Measurement Study of Mobility-Induced Losses in IEEE 802.15.4

Kaishun Wu*[†], Haoyu Tan*, Hoi-Lun Ngan*, Yunhuai Liu* and Lionel M. Ni*

*Department of Computer Science and Engineering, Hong Kong University of Science and Technology Email: {kwinson,hytan,cpeglun,yunhuai,ni}@cse.ust.hk [†]School of Physics and Engineering, Sun Yat-sen University, Guangzhou, China

Abstract-Recent years have seen an increasing need of wireless networks in a mobile environment serving for more complex tasks and applications. Mobility becomes an indispensable factor of the system design and has been widely recognized as a general cause of packet loss. Though many works have been done on mobility study, to the best of our knowledge, they are mainly based on simulations or analytical studies that assume idealized link conditions. In this work, we experimentally investigate the nature of the error characteristics of mobility-induced packet losses at "chip-level" in IEEE 802.15.4. We believe this more understanding of mobility-induced packet losses can bring great potential benefits for further study on channel coding, routing and protocol design. Toward this end, we design and implement an efficient algorithm to distinguish mobility-induced packet losses from other packet losses of static environments such as attenuation. Our algorithm is greatly advantaged as it needs no training data even when environment changes. Collecting three corrupted packets is sufficient to obtain a satisfactory performance. This feature makes our design in particular suitable for dynamic and mobile environments, allowing real-time mobilityinduced loss detection in an online manner. Experiments based on GNU Radio testbed show that our algorithm can provide an accuracy of up to 96%.

Index Terms—mobility; IEEE 802.15.4; pseudo-noise code (PN-Code); chip error characteristics; measurement study

I. INTRODUCTION

In recent years, more and more wireless networks are deployed in a mobile environment serving for more complex tasks and applications. Device mobility becomes unavoidable and is introducing more and more impact on the system design. It is widely accepted as a cause of packet loss because of complex behavior of wireless radios in a practical mobile environment [1].

The majority of previous studies on mobility-induced losses focused on the investigations on packet-level losses. Few efforts have been devoted to access into a lower layer to investigate the error characteristics in a deeper and more comprehensive manner. For instances, the metrics of RSSI (Received Signal Strength Indication), SNR (Signal to Noise Ratio), and etc are widely in use that are all at the packetlevel to reflect the mobility properties. In many real scenarios, these metrics can be very misleading. For example, RSSI is available only when the packet is successfully received packets, while much of more useful information is lost as well when the corrupted packets are discarded [2]. There is very little information provided by the PHY to the upper layer in current design off-the-shelf wireless products.

IEEE 802.15.4 [3] uses the DSSS modulation scheme that is also applied in IEEE 802.11b. In the specification, every 4-bit symbol is encoded into a sequence of 32 chips that are actually transmitted over the air. The 32 chips as a whole is called a pseudo-noise code (PN-Code). A chip is an accessible element at the physical layer. An information bit is repackaged as certain sequences of chips before being transmitted over the air. Hence, a chip is a fraction of a packet or even a bit. Since the chip duration is much shorter than the symbol duration, it is believed that chips are capable of capturing more information from the channel. A symbol error can only reflect whether there is an error or not. In contrast, the chip correlation information in a symbol can give us more valuable information for analysis.

Based on this PN-Code and chip errors, in this paper we conduct extensive experiments to characterize different chip error characteristics owing to attenuation and mobility. To the best of our knowledge, this is the first detailed and systematic experiment study of chip level error characteristics of mobility-induced packet losses under IEEE 802.15.4. Another major contribution is that, according to the different characteristics of mobility-induced and attenuation-induced errors, we are able to distinguish these two kinds with fairly high confidence. Moreover, three packets are sufficient to to obtain an accuracy over 96%. The algorithm provides attractive flexibility as it is universal to different environments without further extensions. In other words no data training is needed when the environment changes.

The rest of this paper is organized as follows. In Section II, we will present our experimental methodology. It follows by the experimental results and analysis of chip error characteristics. Detection algorithm and evaluation will be given in Section IV. Furthermore, a brief overview on related literature is given in Section V. Finally, we will conclude our our work.

II. EXPERIMENTAL METHODOLOGY

In this section we will show the experiment setup and scenarios. We will also describe the metrics for our analysis.

4-bit Symbol	32-chip PN-Code	4-bit Symbol	32-chip PN-Code
0x0	0x744AC39B	0x8	0xDEE06931
0x1	0x44AC39B7	0x9	0xEE06931D
0x2	0x4AC39B74	0xA	0xE06931DE
0x3	0xAC39B744	0xB	0x06931DEE
0x4	0xC39B744A	0xC	0x6931DEE0
0x5	0x39B744AC	0xD	0x931DEE06
0x6	0x9B744AC3	0xE	0x31DEE069
0x7	0xB744AC39	0xF	0x1DEE0693

TABLE I: Symbol to Chip Mapping

A. Preliminaries of PN-Code in IEEE 802.15.4

The IEEE 802.15.4 standard aims at conveying information over relatively short distance in low-rate wireless personal area networks (LR-WPANs). The standard defines 27 channels spreading over 3 main bands: 868MHz, 915MHz and 2.4GHz. Our study will focus on the 2.4 GHz band which contains channels 11 through 26 and provides a transmission rate of 250 Kbps. More information of the standard can be found in its specification [3].

The IEEE 802.15.4 standard employs DSSS at PHY to resist interference and noise. Before the bit sequences are being modulated and transmitted through the antenna, there is an additional process to chop the sequences into 4-bit symbols and replace each symbol with the corresponding 32-bit PN-Code which is also known as chip sequence. The mapping is shown in Table I. The resultant stream of chip sequences is modulated to the carrier using OQPSK [3].

The receiver attempts to decode the 32-bit chip sequence to the corresponding 4-bit symbol. Due to attenuation, mobility and other factors, it is not unusual that one or more bits in the chip sequence are corrupted. The set of chip sequence are designed to cope with such errors. The receiver will correlate the received sequence with each of the 16 desired chip sequences and select one with the minimum number of bit differences. The set can tolerate up to 10 bit differences without resulting in decoding errors.

This results in a maximum likelihood decoder (MLD) in our study. In MLD, a 32-bit chip sequence P is compared with 16 predefined PN-Codes $PN_1, PN_2, \ldots, PN_{16}$ and output k_{min} such that

$$k_{min} = \arg\min_{k} h(P, PN_k),$$

where $k \in \{PN_1, PN_2, ..., PN_{16}\}$ and h is the hamming distance of two bit strings P and PN_k . We define the value of $h(P, PN_{kmin})$ as Chip Errors Per PN-Code (CEPP) for the received sequence P.

B. Experimental Setup

We run GNURadio [4] on Universal Software Radio Peripheral (USRP) [5] devices as the transmitter and receiver in our experiment. Each of the devices is equipped with the RFX2400 daughterboard for 2.3 - 2.4GHz communication. We

TABLE II: Experimental Scenarios

Scenario Number	Location	Mode	Speed
1	Office	Attenuation	Static
2	Passageway	Mobility	Walking speed
3	Campus	Mobility	Vehicular speed

adopted and modified the IEEE 802.15.4 PHY software implementation for GNURadio from [6]. Rather than halting the decoding process of the remaining portion of PN sequences, we allow the receiver to continue the decoding process even if it encounters a chip sequence with more than 10 bit differences until the end of the frame.

The transmitter has a DAC rate of 128M samples/s and a interpolation rate of 128. On the receiver side, the ADC rate is 64M samples/s and the decimation rate is 64. Each frame in our experiments consists of an 11-byte preamble, a 90-byte payload and a 2-byte CRC. Therefore, each frame has 206 CEPPs in accordance to 206 received PN-Codes.

To avoid the interference from IEEE 802.11b/g devices which are operating at 2.4 GHz ISM band, we have switched the communication channel to 2.375 GHz. We have verified that no users were using this channel during our experiments.

C. Experimental Scenario

This work aims at distinguishing packet losses due to attenuation and mobility of transceivers. Table II shows the characteristics of our experiment scenarios. We conduct experiment in an ordinary office to collect packet loss data due to attenuation. In mobility experiments, we analyze the chip error characteristics at walking and vehicular speeds.

1) Mobility: To investigate the mobility properties at walking speed, we conduct the experiment in an ordinary passageway. At the calibration phase, the transmission power is tuned to a minimum while keeping the packet reception ratio (PRR) at 100%. After that, the sender broadcasts at a constant rate of 250 Kbps and moves at walking speed. During the experiment, the sender sends 2000 frames at 10 ms interval with random payload content. The experiment is repeated several times for different walking speeds and locations.

The second experiment examines the mobility effect which the sender is moving at faster, vehicular speed. In this experiment, we put the receiver in the center of the garden and drove a car around it at different speeds. The experiment is repeated using the same procedure as in the first one.

2) Attenuation: Attenuation is the change of signal strength as it propagates through the space. It is caused by a complex set of factors, including the propagation medium, reflection and propagation distance. We conduct the experiment in an ordinary office. We fix the locations of the transmitter and receiver.

The transmitter and receiver are placed in a straight line with a clear line-of-sight (LOS) path between them. Then we calibrate and record the set of transmission powers which creates links in the transitional region [7]. With the same transmission configuration as in mobility experiments, this experiment repeats 100 times with different transmission powers. We have also considered the cases that LOS transmission is impossible. We investigate the error characteristics for nonline-of-sight (NLOS) scenarios by putting a metallic cabinet between the transmitter and the receiver.

D. CEPP Statistics of Corrupted Frames

CEPP measures the correlation between a received PN-Code and its maximum likelihood in pre-defined PN-Codes. If a CEPP value is greater than or equal to the correlator threshold (i.e., 10 in our experiments), its corresponding maximum likelihood PN-Code is unlikely to be the one transmitted by the sender. We refer this PN-Codes as *undecodable* PN-Codes. Correspondingly, we label a PN-Code as *decodable* if its CEPP is less than the correlator threshold. Figure 1 shows complete CEPPs in each of three corrupted frames observed in our experiments. If one concerns only correct frame decoding, the entire frame should be dropped immediately once an undecodable PN-Code is detected. However, the remaining PN-Codes of an undecodable frame are not useless because they may reveal the causes of errors. This motivates us to investigate the following two CEPP statistics of an undecodable frame.



Fig. 1: Three examples of CEPP series in corrupted frames from our experiments, in which the correlator threshold is set to 10. Each frame is composed of an 11-byte header, a 90-byte payload and a 2-byte CRC checksum. Therefore, each frame has 206 PN-Codes and 206 corresponding CEPP values.

1) Position of the First Undecodable PN-Code (PFUP): Position of the First Undecodable PN-Code (PFUP) is defined as

$$PFUP = \frac{k}{L}$$

where k indicates the index of the first undecodable PN-Code in the frame and L is the total number of PN-Codes of the frame. The first undecodable PN-Code in a frame is special because all PN-Codes ahead of it can be correctly translated into information bits. Thus, PFUP is a lower bound of decodable portion of a frame. For the three frames shown in Figure 1, the PFUP values are 54/206 = 0.262, 68/206 = 0.330 and 182/206 = 0.883 respectively.

2) Partial Frequency of Decodable PN-Codes (PFDP): We take the position of the first undecodable PN-Code as a splitting point. It separates the frame into two regions. The first one consists of decodable PN-Codes. The second one, identified as *undecodable portion*, contains the remaining PN-Codes of the frame which have a series of low and high CEPPs. We now define the Partial Frequency of Decodable PN-Codes (PFDP) as

$$PFDP = \frac{d}{L_2} = \frac{d}{L-k+1}$$

where L_2 denotes the total number of PN-Codes and d is the number of PN-Codes with CEPPs smaller than threshold in the undecodable region respectively. It is not unusual that PFDP is exactly zero as shown in Figure 1(c). A possible scenario is that the transmitted signal is completely blocked at the later portion of a frame transmission. The received PN-Code in the undecodable region is 'decoded' from noise which is highly uncorrelated with the transmitted signal.

The value of PFDP is a measurement of the power relationship between received signal and noise. It differs from traditional SNR as PFDP concerns only the undecodable region while SNR averages the whole frame. By taking the error regions into account, we believe PFDF would outperform SNR in identifying and characterizing frame losses.

III. EXPERIMENTAL RESULTS AND DISCUSSION

In this section, we present our experimental results and discuss them in detail. It is worth noting that in our work, the chips under study are over-the-air chips but not the information bits. We believe the chip-level information can give us more details because of a shorter duration for a chip than a symbol. To the best of our knowledge, many other studies on the mobility properties address only the PRR. And most of them just used simulations to investigate the mobility properties.

We have a total of 2,000,000 frames for each of the experiment scenarios. Around 20% of them are corrupted. We compute PFUP and PFDP for every frame and plot their cumulative distributions for each of the scenarios. Figure 2 shows the PFUP distributions in the attenuation and mobility scenarios. PFDP distributions of attenuation and mobility are shown in Figure 3. Figure 4 shows the PFDP distributions of two different mobility scenarios, walking and vehicular speed.



Fig. 2: Empirical CDF of PFUP in sce-Fig. 3: Empirical CDF of PFDP in sce-Fig. 4: Empirical CDF of PFDP in scenario of attenuation and mobility nario of attenuation and mobility

nario of walking speed and vehicle speed mobility

In Figure 2, we observe that PFUP bias toward 0 for both scenarios. This indicates that the first error tends to appear closer to header. There are no significant differences between these two scenarios.

In Figure 3, the distributions of PFDP are quite different in two scenarios. Most of the PFDP are small in the attenuation scenario while decodable PN-Codes appear much more frequently in mobility scenario. In attenuation scenario, when an error appears within the frame, there is a high tendency that the errors will continue to happen until the end of the frame. Its coherence time is believed to be much greater than that of mobility cases.

We then extend the investigation to mobility at walking and vehicular speed. In Figure 4, we find that the decodable PN-Codes appear more often in the walking scenario. Undecodable PN-Codes appear with higher probability with increasing speed since the coherence time will decrease when the speed increases.

IV. DETECTION ALGORITHM AND EVALUATION

Previous section shows a significant statistical differences in the distributions of PFDP between corrupted frames due to attenuation and mobility. In this section, we present an effective algorithm for detecting mobility-induced frame losses based on these findings. It is worth noting that we only consider two causes of frame loss-attenuation and mobility.

The input of our algorithm is a number of PFDPs of consecutive corrupted frames. The output is an inference of whether the losses are induced by attenuation or mobility. If the majority of PFDPs are equal to zeros, it is highly probable that the loss is due to attenuation. Otherwise, the loss is resulted from mobility. For convenience, we assume that the number of input frames (denoted by n) is odd. We will remove one entry automatically if n is even. Although the algorithm is simple, the performance is surprisingly good. We will demonstrate why it is the case. The terminologies we will use are listed in Table III.

The accuracy of the algorithm is defined as the probability of making a correct infernce in the worst case. We define α and β as shown in Table III because they are the parameters in the worst case for our algorithm. If n = 1, it is easy to see that the accuracy is $1 - \max\{\alpha, \beta\}$. If n > 1, the false positive probability of mobility inference is

$$\begin{split} &P\left(\left.r > \frac{n-1}{2} \mid M\right) = \sum_{i=(n+1)/2}^{n} P(r=i|M) \\ &= \sum_{i=(n+1)/2}^{n} P(Z|M)^i P(Y|M)^{n-i} \\ &< \sum_{i=(n+1)/2}^{n} P(Z|M)^i < \frac{\alpha^{(n+1)/2}}{1-\alpha}. \end{split}$$

TABLE III: Terminologies

Term	Significance		
M	event: frame losses are caused by mobility		
A	event: frame losses are caused by attenuation		
Y	event: the value of PFDP of a corrupted frame is NOT zero		
Z	event: the value of PFDP of a corrupted frame is zero		
n	the number of input PFDPs		
r	the number of zero PFDPs in input		
α	$\sup\{P(Z M)\}$		
β	$\sup\{P(Y A)\}$		

Similarly, we have

$$P\left(\left|r < \frac{n+1}{2}\right| A\right) < \frac{\beta^{(n+1)/2}}{1-\beta},$$

for the false positive probability of attenuation inference. Hence, $1 - \frac{\theta^{(n+1)/2}}{1-\theta}$, where $\theta = \max\{\alpha, \beta\}$, is the lower bound of the accuracy of our algorithm with respect to n. The accuracy of our algorithm with various values of n and θ is shown in Figure 5.



Fig. 5: The lower bound of accuracy with respect to n and θ .

When n increases, the accuracy converges rapidly to 1. If n is greater than 6, the accuracy is always above 90% and is insensitive to θ . However, we can see from Figure 1 that 0.4 is a conservative estimation of θ , which implies that setting n to 3 would be good enough for most cases. Thus, our algorithm has a superior performance even the input size n is small.

To verify the theoretical analysis, we apply our algorithm to the set of corrupted frames collected in section III. Each corrupted frame is labeled with its cause. We then compare the label with the algorithm output. The accuracy of our algorithm is 96.30%, 99.72% and 99.97% when n is 3, 5 and 7, respectively. The result is even better than the theoretical analysis because derived lower bound is not very tight.

V. RELATED WORKS

There are a number of measurement-based studies to examine the characteristics and causes of packet loss. Eckhardt et al. characterized packet errors and evaluated the effects of interference and attenuation (owing to propagation distance and obstacles) based on the transmission error and loss rate [8]. Willig et al. [9] used an IEEE 802.11-compliant radio to present bit error patterns obtained from an industrial environments. They showed that the Gilbert/Elliot model and its variant are useful tools for simulating bit errors on wireless links. Aguayo et al. [10] analyzed the packet loss in Roofnet network and stated that link errors stayed relatively uniform in static wireless networks. Jamieson et al. [11] used PHYindependent hints which based on Hamming distance in block decoding to develop a partial packet recovery system.

A work from Kaishun Wu et al. [12] presented some chip error patterns in static environments. It showed that recognizing these patterns could infer the channel condition. However, mobility is another cause of frame losses which occur in mobile channel. This work fills this important gap.

VI. CONCLUSION

In this paper, we reported sets of mobility-induced losses experiments in real environments. Different from other previous work, we accessed more deeply into PHY layer for investigation. We experimentally examined the nature of the error characteristics of mobility-induced frame loss at "chiplevel" in IEEE 802.15.4. Our experiments showed that these mobility-induced frame losses have unique chip error characteristics which are different from those due to attenuation in static environments. Finally, we proposed a detection algorithm for receivers to distinguish these frame losses. Our algorithm has a superior performance by providing a detection accuracy of over 96% in our evaluations.

ACKNOWLEDGMENTS

This research was supported in part by Hong Kong RGC Grant HKUST617908, China NSFC Grant 60933011, the National Basic Research Program of China (973 Program) under Grant No. 2006CB303000, the National Hi-Tech R&D Program of China (863 Program) under Grant No. 2008AA01A324, the National Science and Technology Major Project of China under Grant No. 2009ZX03006-001, and the Science and Technology Planning Project of Guangdong Province, China under Grant No. 2009A080207002.

REFERENCES

- K. Dantu, S. Kapadia, R. Sinha, and A. Helmy, "Modeling of mobilityinduced losses in MANETs (MILMAN)," in SIGCOMM, 2002.
- [2] C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Measurement-based models of delivery and interference in static wireless networks," in *SIGCOMM*, 2006.
- [3] "IEEE Standard for Information technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs)," IEEE Std 802.15.4–2006 (Revision of IEEE Std 802.15.4–2003), 2006.
- [4] E. Blossom, "Gnu software defined radio," http://www.gnu.org/software/ gnuradio.
- [5] M. Ettus, "The Universal Software Radio Peripheral or USRP, 2008."
- [6] T. Schmid, "GNU Radio 802.15. 4 En-and Decoding," UCLA NESL Technical Report, Tech. Rep., 2005.
- [7] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in SECON, 2004.
- [8] D. Eckhardt and P. Steenkiste, "Measurement and analysis of the error characteristics of an in-building wireless network," SIGCOMM, 1996.
- [9] A. Willig, M. Kubisch, C. Hoene, A. Wolisz *et al.*, "Measurements of a wireless link in an industrial environment using an IEEE 802. 11compliant physical layer," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 6, pp. 1265–1282, 2002.
- [10] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," in SIGCOMM, 2004.
- [11] K. Jamieson, "PPR: Partial packet recovery for wireless networks," in SIGCOMM, 2007.
- [12] Kaishun WU, Haoyu Tan, Hoi-Lun Ngan, Lionel M.Ni, "Chip Error Pattern Analysis in IEEE 802.15.4," HKUST Technical Report, Tech. Rep., 2009.