the sensitivity threshold (about -87 dBm), the packet reception rate (PRR) is atleast 85%. Around this sensitivity threshold, however, the PRR is not correlated possibly due to variations in local phenomena such as noise. LQI, on the other hand, varies over a wider range over time for a given link. However, the mean LQI computed over many packets has a better correlation with PRR.

1. Introduction

Link estimation is a critical part of almost every sensor network routing protocol. Knowing the packet reception rate of candidate neighbors lets a protocol choose the most energy efficient next routing hop.

The earliest sensornet link estimators assumed link symmetry, establishing routes either through flooding [5] or snooping [7]. Later it was shown that this assumption was invalid and it led to terrible performance. Second generation estimators [13] used packet sequence numbers to count lost packets, but required a low rate of control traffic to ensure that nodes could detect when links died. While this approach can detect good links, it adapts slowly to changes in link quality. This limitation led to a number of approaches that use information from the radio hardware. One basic indicator is RSSI, which is the strength of a received RF signal. Conventional wisdom in the sensornet community says that RSSI is not a good indicator. This belief is based on experimental work with early platforms [14], which showed that while detecting good links was possible with RSSI, with imperfect links it is difficult to make good estimates.

Many current platforms — e.g., the micaZ, Telos, and Intel Mote2 — use the same radio chip, the CC2420 [3]. In addition to RSSI, CC2420 provides an additional hardware indicator, LQI, which is effectively a measure of chip error rate. The observed limitations of RSSI in earlier platforms, combined with radio homogeneity, has led several routing layers [12,11] to adopt LQI, based on the last packet, as the indicator of choice. Despite its wide adoption, as far as we are aware there has so far been no evaluation of whether LQI is a good indicator. In this paper, we present an initial investigation into this question. Our preliminary results indicate that RSSI varies very little over time compared to LQI. Our results also indicate that RSSI is highly correlated with PRR except when operating at the edge of receiver sensitivity. While LQI can make more precise estimates, doing so requires averaging over many readings, which decreases agility and increases estimation cost. These results suggest that there is a disconnection between the behavior of hardware indicators and current protocols that use them.

The rest of the paper is organized as follows. We briefly talk about prior work in Section 2. We then discuss our experimental methodology in Section 3 and then present our experimental results in Section 4 with conclusions in Section 5.

2. Prior Work

Many researchers have done thorough evaluation of previous generation motes with radio chips TR1000 [9] and CC1000 [3]. We briefly present some of these works and then introduce our own.

Ganesan et al [4] analyzed performance of Rene motes (based on TR1000) and showed that even a simple algorithm such as flooding had significant complex-

ity at scale. They attributed this complexity partly to the link asymmetries and suggested that these asymmetries were due to sensitivity mismatch at different nodes. Woo et al [13] also confirmed a "transitional" region where the PRR had no correlation with distance. Cerpa et al [2] also showed that there was no correlation between PRR and distance beyond 50% of a node's communication range. They also found many link asymmetries (up to 30%). They showed that these asymmetries were not due to the environment and so they might be due to hardware miscaibration.

Zhao et al [14] measured packet delivery of Mica motes (based on TR1000). From their measurements without any encoding they showed that links with PRR of at least 95% had high RSSI while the converse was not true.

Son et al [10] analyzed concurrent packet transmissions for mica2 motes. They showed that the signal to interference and noise ratio (SINR) threshold was different for different nodes and that it also depends on the RSSI. They suggested that these were due to hardware miscalibration. Aguayo et al [1] observed similar packet delivery behaviors in 802.11 networks, but concluded that they were most likely due to multipath effects as there was little correlation between PRR and SINR. However, their experimental methodology differs from those of the sensor network studies. For example, they consider average SINR ratios over second-long periods rather than on a per-packet basis.

Polastre et al [8] presented preliminary evaluation results for Telos motes (based on CC2420) and suggested that the average LQI was a better indicator of PRR and that RSSI was a bad indicator. We believe that the correlation between RSSI, LQI and PRR may be more easily understood through plots of PRR against LQI and RSSI.

In all the previous work presented above for sensor networking, the authors have suggested that the hardware miscalibration with the older radios was possibly the cause of link asymmetries and transitional regions. In our work presented here, we have evaluated the newer set of motes called micaZ motes (based on CC2420) after carefully choosing evaluation and plotting strategies, thanks to the knowledge gained from the previous work listed above.

3. Experimental Methods

In our experiments we used micaz motes that are based on the CC2420 radio chip, which apart from being more advanced than older radios is based on IEEE 802.15.4, an emerging WSN standard.

Radio	Platforms	Data rate	Keying	Encoding
TR1000	mica	13.3kbps	ASK	SEC_DED
CC1000	mica2	19.2kbps	BFSK	Manchester
CC2420	micaZ, Telos, Intel Mote2	256kbps	OQPSK	Spread Spectrum

Figure 1. Comparison of the RFM TR1000, CC1000, and CC2420 radios.

3.1. CC2420

CC2420 operates in 2.4 GHz ISM band with an effective data rate of 256 kbps, a much higher rate than older radios. In the 2.4 GHz band, it has 16 channels (numbered 11 through 26) with each channel occupying a 3 MHz bandwidth with a center frequency separation of 5 MHz for adjacent channels. CC2420 uses an encoding scheme that encodes 32 chips for a symbol of 4 bits. This encoded data is then OQPSK (offset quadrature phase shift keying) modulated. Figure 1 compares the CC2420 with two earlier mote radios, the RFM TR1000 and the ChipCon CC1000.

CC2420 provides two useful measurements: RSSI and LQI. RSSI is the estimate of the signal power and is calculated over 8 symbol periods and stored in the RSSI_VAL register. Chipcon specifies the following formula to compute the received signal power (P) in dBm: P = RSSI_VAL + RSSI_OFFSET, where RSSI_OFFSET is about -45. We refer to this power, P (in dBm) as RSSI throughout this article.

LQI can be viewed as chip error rate and is calculated over 8 bits following the start frame delimiter (SFD). LQI values are usually between 110 and 50, and correspond to maximum and minimum quality frames respectively.

3.2. Experimental Setup

We carried out our evaluations on a testbed (mirage) provided by Intel Research at Berkeley [6]. We picked 30 micaZ motes placed at different locations from a total of 100. Each micaz mote was connected to the mirage server over the Ethernet.

We programmed the motes to accept commands from the server to send 200 packets to a chosen receiver node at the rate of 1 packet every 10 milliseconds in channel 11 and at a particular transmission power level. To avoid collisions, at any time only one mote transmitted. Every packet sent was a unicast instead of a broadcast because unicast is the general form of traffic that is observed in a real WSN and so we wanted to see how the nodes perform for this traffic pattern. One drawback, however, with the unicast traffic evalu-

ation was that we were not able to look at spatial correlation of PRR, RSSI and LQI among near-by nodes.

Every mote sent 200 packets to every other node, one node at a time. Upon sending 200 packets, the transmitter node informed the server. The server then sent a command to the corresponding receiver node to send statistics (sequence number, RSSI and LQI) it collected for the packets it received. Once the statistics were collected, the server sent command to another node to send 200 packets to a specific receiver node. Once all the nodes finished sending packets to all other nodes we repeated the same experiment at a different transmission power level. We carried out the same experiment at 5 different power levels: 0, -3, -7, -15 and -25 dBm. The experiments with different power levels were conducted one after the other. Two consecutive experiments were separated by the time it took to program all the motes (about 5 minutes). We present our evaluation results and observations in the following sections.

4. Results

Figure 2 has plots of PRR against RSSI on the top and plots of PRR against LQI at the bottom for different transmission power levels. For the PRR vs. RSSI plots, we took a link (m to n) and first plotted its PRR and average RSSI (computed over all packets received). We then plotted the distribution (mean - standard deviation and mean + standard deviation) of RSSI corresponding to that link. We repeated the same for every link (note that link n to m is different from link m to n). We did the same for the plots of PRR vs. LQI.

From the plots it is clear that the RSSI has very small variance compared to LQI for any link over time suggesting that the RSSI in a single packet is a good estimate of the average RSSI over many packets in that link. Moreover, it is also clear that generally for RSSI values greater than -87 dBm PRR is atleast 85% indicating a very good link. But for RSSI values less than -87 dBm we enter a grey region where the PRR varies rather radically. It is important to note that -87 dBm is very close to the sensitivity threshold of CC2420 which is about -90 dBm. This is why above this value we are seeing PRR above 85%. However when the RSSI is below this sensitivity threshold, we believe that some local factors such as noise play the dominant role and so depending on this noise level the PRR may very well vary between 0 and 100%.
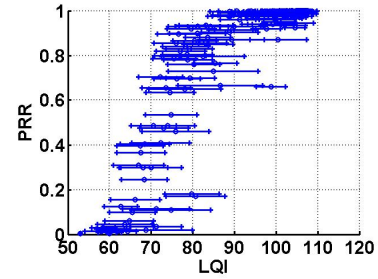
Note that the two outliers in the first plot that have RSSI value above -87 dBm but have PRR lower than 85% and have higher variance than all others. This is possibly due to changes in multipath pattern that
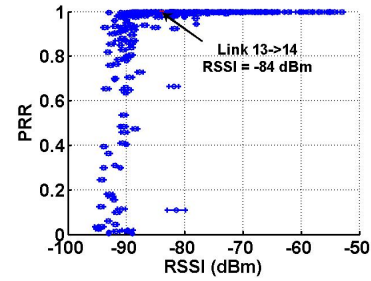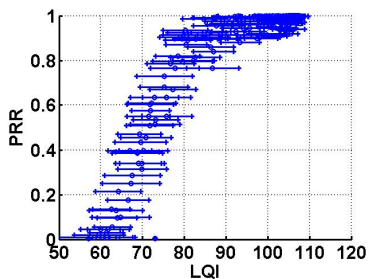

(a) PRR vs. RSSI at Power Level 0 dBm
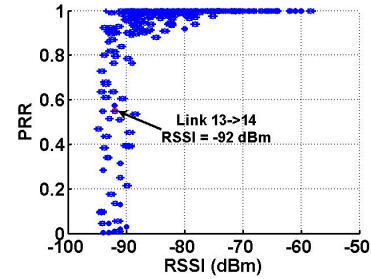
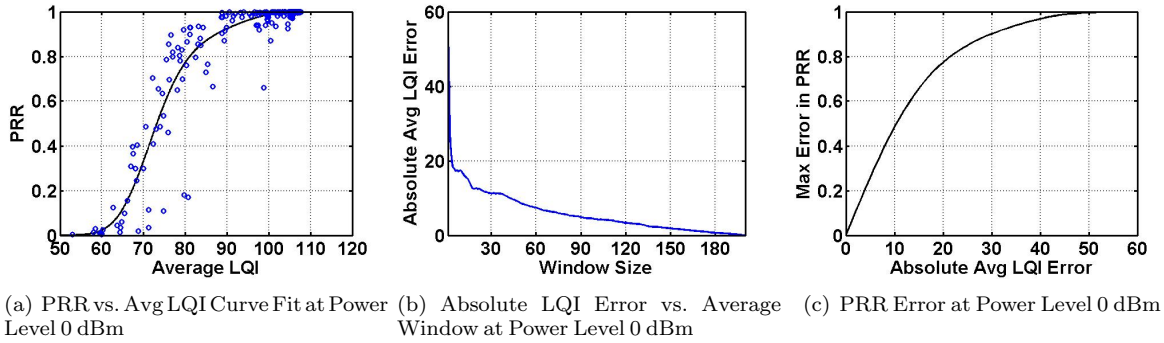(b) PRR vs. RSSI at Power Level -7 dBm

(c) PRR vs. LQI at Power Level 0 dBm

(d) PRR vs. LQI at Power Level -7 dBm

Figure 2. PRR of directional node pairs versus the RSSI and LQI. RSSI shows very low variation for a link over time compared to LQI. PRR is atleast 85% for links with RSSI above -87 dBm. "Grey" region about -87 dBm.

(a) PRR vs. Avg LQI Curve Fit at Power Level 0 dBm

(b) Absolute LQI Error vs. Average Window at Power Level 0 dBm

(c) PRR Error at Power Level 0 dBm

Figure 3. Average LQI and PRR have nice correlation. The absolute error in the estimate of average LQI through averaging LQI over a window of packets is large even for large window sizes (window size is 60 for an error of 10). An error in the average LQI estimate of 10 can result in an error of upto 0.5 in the PRR estimate.

changed RSSI over time. This suggests that if there is a change in RSSI over time for a link then our estimation of PRR may not be accurate. To illustrate this point, we have shown the RSSI and PRR for a link (13 to 14) for two different transmission power levels in Figure 2. Note that when the link had an RSSI of -84 dBm it had a PRR close to 1 but when the same link (at a lower transmission power) had an RSSI of -92 dBm it enters the "grey" region and has a lower PRR (about 0.5). This suggests that a variation in RSSI can possibly change your PRR. This is what, we believe, has happened with the outliers.

It is also interesting to note that the width of the grey region is smaller than what Son et al [10] saw with the older mote (mica2). We do not completely understand why mica2 motes have wider transition region and we leave this as an open issue.

Now, if we look at the plots for LQI, it has a very high variance over time for a given link. At high transmit power level (0 dBm), an LQI value of 85 could actually mean anything between 10% and 100% PRR. At lower transmit power (-7 dBm), although only fewer links are involved the variance of LQI in any link is still high. However, if we look at the average LQI values marked by small circles in the middle of every horizontal line, it follows a rather smooth curve suggesting a better correlation with PRR. This suggests that may be averaging LQI values over a window of packets may better predict PRR than RSSI. To check this, we first fitted a curve to the average LQI vs PRR as shown in Figure 3(a). The curve fitting was done by first converting LQI into a chip error rate between 0 and 1 followed by calculating the corresponding bit error rate (8 chips/bit) and then the PRR (40 bytes/packet). Although the curve fits the data quite good there are still

a few outliers. Although we do not completely understand what might cause them, we believe that environmental changes and also interference from 802.11 networks might have been at work.

In an effort to calcuate the optimal window size (number of packets) over which to average the LQI values, we vary the window size and calculate the absoulte difference between the actual average LQI and the average computed over the window of packets. We plot the maximum absolute error in average LQI for all the window sizes in Figure 3(b). Clearly, LQI from a single packet (window size of 1) can be off by upto 50. To be as close as 10 from the actual average LQI, we need a window size of about 40. We then see how much error is possible in the estimate of PRR (calculated from the fitted curve) for various absoulte errors in average LQI. We plot the absolute error in average LQI vs maximum possible error in PRR estimate in Figure 3(c). An error in average LQI estimate of just 10 can have an error of about 0.5 in the PRR estimate. This means even when the window size is as large as 40 we can be off by 0.5 in estimating the PRR of a link. However, when the window size is about 120 we could see an error in the average LQI estimate of about 5, which can have an error of only 0.1 in the estimate of PRR. Such large window sizes makes such an estimator slow to adapt to environmental changes.

Figure 4 shows the plot of RSSI measured by both the nodes of a link for all the links. It is very symmetric. There are very small variations in the RSSI measured by the two nodes which are attributed to channel variations. This suggests that newer radios have insignificant or low hardware miscalibration issues.

Overall, we believe that the RSSI is a good candidate as an indicator of link quality if its value is above
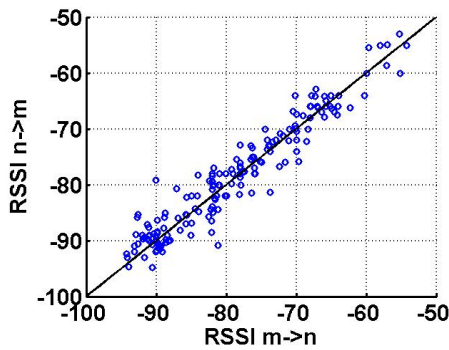
Figure 4. Plot of RSSI measured by Node n for packets from Node m vs. RSSI measured by Node m for packets from Node n.

the sensitivity threshold. It is also an attractive agile link estimator, due to its consistency over time. On the other hand, average LQI can provide a more accurate estimation of the link but will be slow to adapt to changes and also may be expensive. These observations and results are similar at other power levels that not shown in this article for brevity.

5. Conclusion

Link estimation is an important factor in protocol design. Hardware miscalibration in older radios forced RSSI to be excluded from link estimation. The absence of a comprehensive evaluation of newer radios motivated us to carry out one for CC2420 to see if hardware miscalibration is still an issue. Our results from this evaluation show RSSI as a promising indicator when its value is above the sensitivity threshold of CC2420 (-87 dBm). At the edge of this threshold, however, it does not have a good correlation with PRR. We believe that this is due to local noise variations at different nodes. We also found that the LQI when averaged over many packets (about 120) has better correlation with PRR.

Our observations are clearly different from the general belief that LQI (not average LQI) is a good indicator of link quality and that RSSI is a bad indicator. We believe that our observations are very important to the WSN community as it may help redefine some of the research directions. For instance, protocol designers looking for inexpensive and agile link estimators may choose RSSI over LQI. We also found that hardware miscalibartion is low or insignificant in CC2420 due to RSSI symmetry in links. Hence, it may be worthwhile revisiting those wireless models for wireless sensor network analysis.

References

[1] D. Aguayo, J. C. Bicket, S. Biswas, G. Judd, and R. Morris. Link-level measurements from an 802.11b mesh network. In *SIGCOMM*, pages 121–132, 2004.

[2] A. Cerpa, N. Busek, and D. Estrin. Scale: A tool for simple connectivity assessment in lossy environments. Technical Report 0021, Sept. 2003.

[3] ChipCon Inc. `http://www.chipcon.com`.

[4] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker. Complex behavior at scale: An experimental study of low-power wireless sensor networks, 2002.

[5] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. E. Culler, and K. S. J. Pister. System architecture directions for networked sensors. In *Architectural Support for Programming Languages and Operating Systems*, pages 93–104, 2000. TinyOS is available at http://webs.cs.berkeley.edu.

[6] Intel Research Berkeley. Mirage testbed. `https://mirage.berkeley.intel-research.net/`.

[7] P. Levis, S. Madden, D. Gay, J. Polastre, R. Szewczyk, A. Woo, E. Brewer, and D. Culler. The emergence of networking abstractions and techniques in tinyos. In *Proceedings of the First USENIX/ACM Symposium on Network Systems Design and Implementation (NSDI)*, 2004.

[8] J. Polastre, R. Szewczyk, and D. E. Culler. Telos: enabling ultra-low power wireless research. In *IPSN*, pages 364–369, 2005.

[9] RF Monolithics. `http://www.rfm.com`.

[10] D. Son, B. Krishnamachari, and J. Heidemann. Experimental analysis of concurrent packet transmissions in low-power wireless networks. Technical Report ISI-TR-2005-609, Nov. 2005.

[11] TinyOS. MultiHopLQI. `http://www.tinyos.net/tinyos-1.x/tos/lib/MultiHopLQI`, 2004.

[12] G. Tolle and D. Culler. Design of an application-cooperative management system for wireless sensor networks. In *Proceedings of Second European Workshop on Wireless Sensor Networks (EWSN 2005)*, 2005.

[13] A. Woo, T. Tong, and D. Culler. Taming the underlying challenges of reliable multihop routing in sensor networks. In *Proceedings of the First International Conference on Embedded Networked Sensor Systems (SenSys)*, 2003.

[14] J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *Proceedings of the First ACM Conference on Embedded Network Sensor Systems*, 2003.