WiZig: Cross-Technology Energy Communication over a Noisy Channel

Xiuzhen Guo, *Student Member, IEEE,* Yuan He, *Member, IEEE,* and Xiaolong Zheng*, *Member, IEEE*

***Abstract*—The proliferation of IoT applications brings the demand of ubiquitous connections among heterogeneous wireless devices. Cross-Technology Communication (CTC) is a significant technique to directly exchange data among heterogeneous devices that follow different standards. By exploiting a side-channel like frequency, amplitude, or temporal modulation, the existing works enable CTC but have limited performance under channel noise. In this paper, we propose WiZig, a novel CTC technique from WiFi to ZigBee that employs modulations in both the amplitude and temporal dimensions to optimize the throughput over a noisy channel. We establish a theoretical model of the energy communication channel to clearly understand the channel capacity. We then devise an online rate adaptation algorithm to adjust the modulation strategy according to the channel condition. Based on the theoretical model, WiZig controls the number of encoded energy amplitudes and the length of a receiving window, so as to optimize the CTC throughput. We implement a prototype of WiZig on a software radio platform and a commercial ZigBee device. The evaluation shows that WiZig achieves a throughput of 153.85bps with less than 1 % symbol error rate in a real environment.**

***Index Terms*—Wireless Communication, Cross-technology, Protocol**

I. Introduction

THE ever-developing Internet of Things (IoT) brings the prosperity of wireless sensing and actuation applications [1] [2] [3]. In many scenarios, different IoT applications co-exist and deploy heterogeneous devices in the shared medium as well in the physical space [4] [5] [6] [7] [8]. Timely and efficient data exchange among those devices is therefore a fundamental requirement to ensure the usability, interoperability, and dependability of the IoT [9] [10] [11] [12] [13] [14]. Those devices operate on the same frequency band but follow different technologies, e.g. WiFi, ZigBee, and Bluetooth on the 2.4GHz ISM band. How to deliver data across different technologies remains an open problem.

Early works to address the above problem propose to build indirect connections among devices. Cloud [15] is a choice to gather data from different devices via the Internet. Another proposal is to connect the devices via a local gateway [16] [17]. Such a gateway is equipped with various radio interfaces, enabling it to communicate with devices of different wireless technologies. But the need of extra hardware prevents the pervasive uses, not to mention the high deployment and maintenance cost.

Direct communication among different technologies appears to be a more promising direction. Under this circumstance, Cross-Technology Communication (CTC) technique is proposed, which aims at directly exchanging mutually understandable data between two different technologies. This is a challenging task because a device cannot directly decode the standardized data from another technology. The existing proposals try to exploit free side-channels as information carriers. Regarding the wireless medium, a side channel typically exists in the following three dimensions: frequency, amplitude, and time [18]. For example, by intentionally control the absence or presence of data packets, the works in [19] and in [20] encode 0/1 bit in the amplitude dimension. Decoding the modulated bit, however, is highly susceptible to the channel noise. FreeBee [21] embeds CTC symbols into the transmission timing of beacons, which can be detected by the RSSI of the received signal. The achievable data rate of FreeBee is bounded by the beacon rate of WiFi devices. To tolerate the channel noise, FreeBee mainly adopts the folding technique to improve the decoding reliability. BlueBee [22] is a physical-level CTC method, which modifies the payload of Bluetooth to emulate the signal of ZigBee. Due to the imperfect emulated signal, BlueBee is prone to interference from the noise. When noise is present, both the amplitude and the phase of the received signal can have errors. BlueBee mainly adopts redundancy techniques like repeated transmissions, link layer coding, and multiple preambles to improve the transmission reliability.

Based on the above facts, we realize that amplitude-modulation-based CTC is easy to implement but prone to packet corruptions over a noisy channel. Temporal modulation is relatively more robust to noise, while its throughput is generally restricted by various technological specifications. Can we achieve high-throughput CTC over a noisy channel? This is a crucial problem with great practical significance.

We explore the answer to the above question in this work and propose WiZig, a practical CTC protocol from WiFi to ZigBee. By regulating the transmission power levels, a WiZig sender encodes one or more bits by means of multiple energy levels. A WiZig receiver detects the energy levels of the received signal sequences and then decodes data. The key point of WiZig is to explore the ability and design space of using packet energy as a side channel to realize CTC. WiZig aims to show the feasibility of energy-based CTC, reveal the importance to deal with noise channel, and demonstrate the effectiveness of our design. Some general techniques, e.g. the

Xiuzhen Guo and Yuan He are with the School of Software and TNLIST, Tsinghua University, P.R. China. Xiaolong Zheng is with the School of Computer Science, Beijing University of Posts and Telecommunications, P.R. China.

E-mail: guoxz16@mails.tsinghua.edu.cn, heyuan@mail.tsinghua.edu.cn, zhengxiaolong@bupt.edu.cn

*Yuan He and Xiaolong Zheng are the co-corresponding authors.

rateless codes, can also be applied to WiZig to further improve the anti-noise ability. In the design of WiZig, we address both theoretical and practical challenges of CTC. The main contributions of this work are summarized as follows.

- We present a general framework of CTC from WiFi to ZigBee, which jointly employs modulation techniques in both the amplitude and temporal dimensions. Based on this framework, we establish a theoretical model to clearly describe the relationship between BER (Bit Error Rate) and SNR (Signal to Noise Ratio).
- We devise the WiZig protocol, which mainly consists of two modulations and an online rate adaptation algorithm. The rate adaptation algorithm optimizes the throughput of CTC against dynamic noise, according to the theoretical foundation we build.
- We implement WiZig on a software radio platform and a commercial ZigBee device. We evaluate the performance of WiZig using different experimental settings. The throughput of WiZig is 153.85bps with less than 1% symbol error rate in the real office environment.

The rest of this paper is organized as follows. Section II discusses the related work. Section III verifies the feasibility of energy communication and Section IV presents the design overview of WiZig. Section V illuminates the theoretical fundamentals of this work. Section VI presents the modulation of WiZig and an online rate adaptation algorithm. In Section VII, we evaluate the performance of WiZig. In Section VIII, discussions about cross-technology communication have been given. We conclude this work in Section IX.

II. Related work

Realizing interconnection of all the smart things has become an inevitable trend in the era of IoT. A cloud solution builds an indirect communication between two different technologies, which requires devices to have access to the Internet [23] [24] [25]. Besides, the exchange has considerable transmission delay between the sensor to the server. Gateway can also enable the communication between WiFi and ZigBee [16] [17]. It must have two interfaces, one is WiFi and the other is ZigBee. The gateway shuttles traffic between two interfaces (and two networks as well). A CTC message coming from the WiFi device has to detour to the gateway. The gateway must translates this message first before it forwards the message to the ZigBee device via the ZigBee interface. Besides, the dedicated gateway increases the deployment and maintenance cost.

Recently, Cross-Technology Communication (CTC) technique is proposed to enable direct communication between heterogeneous wireless devices [26]. FreeBee [21] embeds symbols into beacons by shifting their transmission timings. The date rate of FreeBee is limited by the beacon rate which is usually 102.4ms per beacon for commercial WiFi devices, however. Other works propose the energy profile as a new information carrier to exchange the data without a gateway. Esense [19] is the first work that uses energy sampling realizing data transmission from the WiFi to the ZigBee device. It aims at building an alphabet of implicit data by using the packet duration information. HoWiEs [27] improves the Esense mechanism and uses the combination of predefined packets sizes form the alphabet to realize delivery. Gap Sense [28] transmits legacy packets with a customized preamble and constructs sequences of energy pulses. Then the receiver senses the gaps between the energy pulses to decode the data. The absence or presence of packets is used to transmit data for interconnection between heterogeneous wireless devices [20]. [29] realizes CTC from BLE to WiFi by using energy burst patterns to encode symbols on overlapping channels.

Since the communication channel is intrinsically noisy, it is not a trivial to reduce the harmful impact of noise and realize efficient communication. The impact of noise on throughput is analyzed in [21] but how to resist random and uncontrolled noise is not clear. In [19] [20] [27] [28], it has been revealed that energy profile is promising as a new channel to realize direct communication between heterogeneous wireless devices. Those works only exchange data via the energy channel, however, it is difficult to apply them in the practical noisy environments.

Compared with FreeBee or BlueBee, WiZig presents a new medium (namely packet energy levels) to convey CTC symbols. Compared with other energy-based CTC works, such as Esense and Gap sense, WiZig improves the efficiency and reliability of CTC in a noisy channel, by using the two modulation techniques and the online rate adaptation algorithm.

III. Preliminary study

In this section, we investigate the feasibility of using energy as a CTC medium. We also study whether there is enough coding space to achieve energy communication.

A. The feasibility of energy communication

Energy can be used as a medium of communication. We can leverage the absence or presence of packets to transmit data and realize connection between heterogeneous wireless devices, which can be considered as Amplitude Modulation (AM). The BER is the bit error rate when the receiver misjudges a single sample point. The SER is the symbol error rate when the receive misjudges a symbol in a receiving window. The relationship between BER and SNR can be calculated by

$$p = \frac{1}{2} erfc(\sqrt{\frac{r}{4}}) \tag{1}$$

where p is BER, r is SNR, and the complementary error function is defined as $erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^{+\infty} e^{-u^2}\, du$. Then we can deduce:

$$r = 4(erfc^{-1}(2p))^2 \tag{2}$$

SNR also can be written as

$$r = 10lg\frac{P_{signal}}{P_{noise}} = 10lgP_{signal} - 10lgP_{noise} \tag{3}$$

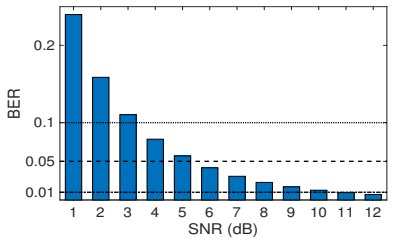


Fig. 1. The relationship of BER and SNR

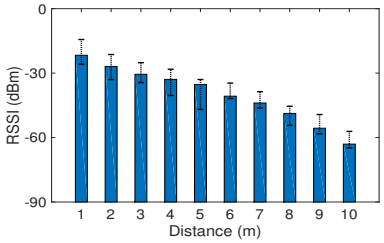


Fig. 2. RSSI values of commercial AP with the increase of distance between the sender and the receiver

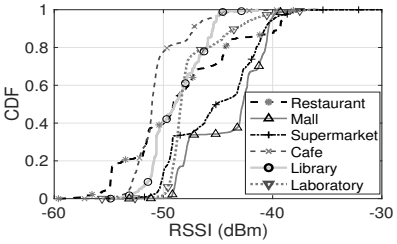
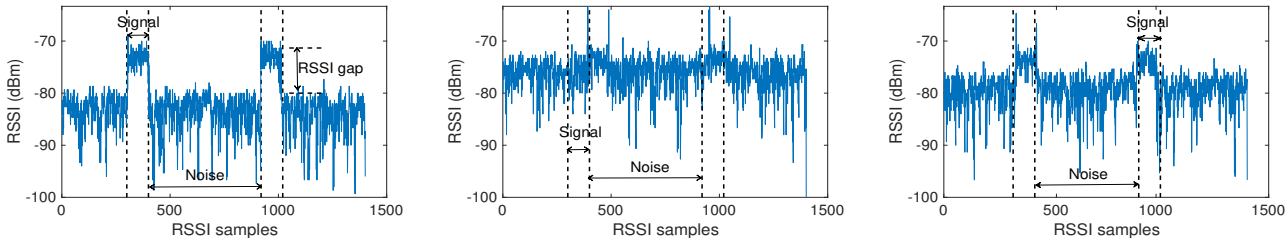


Fig. 3. The CDF curves of RSSI values in different environments



(a) SNR=7dB (b) SNR=1dB (c) SNR=4dB

Fig. 4. RSSI sequences with different SNR

where P_{signal} is the power of signal and P_{noise} is the power of noise.

A concrete example of the relationship between BER and SNR is shown in Fig. 1. We find that different SNR thresholds are required to satisfy different BER requirements. For example, for the BER of 0.01, SNR needs to be 10dB. But for the BER of 0.1, SNR only needs 3dB. Given a BER requirement, it is only needs SNR to be higher than a threshold. A strictly high SNR is not necessary.

In practice, SNR is likely to be much higher than the threshold for a BER requirement. If we control the transmission power to satisfy the minimum SNR, there will be a large unexploited coding space. We can consider to encode multiple bits into multiple energy levels. To fully exploit channel resource, energy levels can be divided as many as possible. The maximum number of energy levels relies on the strength of the received signal. In this paper, we use the Received Signal Strength Indication (RSSI) to translate the received signal power in dBm as $RSSI(dBm) = 10 * log_{10}\frac{P(mw)}{1mw}$, where P is the power of the received signal. The RSSI gap between adjacent energy levels needs to satisfy SNR requirement and guarantee required BER.

How many energy levels can be divided at most in theory? To answer this question, we first investigate how large the RSSI gap between two adjacent energy levels can be. We control one AP as the transmitter and three other APs as noise sources to observe the effectiveness of energy coding under different RSSI gaps. First, we turn on one noise AP to generate jamming and the SNR is close to 7dB. The RSSI sequence is shown as Fig. 4(a). We can find that the signal and the noise can be separated clearly because the RSSI gap between the signal and the noise is large. In this condition, the energy space seems to have not been fully utilized because there is still a large gap between the signal and the noise, however. Second, we turn on three APs to generate noise and the SNR is close to 1dB. From the RSSI sequence in Fig. 4(b), we can find that the RSSI gap between the signal and the noise is too small. As a result, we cannot judge the signal correctly and have a high BER in this case. Third, we turn off one AP and leave two APs to generate a SNR of 4dB. The resultant RSSI sequence is shown in Fig. 4(c). We can find the SNR value in this case is proper because the RSSI gap is enough to separate the signal from the noise but not too large to waste the coding space. The proper RSSI gap to separate two energy levels is decided by SNR. Given a required BER, a minimum RSSI gap $\triangle RSSI$ can be calculated to obtain an appropriate SNR.

B. Space of energy coding

From the above analysis, the maximum number of energy levels that can be divided in theory is:

$$N = \frac{RSSI_{max}}{\triangle RSSI} \tag{4}$$

where $RSSI_{max}$ is the max signal strength at the receiver and $\triangle RSSI$ is the minimum RSSI gap between two adjacent energy levels. From Eq. (4), we can find that the number of available energy levels depends on both $RSSI_{max}$ and $\triangle RSSI$. Once given a BER, the SNR requirement and $\triangle RSSI$ are determined. So we measure the RSSI of commercial devices to investigate the coding space of the multi-energy levels in controlled and real environments.

First, we study whether there is enough coding space when the distance between the sender and the receiver varies. In the experiment, we choose the TL-WR742N AP, whose the maximum transmission power is 100mW. We control the AP transmits packets with the power setting corresponding to

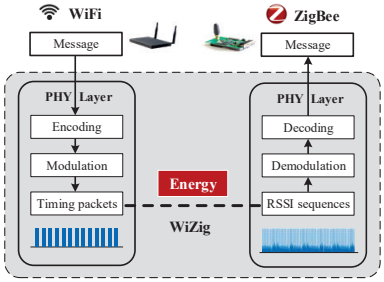


Fig. 5. The framework of WiZig

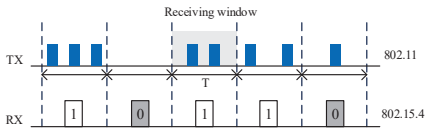


Fig. 6. The basic schematic diagram of WiZig

"high". A TelesB node receives the signal and records the RSSI values. We vary the distance between the sender and the receiver from 1 to 10 meters with a step size of one meter. The experiment result is shown in Fig. 2. We can find that the RSSI decreases when increasing the distance between the AP and the node, which means the number of available energy levels decreases. There are still multiple energy levels available to improve the data rate, however. We assume the strength of noise base is -80dBm and the RSSI gap between two adjacent energy levels is 5dBm. When the distance between the AP and the node is 9m, the RSSI decreases to -60dBm and there are four energy levels available, according to Eq. (4).

We also study the RSSI distribution in various real environments. We conduct RSSI measurements in six different environments, including restaurant, library, cafe, mall, laboratory, and supermarket. The distance from the WiFi AP to the ZigBee node is set at 5m. The ZigBee node records RSSI values for ten minutes. The measurement results are shown in Fig. 3. No matter in which environments, there are nearly 80% of RSSI values larger than -50dBm.

In summary, we can find large unexploited coding space in real environments. The preliminary study shows it is possible to use multiple energy levels to improve the data rate. The practical channel condition is noisy and dynamic. How to decide the appropriate number of used energy levels accordingly in an online manner still needs further study.

IV. Overview of WiZig Framework

In this section, we present the framework of WiZig, a novel CTC technique that enables direct communication among wireless devices with different PHY/MAC standards. Fig. 5 presents the overview of WiZig. Without losing generality, we use the transmission from WiFi to ZigBee as an example. The WiZig sender modulates the presence of WiFi packets as the symbol 1 and the absence of WiFi packets as the symbol 0. After encoding and modulation, the sender transmits these WiFi packets without modifying the PHY layer. On the receiver side, the WiZig receiver detects the RSSI sequences in the overlapping communication channel that is saturated with WiFi packets. Then the receiver decodes the data according to RSSI values within a receiving window.

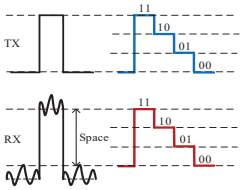


Fig. 7. The schematic diagram of energy communication when communication with multiple energy channel is noisy

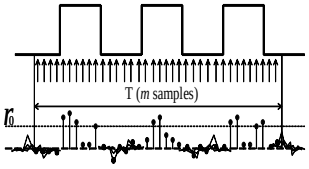


Fig. 8. Energy communication when levels (4 energy levels in the example)

The basic communication scheme of WiZig is shown in Fig. 6. The sender transmits packets and the receiver detects the RSSI values within a receiving window T. If the receiver detects a proportion of packet sample points, then it will decode the symbol as 1. Otherwise, the receiver decodes the symbol as 0. The channel resource is not fully exploited if only transmitting one symbol in a fixed receiving window as shown in our preliminary study. The sender increases the number of energy levels to encode multiple symbols.

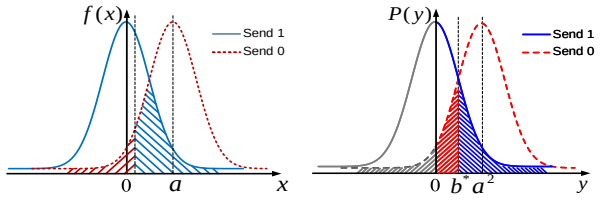
We use the communication with four energy levels as an example shown in Fig. 7. If the sender transmits symbols with different powers and encodes these symbols as 00, 01, 10, 11, the receiver detects RSSI sequence with four different levels and realizes two-bit symbol communication within a receiving window. We adopt the encoding of OOK when there are two energy levels. When we use multiple energy levels, we can use the encoding of PAM. In order to improve the reliability, Gray code can also be used.

The energy communication channel is intrinsically noisy, however. We model the energy channel and theoretically analyze the relationship among the BER, SER, SNR, and the number of energy levels. Based on the theoretical model, we carefully design our modulation/demodulation strategies and set parameters to reduce the SER under a noisy channel.

V. Theoretical Analysis

The RSSI samples that the receiver receives are corrupted due to the intrinsic noise in wireless channels as shown in Fig. 8. There are m samples in a receiving window. Each sample has a RSSI value corresponding to the energy in the channel. A sample will be detected as a packet sample if its RSSI value is higher than a threshold. A packet sample may be misjudged as noise because the channel noise corrupts its RSSI value to belower than the threshold, however. To understand the potential of the energy communication channel and improve its ability of resisting interference, we first build the model of the energy communication channel and describe the constraint relationship between the BER and SNR. Then we consider all the samples in a receiving window to describe the relationship between SER and SNR.

First, we build a model that describes the relationship between the BER and the SNR for one sample. We suppose



(a) Signal amplitude (b) Signal power

Fig. 9. The curves of one-dimensional probability density

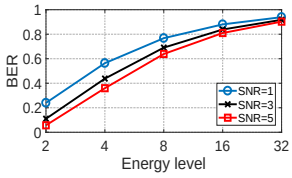


Fig. 10. The relationship between BER and energy level

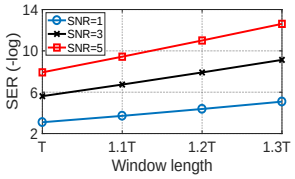


Fig. 11. The relationship between SER and window length

the transmitted signal is $u(t) = a\cos\omega_c t$. The Gaussian noise can be denoted as $n(t) = n_c(t)\cos\omega_c t - n_s(t)\sin\omega_c t$ and $n(t), n_c(t), n_s(t) \sim N(0, \sigma^2)$. $x(t)$ is the received signal, which is the superposition of the transmitted signal and the noise.

$$x(t) = \begin{cases} n_c(t)\cos\omega_c t - n_s(t)\sin\omega_c t, & send\ 0 \\ (a + n_c(t))\cos\omega_c t - n_s(t)\sin\omega_c t, & send\ 1 \end{cases} \quad (5)$$

So $x(t)$ is also Gaussian. Its mean value is a (send 1) or 0 (send 0), and the variance is σ^2. Its one-dimensional probability density is $f_0(x)$ (send 0) or $f_1(x)$ (send 1) as follows.

$$f_0(x) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{x^2}{2\sigma_n^2}} \quad (6)$$

$$f_1(x) = \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(x-a)^2}{2\sigma_n^2}} \quad (7)$$

We plot the curves of $f_0(x)$ and $f_1(x)$ in Fig. 9(a). The signal amplitude $x(t)$ follows Gaussian distribution and satisfies that $x(t) \sim N(a, \sigma^2)$, where a is the mean value and σ^2 is the variance. We denote the signal power is $y = x^2(t)$, which follows (non-central) Chi-squared distribution with one degree of freedom shown as $y \sim \chi^2(n, \lambda) = \chi^2(1, \lambda)$ and $\lambda = \frac{a^2}{\sigma^2}$. The probability density $P(y)$ is

$$P(y) = \frac{1}{2\sigma^2}\left(\frac{y}{\lambda\sigma^2}\right)^{-\frac{1}{4}} e^{-\left(\frac{y}{2\sigma^2}+\frac{\lambda}{2}\right)} I_{-\frac{1}{2}}\left(\sqrt{y\lambda}\right) \quad (8)$$

The SNR range in our evaluation is from 5dB to 36dB, we can obtain that

$$5 \le SNR = \frac{a^2}{2\sigma^2} = \frac{\lambda}{2} \le 36 \quad (9)$$

So the range of λ satisfies that

$$10 \le \lambda \le 72 \quad (10)$$

We plot the curves of probability density $P(y)$ with different SNR values as shown in Fig. 12. We find that the curve of $P(y)$ is more similar to Gaussian shape with the increase of SNR. Moreover, λ is larger than σ^2 as shown in Eq. 9 and Eq. 10. Therefore, the (non-central) Chi-squared distribution of signal power tends towards Gaussian distribution. We leverage the simplified Gaussian model to further deduce the relationship between SER and SNR. We denote the decoding threshold as b, and the decision rule based on signal power x^2 is as follows:

(1) $x^2 > b$, the sample point is decoded as 1.

(2) $x^2 \le b$, the sample point is decoded as 0.

We use $P(0|1)$ represents the probability that the receiver decodes 1 as 0 and $P(1|0)$ represents the probability that the receiver decodes 0 as 1. Denote the probability that the sender sends 1 and 0 as $S(1)$ and $S(0)$, respectively. If $S(1) = S(0) = \frac{1}{2}$ and $b^* = \frac{a^2}{2}$, the total error rate of the receiver denoted by p is shown in the shaded area in Fig. 9(b) and it can be calculated as follows.

$$p{=}S(1)P(0|1){+}S(0)P(1|0){=}\frac{1}{2}erfc(\sqrt{\frac{r}{4}}), r{=}\frac{a^2}{2\sigma_n^2} \quad (11)$$

Eq. (11) reveals the relationship between the BER and the SNR for one RSSI sample. We assume the receiver gets m samples in a receiving window. m_0 is a predefined threshold used for sampling. If the number of misjudged sample points in a receiving window is less than m_0, the receiver can decode the symbol correctly. m_0 is usually the half of the total number of sample points in a receiving window. Then the SER P_e is calculated by

$$P_e = \sum_{q=0}^{m_0} C_m^q (1-p)^q p^{m-q} \quad (12)$$

If M energy levels are used, the BER and the SER can be represented as Eq. (13) and Eq. (14), respectively.

$$p_m = (1 - \frac{1}{M})erfc(\sqrt{(\frac{3}{M^2-1})\frac{r}{4}}), r = \frac{a^2}{2\sigma_n^2} \quad (13)$$

$$P_{em} = \sum_{q=0}^{m_0} C_m^q (1-p_m)^q {p_m}^{m-q} \quad (14)$$

We use p_m and P_{em} denote the BER and the SER with multiple energy levels.

We can observe the relationship among BER (p_m), SNR (r), and the number of energy levels (M) in Fig. 10. First, the BER increases with the decrease of the SNR. For example, if the number of energy levels is 2, the BER is 0.0569 when the SNR is 5dB. The BER increases to 0.2398 when the SNR is 1dB. Second, the BER increases when the number of used energy levels increases. For example, when SNR is 3dB, the BER is 0.1103 if the number of energy levels is 2. The BER increases to 0.6906 if we use 8 energy levels. Moreover, if we use 32 energy levels, the BER increases to 0.9175. Therefore, the BER is much higher when we use multiple energy levels. In such a case, we can adjust the length of receiving window to reduce the error rate. We extend the receiving window length

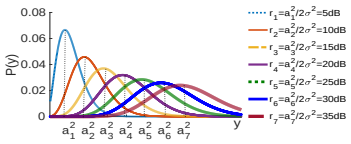


Fig. 12. Probability density $P(y)$ with different SNR values

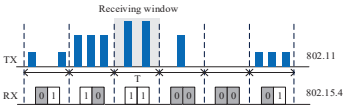


Fig. 13. The schematic diagram of amplitude modulation

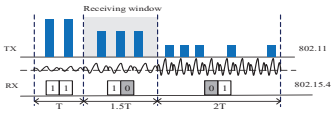


Fig. 14. The schematic diagram of temporal modulation

using the step-by-step method. The receiving window length is calculated by

$$K = T + n * \frac{T}{10}, n = 0, 1, ..., 10 \tag{15}$$

where the K is the receiving window length, T is the initial receiving window length, and $\frac{T}{10}$ is the adjusting step. The receiving window length K can influence the adopted energy levels. The longer the receiving window is, the stronger the noise-tolerant ability is. The SER decreases with the increasing length of receiving window, as shown in Fig. 11 Note that $-log(SER)$ increases monotonically. For example, if the SNR is 1dB, the SER is 0.0453 when $n = 0$. The SER decreases to 0.0062 when $n = 3$.In short, we can adjust the receiving window length to adapt the requirement of SER.

Notation	Description	Type
r_0	The original SNR	Measured
r_1	The new SNR	Measured
M_0	The original number of energy levels	User defined
M'	The new number of energy levels	User defined
K_0	The original receiving window length	User defined
K'	The new receiving window length	User defined
d_s	The SER requirement	Fixed (0.01)
d_r	The SNR difference threshold	Fixed (3dB)
T	The initial receiving window length	Fixed (5ms)
T_{max}	The maximum receiving window length	Fixed (10ms)
N_{min}	The minimum number of energy levels	Fixed (2)
N_{max}	The maximum number of energy levels	Measured
$RSSI_{max}$	The maximum RSSI value	Measured
$\triangle RSSI$	The RSSI required gap between two adjacent energy levels	Measured

TABLE I
PARAMETERS IN WIZIG

VI. Modulation and Demodulation

A. Amplitude Modulation

We propose amplitude modulation that increases the number of energy levels to improve the data rate. We can encode multiple symbols in a receiving window. As shown in Fig. 13, the WiZig sender transmits packets with three different power levels to provide three energy levels that can be encoded as 01, 10, 11. And the absence of packet is encoded as 00. The WiZig receiver samples the RSSI in the channel and it detects four different energy levels. Then the receiver decodes the 2-bit data.

We take four energy as example. Suppose there are m packet samples $(s_1, s_2, ..., s_m)$ in a receiving window T, the amplitude modulation/demodulation strategies are as follows. (1) The WiZig sender transmits packets with three different power levels W_0, W_1 and W_2. Without losing generality, we assume $W_2 > W_1 > W_0$. Then the sender encodes the packets with power of $W_2/W_1/W_0$ as symbol 11/10/01. The absence of packet is encoded into 00. (2) The receiver detects signal strength on the overlapping channel and obtains the RSSI sequences. Based on the number of used energy levels, the receiver sets three point deciding thresholds th_{e0}, th_{e1} and th_{e2} which satisfy $th_{e2} > th_{e1} > th_{e0}$. For each packet sample point, we denote its logic value is y and y is decided as

$$y = \begin{cases} 11, & RSSI \geq th_{e2} \\ 10, & th_{e1} \leq RSSI < th_{e2} \\ 01, & th_{e0} \leq RSSI < th_{e1} \\ 00, & RSSI \leq th_{e0} \end{cases} \tag{16}$$

(3) For each receiving window, we set a number threshold th_m. When the number of energy levels is M and the number of sample points in a receiving window is m, the value of th_m is $th_m = \frac{m}{M}$. Using this setting, WiZig decodes a CTC symbol to the corresponding energy level, which has the largest number of sample points (which clearly will exceed the threshold th_m).

B. Temporal Modulation

We propose temporal modulation to make the WiZig more resilient to the dynamic noise. Temporal modulation can alleviate the BER increase caused by using multiple energy levels. WiZig extends the length of a receiving window to reduce the error rate, when the channel quality is poor. Similarly, WiZig shortens the length of a receiving window to improve the date rate, when the channel quality is good. Without losing generality, we also use the modulation with four energy levels as an example as shown in Fig. 14. The WiZig sender transmits packets and the length of the receiving window is adjusted according to the channel noise. The stronger the noise is, the larger the window length is.

The temporal modulation/demodulation strategies are as follows. (1) The WiZig receiver samples the signal strength in the channel and estimates the channel noise. If the current SNR becomes higher, then WiZig shortens the receiving window length to improve the data rate. Similarly, if current SNR decreases, then WiZig extends the receiving window length to tolerate errors. We adjust the receiving window as Eq. (15) . (2) After determining the length of a receiving window, the sender modulates the data and the receiver decodes the energy symbols, following the strategies in amplitude modulation.

C. Online Rate Adaptation

The goal of our online rate adaptation algorithm is to optimize the throughput by adjusting the modulation strategies

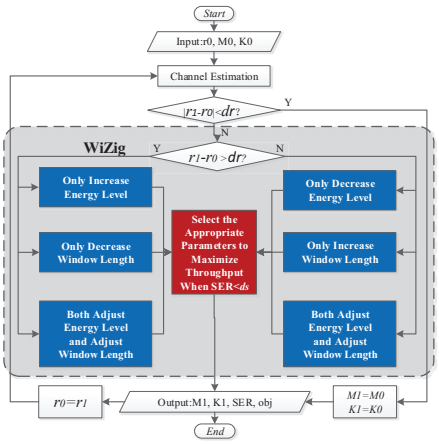


Fig. 15. The flow chart of online rate adaptation algorithm

Fig. 16. The experiment platform: a prototype of WiZig, one USRP acts as WiFi sender, another USRP generates Gaussian noise and TelosB mote is used as ZigBee device

according to the current channel condition. We discretize the values of the receiving window length and the number of energy levels to get the appropriate solution. The throughput is:

$$\max \ Th(M,K) = \frac{\log_2 M}{K} * (1 - P_e) \tag{17}$$

$$s.t. \quad \begin{cases} M = 2, 4, 8, 16 \\ K = T + n * \frac{T}{10}, n = 0, 1, ..., 10 \end{cases} \tag{18}$$

where M is the number of energy levels, K is the length of receiving window. P_e is the SER and $\log_2 M$ represents the number of data symbols in a receiving window. Considering the computing overhead, we search a limited feasible set of (M, K) to get the near-optimal solution. The side condition (feasible set of (M, K)) of this optimization problem is shown in Eq. (18). When the SNR changes from r_0 to r_1, our online rate adaptation algorithm will adjust the parameters from (M_0, K_0) to (M', K') to adapt to the varied channel condition.

The process of our online rate adaptation is illustrated in Fig. 15. There are three different kinds of parameters in WiZig, including fixed parameters, measured parameters, and user defined parameters of the rate adaptation algorithm. The specific description and type are as shown in Table I. We use SNR variation as the heuristic of rate adaptation. The receiver listens to the channel and measures the SNR variation at specified intervals (2s in our implementation). If the difference between the new SNR and the previous SNR exceeds a threshold, we adjust the number of energy levels and/or the receiving window length to optimize the throughput. The threshold of SNR difference is set at 3dB. Multiple iterations may be necessary to achieve a desired SNR at the receiver if the channel is unstable and noisy. The specific working flow of the rate adaptation algorithm is as follows.

(1) The receiver listens to the channel and estimates the values of r_1, $RSSI_{max}$, and $\triangle RSSI$. In this way, the receiver calculates the value of N_{max} by Eq. (4).

(2) If the difference between r_1 and r_0 is less than the threshold of d_r, which means the channel condition doesn't change too much, then the original parameters will be used without any modification.

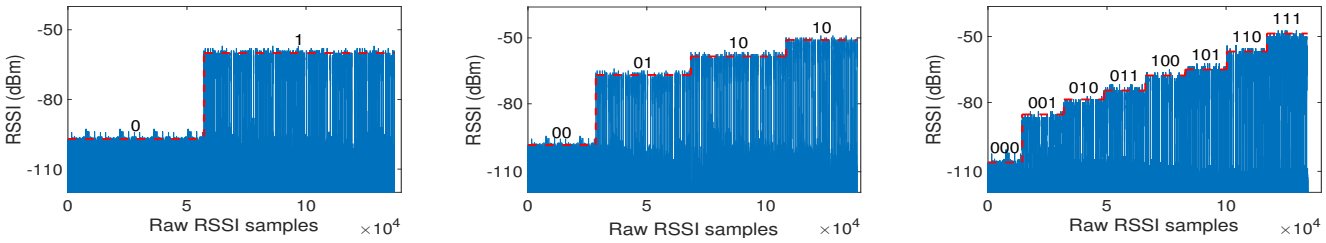
(3) If the r_1 is higher than the r_0, it means the channel becomes better. There are three possible strategies to improve the throughput. First, we keep the receiving window length unchanged and only increase the number of energy levels to M', where $M' = argmax(Th)$ and $M' \leq N_{max}$. Second, we keep the number of energy levels unchanged and only decrease the window length to K', where $K' = argmax(Th)$ and $K' \geq 0$. Third, we adjust the number of energy levels and the receiving window length at the same time to (M', K'), where $(M', K') = argmax(Th)$.

(4) If the r_1 is lower than the r_0, it means the channel becomes worse. There are also three possible strategies to decrease the SER to satisfy the requirement. First, we keep the receiving window length unchanged and only decrease the number of energy levels to M', where $M' = argmax(Th)$ and $M' \geq N_{min}$. Second, we keep the number of energy levels unchanged and only increase the receiving window length to K', where $K' = argmax(Th)$ and $T' \leq T_{max}$. Third, we adjust the number of energy levels and the receiving window length at the same time to (M', K'), where $(M', K') = argmax(Th)$.

(5) Three different strategies can obtain three different throughput. We select the appropriate parameters of the number of energy levels and the receiving window length to maximize the throughput when $SER < d_s$, according to $Th_{max} = max(Th(M', K_0), Th(M_0, K'), Th(M', K'))$. The appropriate parameters are $(M', K') = argmax(Th(M', K_0), Th(M_0, K'), Th(M', K'))$.

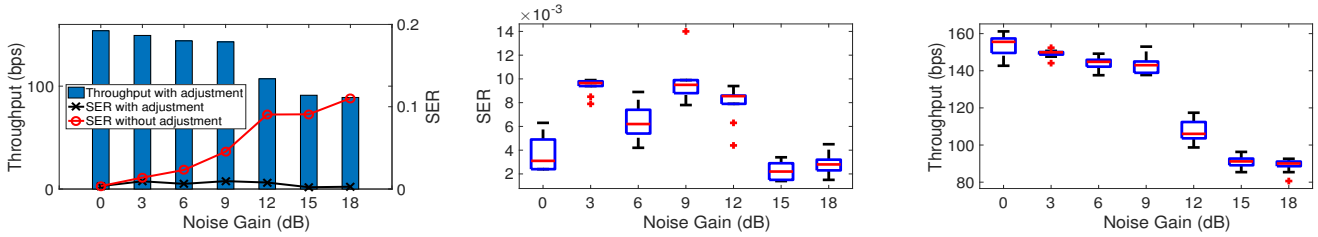
(6) The receiver obtains the new parameters (M', K') and sends back to the sender using the method proposed in FreeBee [21]. Then the sender and the receiver continue the transmissions using new parameters.

Our rate adaptation algorithm searches a limited feasible parameter space to find an appropriate solution. Exploring the full parameter space does obtain the optimal solution. In practice, the channel variation is relatively stable and it is not common to have sudden and sharp SNR variation. Hence, we take a conservative design principle and propose the heuristic algorithm. We gradually change our parameter to avoid the possible system instability caused by responding to the temporally instable SNR which varies sharply in a short time. We admit that the parameters obtained by our rate adaptation algorithm may be not the optimal solution,



(a) Two energy levels (b) Four energy levels (c) Eight energy levels

Fig. 17. Raw RSSI sequences with different number of energy levels, the receiving window length is 7.5ms



(a) Throughput and SER with and without adaptation algorithm (b) SER (c) Throughput

Fig. 18. SER and throughput under different noise intensities

but our solution can approximate the optimal solution and improve throughput. If the SNR indeed becomes very good, our algorithm can also gradually converge to the optimal solution.

For the purposes of CTC, WiZig actually builds a side channel based on packet energy to convey CTC symbols. Note that WiZig preferentially utilizes the existing WiFi traffic. If there are enough WiFi packets queueing, the transmission of WiZig doesn't introduce new traffic. If there are not enough WiFi packets, WiZig needs to inject WiFi packets as the carrier. Such communication is operated just similarly with the regular communication. The only influence WiZig may bring sometimes is the additional traffic. If we look at how it interacts with the existing WiFi networks and ZigBee networks, the conventional MAC will be able to deal with everything. In this regard, there is not much difference between adding a new link of conventional wireless and adding a WiZig link into the network.

VII. Evaluation

A. Evaluation Settings

We implement a prototype of WiZig on TelosB and a USRP platform as shown in Fig. 16. Our prototype uses an USRP N210/GNU Radio to generate WiFi packets following IEEE 802.11 standards. It transmits 3000 packets with length of 250 bytes per second. Spectrum overlap is the prerequisite of energy communication. In our implementation, we choose the 802.11 channel 6 and the 802.15.4 channel 17 to construct the energy communication channel. We control the transmission power levels to generate different energy levels. The difference between two adjacent levels of transmission gain is 5dB because the corresponding RSSI values are enough to be distinguished. An USRP/N210 device transmits WiFi packets without CSMA and the maximum transmission power gain is 20dB. We use another USRP/N210 to generate Gaussian noise with different power levels. The distance between the USRP based Wi-Fi sources and the ZigBee device is 5m, the distance between the USRP based Gaussian noise generator and the ZigBee device is 2m. We conduct the experiment in the laboratory and there are several APs in the vicinity of the frequency bands. The decoding threshold th_e is decided by the difference of the received RSSI values between the noise and particular energy levels. The RSSI sampling rate of TelosB node is 36KHz. The initial receiving window is 5ms. When there are two energy levels, the decision threshold th_m is 9, nearly half of the total WiFi samples in a receiving window. We assume the channel noise is Gaussian noise and the SNR range is from 5dB to 36dB.

B. SER and Throughput

First, the USRP sender transmits packets and the transmission gain of the USRP is 20dB. The TelosB receiver samples the RSSI sequence, the sampling rate is 36kHz and the receiving window length 7.5ms. The receiving window length can be adjusted by Eq. (15). The decoding threshold th_e is -70dBm. Fig. 17(a) shows the raw RSSI sequence sampled by a TelosB mote. When the energy level is 2, the SER is 0 and the throughput is 40.56bps. Due to the hardware capability and software processing delay, the sender is not always transmitting. If the sender is always transmitting, the throughput can be 133.3bps in theory.

Second, the transmission gain is set as 15dB, 20dB, and 25dB, the raw RSSI values sampled by a TelosB mote are shown in Fig. 17(b). The decoding threshold th_{e0}, th_{e1}, and

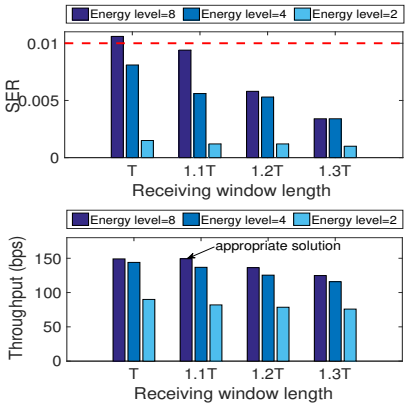


Fig. 19. SER and throughput with different parameters when noise gain is 3dB

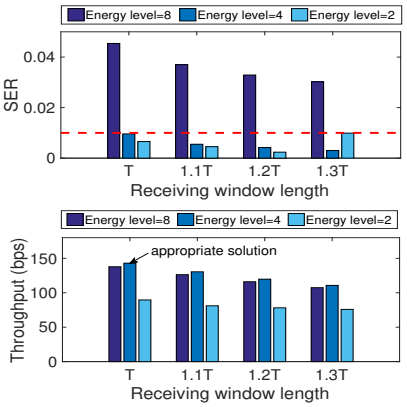


Fig. 20. SER and throughput with different parameters when noise gain is 9dB

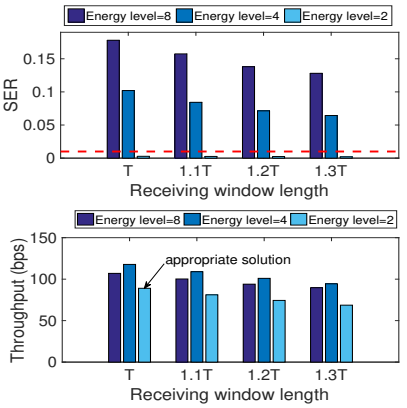


Fig. 21. SER and throughput with different parameters when noise gain is 18dB

th_{e2} are -80dBm, -65dBm, and -55dBm. Obviously, we can find that there are three different levels of the spikes which are encoded as 01, 10 and 11. The SER is also 0 and the throughput is 81.22bps.

Third, the transmission gain is set as 0dB, 5dB, 10dB, 15dB, 20dB, 25dB, and 30dB, the raw RSSI values are shown in Fig. 17(c). The threshold th_{e0}, th_{e1}, th_{e2}, th_{e3}, th_{e4}, th_{e5}, and th_{e6} are -86dBm, -80dBm, -74dBm, -68dBm, -62dBm, -56dBm, and -50dBm. Obviously, there are seven different levels of the spikes which are encoded as 001, 010, 011, 100, 101, 110, and 111. The SER is 0.0036 and the throughput is 118.4bps.

C. Benefit of online rate adaptation

First, we study the performance of WiZig under varied noise intensities. We increase the channel noise intensities and use the original setting with fixed parameters. The number of energy levels is 8 and the receiving window length is 5ms. The gain of the jamming USRP is set as 0dB, 3dB, 6dB, 9dB, 12dB, 15dB, and 18dB. The distance between TelosB node and noise source is twenty centimeters. We can find that SER increases sharply as shown in Fig. 18(a). When the noise gain is 0dB, the SER is only 0.0036. When the noise gain is 3dB, the SER is 0.0139 and larger than 0.01, however. The SER value of 0.01 is the maximum SER that communication system can tolerant. Furthermore, when the noise gain is 18dB, the SER increases to 0.1103 which is more than ten times of 0.01. Our adaptation algorithm reduces the SER and increases the throughput by reducing the number of energy levels and extending the receiving window length to overcome the noisy channel. When the noise gain is 3dB, the SER with the rate adaptation algorithm is 0.0094, which is lower than 0.01 and satisfies the SER requirement. The corresponding throughput is 149.3bps, which is a little lower than original 153.85bps. The SER is always lower than the SER required 0.01. The throughput decreases subtly with the increase of the noise intensity when use our online adaptation algorithm. When the noise gain is 18dB, the SER is 0.0028, which decreases by near 40 times than the SER without adaption algorithm. Our algorithm adaptively changes the number of energy levels from 8 to 2 to be resilient to the noisy channel. Thanks to the adjustment, WiZig achieves a throughput of 89bps. The length of the receiving window is extended to 1.1T to guarantee the SER lower than 0.01.

For each noise intensities, we conduct 10 experiments to study the average performance of our method. As shown as Fig. 18(b) and Fig. 18(c), we can find that the values of SER are always lower than 0.01, except 0.014 when noise gain is 9dB. The throughput decreases with the increase of noise intensities. It is also stable and the fluctuation of throughput is smallest when noise gain is 3dB.

Our rate adaptation algorithm finds the appropriate parameters of the number of energy levels and the receiving window length (M, K) when channel is noisy. We select several parameter combinations to observe the corresponding throughput and the SER under three different noise intensities. The number of energy levels varies from 8 to 2 and the receiving window length varies from T (5ms) to 1.3T. We take that the noise gain is 3dB, 9dB, and 18dB as examples. The results show that adjusting the parameters obtains appropriate throughput under the condition of guaranteeing the SER to be lower than 0.01. When the noise gain is 3dB, the SER is lower than 0.01 except when energy level is 8 and window length is T. In such a condition, we select other parameter combinations to make the throughput maximization as shown in Fig. 19. If we select the number of energy levels as 8 and the receiving window length as 1.1T, the SER is 0.0094 and the throughput is 149.3bps.When the noise gain is 9dB, the SER is larger than 0.01 when the number of energy levels is 8. Since the receiving window length has already been the maximum value, it is necessary to decrease the number of energy levels. We select the number of energy levels is 4 and the receiving window length is T. In such a condition, the SER is 0.0096 and the throughput is 141.0176bps as shown in Fig. 20. When the noise gain is 18dB, the SER is larger than 0.01 when the energy level is 8 and 4. In this condition, we select the number of energy levels is 2 and the receiving window length is T. In this way, we obtain the throughput of 89.0167bps and the SER is 0.0028 at the same time as shown in Fig. 21.

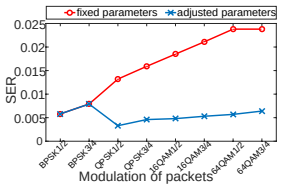


Fig. 22. SER under different WiFi modulations

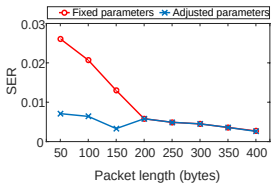


Fig. 23. SER under different WiFi packet lengths

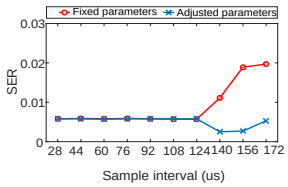


Fig. 24. SER under different RSSI sampling rates

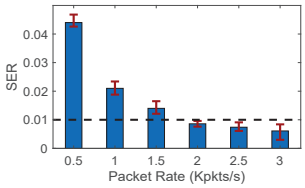


Fig. 25. WiZig SER under different WiFi packet rates

D. Robustness of SER Model

Several other factors, for example, the modulation, the length of the sender packets, and the sample rate of the receiver also influence the SER. In this subsection, we study the influence of these factors on the performance of WiZig. We compare WiZig with the fixed parameters and the rate adaptation algorithm. If we use the fixed parameters, the number of energy levels is 2 and the receiving window length is 5ms.

First, we change the modulation of WiFi packets and observe its impact on the SER. The results are shown in Fig. 22. We find that the SER increases corresponding to eight different methods if we keep other factors unchanged. The SER of BPSK1/2 is 0.0058 and it is the lowest. The performance of BPSK1/2 is much better than 64QAM. This is because that different WiFi modulation mechanisms have different data rate. The higher data rate a modulation has, the shorter its on-air time a packet. Therefore the less RSSI samples a WiZig receiver collects. As a consequence, the decoding accuracy decreases. Generally speaking, all WiFi modulation techniques can be used when WiZig is operated. Whether a change of WiFi modulation technique (potentially sacrifice in WiFi throughput) is deserved depends on the network operator's decision, namely to value more on the WiFi network throughput or the CTC performance.

Second, we let the WiZig sender transmits packets with different lengths. The modulation is BPSK1/2. We find that the SER decreases with the increase of packet length when other factors are fixed as shown in Fig. 23. If the packet length is 50 bytes, the SER is 0.0261. Extending the packet length to 250 bytes, the SER decreases to 0.0058. If the packet length is 400 bytes, the SER is 0.0036. WiFi packet length has a significant influence on the SER. If we enable the online rate adaptation, we find that the SER of WiZig is lower than fixed parameters obviously when the WiFi length is smaller than 200 bytes.

We also verify the robustness of WiZig with considering the sampling rate of ZigBee devices. The USRP/N210 sender transmits packets modulated by BPSK1/2 with length of 250 bytes. We change the sample interval of the ZigBee receiver and observe the variation of the SER as shown in Fig. 24. We find that the relationship of the SER and the sample interval is not an absolutely monotonic relation. The SER is relatively small and stable when the sample interval is less than 140us.

The modulation, the length of the packets, and the sample rate of the receiver have a great influence on the SER. WiZig selects the appropriate parameters to reduce the SER with the theoretical support of energy channel model.

E. Implementation on commercial devices

To implement WiZig on a WiFi card, what we need is merely the ability to transmit packets on demand, with regard to the transmission time and packet intervals. In order to achieve this ability, one may utilize packets that already queue on the WiFi radio and modulate the transmissions, or generate and send packets as required. Correspondingly, one may make modification to the WiFi card (if allowed) or adopt a packet injection tool.

In our implementation, we use the D-ITG [30] to generate packets. The overhead (denoted by C) is measured by the extra traffic induced by WiZig and generated by the tool, which can be calculated by $C = S * W * L$. Where S denotes the WiFi packet sending rate, W denotes the WiZig receiving window length, and L denotes the WiFi packet length. According to the Eq. (13), S, W, L should be appropriately configured such that the SER of WiZig meets the requirement of communication. For example, in order to have a SER=0.01, the typical setting in the implementation is S=2000 pkts/s, W=3ms, and L=200 bytes. Accordingly, the overhead of sending 1-bit WiZig *symbol* is C=1.2 Kbytes. For ease of understanding, Fig. 25 shows the WiZig SER under different WiFi packet sending rates, given W=3ms, and L=200 bytes.

As for the impact of the packet injection tool on the network, we measure the channel holding time (denoted by T) for sending 1-bit WiZig symbol, which can be calculated by $T = C/D$. Where D denotes the data rate of WiFi transmission. For example, when we set the data rate of WiFi transmission at 54Mbps (802.11a/g), the channel holding time for sending 1-bit WiZig symbol is 177.8μs.

F. The impact of WiZig on the WiFi network

WiZig achieves amplitude modulation by adjusting the Tx power of the WiFi sender. The adjustment of WiFi Tx power will affect the modulation and coding rate selection (MCS), and further affect WiFi performance. We conduct experiments, in which WiFi MCS is adapted to the variation of Tx power, so that we can observe the resulting WiFi throughput. Table II lists the corresponding MCS and transmission rate, when the WiFi Tx power gain is varied from 20dB to 0dB.

Theoretically, the WiFi throughput is calculated by the following equation:

$$Th_{WiFi} = \frac{1}{I_W + \frac{N_B}{MC_R}} N_B \tag{19}$$

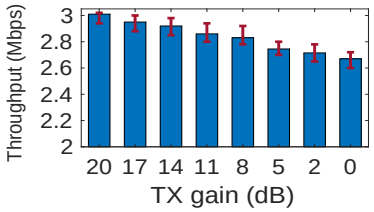


Fig. 26. WiFi throughput under different Tx gains

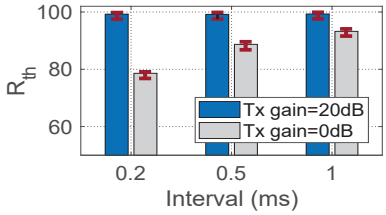


Fig. 27. WiFi throughput under different packet intervals

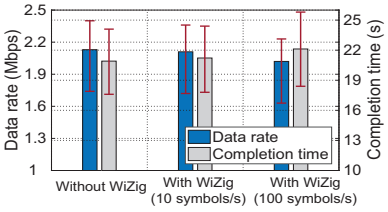


Fig. 28. Performance of WiFi networks w/o WiZig

where I_W is interval between WiFi packets, N_B is the number of bits transmitted by a WiFi packet, and MC_R is the transmission rate.

In the experiment, we set the WiFi packet length at 200 bytes with the packet interval of 500μs. The experimental result is shown in Fig. 26. We can see the WiFi throughput decreases with the decrease of Tx power. Specifically, the WiFi throughput is 2.67Mbps when Tx gain is 0dB, which decreases by 11.3% compared with the WiFi throughput when Tx gain is 20dB (3.01Mbps). Although WiFi Tx power adjustment affects the performance of WiFi throughput, the impact is reasonable and limited.

We further change the packet interval to simulate different traffic pattern of WiFi such as website browsing, online video watching, and file downloading. The packet interval refers to the time between the end of the current packet and the start of the next packet. We define a new metric R_{th} as $R_{th} = \frac{Th_i}{Th_{20}}$, where Th_{20} is theoretical WiFi throughput when Tx power gain is 20dB and Th_i is the WiFi throughput with other Tx power gains idB. The packet interval varies from 200μs to 1ms. When the packet interval is 200μs and Tx power gain is 20dB, the theoretical WiFi throughput is $Th_{20} = 6.67Mbps$. When the packet interval is 200μs and Tx power gain is 0dB, the theoretical WiFi throughput is $Th_0 = 5.54Mbps$. So the value of R_{th} is $R_{th} = \frac{5.54Mbps}{6.97Mbps} = 79.8\%$. Similarly, we can know R_{th} is 89.9% when the packet interval is 500μs, the corresponding WiFi throughputs are 3.02Mbps and 2.72Mbps, respectively for 20dB and 0dB Tx power gain. When the packet interval is 1ms, R_{th} is 94.6%, and the corresponding WiFi throughputs are 1.17Mbps and 1.10Mbps, respectively for 20dB and 0dB Tx power gain.

We conduct experiments and the evaluation result is shown in Fig. 27. When the Tx power gain is 20dB, the value of R_{th} is 99.2% and relatively stable, no matter what the interval is. When the Tx power gain is 0dB, the value of R_{th} is 78.6%, 88.7% and 93.3% when the packet interval is 200μs, 500μs and 1ms, respectively. We find that the impact of WiFi Tx power on WiFi throughput decreases with the increase of packet interval. Because the variation of packet duration affected by the modulation scheme is usually dozens of microseconds. If the packet interval is large, the variation of packet duration can be ignored. In other words, WiFi Tx power has impact on the WiFi throughput for the data-intensive applications. For other non-intensive applications, the impact of Tx power is relatively small.

We use iperf to evaluate the impact of WiZig on the WiFi network. Two laptops installed iperf act as a pair of WiFi server and client. Around WiFi devices, we deploy a pair of devices for WiZig transmissions. An USRP device is used as the WiZig sender and a TelosB node is used as the WiZig receiver. The WiFi packet sending rate of the WiZig sender is 2000 pkts/s and the packet length is 200 bytes. The symbol receiving window length of WiZig is 3ms. In the experiments, the WiFi client transmits 5 MBytes data to the WiFi server. We measure the data rate and completion time of this WiFi network in three different cases: 1) without WiZig, 2) with WiZig at the symbol rate of 10 symbols/s, and 3) with WiZig at the symbol rate of 100 symbols/s. In cases 2 and 3, the WiZig transmission is kept on from the beginning to the end of the experiment. In each case, we repeat the experiments for 20 times.

Tx gain	Modulation	Coding rate	Transmission rate
20dB	64QAM	3/4	54Mbps
17dB	64QAM	2/3	48Mbps
14dB	64QAM	2/3	48Mbps
11dB	16QAM	3/4	36Mbps
8dB	16QAM	3/4	36Mbps
5dB	16QAM	1/2	24Mbps
2dB	16QAM	1/2	24Mbps
0dB	4QAM	3/4	18Mbps

TABLE II
MODULATION, CODING RATE AND TRANSMISSION RATE UNDER DIFFERENT TX GAINS

The experimental results are shown in the Fig. 28. When there is no WiZig transmission, the average data rate of WiFi network is 2.13Mbps and the average completion time of WiFi network is 20.91s. When the symbol rate of WiZig is 100 symbols/s, the average data rate of WiFi network decreases to 2.02Mbps and the average completion time of WiFi network increases to 22.12s. Although the performance of the WiFi network degrades under the impact of WiZig, the impact is limited.

G. WiZig Performance under Mobility

We conduct new experiments to evaluate the performance of WiZig under mobility. In the experiments, an USRP device transmits WiFi packets as the CTC sender. The WiFi packet sending rate is 500 pkts/s and the packet length is 200 bytes. A volunteer carrying the ZigBee receiver walks, jogs, and runs at a speed of 1 m/s, 2 m/s, and 4 m/s, respectively. WiZig adopts the online rate adaptation algorithm to adjust parameters during the experiments. The results show that when the movement speed is 1 m/s, 2 m/s, 4 m/s, the number of energy levels is 4, 2, 2, and the WiZig receiving window

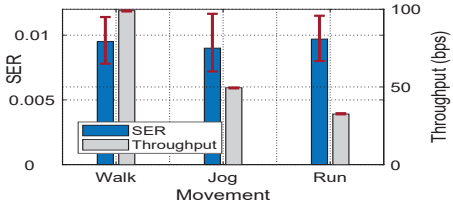


Fig. 29. WiZig performance under mobility

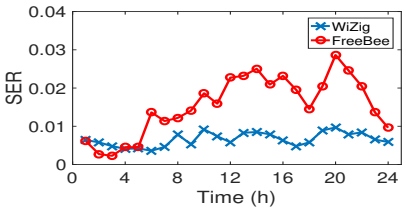


Fig. 30. SER of WiZig and FreeBee during our 24-hours experiment

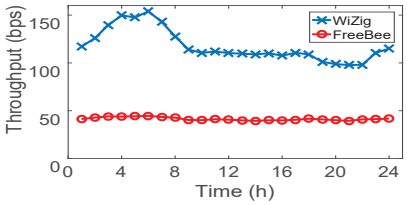
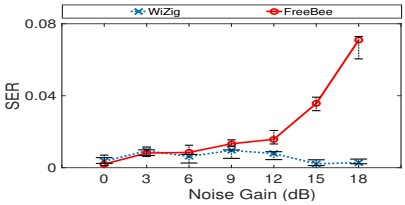
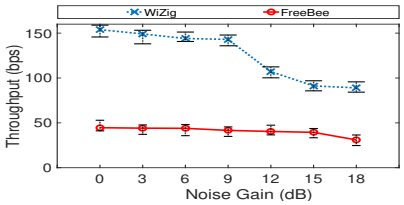


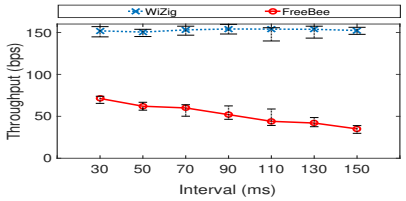
Fig. 31. Throughput of WiZig and FreeBee during our 24-hours experiment



(a) SER under different noise intensities



(b) Throughput under different noise intensities



(c) Throughput under different beacon intervals

Fig. 32. Performance comparison between WiZig and FreeBee

length is 20ms, 20ms, 30ms. The completion time of parameter adjustment corresponding to the three different movements is 0.081s, 0.164s, and 0.244s, respectively. Fig. 29 shows the SER and throughput of WiZig under mobility. The SER is lower than 0.01 in all of our mobility experiments. The throughput is 99.1bps, 49.5bps, and 32.6bps for walking, jogging and running, respectively. The results demonstrate WiZig can work well in different mobile environments due to our rate adaption algorithm.

H. WiZig Vs Freebee

We also compare WiZig with FreeBee, a state-of-art cross-technology communication scheme. FreeBee modulates symbol data by shifting the timing of periodical beacon frames. It demodulates the data by RSSI values based on the method of folding [31]. We implement Freebee following the design described in the paper [21]. We set the number of beacon repetitions for statistic demodulation ρ is 2. The beacon interval is 100 ms. We observe the SER and the throughput performance of FreeBee and WiZig. The results are shown in Fig. 32(a) and Fig. 32(b).

First, we compare WiZig and FreeBee in terms of the SER as shown in Fig. 32(a). When channel is clean, the performances of both FreeBee and WiZig are good. For example, when the noise gain is 0dB, the SER of FreeBee is 0.002 and WiZig is 0.0036. With the increase of the noise, the SER of FreeBee increases to be larger than the required threshold 0.01, while the SER of WiZig is always lower than 0.01. For example, the SER of FreeBee and WiZig are 0.0133 and 0.0096 respectively when the noise gain is 9dB. Further more, when the noise gain increases to 18dB, the SER of FreeBee increases to 0.0709, while the SER of WiZig is 0.0028, which is still lower than the required 0.01.

Second, we compare WiZig and FreeBee in terms of the throughput as shown in Fig. 32(b). The throughput of WiZig is always higher than FreeBee. For example, when the noise gain is 0dB, the throughput of WiZig is 153.85bps and FreeBee is 44.6bps. With the increase of the channel noise, although the throughput of FreeBee and WiZig both decreases monotonously. The throughput of WiZig is higher than FreeBee. For example, the throughput of FreeBee and WiZig are 41.6bps and 143.1bps respectively when the noise gain is 9dB. Further more, when the noise gain increases to 18dB, the throughput of FreeBee increases to 30.9bps, while the throughput of WiZig is 89.1bps, which is near three times of FreeBee.

The beacon interval has a significant influence on the performance of FreeBee as shown in Fig. 32(c). Enlarging the beacon interval has two effects. On the one hand, it offers more space for timing shift and yields more bits per symbol. On the other hand, it requires more time to reach the same ρ. The throughput of FreeBee decreases with the increase of beacon interval. The throughput of WiZig is stable and higher than FreeBee, however.

Furthermore, we observe the performance of FreeBee and WiZig within 24 hours in a laboratory environment. As shown in Fig. 30, we find that the SER of WiZig is always lower than 0.01. The SER of FreeBee is larger than 0.01 when the channel is noisy. When the time is 0 a.m. to 6 a.m., the SER is lower than 0.01 both for WiZig and FreeBee. It maybe because that the channel is clean in this time period. During the daytime and 7 a.m. to 11 p.m. , the SER of FreeBee is higher than WiZig. Especially in 8 p.m., the SER of FreeBee is 0.0286 and WiZig is 0.0097 due to numerous people and wireless interference. The throughput of FreeBee and WiZig is shown in Fig. 31. We find that although the throughput variation of WiZig is larger than FreeBee, the throughput of WiZig is better than FreeBee. Even in 8 p.m., the throughput of WiZig is 98.8bps and it is more than two times than Freebee, which is 40.1bps.

VIII. Discussion

Feasibility of the reverse communication. There are multiple methods among the existing proposals that support reverse communication, i.e. the communication from ZigBee to WiFi. One option is FreeBee [21]. We can borrow the idea of FreeBee and modulate ZigBee packets in the temporal dimension. Another option is ZigFi [32], which affects the CSI readings of WiFi packets to convey data. When applied as a back-channel for WiZig, FreeBee has better applicability while ZigFi has better throughput. As WiZig doesn't pose any additional restriction on the reverse communication, we believe future proposals on communication from ZigBee to WiFi can also be integrated with WiZig.

Medium access control. The CTS-to-self mechanism is an appropriate solution for WiZig to capture the channel and make sure nobody else transmits for a given period. Before transmitting CTC packets, a WiZig sender (WiFi device) can first broadcast a CTS-to-self control frame that contains information of the occupied duration of the network. Other nodes that overhear the frame will backoff and keep silent until the channel is clear again. The CTS-to-self mechanism may impact the system throughput, however, such impact is limited in theory. First, the 14-bit CTS-to-self control frame is light-weight and has negligible impact on the performance of WiZig. Second, the impact of the CTS-to-self mechanism on the other WiFi nodes is limited and tolerable. Most of the existing CTC methods have to suspend the other transmissions for the CTC channel with less interference. Due to the high throughput of WiZig, the transmission for tens of bits needs only tens of milliseconds, which is affordable for enabling the direct communications.

The impact of WiFi beamforming. A direct impact of beamforiming is that the energy is no longer uniformly distributed in the communication range of a WiFi device. We discuss the potential impact in two cases. First, if the WiZig devices are all stationary, beamforming has no impact on our solution, because WiZig can be adapted to the channel condition to optimize its throughput. Second, if WiZig is operated under device mobility, we may anticipate that the perceived SNR of the WiZig device may change significantly and frequently. According to the online rate adaptation algorithm, WiZig can monitor the changes of SNR and leverage the algorithm to optimize its throughput.

The networking aspects of the WiZig. WiZig is a general communication framework for CTC from WiFi to ZigBee. The users can define customized packet format in the network layer. In our current implementation, we define the source/destination addresses, packet sequence number, payload, and CRC fields in a packet. The address can be defined as the MAC addresses of devices or self-defined by the application. When a WiZig sender has CTC data to transmit, it may first broadcast a CTS-to-self frame to inform other concurrent senders and reserve the channel. Then the receivers use existing signal identification methods [12] to identify WiFi devices and achieve the CTC neighbor discovery. Then address checking is used to judge whether the on-going transmission should be received by the device. The receiver will decode the CTC packets that pass the address checking and reply an ACK by the reversed communication link after the packet is successfully decoded. Furthermore, rate adaptation is proposed to improve the throughput by adjusting the parameters of the number of energy levels and the receiving window length.

WiZig is not limited to work between two nodes. If there are more than two nodes in the network, the WiZig link can be established between any pair of WiFi and ZigBee nodes. In such condition, distinguishing multiple nodes is an interesting and meaningful issue. For this purpose, we may find suitable solutions from the existing works. For example, ZiSense [12] provides an effective and efficient solution, which utilizes the physical layer signal features to distinguish different devices. Moreover, at the MAC layer and above, the source address information of devices may be included in the packets, so as to distinguish different nodes.

IX. Conclusion

We propose WiZig, a novel cross-technology communication mechanism that enables wireless devices with different PHY/MAC standards to communicate directly. We model the energy channel and analyze the relationship among the BER, SER, SNR, and the energy levels in theory. Based on the theoretical model, we carefully design our amplitude/temporal modulation/demodulation strategies. An online rate adaptation algorithm is further proposed to dynamically adjust the number of energy levels and the length of receiving window to realize a high data rate under the dynamic channel. We implement a prototype of WiZig on a software radio platform and a commercial ZigBee device. The evaluation results show that WiZig achieves a throughput of 153.85bps with less than 1% symbol error rate in the real office environment.

Acknowledgment

This work is supported in part by National Science Fund under grant No. 61672320, NSFC for Excellent Young Scientist No. 61422207, National Key R&D Program of China No. 2017YFB1003000, and NSFC No. 61772306.

References

[1] D. Mattiacci, S. Kosta, A. Mei, and J. Stefa, "Supporting interoperability of things in iot systems," in *Proceedings of ACM Sensys*, 2013.

[2] K. Mikhaylov, J. Petäjäjärvi, M. Mäkeläinen, A. Paatelma, and T. Hänninen, "Extensible modular wireless sensor and actuator network and iot platform with plug&play module connection," in *Proceedings of ACM IPSN*, 2015.

[3] V. L. Erickson and A. E. Cerpa, "Thermovote: participatory sensing for efficient building hvac conditioning," in *Proceedings of ACM BuildSys*, 2012.

[4] A. Hithnawi, H. Shafagh, and S. Duquennoy, "Tiim: technology-independent interference mitigation for low-power wireless networks," in *Proceedings of ACM IPSN*, 2015.

[5] C.-J. M. Liang, K. Chen, N. B. Priyantha, J. Liu, and F. Zhao, "Rushnet: practical traffic prioritization for saturated wireless sensor networks," in *Proceedings of ACM Sensys*, 2014.

[6] S. Gollakota, F. Adib, D. Katabi, and S. Seshan, "Clearing the rf smog: making 802.11 n robust to cross-technology interference," in *Proceedings of ACM SIGCOMM*, 2011.

[7] Y. Wang, Q. Wang, Z. Zeng, G. Zheng, and R. Zheng, "Wicop: Engineering wifi temporal white-spaces for safe operations of wireless body area networks in medical applications," in *Proceedings of IEEE RTSS*, 2011.

[8] X. Zheng, Z. Cao, J. Wang, Y. He, and Y. Liu, "Interference resilient duty cycling for wireless sensor networks under co-existing environments," *IEEE Transactions on Communications*, vol. 65, no. 7, pp. 2971–2984, 2017.
[9] X. Zhang and K. G. Shin, "Enabling coexistence of heterogeneous wireless systems: case for zigbee and wifi," in *Proceedings of ACM MobiHoc*, 2011.
[10] X. Zhang and K. G. Shin, "Adaptive subcarrier nulling: Enabling partial spectrum sharing in wireless lans," in *Proceedings of IEEE ICNP*, 2011.
[11] C.-J. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis, "Surviving wi-fi interference in low power zigbee networks," in *Proceedings of ACM SenSys*, 2010.
[12] X. Zheng, Z. Cao, J. Wang, Y. He, and Y. Liu, "Zisense: towards interference resilient duty cycling in wireless sensor networks," in *Proceedings of ACM SenSys*, 2014.
[13] Z. Zhao, W. Dong, G. Chen, G. Min, T. Gu, and J. Bu, "Embracing corruption burstiness: Fast error recovery for zigbee under wi-fi interference," *IEEE Transactions on Mobile Computing*, vol. 16, no. 9, pp. 2518–2530, 2017.
[14] Z. Zhao, W. Dong, J. Bu, T. Gu, and G. Min, "Accurate and generic sender selection for bulk data dissemination in low-power wireless networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 948–959, 2017.
[15] K. Jang, J. Sherry, H. Ballani, and T. Moncaster, "Silo: Predictable message latency in the cloud," in *Proceedings of ACM SigComm*, 2015.
[16] T. Zachariah, N. Klugman, B. Campbell, J. Adkins, N. Jackson, and P. Dutta, "The internet of things has a gateway problem," in *Proceedings of ACM HotMobile*, 2015.
[17] V. Iyer, V. Talla, B. Kellogg, S. Gollakota, and J. R. Smith, "Inter-technology backscatter: Towards internet connectivity for implanted devices," *arXiv preprint arXiv:1607.04663*, 2016.
[18] R. G. Cid-Fuentes, M. Y. Naderi, S. Basagni, K. R. Chowdhury, A. Cabellos-Aparicio, and E. Alarcon, "On signaling power: Communications over wireless energy," in *Proceedings of IEEE INFOCOM*, 2016.
[19] K. Chebrolu and A. Dhekne, "Esense: communication through energy sensing," in *Proceedings of ACM MobiCom*, 2009.
[20] S. Yin, Q. Li, and O. Gnawali, "Interconnecting wifi devices with ieee 802.15. 4 devices without using a gateway," in *Proceedings of DCOSS*, 2015.
[21] S. M. Kim and T. He, "Freebee: Cross-technology communication via free side-channel," in *Proceedings of ACM MobiCom*, 2015.
[22] W. Jiang, Z. Yin, R. Liu, Z. Li, S. M. Kin, and T. He, "Bluebee: a 10,000x faster cross-technology communication via phy emulation," in *Proceedings of ACM SenSys*, 2017.
[23] R. Petrolo, V. Loscrì, and N. Mitton, "Towards a smart city based on cloud of things, a survey on the smart city vision and paradigms," *Transactions on Emerging Telecommunications Technologies*, 2015.
[24] Z. Shelby and C. Bormann, *6LoWPAN: The wireless embedded Internet*, vol. 43. John Wiley & Sons, 2011.
[25] D. Culler and S. Chakrabarti, "6lowpan: Incorporating ieee 802.15. 4 into the ip architecture," 2009.
[26] X. Zhou, R. Zhang, and C. K. Ho, "Wireless information and power transfer: Architecture design and rate-energy tradeoff," *IEEE Transactions on Communications*, vol. 61, no. 11, pp. 4754–4767, 2013.
[27] Y. Zhang and Q. Li, "Howies: A holistic approach to zigbee assisted wifi energy savings in mobile devices," in *Proceedings of IEEE INFOCOM*, 2013.
[28] X. Zhang and K. G. Shin, "Gap sense: Lightweight coordination of heterogeneous wireless devices," in *Proceedings of IEEE INFOCOM*, 2013.
[29] A. Bereza, U. Wetzker, C. Herrmann, C. A. Boano, and M. Zimmerling, "Demo: Cross-technology communication between ble and wi-fi using commodity hardware," in *Proceedings of ACM EWSN*, 2017.
[30] A. Botta, A. Dainotti, and A. Pescapè, "A tool for the generation of realistic network workload for emerging networking scenarios," *Computer Networks*, vol. 56, no. 15, pp. 3531–3547, 2012.
[31] D. H. Staelin, "Fast folding algorithm for detection of periodic pulse trains," *Proceedings of the IEEE*, vol. 57, no. 4, pp. 724–725, 1969.
[32] X. Guo, Y. He, X. Zheng, L. Yu, and O. Gnawali, "Zigfi: Harnessing channel state information for cross-technology communication," in *Proceedings of IEEE INFOCOM*, 2018.

Xiuzhen Guo received the B.E. degree in the School of Electronic and Information Engineering from Southwest University in 2016. She is currently a PhD. student in Tsinghua University. Her research interests include wireless sensor networks and wireless network co-existence.

Yuan He is an associate professor in the School of Software and TNLIST of Tsinghua University. He received his B.E. degree in the University of Science and Technology of China, his M.E. degree in the Institute of Software, Chinese Academy of Sciences, and his PhD degree in Hong Kong University of Science and Technology. His research interests include wireless networks, Internet of Things, pervasive and mobile computing. He is a member of the IEEE and ACM.

Xiaolong Zheng is currently an associate researcher with the School of Computer Science, Beijing University of Posts and Telecommunications, China. He received his B.E. degree from the Dalian University of Technology, China, in 2011, and his Ph.D. degree from the Hong Kong University of Science and Technology, China, in 2015. His research interests include the Internet of Things, wireless networks, and ubiquitous computing.