

Towards Accurate and Agile Link Quality Estimation in Wireless Sensor Networks

Yian Qin, Zhitao He, Thiemo Voigt
Swedish Institute of Computer Science
Box 1263, SE-164 29
Kista, Sweden
Email: {yian, zhitao, thiemo}@sics.se

Abstract—Link quality estimation has been an important research topic in the wireless sensor networking community and researchers have developed a large number of different methods to estimate link quality. The commonly used CC2420 radio provides simple signal quality indicators. These are agile in that they react fast to changing link quality but they are inaccurate under complicated channel conditions. More sophisticated link quality estimators combine these simple metrics with packet reception statistics collected by the network stack. These approaches compensate the hardware-based metrics to a limited degree but they compromise agility and incur extra overhead. In this paper, we take a novel approach and develop a number of link quality metrics using a software defined radio. We evaluate our metrics under several channel conditions. The results show that they have good accuracy and can handle complicated channel conditions if combined properly.

I. INTRODUCTION

Link quality estimation is often an indispensable component of a wireless sensor network's communication protocol stack, because it provides crucial information to trigger other operations such as topology change, route update, and local retransmission. These operations are designed into the protocol in order to maintain good network performance in terms of latency, data loss and energy consumption. To support these operations, link quality estimation must be accurate in prediction of probable packet loss, agile to reflect changes in the environment, and simple to implement on low cost sensor node hardware.

Researchers have used two main information sources for estimation of a link's current quality: hardware-based signal quality indicators and recent history of packet losses. Hardware-based signal quality indicators, such as the Received Signal Strength Indicator (RSSI) and the Link Quality Indicator (LQI) available on CC2420 [1], benefit from their high agility to changes in channel conditions and low software processing overhead. They, however, have been shown to have inadequate accuracy under various channel conditions [2] [3] [4] [5] [6]. Relating these indicators directly to system-level performance metrics such as packet reception rate (PRR) yields unreliable and unstable predictions, because the channel condition is unknown. Some approaches thus try to correct the inherent limitations of the hardware-based indicators by giving some weight to a second source of information: PRR measurements in the past. This improves

stability but decreases agility at the same time, so protocols react more slowly to changes in link quality. Extra packets need to be exchanged over the link if faster reaction time is desired.

Future low power radios used by WSNs will likely incorporate more digital signal processing (DSP) capability as a result of lowered hardware costs which enables new hardware-based link quality estimation metrics. To explore opportunities in this direction, we use Software Defined Radio (SDR) as a convenient platform to develop new link quality estimation metrics which paves the way towards future hardware implementations. Our SDR is compatible to the IEEE 802.15.4 PHY standard [7] commonly used in WSNs. The high programmability and processing power of SDR allows us to implement our metrics without any platform dependence.

We implement three signal quality metrics, obtained on the radio's reception path. The three metrics are Chip Error Rate (CER), Error Vector Magnitude (EVM) and Spectrum Factor (SF). CER is derived from the IEEE 802.15.4 standard's spread spectrum modulation scheme. EVM is described by the standard as a test parameter for transmitter verification, usually using a signal analyzer, but has not been implemented as a user feature on the radio transceiver itself. SF is a new metric that we define, used to show signal distortion in the frequency domain. CER and EVM are updated upon each packet received; SF can be sampled at any moment. Therefore, all our metrics are very agile to changes in channel conditions.

We evaluate the three metrics under four characteristic channel conditions. Our results show that the metrics have an overall high accuracy in prediction of PRR on a stable channel, based on information collected from a single received packet. They can further be combined to handle harsh or changing channel conditions.

Our main contributions are the following:

- We develop three agile and accurate link quality estimation metrics.
- We evaluate the performance of the individual metrics.
- We propose a heuristic approach that combines the use of the metrics to estimate a link's quality.

In an earlier position paper [8], we have presented our results. In this paper we present in more detail the motivation and implementation of our approach as well as our experimental

setup that consists of novel combinations of hard- and software necessary to interpret our results.

The rest of this paper is organized as follows. Section II lays out the background for our work. We present the design and implementation of the metrics in Section III and the evaluation in Section IV. We discuss related work in Section V and draw our conclusions in Section VI.

II. BACKGROUND

In this section we present background knowledge on physical layer encoding and modulation in the IEEE 802.15.4 standard, software defined radio, interference sources and the CC2420 radio's hardware estimators for link quality.

A. IEEE 802.15.4 PHY encoding and modulation

The IEEE 802.15.4 2.4 GHz physical layer uses a direct sequence spread spectrum (DSSS) technique to improve robustness against interference from other ISM transmitters such as WiFi and Bluetooth devices [7] [9]. Along the transmission path, information bits from a PHY packet spread into a binary sequence. The binary sequence consists of bits that are called chips, generated from a code map matching every 4 information bits to a 32-chip pseudo noise (PN) code. The 16 code words are orthogonal, whose large hamming distances ensure that a moderately corrupted PN-code can be reconstructed by a correlation function at the receiver.

Every two chips are encoded as an O-QPSK symbol before being up-converted to 2.4 GHz and transmitted. An O-QPSK symbol is represented by a two-dimensional spatial vector in the complex signal domain, whose length and direction represent its amplitude and phase respectively. The four possible symbols' signal vectors point at four different directions 90° apart, making it relatively easy for the receiver to demodulate using obtained phase information.

B. IEEE 802.15.4 demodulation and PHY decoding in SDR

On the receiving side, reconstruction of the original packet basically involves reversing the operations of the transmitter. With current circuit technologies, the 2.4 GHz received signal needs to be down-converted to an baseband signal by an analog mixer. The baseband signal is then digitized by an A/D converter. Successive digital demodulation and decoding are then performed, with additional steps for correcting the phase and frequency offsets between the clocks of the transmitter and the receiver.

Our IEEE 802.15.4 receiver is based on the GNU Radio [10] SDR tool chain and its USRP hardware [11]. The USRP performs down-conversion and A/D conversion of the radio signal. The GNU Radio provides a mixed Python and C++ programming environment with common DSP library modules. We use the UCLA's ZigBee PHY library [12] [13], a GNU Radio extension, to carry out IEEE 802.15.4 PHY functions. This full digital receiver [14] has an architecture similar to the CC2420, but implements O-QPSK as FSK [9].

We show the receiver's digital signal processing flow in Fig. 1:

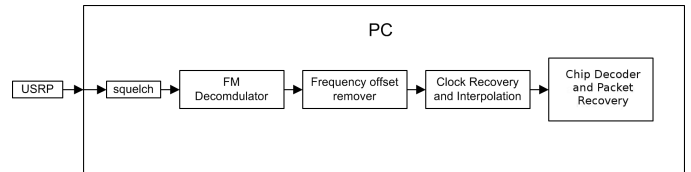


Fig. 1. Block Diagram of UCLA ZigBee PHY Receiver

- 1) The computer filters out digital samples representing weak signals (or noise) with a threshold-based squelch.
- 2) The complex data stream is demodulated to a frequency signal by the FM Demodulator for FSK demodulation.
- 3) The computer high-passes the frequency signal for removing the frequency offset.
- 4) the FSK samples are recovered by a clock recovery module [15] and an interpolator [16].
- 5) Finally, the recovered samples are sent to the chip decoder that regenerates the packet.

In contrast to CC2420's silicon solution, our SDR tool chain allows us to modify receiver parameters and to insert custom functions at various stages of the receiving path.

C. Common sources of noise and interference

In the frequency domain, an IEEE 802.15.4 signal manifests itself as a bell shape spectrum on a 5 MHz-wide channel. As shown in Fig. 2, the main lobe at +/- 1.5 MHz of the center frequency is the useful signal component for demodulation. The two symmetrical side lobes are signal components on the side of and separated from the main lobe by peaks.

Adjacent channel interference is a troublesome issue for multi-channel protocols [17]. When an interferer operating on an adjacent channel is located much closer to the receiver than the transmitter, a significant portion of the side lobe energy from the interferer may leak into the receiver's channel, causing distortion to the received signal.

Fig. 3 shows a received signal spectrum affected by such an interferer from the lower adjacent channel, which causes an elevated noise floor on the lower half band.

Background noise in a radio channel is often modeled as additive white Gaussian noise (AWGN). AWGN raises the noise floor evenly across the channel, as shown in Fig. 4.

An IEEE 802.15.4 interferer using the same channel, but located farther away from the receiver than the transmitter, degrades the received signal in a form of in-band noise. Because both the interference and the signal share the same spectrum characteristics, the resulted spectrum looks deceptively normal, as shown in Fig. 5.

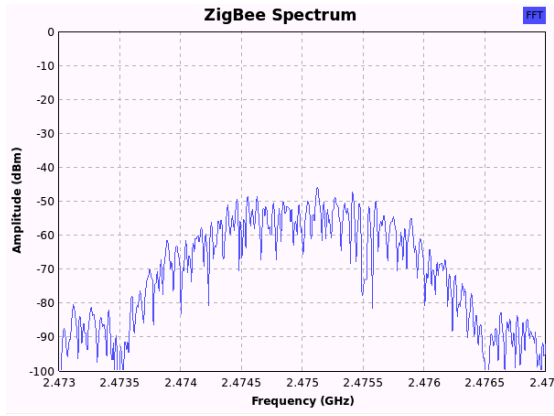


Fig. 2. Normal signal spectrum

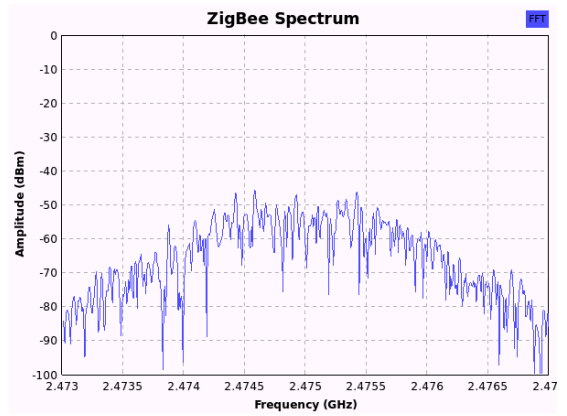


Fig. 5. Signal spectrum overlapped with noise from an in-band interferer.

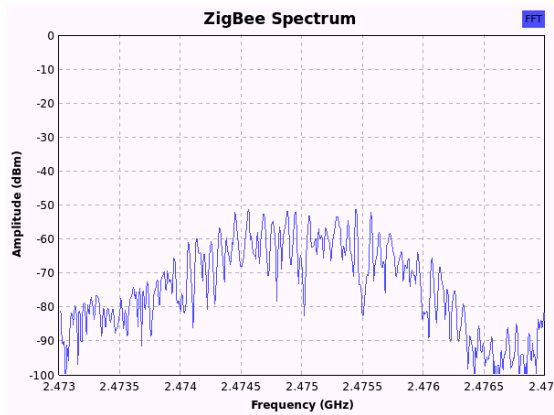


Fig. 3. Signal spectrum with adjacent channel interference that causes an elevated noise floor on the lower half band.

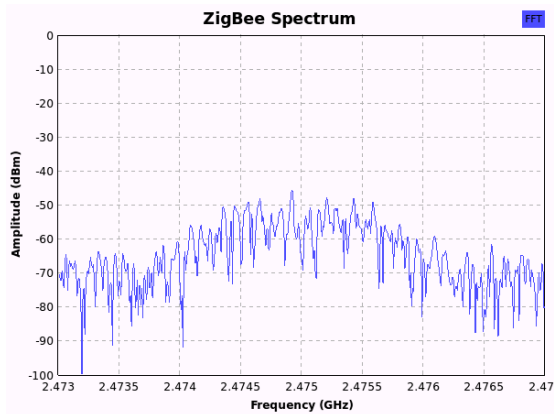


Fig. 4. Signal spectrum in AWGN channel. Background noise raises the noise floor evenly across the channel.

D. CC2420 RSSI and LQI characteristics and limitations

The CC2420 has a built-in RSSI providing the in band radio power that is always averaged over 8 symbol periods (128 μ s). Since this indicator cannot distinguish random interference from signals, it has a low correlation with link quality [2] [3]. More fundamentally, signal strength itself is not a good metric to estimate the PRR, because of the poor sensitivity. We will provide more analysis about signal strength and PRR in Section IV-B1.

CC2420's LQI indicates the average correlation value of the 8 first symbols of the received PHY header. The data sheet does not mention the exact algorithm. Some researchers argue that the LQI is more relative to PRR than RSSI [3]. But most of these studies give different conclusions. The correlation coefficient between LQI to PRR varies a lot in these studies [3] [2]. Hence, the trustworthiness of the LQI is at least unclear.

III. DESIGN AND IMPLEMENTATION OF THREE METRICS

In this section we describe the design and implementation of the three metrics CER, EVM and SF with GNU Radio and the UCLA ZigBee library.

A. Link quality estimation by CER

Due to attenuation and interference, the receiver might observe errors in the decoded chip sequence. Usually a weak link has a higher number of chip errors than a good link. A sufficient number of the uncorrupted chips in the sequence ensures correct decoding. The corrupted chips however contain useful information about the severity of signal attenuation and distortion. Assuming correct chip-to-bit decoding, we can mark the number of chip errors in a received chip sequence by comparing it with the decoded PN code. We then derive the received packet's chip error rate by taking the ratio between the number of error chips N_e and the total number of chips N in the packet.

$$CER = \frac{N_e}{N} \times 100\% \quad (1)$$

We implement the CER counter as a C++ function and insert it into the chip-to-bit decoder in the SDR receiver.

B. Link quality estimation by EVM

The FSK demodulator is always observing a certain degree of signal distortion, visible as the difference between the received symbol vector and the ideal vector. To quantify the degree of distortion, an error vector can be drawn between the two vectors. Its length is termed Error Vector Magnitude [18]. A large EVM might cause incorrect symbol decoding, which in turn leads to a chip decoding error.

Since our SDR's demodulator does not provide the received symbol's amplitude information, we define the symbol distortion as the phase error between the received and ideal symbol vectors instead, while keeping the common term EVM:

$$EVM = \sqrt{2 - 2\cos\phi} \times 100\% \quad (2)$$

We implement and integrate the EVM calculator in the FSK demodulator. Because IEEE 802.15.4's DSSS technique makes it tolerant with noise impulses, a large EVM in a few symbols may not necessarily cause a lost packet. Therefore, we believe that the average EVM measured over the whole packet is more correlated to PRR than the peak EVM. Hence our EVM metric is obtained by computing the average of all received symbols' EVMs over the received packet.

C. Link quality estimation at frequency domain

Since a signal spans over a certain frequency band, and various types of noise and interference are often frequency specific, we believe it is interesting to show the signal distortion in the frequency domain as well, by means of spectrum analysis. As we conceive of a measurement of the frequency-domain signal distortion, we take two factors into consideration: meaningful signal spectrum components; and computation complexity. We come up with a signal quality metric for 5 MHz-wide IEEE 802.15.4 signals, taking the ratio between the energy of the main lobe and that of the sum of the two side lobes, and call this the "Spectrum Factor" (SF). In the equation below, we denote the received signal in the frequency domain as $X(f)$ and its central frequency as f_c . Because the computation consists of only bandpass filtering and integration, SF should be simple enough to be implemented by low cost digital hardware as well. We sample 2 bytes from a packet to determine its SF.

$$SF = \frac{\int_{f_c-1.5MHz}^{f_c+1.5MHz} |X(f)|^2}{\int_{f_c+1.5MHz}^{f_c+2MHz} |X(f)|^2 + \int_{f_c-2MHz}^{f_c-1.5MHz} |X(f)|^2} \quad (3)$$

We use the GNU Radio's FFT module to perform spectrum analysis on the band. We set the sampling rate to 4 MHz, equivalent to twice the chip rate, which yields the best time recovery and demodulation.

IV. EVALUATION

We evaluate the three metrics under four different channel conditions: clean channel, adjacent channel interference, additive white Gaussian noise (AWGN), and in-band noise. We construct experimental setups to emulate the channel

conditions, by novel combinations of hardware and software. In each run, a transmitter sends 500 packets to the receiver to calculate the packet reception rate (PRR) and other statistics.

A. Experimental Setup

To emulate a clean channel, we connect two USRPs, one transmitter and one receiver, by a coaxial cable and an -20 dB attenuator. In this experiment, we decrease the transmission power by 20% (about 1 dBm) between runs. The received signal power decreases from -54.89 dBm to -66.82 dBm. Since the results change sharply when the received signal strength decreases from -66.56 dBm to -66.86 dBm we make additional runs by decreasing power by 1% within this range.

AWGN channel is one of the most common channel model in wireless communication. As shown in Fig. 6, for emulating an AWGN channel, a -70dBm white Gauss noise source from GNU Radio is added to the stream of received data from the USRP receiver. We perform more runs in the ranges where the results change sharply.

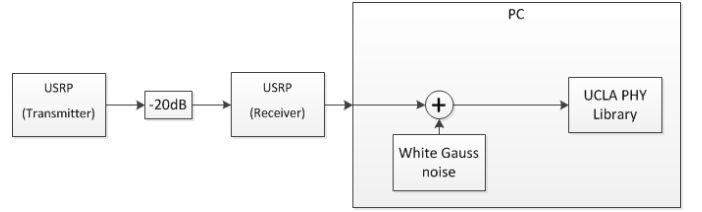


Fig. 6. Testing System for AWGN channel

The in-band noise is generated by adding a -50dBm Gaussian noise at the output of UCLA ZigBee PHY transmitter. Since the noise and signal pass through the same smoothing filter in the USRP, they have the same bandwidth. The connection diagram is shown in Fig. 7. The first runs are done by decreasing the transmission power by 20% (about 1 dBm) each time. After the first runs we do more experiments in the area where the results change sharply.

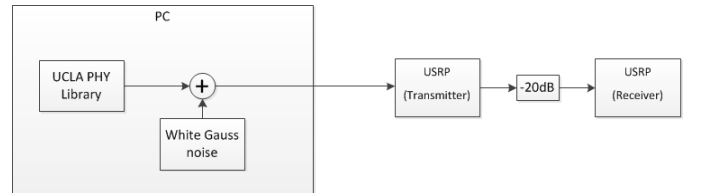


Fig. 7. Testing System for In-Band Noise

For emulation of adjacent channel interference, as indicated in Fig. 8, we connect a Tmote Sky node that transmits random data on the adjacent channel continuously at -13 dBm, to our system by a T-connector and an extra 20 dB attenuator.

We decrease the transmission power by 20% (about 1 dBm) each time firstly. Then, we do more precise experiments by decreasing power by 1% each time in the area where the results change sharply.

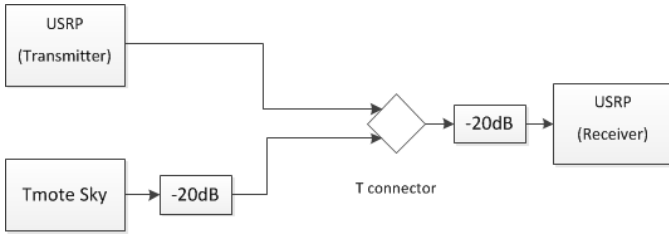


Fig. 8. Testing System for Adjacent Channel Interference

B. Experimental Results

1) *Signal strength VS. PRR*: Fig. 9 indicates the relation of signal strength with PRR. For calculating the PRR, we transmit 500 packets in each experiment.

In the clean channel the PRR declines sharply at about -67 dBm. Since the receiving performance in a clean communication channel is mainly limited by the sensitivity of the receiver, our results show that the minimum signal the USRP can receive is about -67 dBm.

The PRRs are larger than 98% when the signal is stronger than -57 dBm in the AWGN channel, in-band noise channel and the channel with adjacent channel noise. The PRR decreases to 0% when the signal is weaker than -63 dBm.

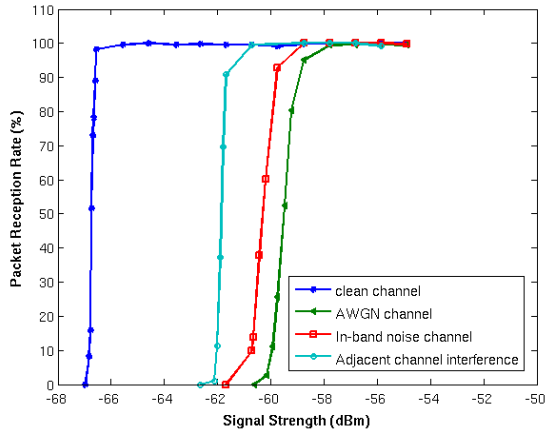


Fig. 9. Signal strength VS. PRR

2) *CER VS. PRR*: Fig. 10 reveals the relationship between the CER and the PRR. The standard deviation of the CER is plotted as error bars on each curve.

CER has the smallest standard deviation among the three metrics under all channel conditions. The error bars marked on its curves are almost invisible. The range of the CER depends on the channel conditions. For the same PRR, CER in AWGN channel is significantly higher than that in other three channels. In the AWGN channel, the CER increases to 4.5% when the PRR decreases to 0%. In contrast, the CER in the clean channel is always below 0.1%.

Loss of packet does not only depend on the chip error, but also the failure of radio synchronization within the 4 bytes preamble. Furthermore, the weak signal strength slows down

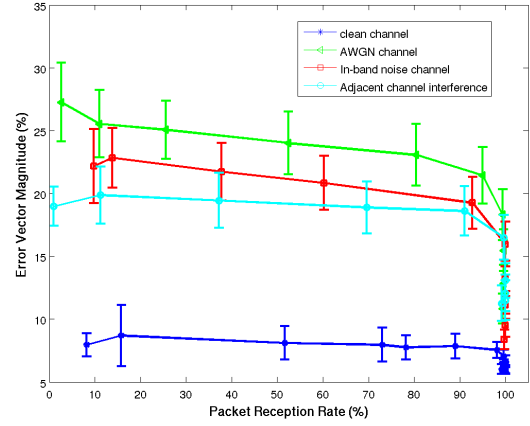


Fig. 11. PRR VS. EVM

the speed of synchronization. Above reasons can possibly explain why PRR in clean channel drops off sharply at -66 dBm and keeps a long CER.

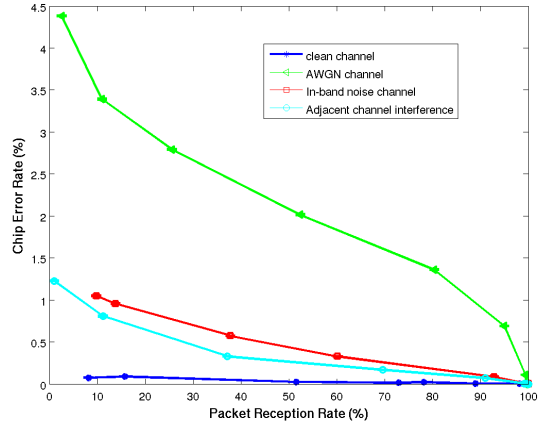


Fig. 10. PRR VS. CER

3) *EVM VS. PRR*: Fig. 11 shows the relation between the average EVM and the PRR.

EVM has very good sensitivity for changes in the PRR, when the PRR is higher than 95%. The EVM increases from 6% up to 21% as the PRR degrades from 100% to 95%. This can be used, for example, to track a fading link between two mobile nodes. Furthermore, the standard deviation of EVM in this range is low.

4) *SF VS. PRR*: The SF does not rely on successful synchronization or demodulation as EVM and CER. Therefore, the SF can be computed even on unsuccessfully demodulated packets. This enables SF to estimate links faster than CER and EVM when the link is very weak. Fig. 12 shows the relation between PRR and SF. The SF has a fairly small standard deviation and reasonable sensitivity for a PRR lower than 10% under the three noisy channel conditions.

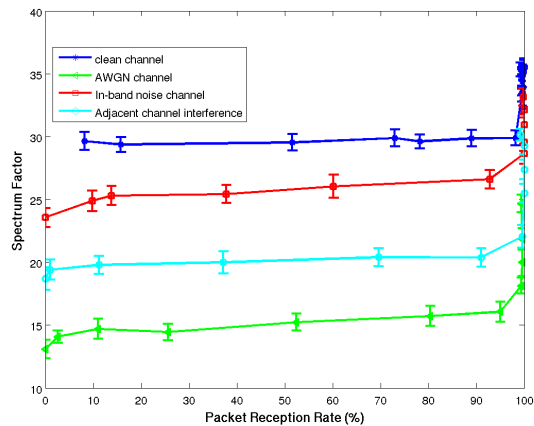


Fig. 12. PRR VS. SF

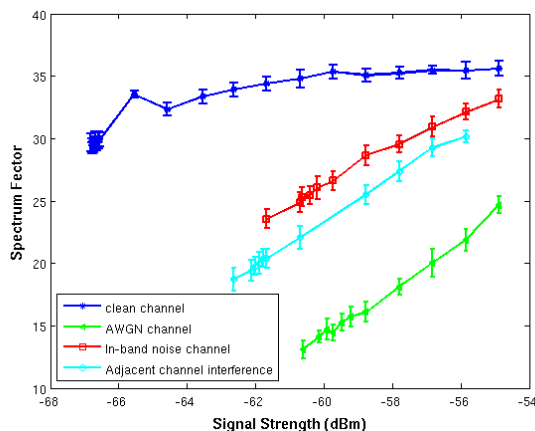


Fig. 13. Signal Strength VS. SF

Fig. 13 shows the relation between signal strength and SF. In all of the four channel conditions, the SF is almost linear to the signal strength and has a fairly small standard deviation.

C. Handling complex channel conditions

An important observation from the results described above is that the three metrics behave differently under different channels conditions, suggesting us to apply them carefully. In order to improve accuracy, we need to characterize a link's channel condition, by identifying its dominant spectral pattern, before we estimate the link quality. A heuristic algorithm can be developed to match a signal's spectrum to one of the four channel conditions we have described. We can then apply the three metrics together: CER is used to approximately estimate the PRR; EVM can be applied to obtain a highly accurate PRR estimation for a PRR range between 95% and 100%; On the other hand, SF can be used to estimate a weaker link.

V. RELATED WORK

We divide the related work in two parts. First we discuss link quality estimation in wireless sensor networks focusing

on the hardware metrics. In the second part, we discuss other WSN research that has used software defined radios.

A. Link quality estimation

Most of the research on link quality estimation in wireless sensor networks has made use of the estimators provided by the CC2420 radio. This radio provides the LQI and RSSI. Another metric that is easy to access is the PRR. When we sample the RSSI in absence of packet transmissions, we can also obtain the noise floor and the SNR. Typically, this is done directly after a packet has been received.

A PRR-based metric is the well-known expected transmission count metric (ETX) [19], [20] which in WSN corresponds to the number of transmissions, including retransmissions, required to transmit a packet to the sink. Cerpa et al. extend ETX to also include temporal information such as the number of subsequent packet losses [21]. Boano et al. show that only a few LQI samples are necessary to identify very good links [22]. The multi-hop LQI protocol base is link estimation on the LQI [23]. Using the RSSI is proposed by Srinivasan and Levis [24] as well as by Lin et al. [25]. Researchers have also combined these metrics. For example, Rondinone et al. combine PRR and RSSI [26] while the Triangle metric by Boano et al. combines RSSI, LQI, SNR into one single metric [27]. Another group of estimators uses additional information. For example, Baccour et al.'s fuzzy estimator also includes information about link stability and link asymmetry [28]. Most of these studies make use existing technology such as the CC2420 radio. In contrast, we use a clean slate approach and use software defined radio to evaluate new estimators that could be provided by future radios.

B. SDR applications in WSNs

Using SDR, Wu et al. observe that chip errors in IEEE 802.15.4 frames caused by attenuation and a hidden IEEE 802.15.4 interferer show two distinctive distribution patterns [29]. But such a distinction is not hard to make using RSSI alone. It would be interesting, however, to consider both chip error rate and chip error distribution when characterizing a link. The same authors show that chip error pattern detection can also be applied to detect packet loss due to mobile nodes [30].

FHMESH [31] builds a mesh networking stack on top of the UCLA ZigBee library, which interacts with common sensor nodes. Sundeep et al. apply the technology to medical body area network [32]. Both works use packet error rate (PER), the complement of PRR, as their sole performance metric.

VI. CONCLUSIONS

In this paper we have presented three link quality estimation metrics that are easy to implement by modern micro electronics. After evaluating our metrics under several channel conditions, we observe that all the three metrics behave differently under these conditions. Moreover, combining the three metrics integrates their advantages and provides better accuracy. This makes the metrics attractive for future sensor

networks that we expect to feature more advanced SDR-like low power radios.

ACKNOWLEDGMENTS

This work has been performed within the SICS Center for Networked Systems funded by VINNOVA, SSF, KKS, ABB, Ericsson, Saab SDS, TeliaSonera, T2Data, Vendolocus and Peerialism. This work has been supported by VINNOVA and the European Commission with contract FP7-2007-2-224053 (CONET).

REFERENCES

- [1] Texas Instruments, "CC2420 datasheet."
- [2] L. Tang, K.-C. Wang, Y. Huang, and F. Gu, "Channel characterization and link quality assessment of IEEE 802.15.4-compliant radio for factory environments," *Industrial Informatics, IEEE Transactions on*, vol. 3, pp. 99–110, May 2007.
- [3] Z. Jian and Z. Hai, "A link quality evaluation model in wireless sensor networks," *Sensor Technologies and Applications, 2009. SENSORCOMM '09. Third International Conference on*, pp. 1–5, Jun. 2009.
- [4] M. Holland, R. Aures, and W. Heinzelman, "Experimental investigation of radio performance in wireless sensor networks," in *WiMesh'06*, (Reston, Virginia, USA), 2006.
- [5] T. Rein, "Energy and time efficient link-quality estimation for wireless sensor networks," Master's thesis, ETH, Zurich, Switzerland, April 2007.
- [6] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, "An empirical study of low power wireless," in *SING Tech Report*, October '08.
- [7] *IEEE Std 802.15.4-2006 (Revision of IEEE Std 802.15.4-2003)*, pp. 0_1–305, 2006.
- [8] Y. Qin, Z. He, and T. Voigt, "Link Quality Estimation for Future Cooperating Objects," in *The Second International Workshop on Networks of Cooperating Objects (CONET 2011)*, (Chicago, USA), Apr. 2011.
- [9] J. Notor, A. Caviglia, and G. Levy, "Cmos rfc architectures for IEEE 802.15.4 networks," *Cadence Design Systems, Inc*, 2003.
- [10] "GNU radio," <http://gnuradio.org>. Visited 2011-03-12.
- [11] "Universal software radio peripheral," <http://gnuradio.org/redmine/wiki/gnuradio/USRP>. Visited 2011-03-12.
- [12] T. Schmid, L. Choong, and S. Leidelöf, "UCLA ZigBee PHY," <https://www.cgran.org/wiki/UCLAZigBee>. Visited 2011-03-12.
- [13] T. Schmid, "Gnu radio 802.15.4 en- and decoding," Tech. Rep. TR-UCLA-NESL-200609-06, Networked & Embedded Systems Laboratory, University of California, Los Angeles, September 2006.
- [14] H. Meyr, M. Moeneclaey, and S. Fechtel, *Digital Communication Receiver: Synchronization, Channel Estimation and Signal Processing*. John Wiley & Sons, 1998.
- [15] G. Danesfahani and T. Jeans, "Optimisation of modified mueller and muller algorithm," *Electronics Letters*, vol. 31, pp. 1032–1033, Jun. 1995.
- [16] G. Oetken, "A new approach for the design of digital interpolating filters," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 27, pp. 637–643, Dec. 1979.
- [17] G. Xing, M. Sha, J. Huang, G. Zhou, X. Wang, and S. Liu, "Multi-channel interference measurement and modeling in low-power wireless networks," in *30th IEEE Real-Time Systems Symposium (RTSS)*, (Vienna, Austria), Dec. 2009.
- [18] J. Pinto and I. Darwazeh, "Error vector magnitude relation to magnitude and phase distortion in 8-psk systems," *Electronics Letters*, vol. 37, pp. 437–438, Mar. 2001.
- [19] D. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A High-throughput Path Metric for Multi-hop Wireless Routing," in *Proc. of the 9th Conference on Mobile Computing and Networking (Mobicom)*, (San Diego, CA, USA), pp. 134–146, Sept. 2003.
- [20] D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.
- [21] A. Cerpa, J. Wong, M. Potkonjak, and D. Estrin, "Temporal Properties of Low-power Wireless Links: Modeling and Implications on Multi-hop Routing," in *Proc. of the 6th Symposium on Mobile ad hoc networking and computing (MobiHoc)*, (Urbana-Champaign, IL, USA), pp. 414–425, May 2005.
- [22] C. Boano, T. Voigt, A. Dunkels, F. Österlind, N. Tsiftes, L. Mottola, and P. Surez, "Exploiting the LQI Variance for Rapid Channel Quality Assessment," in *Proc. of the 8th Conference on Information Processing in Sensor Networks (IPSN), poster session*, (San Francisco, CA, USA), pp. 369–370, Apr. 2009.
- [23] TinyOS MultiHopLQI routing algorithm. <http://www.tinyos.net/tinyos-1.x/tos/lib/MultiHopLQI>, 2004.
- [24] K. Srinivasan and P. Levis, "RSSI is Under Appreciated," in *Proc. of the 3rd Workshop on Embedded Networked Sensors (EmNets)*, (Cambridge, MA, USA), May 2006.
- [25] S. Lin, J. Zhang, G. Zhou, L. Gu, T. He, and J. Stankovic, "ATPC: Adaptive Transmission Power Control for Wireless Sensor Networks," in *Proc. of the 4th Conference on Embedded Networked Sensor Systems (SenSys)*, (Boulder, Colorado, USA), Nov. 2006.
- [26] M. Rondinone, J. Ansari, J. Riihijärvi, P. Mähönen, "Designing a Reliable, Stable Link Quality Metric for Wireless Sensor Networks," in *RealWSN*, 2008.
- [27] C. Boano, M. A. Zúniga, T. Voigt, A. Willig, K. Römer, "The Triangle Metric: Fast Link Quality Estimation for Mobile Wireless Sensor Networks," in *ICCCN*, 2010.
- [28] N. Baccour, A. Koubâa, H. Youssef, M. Jamâa, D. Rosário, M. Alves, L. B. Becker, "F-LQE: A Fuzzy Link Quality Estimator for Wireless Sensor Networks," in *EWSN*, 2010.
- [29] K. Wu, H. Tan, H.-L. Ngan, and L. Ni, "Chip error pattern analysis in IEEE 802.15.4," in *INFOCOM, 2010 Proceedings IEEE*, pp. 1–5, 2010.
- [30] K. Wu, H. Tan, H. Ngan, Y. Liu, and L. Ni, "Measurement study of mobility-induced losses in IEEE 802.15.4," in *Communications (ICC), 2010 IEEE International Conference on*, pp. 1–5, IEEE, 2010.
- [31] L. You, C. Dong, G. Chen, Y. Dai, and W. Zhou, "FHMesh: A Flexible Heterogeneous Mesh Networking Platform," in *2010 Sixth International Conference on Mobile Ad-hoc and Sensor Networks*, pp. 189–192, IEEE, 2010.
- [32] C. Sundeep Prabhakar, R. de Francisco, and G. Leus, "Performance Evaluation of an IEEE 802.15.4 Cognitive Radio Link in the 2360-2400 MHz Band," in *IEEE WCNC, IEEE*, 2011.