SledZig: Boosting Cross-Technology Coexistence for Low-Power Wireless Devices

Junmei Yao
Shenzhen University
Shenzhen, China
yaojunmei@szu.edu.cn

Haolang Huang
Shenzhen University
Shenzhen, China
huanghaolang2020@email.szu.edu.cn

Ruitao Xie
Shenzhen University
Shenzhen, China
xie@szu.edu.cn

Xiaolong Zheng
Beijing University of Posts and Telecommunications
Beijing, China
zhengxiaolong@bupt.edu.cn

Kaishun Wu
Shenzhen University
Shenzhen, China
wu@szu.edu.cn

***Abstract*—With the rapid growth of Internet of Things, the number of heterogeneous wireless devices working in the same frequency band increases dramatically, leading to severe cross-technology interference. To enable coexistence, researchers have proposed a large number of mechanisms to manage interference. However, existing mechanisms have severe modifications in either the physical or MAC (medium access control) layers, making them hard to be deployed on commercial devices. In this paper, we design and implement SledZig to boost cross-technology coexistence for low-power devices through both enabling more transmission opportunities and avoiding interference. SledZig is fully compatible with the standard in both physical and MAC layers. It decreases the WiFi signal power on the channel of low-power devices while keeps the WiFi transmission power unchanged, through making constellation points in the overlapped subcarriers have the lowest power, which can be achieved by just encoding the WiFi payload. We implement SledZig on hardware testbed and evaluate its performance under different settings. Experiment results show that SledZig can effectively increase ZigBee transmissions and improve its performance over a WiFi channel under various WiFi data traffic, with as low as 6.94% WiFi throughput loss.**

***Index Terms*—Keywords: Heterogeneous Wireless Networks; Coexistence; WiFi; ZigBee.**

I. Introduction

The prosperity of Internet of Things (IoT) increases the number of wireless devices exponentially. Wireless devices adopt heterogeneous wireless technologies, as each technology has its own suitable application scenarios due to its strengths and weaknesses. In the crowded ISM (industrial, scientific and medical) frequency band, the heterogeneous wireless devices inevitably work in the overlapped channels, leading to severe cross-technology coexistence problem.

WiFi and ZigBee are the two most common wireless technologies in IoT. WiFi is used for wireless local area networks (WLAN), while its market has stable increase now and in the future. Cisco predicts that the number of WiFi hotspots will reach 628 Million by 2023 [1]. Meanwhile, ZigBee plays an important role in providing low cost, low data rate, and low energy consumption characteristics for wireless sensor networks. The ZigBee market also increases steadily these years. It was valued at USD 2.81 Billion in 2018 and is projected to reach USD 5.38 Billion by 2026 [2]. WiFi and ZigBee has asymmetry power levels. The ZigBee signal is always transmitted at less than $1mW$ for energy saving, while the WiFi signal is transmitted at up to $100mW$ for large coverage. Meanwhile, when the devices are contending channel, WiFi has higher priority than ZigBee and can always win the channel for data transmission, due to their MAC layer design. Thus, the WiFi devices induce severe coexistence problems to ZigBee devices, through either prohibiting the ZigBee devices from data transmission or interfering the ongoing ZigBee data transmission.

The coexistence problem has attracted much research interest in past years. The related works can be categorized into two groups: cross-technology interference avoidance and interference resistance. Interference avoidance mechanisms always mitigate cross-technology interference (CTI) through designing physical (PHY) or MAC layer protocols. For example, EmBee [3] lets a WiFi device identify the channel of ZigBee signals and then reserves the corresponding channel for ZigBee transmission through designing null subcarriers. Interference resistance mechanisms try to recover the collided signal through PHY layer design, such as CrossZig [4], which utilizes packet merging and adaptive forward error correction (FEC) coding to recover packets under CTI. Both kinds of mechanisms require modifications on either the MAC layer or the PHY layer, thus cannot be applied to current commercial devices directly.

In this paper, we propose SledZig, a subcarrier-level energy decreasing mechanism on WiFi to boost ZigBee transmission. SledZig is fully compatible with the standard PHY and MAC layer processes, and requires no change on commercial WiFi and ZigBee devices. It decreases the WiFi signal energy on the ZigBee channel while keeps the WiFi transmission power unchanged, through exploiting the features of QAM (quadrature amplitude modulation) modulation in WiFi. QAM is a combination of phase and amplitude modulation methods, making the QAM constellation points have different power levels. By inserting extra bits to original WiFi data bits, we let the QAM points in subcarriers overlapped with the ZigBee channel have the lowest power, while those out of the ZigBee channel remain unchanged, leading to up to 14dB energy decreasing on the ZigBee channel. With this energy decreasing, the ZigBee network performance can be improved dramatically through both enabling more transmission opportunities and avoiding interference.

From the perspective of usage, SledZig is quite simple. With the original data bits, the WiFi transmitter first inserts extra bits to generate the transmit bits. When the transmit bits are passed through the standard WiFi transmission process, the signal energy on the ZigBee channel can be automatically decreased, thus to boost ZigBee transmissions. Meanwhile, the WiFi receiver can easily obtain the original data bits through remove the extra bits from the received bits.

This paper makes the following main contributions:

- We design SledZig, a subcarrier-level energy decreasing mechanism on WiFi to decrease the signal power on ZigBee channels, thus to increase the ZigBee network performance from both enabling more transmission opportunities and avoiding CTI.
- To the best of our knowledge, SledZig is the first mechanism that mitigate CTI through just encoding the WiFi payload. It is compatible with WiFi and ZigBee standards in both PHY and MAC layers, and can be easily deployed to commercial devices.
- We implement SledZig on hardware testbed based on USRP N210 and TelosB platforms. Experimental results indicate that SledZig can decrease the WiFi signal power on a ZigBee channel by up to 14dB. Meanwhile, it can improve the ZigBee performance dramatically with as low as 6.94% WiFi throughput loss.

This paper is organized as follows: Section II briefly describes the basic knowledge of WiFi transmission and the main differences of WiFi and ZigBee in the PHY and MAC layers. Section III illustrates the coexistence problem and presents the opportunity to solve the problem. Section IV presents the detailed design of SledZig. Section V evaluates the performance of SledZig comparing with the standards through hardware experiments. Section VI introduces related works. Section VII concludes this paper and puts forward future works.

II. Background

In this part, we introduce the background knowledge that is important for the SledZig design.

A. WiFi Transmission

Fig. 1 depicts the standard WiFi transmission process. The data bits are first passed through the channel coding module to combat interference, and transformed to complex symbols after QAM modulation; the QAM points are then mapped into OFDM (orthogonal frequency division multiplexing) subcarriers after the S/P (serial-to-parallel) module, and output as the time-domain OFDM symbols after IFFT (inverse fast fourier transform) and P/S (parallel-to-serial) processes; each OFDM symbol is inserted with CP (cyclic prefix) to eliminate the inter-symbol interference; the signal will finally be transmitted after RF front end.

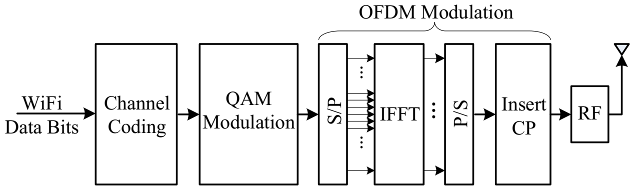


Fig. 1. The standard WiFi transmission process.

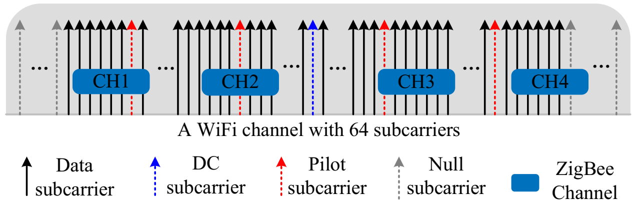


Fig. 2. An illustration of the WiFi channel overlapping with four ZigBee channels.

It is worth noting that OFDM makes a device transmit multiple orthogonal subcarrier signals which are closely spaced to carry data in parallel. In the WiFi system, each 20MHz WiFi channel is divided into 64 subcarriers, including 48 data subcarriers, 4 pilot subcarriers and 12 null subcarriers, as shown in Fig. 2.

B. Differences of WiFi and ZigBee

1) The PHY Layer Specifications: WiFi and ZigBee working in the 2.4 GHz ISM band have distinct specifications. They adopt different PHY layer technologies, as WiFi adopts OFDM and QAM modulations but ZigBee adopts DSSS (direct sequence spread spectrum) and OQPSK (offset quadrature phase shift keying) modulations. Besides that, they have different channel bandwidth. ZigBee has sixteen 2MHz channels with 5MHz channel spacing, numbering from 11 to 26. WiFi has thirteen 20MHz channels with 25MHz channel spacing [1]. Thus, one WiFi channel overlaps with four ZigBee channels. Each WiFi channel which contains 64 subcarriers overlaps with four ZigBee channels in the same pattern, as shown in Fig. 2. For the ease of description in the following part, we call the four ZigBee channels as CH1, CH2, CH3 and CH4 for short. We see that CH1-CH3 overlap with a pilot subcarrier and CH4 overlaps with null subcarriers.

Moreover, the two kinds of devices have asymmetry transmission power. ZigBee devices have the transmission power of no more than 0dBm to cut down energy consumption, while the WiFi transmission power can be up to 20dBm with the purpose of large coverage.

2) The MAC Layer Specifications: Both the WiFi and ZigBee networks adopt CSMA/CA mechanism to contend the channel. The detailed CSMA/CA mechanism is shown in Fig. 3. When a device begins to transmit a data packet, it first waits for DIFS time; if the channel is idle during DIFS,

[1]The WiFi channel can be up to 40MHz in 802.11n and 160MHz in 802.11ax. This paper focuses on the 20MHz channel, while the similar idea can be easily extended to wider channel scenarios.

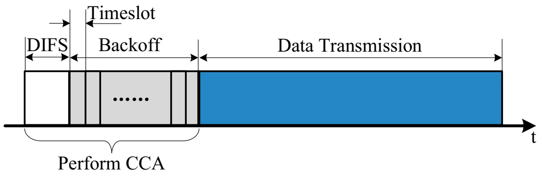
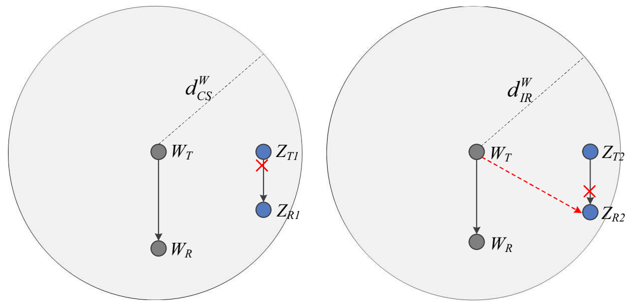
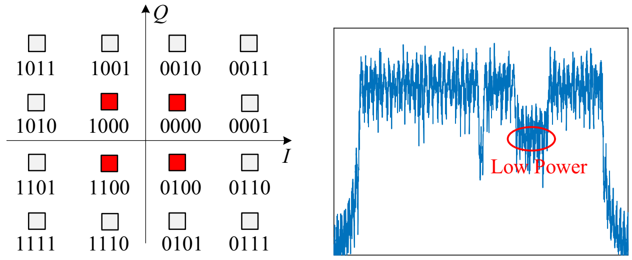


Fig. 3. An illustration of CSMA/CA.



(a) ZigBee devices within the WiFi carrier sense range d_{CS}^{W} are prohibited to transmit data. (b) The ZigBee transmission is interfered by the WiFi transmission.

Fig. 4. Two scenarios that WiFi affects the ZigBee performance.



(a) QAM-16 constellation points (b) WiFi Frequency spectrum

Fig. 5. An example of the QAM-16 lowest points and the frequency spectrum when all the overlapped subcarriers are filled with the lowest points.

the device then waits for a random duration which consists of multiple backoff timeslots to contend for the channel; the backoff timer is decreased by one when the channel is idle for a backoff slot, and is frozen when the channel is busy; the device can finally transmit a data packet if the backoff timer reaches zero. During DIFS or each backoff timeslot, the device should perform CCA (clear channel assessment) to determine whether the channel is idle. The channel is determined to be idle if the detected signal energy is below a predefined threshold; otherwise it is busy.

The main difference here between WiFi and ZigBee is that, the WiFi DIFS is $28\mu s$ [5] while ZigBee DIFS is $320\mu s$ [6], meanwhile, WiFi backoff slot is 9 or $20\mu s$ while ZigBee backoff slot is $320\mu s$. This leads to extreme unfairness in the channel competition, as the WiFi device can always win the channel for transmission.

III. Motivation

Here we first illustrate the cross-technology coexistence problem, then explain the opportunity on SledZig design.

A. Cross-Technology Coexistence Problem

The WiFi and ZigBee differences on PHY and MAC layers lead to severe cross-technology coexistence problem. Actually, with the asymmetry transmission power and MAC parameters, WiFi always affects the ZigBee network performance from two scenarios.

The first scenario lies in the fact that the high WiFi transmission power leads to a large carrier sense range d_{CS}^{W} and prohibits some ZigBee transmissions. As shown in Fig. 4(a), when the WiFi link $W_T \longrightarrow W_R$ and ZigBee link $Z_{T1} \longrightarrow Z_{R1}$ coexist in the network, the ZigBee device Z_{T1} is always prohibited from transmitting data to Z_{R1}. The reason comes from the unfairness in channel competition. As discussed in the previous part, the duration of WiFi DIFS or backoff timeslot is much shorter than that of ZigBee. Thus, when both W_T and Z_{T1} have data packets for transmission and contend the channel, W_T can always win, making ZigBee with extremely poor performance in this situation. Our preliminary experiments indicate that, the ZigBee link can proceed its data transmission only when the WiFi link is very unsaturated, that is, the WiFi application layer data rate should be below 20% of the PHY layer data rate.

The second scenario is that the WiFi transmission may interfere with the ZigBee transmission. As shown in Fig. 4(b), when the ZigBee link $Z_{T2} \longrightarrow Z_{R2}$ proceeds its data transmission, it still has a high probability to be interfered by the WiFi transmission $W_T \longrightarrow W_R$, since it is within the WiFi interference range d_{IR}^{W}. Here Z_{T2} may transmit its data packets either because it is out of d_{CS}^{W} of the WiFi link or because it wins the channel although it is within d_{CS}^{W}. The strong WiFi signal can easily interfere with the ZigBee transmission.

B. Opportunity

Our analysis on the two scenarios in Fig. 4 reveals that, decreasing the WiFi transmission power will obviously increase the ZigBee network performance. In Fig. 4(a), the WiFi carrier sense range d_{CS}^{W} will be shortened, allowing the ZigBee device Z_{T1} to be out of d_{CS}^{W} and have the opportunity to transmit data to Z_{R1}. In Fig. 4(b), the signal from W_T with lower power will have less interference on the ZigBee link $Z_{T2} \longrightarrow Z_{R2}$, leading to successful ZigBee transmissions.

One intuitive way to decrease the WiFi signal power is to adjust the transmit gain to decrease the transmission power, but it will obviously decrease the WiFi performance significantly. Some other methods try to reserve the channel for ZigBee, such as EmBee [3] which designs null subcarriers on the overlapped channel; however, these methods cannot be applied to commercial devices due to the requirement of hardware modification.

We observe that the WiFi power on the overlapped subcarriers can be decreased through designing low power constellation points. As shown in Fig. 1, a WiFi device conducts QAM modulation before the OFDM module. QAM modulation is a combination of phase and amplitude modulations. Fig. 5(a) shows the QAM-16 constellation points, each of which represents four data bits. Among the 16 points, the red points have the lowest power. When the QAM points in the overlapped subcarriers are all the red ones, the signal power in the ZigBee channel can be reduced significantly,

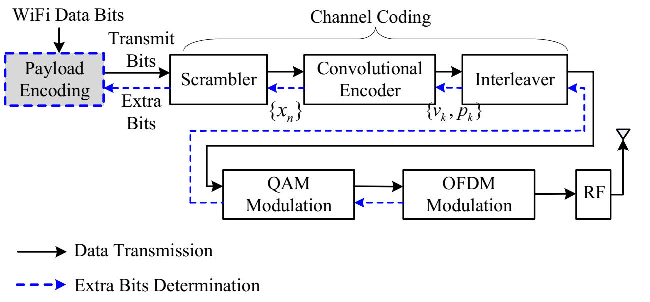


Fig. 6. An overview of SledZig.

as shown in Fig. 5(b). Since this method keeps the WiFi transmission power unchanged, it has limited impact on the WiFi performance.

How much power can be decreased through this way can be derived theoretically. Specifically, the QAM-M modulation encodes groups of $\sqrt{M}$ bits into M constellation points. Each point is a complex symbol which can be denoted as $\mathbf{s}_i = (I_i, Q_i)$, where $I_i, Q_i \in \{\pm(2 \times m - 1)\}$, $i \in [1, M]$ and $m \in [1, \sqrt{M}/2]$. In each QAM modulation, the four lowest points are always $(\pm 1, \pm 1j)$. That means, the low power $P_{low} = 2$. Considering that each point has the equal probability to show in a packet, the average power level of the WiFi signal is $P_{avg} = \sum_i \mathbf{s}_i^2 / 2^M$. Thus, the power decreased through putting lowest points in the overlapped subcarriers is calculated as P_{avg}/P_{low}. More concretely, that value under QAM-16, QAM-64 and QAM-256 is $7.0dB$, $13.2dB$ and $19.3dB$ respectively.

IV. System Design

As shown in Fig. 6, the SlegZig design is to encode the WiFi payload through inserting extra bits to the WiFi data bits, so as to generate the transmit bits; when the transmit bits are passed through the standard WiFi transmission process, the overlapped subcarriers are filled with the lowest constellation points to decrease the signal power on the ZigBee channel. The key issue here is to determine where and what extra bits should be inserted into the WiFi data bits. In this section, we start from the QAM points in the OFDM subcarriers, then follow the reverse WiFi transmission process step by step to achieve this goal.

A. QAM points

According to the design, the QAM points in the overlapped subcarriers should be the four ones with lowest power. For QAM-16, each point carries four bits, and only two bits are significant to make the power lowest. We call them as significant bits, as the shadowed ones shown in Table I. Similarly, each QAM-64 and QAM-256 point has four and six significant bits, respectively. The extra bits should be inserted only to make the significant bits be the designated ones, while the other bits in the QAM points can be arbitrary ones.

B. Overlapped Subcarriers

The more subcarriers used, the greater the impact on WiFi performance, since more extra bits should be inserted into the original WiFi data bits. Here the question is how many subcarriers are required for each ZigBee channel to achieve the lowest power.

TABLE I
An Illustration of Significant Bits.

	QAM-16	QAM-64	QAM-256
Bits	0 0 0 0	0 1 0 0 1 0	0 1 0 0 0 1 0 0
	0 1 0 0	0 1 0 1 1 0	0 1 0 0 1 1 0 0
	1 0 0 0	1 1 0 0 1 0	1 1 0 0 0 1 0 0
	1 1 0 0	1 1 0 1 1 0	1 1 0 0 1 1 0 0

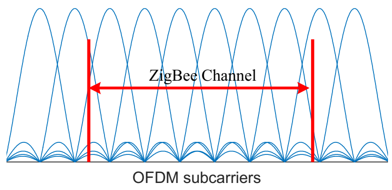


Fig. 7. An illustration of OFDM subcarriers overlapping with a ZigBee channel.

The ZigBee channel is $2MHz$, while each OFDM subcarrier occupies $312.5KHz$. It is easy to take for granted that the number of overlapped subcarriers is $\lceil \frac{2MHz}{312.5KHz} \rceil = 7$. However, this will lead to suboptimal performance. As shown in Fig. 7, the OFDM signal contains multiple closely spaced orthogonal subcarriers. Each subcarrier still has energy leaked into the adjacent subcarriers. Thus, besides the six subcarriers fully overlapped with a ZigBee channel, the two adjacent subcarriers should also be filled with the lowest points. Therefore, we let each ZigBee channel overlap with eight subcarriers, among which one is pilot subcarrier in CH1-CH3, and three are null subcarriers in CH4.

C. Scrambler and Interleaver

The channel coding process includes interleaver, convolutional encoder and scrambler. Interleaver is used in wireless communication system to reduce the decoding errors, and SledZig design here is to generate the significant bits before interleaver through deinterleaving, according to those bits before QAM modulation. As shown in Fig. 6, we denote the significant bits before interleaver as $\{v_k, p_k\}(k \in [1, K])$, where v_k and p_k indicate the value and position of the k-th significant bit. It is worth mentioning that, this process brings additional bonus for SledZig: the significant bits which are gathered together before deinterleaving are scattered to different locations far away, providing feasibility for the extra bits determination in convolutional encoding.

Scrambler is used to avoid long sequences of bits with the same value. SledZig design for this module is to obtain the transmit bits according to the scrambled transmit bits $\{x_n\}$. Since both modules are one-by-one mapping from input bits to output bits, the reverse processes for SledZig are quite easy.

D. Convolutional Encoder

The main objective of SledZig design here is to determine the extra bits required to be inserted to the WiFi data bits according to the significant bits $\{v_k, p_k\}$, as shown in Fig. 6. This process is challenging because convolutional encoder

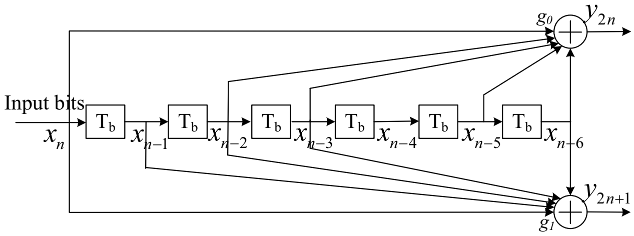


Fig. 8. The process of 1/2-rate convolutional encoding.

TABLE II
AN EXAMPLE OF SIGNIFICANT BITS IN THE FIRST OFDM SYMBOL.

k	1	2	3	4	5	6	7
p_k	29	30	41	42	77	78	89
n	15	15	21	21	39	39	45
k	8	9	10	11	12	13	14
p_k	90	125	138	172	173	183	186
n	45	63	69	86	87	92	93

adds redundancy to the data bits, and it cannot generate arbitrary bit sequence. We achieve this goal through analyzing the convolutional encoding process, summarizing its characteristic to determine the extra bits.

The 802.11 standard recommends several coding rates under each QAM modulation, leading to different WiFi data rates. The 1/2-rate encoding is the basic process in convolutional encoding, where one input bit generates two output bits. The other coding rates like 2/3, 3/4 and 5/6 are achieved by employing puncturing on the 1/2-rate coded bits: some of the coded bits are omitted to increase the coding rate. Here we focus on the 1/2-rate encoding, the process for other coding rates are similar.

The 1/2-rate convolutional encoding process is shown in Fig. 8. It uses two generator polynomials $g_0 = (1011011)_2$ and $g_1 = (1111001)_2$. One input bit x_n triggers two coded bits y_{2n-1} and y_{2n}. The output coded bits are determined by not only the present input bit x_n but also a small number of previous bits from x_{n-1} to x_{n-6}. For the easy of description, we let $X_n = [x_n\ x_{n-1}\ x_{n-2}\ x_{n-3}\ x_{n-4}\ x_{n-5}\ x_{n-6}]'$. Then this one step encoding process to generate two output bits can be formulated as:

$$\begin{aligned} g_0 \times_{GF(2)} X_n &= y_{2n-1}, \\ g_1 \times_{GF(2)} X_n &= y_{2n}, \end{aligned} \tag{1}$$

where $GF(2)$ means the calculation is in the Galois Field GF(2).

We have the significant bits $\{v_k, p_k\}$ after encoder, then the extra bits in the uncoded bits $\{x_n\}$ can be determined through Eq. 1 one by one. To make the description easier, we list an example of the significant bits in the first OFDM symbol in Table II, where QAM-16 is adopted and the ZigBee channel is CH2. There are 14 significant bits in total. The significant bits have two situations, which are very important for the following analysis. One situation is that, given a n, either y_{2n-1} or y_{2n} in Eq. 1 is a significant bit, and the other one can be arbitrary bit, such as the case of $k = 9$, where $n = 63$ and $p_k = 2n - 1 = 125$ in Table II. We call this kind of bit as *single significant bit*. The other situation is that, both the two bits y_{2n-1} and y_{2n} are significant bits, such as the case of $k = 1$ and $k = 2$, where $n = 15$. We call this kind of bits as *twin significant bits*.

For the case of *single significant bit*, we let x_n be the extra bit, which should be inserted to make the equations 1 hold. Here the bits from x_{n-6} to x_{n-1} may be scrambled WiFi data bits or extra bits determined in the previous steps, they are all known in the current step. x_n can be obtained easily through solving the corresponding equation in Eq. 1.

For the case of *twin significant bits*, two extra bits are required to be unknowns in X_n to make Eq. 1 hold. We let x_{n-1} and x_{n-5} be the extra bits, and they can be determined through solving Eq. 1. We note that the bit x_{n-5} are also used to calculate the previous coded bits from $y_{2(n-5)-1}$ to $y_{2(n-1)}$. Once there are *twin significant bits* among them, Eq. 1 may have no solution, as there will be about ten equations together but only three unknowns. However, we find this situation does not happen in the whole extra bits determination process, as the deinterleaving process has scattered the significant bits far way enough to avoid this situation, no matter in which combination of QAM modulations and ZigBee channels. The *twin significant bits* can be always satisfied through inserting two extra bits in the designated positions. We see that no matter in which situation, one significant bit can be satisfied through inserting one extra bit to the WiFi data bits.

The transmit bits $\{x_n\}(n \in [1, N])$ can be generated through inserting extra bits to WiFi data bits $\{x'_i\}$. We formulate the general process in Algorithm 1. Please note that both $\{x'_i\}$ and $\{x_n\}$ are the scrambled bits. The final transmit bits will be obtained through descrambling $\{x_n\}$. From the first bit in $\{x'_i\}$, the device determines whether it triggers a significant bit. If yes, it calculates the extra bits etr_0 or etr_1, then adjusts the values of $\{x_n\}$; if not, it simply assigns current x'_i to x_n. The process is conducted until all the data bits $\{x'_i\}$ are traversed.

E. Impact of Pilot

Each ZigBee channel in CH1-CH3 overlaps with a pilot subcarrier. Since the pilot subcarrier has much higher power than the data subcarriers with the lowest power, it obviously deteriorates the performance of SledZig since the averaged signal power at ZigBee is increased.

In addition, one may argue that, although the averaged signal power at the ZigBee channel decreases, the high power within this short channel band would have much stronger interference to Zigbee, making its transmission unsuccessful. However, the DSSS modulation adopted by ZigBee can naturally tolerate this kind of interference. DSSS makes the transmitted signal wider in bandwidth than the original data bandwidth. If part of the transmission is corrupted, the data can still be recovered from the remaining part of the signal. Thus, as long as the WiFi signal can be decreased to make the ZigBee SNR (signal to noise ratio) meet the requirements of decoding, the ZigBee transmission can be successful.

F. Impact of WiFi Preamble

The previous design only changes the WiFi payload. Actually, each WiFi packet includes a preamble for synchronization and CFO (crucial frequency offset) estimation. The

Algorithm 1: Transmit bits generation process.

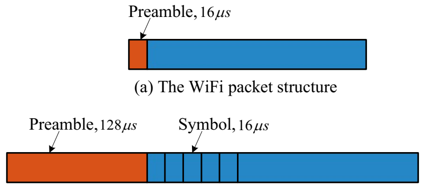
Input : Data bits $\{x'_i\}, i \in [1, N']$;
Significant bits $\{v_k, p_k\}, k \in [1, K]$.
Output: Transmit Bits $\{x_n\}, n \in [1, N]$.

$k \leftarrow 1$; $n \leftarrow 1$; $etr_0 \leftarrow 0$; $etr_1 \leftarrow 0$; $tmp \leftarrow 0$.
for $i = 1 : N'$ **do**
  if $(2n-1) == p_k$ *or* $2n == p_k$ **then**
    $X_n = [etr_0\ x_{n-1}\ x_{n-2}\ x_{n-3}\ x_{n-4}\ x_{n-5}\ x_{n-6}]'$;
    if $(2n-1) == p_k$ **then**
      $y_{2n-1} \leftarrow v_k$.
    else
      $y_{2n} \leftarrow v_k$.
    Calculate etr_0 through Eq. 1;
    $x_n \leftarrow etr_0$;
    $n \leftarrow n+1$; $k \leftarrow k+1$;
    $x_n \leftarrow x'_i$;
    $n \leftarrow n+1$.
  else if $(2n-1) == p_k$ *and* $2n == p_{k+1}$ **then**
    $X_n = [x_{n-2}\ etr_0\ x_{n-3}\ x_{n-4}\ x_{n-5}\ etr_1\ x_{n-6}]'$;
    $y_{2n-1} \leftarrow v_k$, $y_{2n} \leftarrow v_{k+1}$;
    Calculate etr_0 and etr_1 through Eq. 1;
    $tmp \leftarrow x_{n-1}$;
    $x_n \leftarrow x_{n-2}$;
    $x_{n-1} \leftarrow etr_0$;
    $x_{n-2} \leftarrow x_{n-3}$;
    $x_{n-3} \leftarrow x_{n-4}$;
    $x_{n-4} \leftarrow x_{n-5}$;
    $x_{n-5} \leftarrow etr_1$;
    $n \leftarrow n+2$; $k \leftarrow k+2$;
    $x_{n-1} \leftarrow tmp$;
    $x_n \leftarrow x'_i$;
    $n \leftarrow n+1$.
  else
    $x_n \leftarrow x'_i$;
    $n \leftarrow n+1$.

preamble contains 10 repetitive STS (short training symbols) and two repetitive LTS (long training symbols); it lasts for $16\mu s$ in total, as shown in Fig. 9. Meanwhile, the ZigBee devices adopt DSSS and OQPSK modulations. Every group of four bits are spread to specific 32 chips by DSSS; the chips are then modulated through OQPSK for transmission. Each ZigBee symbol lasts for $16\mu s$. The ZigBee preamble contains eight '0000' symbols, corresponding to $128\mu s$. We analyze the impact of WiFi preamble from the two scenarios shown in Fig. 4.

For the scenario of Fig. 4(a) where SledZig decreases the WiFi carrier sense range to enable more ZigBee transmissions, the impact is negligible. The ZigBee CCA period must be eight symbols [6], that is $128\mu s$. Thus, in case the WiFi preamble is within a ZigBee CCA period, this $16\mu s$ high power signal has very limited impact on the CCA result, comparing with the $112\mu s$ low power signal.

For the scenario of Fig. 4(b) where SledZig reduces the WiFi interference to ZigBee transmission, the impact is more complicated. In case the WiFi preamble interferes with the ZigBee preamble, this sudden interference will not affect the detection of ZigBee preamble due to its redundancy design. However, in case the WiFi preamble interferes with a ZigBee



(a) The WiFi packet structure

(b) The ZigBee packet structure

Fig. 9. The packet structure.

symbol in the payload, this symbol will not be detected correctly with a high probability.

Despite this limitation, we will show in section V that SledZig can still improve the ZigBee network performance dramatically.

G. Process at the WiFi Receiver

The process at the WiFi receiver side is quite simple: the receiver first conducts the standard WiFi receiving process to obtain the transmit bits, then removes the extra bits to get the original WiFi data bits. The positions of the extra bits are fixed in the transmit bits, and they are determined by three kinds of information: the ZigBee channel, QAM modulation and coding rate. The latter two information can be obtained directly from the PLCP (physical layer convergence protocol) header of the WiFi packet [5]. The key issue here is to obtain the ZigBee channel. With the transmit bits, the WiFi receiver can conduct the channel coding and modulation process shown in Fig. 4, then it can observe the QAM points and determine the ZigBee channel: the QAM points in the overlapped subcarriers are all lowest ones. This process is fully compatible with the 802.11 standard.

V. Experiments

In this section, we evaluate the the performance of both ZigBee and WiFi networks affected by SledZig through hardware experiments.

A. Experimental Setup

We implement a prototype of SledZig based on USRP (universal software radio peripheral) N210 and TelosB. As shown in Fig. 10, we use one USRP as the WiFi transmitter (WiFi Tx) to generate the WiFi signals following the IEEE 802.11 standard, and use another USRP as the WiFi receiver (WiFi Rx). For a WiFi packet, we first insert extra bits to it according to the SledZig design to generate the transmit bits, then feed the transmit bits to the WiFi transmission process in WiFi Tx to generate the required signal. We use two TelosB devices as the ZigBee Tx and Rx to test the ZigBee performance.

Experiments are conducted in a $10m \times 15m$ open space office. The background noise is tested to be $-91dB$. The USRP Tx and Rx work at the $13th$ WiFi channel. The two TelosB devices work at the four overlapped ZigBee channels numbered from 23 to 26. Here the ZigBee channels 23-25 are CH1-CH3, and the channel 26 is CH4. Since a WiFi channel overlaps with four ZigBee channels in the same pattern, the performance investigated in this WiFi channel can also represent the performance in other channels.

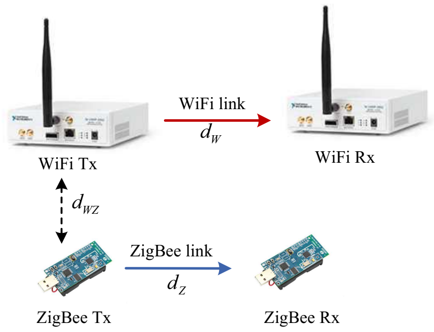


Fig. 10. Experimental setup.

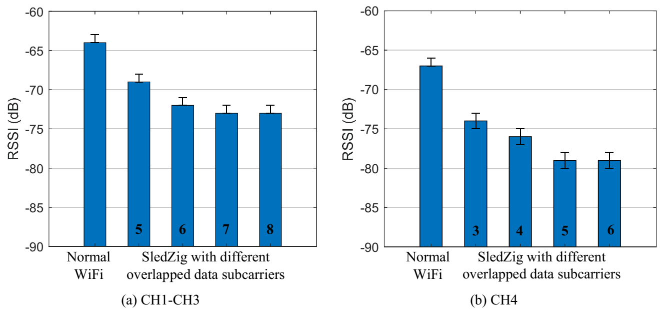


Fig. 11. Impact of the number of data subcarriers on RSSI at ZigBee.

For the easy of description in the following parts, we denote the distance between the WiFi and ZigBee links as d_{WZ}, denote the link distance between WiFi Tx and Rx as d_W, and denote the link distance between ZigBee Tx and Rx as d_Z, as shown in Fig. 10.

B. RSSI at ZigBee

TelosB uses RSSI (received signal strength indication) to measure the received signal power. Since the SledZig design is to decrease the WiFi signal power on the ZigBee channel, this leads to a lower RSSI at ZigBee compared to the standard WiFi signal. Actually, how much RSSI can be reduced will finally affect how much ZigBee performance can be improved. We first investigate RSSI based on the prototype.

According to the theoretical analysis in section IV-B, the optimal number of overlapped data subcarriers with a ZigBee channel is seven for CH1 to CH3, and five for CH4. We test it through experiments. Here the distance between WiFi Tx and ZigBee Rx is fixed at $1m$, and the transmission gain of WiFi Tx is 15. Fig. 11 shows the collected RSSI in four ZigBee channels under QAM-64 as an example. Due to the varied environment and the limitation on the hardware testbed, the collected RSSI under the same situation is not fixed but has $1 \sim 3dB$ variation. We see that in CH1-CH3, the RSSI with seven data subcarriers is about $1 \sim 2dB$ lower than that with six subcarriers, and it remains unchanged when the number of subcarriers increases to eight. We also see that five data subcarriers are suitable for CH4. Besides that, the RSSI from SledZig signal with QAM-64 has about $7dB$ decrease in CH1-CH3, and about $12dB$ decrease in CH4, comparing with the normal WiFi signal where the transmit bits is the randomly generated data bits.

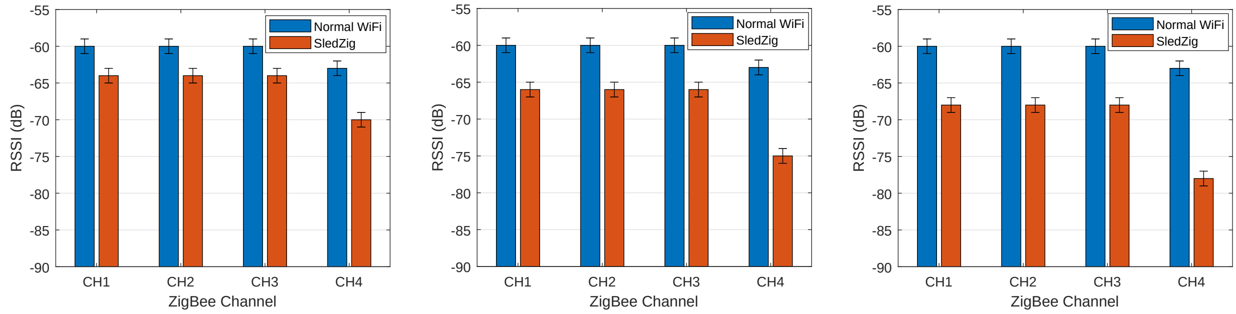
We then conduct experiments to investigate the decrease of RSSI under different QAM modulations and ZigBee channels, the results are shown in Fig. 12. We note that the RSSI from normal WiFi signal has little change when the QAM modulation varies due to the similar averaged signal power. Meanwhile, RSSI collected on CH1, CH2 and CH3 nearly remains unchanged, because the three channels have the similar feature: they are all overlapped with one pilot and seven data subcarriers. In addition, RSSI collected on CH4 is about $3 \sim 4dB$ lower than that on CH1-CH3, since there are two null subcarriers with no power in CH4. In CH1-CH3, SledZig can decrease RSSI from about $-60dB$ to $-64dB$ under QAM-16, to $-66dB$ under QAM-64, and to $-68dB$ under QAM-256. The situation in CH4 is much better, RSSI can be decreased from about $-64dB$ to $-70dB$ under QAM-16, to $-75dB$ under QAM-64, and to $-78dB$ under QAM-256. That is because the pilot subcarrier in CH1~CH3 can largely increase the averaged signal power. From these results, we see that a ZigBee network can have the highest performance when it works on CH4.

C. ZigBee Performance

The main objective of this paper is to decrease the WiFi signal power in the ZigBee channel to improve the ZigBee network performance, through both avoiding interference and exploiting transmission opportunities. Here we conduct experiments to quantify the performance.

1) ZigBee Throughput without Interference: Before investigating the ZigBee performance under interference, we first figure out the ZigBee performance without interference as a reference. We let the WiFi Tx not transmit packets, but let the ZigBee Tx transmit packets continuously. The TelosB transmission gain (Tx gain) can be set from 0 to 31, while 31 is the maximum gain and corresponds to the maximum transmission power. We conduct experiments to investigate the ZigBee power level in terms of the link distance d_Z and Tx gain. As shown in Fig. 13, we see that even when d_Z is $0.5m$, the RSSI is only about $-75dB$ under the maximum transmission power (Tx gain is 31). When d_Z is $1m$ and Tx gain is below 15, the signal is submerged in background noise, that is $-91dB$. When d_Z is $3m$ or larger, the collected RSSI decreases to the background noise even when Tx gain is 25. We set the ZigBee Tx gain as 31 in the following experiments. In addition, the ZigBee throughput without interference is about $63Kbps$, which is much lower than the $250Kbps$ data rate in the PHY layer. Many reasons may lead to this result, such as the long duration of DIFS and backoffs in CSMA/CA, the delay induced by serial communication between TelosB and the laptop, and etc.

2) Impact of d_{WZ}: We then evaluate the ZigBee performance under continuous WiFi transmissions in the same frequency spectrum with the change of d_{WZ} shown in Fig. 10. The WiFi Tx gain is set to be $15dB$. The link distance d_Z is set to be $1m$. Fig. 14 shows the ZigBee throughput of SledZig under three QAM modulations compared with normal WiFi. We see that with SledZig, the ZigBee transmission can be



(a) QAM-16. (b) QAM-64. (c) QAM-256.

Fig. 12. RSSI collected at ZigBee under each channel and modulation type.

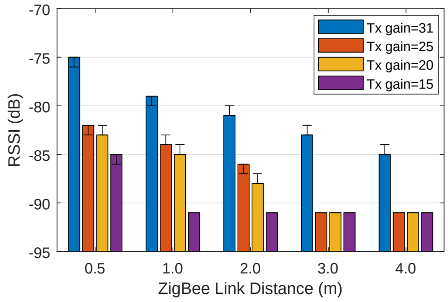
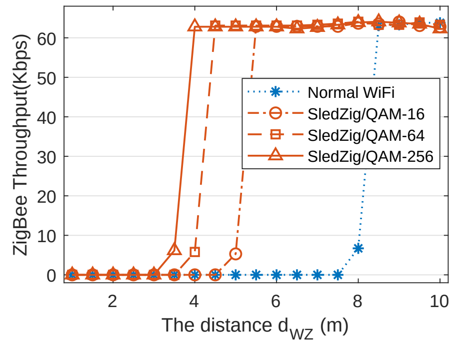


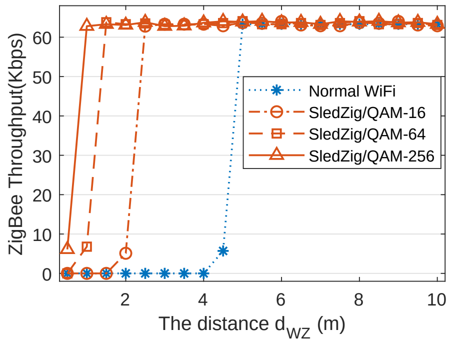
Fig. 13. RSSI in terms of ZigBee link distance d_Z and Tx gain.

successful when the Zigbee link is closer to the WiFi link. Specifically, for ZigBee link in the channels of CH1-CH3, the ZigBee throughput can be about 63$Kbps$ under normal WiFi interference only when d_{WZ} is at least 8.5m, while this distance can be shortened to about 3.5m, 4.5m and 5m with SledZig under QAM-256, QAM-64 and QAM-16 respectively, because the WiFi signal power in the channel can be largely reduced by SledZig. The situation is a little different in CH4, as the overall WiFi signal power in this channel is about 4dB lower than that in CH1-CH3. We see from Fig. 14(b) that SledZig can make Zigbee transmission successful under QAM-256 even when d_{WZ} is as short as 1m. When the Tx gain increases or decreases, the ZigBee throughput varies, but the general trend does not change. With SledZig, ZigBee links which are nearer the WiFi transmitter have more opportunities to transmit packets successfully. The main reason is that the decreased WiFi signal power shortens the WiFi carrier sense range for ZigBee (d_{CS}^W in Fig. 5(a)).

3) Impact of d_Z: We also conduct experiments to investigate how the ZigBee performance can be affected by the ZigBee link distance d_Z under continuous WiFi transmissions. We use the ZigBee channel of CH4, and set d_{WZ} to be 6m to make ZigBee Tx have the opportunity to transmit packets even under the normal WiFi signal. We then change the distance d_Z slightly from 1m to 2m to test the ZigBee throughput. The results are shown in Fig. 15. We see that when d_Z decreases to 1.6m, the ZigBee throughput is nearly zero, as the ZigBee signal is too weak compared to the WiFi signal, making SINR (signal to interference and noise ratio) below the required threshold. SledZig brings little throughput improvement in this case even under QAM-256 due to the high power of WiFi preamble. This experiment is under continuous WiFi transmissions where the ZigBee



(a) CH1-CH3.



(b) CH4.

Fig. 14. The ZigBee throughput in terms of d_{WZ} under continuous WiFi transmission.

payload inevitably overlaps with the WiFi preamble. In the situation with lower WiFi traffic, SledZig can still mitigate the interference to improve ZigBee network performance.

4) Impact of WiFi Traffic: The previous experiments are conducted under continuous WiFi transmissions. Actually, when the WiFi data rate decreases, the ZigBee throughput can be further improved. In Fig. 14(a) we see that, when the distance d_{WZ} is less than 3m in CH3, all the mechanisms have very poor performance under continuously WiFi transmission. We then conduct experiments to investigate the impact of WiFi data traffic. We fix d_{WZ} to be 1m, fix d_Z to be 0.5m, where the ZigBee link has high probability to be interfered by the WiFi signal according to the tested RSSI. We change the parameter of duration ratio to measure the ZigBee performance in this situation. The duration ratio is defined as the ratio of the WiFi data transmission duration in the channel. The value represents the amount of data traffic in the application layer. We change the ratio from

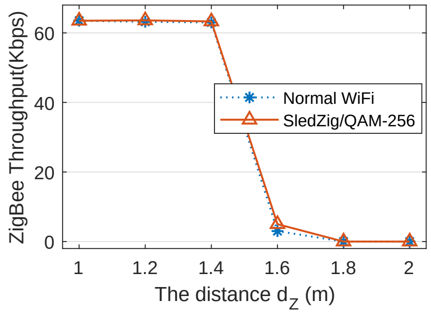
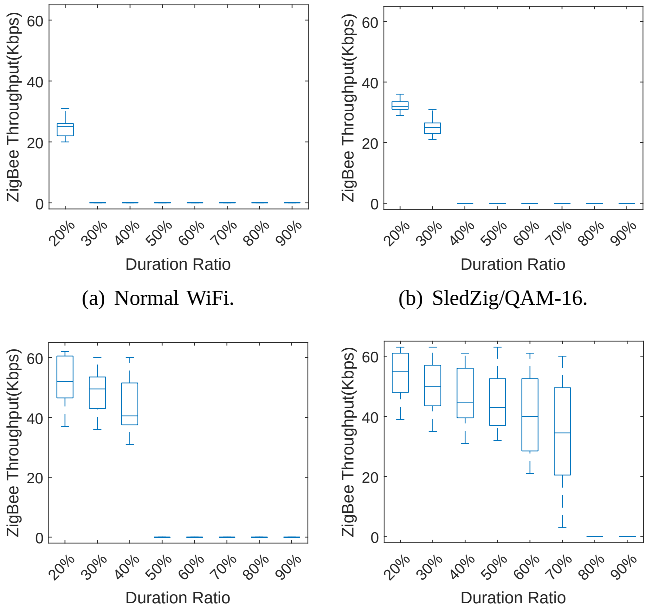


Fig. 15. The ZigBee throughput in terms of d_Z under continuous WiFi transmission.



(a) Normal WiFi.

(b) SledZig/QAM-16.

(c) SledZig/QAM-64.

(d) SledZig/QAM-256.

Fig. 16. The ZigBee throughput under different WiFi data traffic.

20% to 90%, making the WiFi traffic increase gradually. Since the throughput changes in a large range due to random interference situations, we use box plots to show the results, as depicted in Fig. 16. We see that SledZig can improve ZigBee throughput significantly under lower data traffic. The throughput under normal WiFi interference is only about $23Kbps$ when the ratio is 20%, and it is nearly zero when the ratio increases. However, SledZig has high throughput even when the ratio is 70% under QAM-256, 40% under QAM-64 and 20% under QAM-16. Specifically, the average throughput is $34.5Kbps$ when the ratio is 70% under QAM-256, while the lower quartile can still be about $20Kbps$.

D. WiFi Performance

1) Throughput Loss: SledZig requires the WiFi transmitter insert some extra bits to the original WiFi data bits, this process will obviously affect the WiFi throughput. We first make analysis on it.

Table III shows the number of extra bits in one OFDM symbol under different combinations of modulation and coding rate. According to the 802.11 standard, there are two coding rates recommended for QAM-16 and QAM-256, and three coding rates recommended for QAM-64. We see that the number of extra bits is only affected by the QAM modulation and the ZigBee channel, which together determine the positions of significant bits. The number is not affected by the coding rate, because the encoding processes of all the coding rates are based on the 1/2-rate encoding. Other coding rates are achieved through omitting some of the 1/2-rate encoded bits, and the omitted bits have no effect on the significant bits.

TABLE III
THE NUMBER OF EXTRA BITS UNDER DIFFERENT SETTINGS.

Modulation	Coding Rate	No. of bits per OFDM symbol	No. of extra bits (CH1-CH3)	No. of extra bits (CH4)
QAM-16	1/2	96	14	10
	2/3	144	14	10
QAM-64	2/3	192	24	20
	3/4	216	28	20
	5/6	240	28	20
QAM-256	3/4	288	42	30
	5/6	320	42	30

The throughput loss of WiFi data transmission under the combination of three QAM modulations and the possible coding rates is shown in Table IV. We see that the throughput loss ranges from 6.94% to 14.58%. It decreases with the coding rate under each QAM modulation, because the number of WiFi data bits in each OFDM symbol increases while the number of extra bits remains unchanged. Specifically, the situations of QAM-16 with 1/2-rate encoding, QAM-64 with 2/3-rate encoding, and QAM-256 with 3/4-rate encoding under CH1-CH3 have the highest loss of 14.58%, while QAM-16 with 2/3-rate encoding under CH4 has the lowest loss of 6.94%. In general, the throughput loss for CH4 is lower than that for CH1-CH3, due to fewer extra bits.

2) Impact of ZigBee Interference: According to SledZig design, the decreased WiFi signal power leads to more concurrent ZigBee transmissions. Another question here is, whether the ZigBee transmission can in turn interfere with WiFi data transmission. Actually, we do not see the BER (bit error rate) increase of WiFi transmission in the experiments. We then investigate why this happens.

The minimum WiFi SNR that is required to achieve successful transmission for different WiFi settings has been thoroughly studied, as shown in Table. IV. The SNR ranges from $11dB$ to $31dB$. To figure out the impact of ZigBee signal on a WiFi receiver, we let WiFi Tx and ZigBee Tx transmit packets respectively, and let WiFi Rx collect RSSI for each kind of signals. Fig. 17 shows the collected RSSI at the WiFi Rx in terms of the distance from WiFi Tx or ZigBee Tx. We find that the receiving power from ZigBee is much lower than that from WiFi. Specifically, when the distance is $0.5m$, the ZigBee signal power at WiFi Rx is as low as $-85dB$, which is $30dB$ lower than the WiFi signal. The value approximates to the background noise when the distance reaches $1m$. This extremely low power of ZigBee signal at the WiFi device is not only due to its low transmission power, but also because that the ZigBee signal power within the $2MHz$ channel is averaged in the WiFi $20MHz$, making it about $10dB$ lower

TABLE IV
THE WIFI THROUGHPUT LOSS UNDER DIFFERENT SETTINGS.

Modulation	Coding Rate	Min. SNR (dB)	Throughput Loss (CH1-CH3)	Throughput Loss (CH4)
QAM-16	1/2	11	14.58%	10.42%
	2/3	15	9.72%	6.94%
QAM-64	2/3	18	14.58%	10.42%
	3/4	20	12.96%	9.26%
	5/6	25	11.67%	8.33%
QAM-256	3/4	29	14.58%	11.72%
	5/6	31	13.12%	9.37%

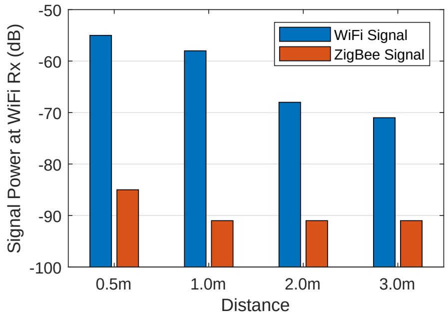


Fig. 17. The collected RSSI at the WiFi receiver with WiFi and ZigBee signals.

than that in the $2MHz$ channel. Thus, the ZigBee signal has little impact on the WiFi data transmission. In extreme cases when ZigBee may interfere with the WiFi transmission, the WiFi link can adapt to the settings with lower SNR threshold to enable data transmission.

VI. Related Works

A. Cross-Technology Coexistence

Cross-Technology coexistence has been an important issue for a long time. Existing works can be divided into two categories: interference avoidance and interference resistance.

Interference resistance mechanisms utilize PHY layer solutions to combat CTI. BuzzBuzz [7] designs new ZigBee packet with more redundancy to mitigate WiFi interference. ZIMO [8] separates WiFi and Zigbee signals into different data streams by using the technologies of MIMO (Multiple-Input Multiple-Output) and interference cancellation. CrossZig [4] and PolarScout [9] make ZigBee devices detect the presence of CTI in a corrupted packet and then recover the packet. These schemes always require hardware modifications or even new transceiver design, which cannot be applied to current devices.

Interference avoidance has attracted much more research interest. Some methods avoid CTI through exchanging coordinated information among heterogeneous devices for protocol design. For example, CBT [10], Weeble [11] and WiCop [12] improve the visibility of ZigBee to WiFi through making ZigBee devices transmit specially designed signals, so that WiFi devices can keep silence during ZigBee transmissions. Gsense [13] makes a WiFi device transmit coordination information to ZigBee devices through a customized preamble, thus to schedule their transmissions. In recent years, some methods utilize the emerging cross-technology communication (CTC) [14]–[17] to achieve interference management by enabling explicit coordination between heterogeneous devices [18]–[23]. For instance, ECC [18] makes a WiFi AP coordinate data transmissions of all the WiFi and ZigBee devices to avoid interference, thus achieves high network throughput; ECT [19] designs the network layer for CTC and lets a server schedule ZigBee transmissions; Chiron [20] designs a customized gateway to enable concurrent transmissions of WiFi and ZigBee data streams in the same frequency band to reduce the transmission delay; BiCord [23] utilizes bidirectional coordination among heterogeneous devices for efficient RF channel allocation. These mechanisms always induce extra packet transmission and require substantial modifications on the standard.

Some other methods avoid CTI through making heterogeneous devices working on different frequency bands [24]–[26]. For example, G-Bee [25] lets a ZigBee device first identify the 802.11b WiFi channel and then transmit its own data packets on the guard band of WiFi traffic to avoid CTI; it requires all the WiFi devices to work on non-overlapped channels, which is hard to be satisfied in the crowded ISM band. EmBee [26] makes a WiFi device reserve the channel for ZigBee transmission through designing null subcarriers; it requires hardware modification as this process is incompatible with the standard WiFi transmission process. By comparison, SledZig can still work in the crowded ISM band without any PHY or MAC modification.

We also see some works focusing on identifying heterogeneous signals to make proper channel access decisions. For example, SoNIC [27], TIIM [28], Smoggy-Link [29] and E-CCA [30] make a device detect the type of interference through using machine learning classifiers. EmBee [26] and LoFi [31] identify the heterogeneous signals and channels through analyzing the signal features. We consider that these mechanisms can work with SledZig to make it more flexible to use, as the WiFi devices can decrease signal power adaptively according to the identified ZigBee channel.

B. WiFi payload encoding

Recent years have seen several works on designing signals through encoding WiFi payload for CTC data transmission. WEBee [14] designs the WiFi payload to make the WiFi signal emulate a ZigBee signal, which can be detected correctly by a standard ZigBee receiver. BlueFi [15] extends the similar idea to the WiFi-to-Bluetooth scenario, where a WiFi device can transmit a standard Bluetooth signal through carefully designing the WiFi payload. These methods have rigorous requirements for the encoded signal, thus all the WiFi payload are used for CTC data transmission, although the WiFi channel is $20MHz$ or more but the ZigBee and Bluetooth channels are only $2MHz$ and $1MHz$, respectively. SLEM [32] and OfdmFi [33] achieves symbol-level energy modulation to deliver CTC information through inserting extra bits to the original WiFi data bits. However, these methods have the limitation that, the QAM points cannot always be the designated lowest or highest ones, which may significantly affect the performance. On the contrary, SledZig can make the QAM points be the ideal ones through inserting a few bits, and the WiFi signal can still deliver the original WiFi packets successfully with significant power decreasing on the ZigBee channel. We note that SymBee [34] adopts payload encoding to achieve ZigBee to WiFi CTC transmission; it works at ZigBee devices and its basic idea is totally different from this work.

VII. Conclusion

In this paper, we propose SledZig to enable coexistence of heterogeneous wireless devices, so as to improve the network performance. SledZig decreases the WiFi signal power on the ZigBee channel through making constellation points in the overlapped subcarriers with the lowest power. It can be achieved through encoding the WiFi payload to generate the transmit bits; when the transmit bits are passed through the WiFi transmission process, the signal power on the ZigBee channel can be decreased naturally. SledZig is fully compatible with WiFi and ZigBee standard, thus can be deployed to commercial devices easily. We implement and evaluate SledZig on hardware testbed, and experimental results show that SledZig can effectively increase ZigBee transmissions and improve its performance over a WiFi channel with as low as 6.94% WiFi throughput loss.

Acknowledgments

This research was supported in part by the China NSFC Grant (62072317, 61802263, U2001207, 61872248), Guangdong NSF 2017A030312008, Shenzhen Science and Technology Foundation (ZDSYS20190902092853047, R2020A045), Faculty Research Fund of Shenzhen University (2019052, 860/000002110322), the Project of DEGP (No.2019KCXTD005, 2021ZDZX1068), the Guangdong “Pearl River Talent Recruitment Program” under Grant 2019ZT08X603. Ruitao Xie is the corresponding author.

References

[1] Cisco, “Cisco annual internet report (20182023) white paper,” 2020.

[2] Verified Market Research, “Global zigbee market size by standards, by application, geographic scope and forecast,” 2020.

[3] R. Chen and W. Gao, “Enabling cross-technology coexistence for extremely weak wireless devices,” in *Proc. of the IEEE INFOCOM*, 2019.

[4] A. Hithnawi, S. Li, H. Shafagh, J. Gross, and S. Duquennoy, “CrossZig: combating cross-technology interference in low-power wireless networks,” in *Proc. of the ACM/IEEE IPSN*, 2016.

[5] IEEE Computer Society. 802.11, “Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 5: Enhancements for higher throughput,” 2009.

[6] IEEE Computer Society. 802.15.4, “IEEE Standard for low-rate wireless networks,” 2016.

[7] C.-J. M. Liang, N. B. Priyantha, J. Liu, and A. Terzis, “Surviving wifi interference in low power ZigBee networks,” in *Proc. of the ACM SenSys*, 2010.

[8] Y. Yan, P. Yang, X. Li, Y. Tao, L. Zhang, and L. You, “ZIMO: building cross-technology MIMO to harmonize zigbee smog with WiFi flash without intervention,” in *Proc. of the ACM MobiCom*, 2013.

[9] C. Shao, H. Park, H. Roh, W. Lee, and H. Kim, “PolarScout: Wi-Fi interference-resilient ZigBee communication via shell-shaping,” *IEEE/ACM Transactions on Networking*, vol. 28, no. 4, pp. 1587–1600, Aug. 2020.

[10] X. Zhang and G. S. Kang, “Enabling coexistence of heterogeneous wireless systems: case for ZigBee and WiFi,” in *Proc. of the ACM MobiHoc*, 2011.

[11] R. Boidar, R. Chandra, and D. Gunawardena, “Weeble: Enabling low-power nodes to coexist with high-power nodes in white space networks,” in *Proc. of the ACM CoNEXT*, 2012.

[12] 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 *Proc. of the IEEE RTSS*, 2011.

[13] X. Zhang and K. G. Shin, “Gap sense: lightweight coordination of heterogeneous wireless devices,” in *Proc. of the IEEE INFOCOM*, 2013.

[14] Z. Li and T. He, “WEBee: physical-layer cross-technology communication via emulation,” in *Proc. of the ACM MobiCom*, 2017.

[15] H.-W. Cho and K. G. Shin, “BlueFi: Bluetooth over WiFi,” in *Proc. of the ACM SIGCOMM*, 2021.

[16] D. Xia, X. Zheng, L. Liu, C. Wang, and H. Ma, “c-Chirp: towards symmetric cross-technology communication over asymmetric channels,” *ACM Transactions on Sensor Networks*, vol. 29, no. 3, pp. 1169–1182, Jun. 2021.

[17] X. Zheng, D. Xia, X. Guo, L. Liu, Y. He, and H. Ma, “Portal: transparent cross-technology opportunistic forwarding for low-power wireless networks,” in *Proc. of the ACM Mobihoc*, 2020.

[18] Z. Yin, Z. Li, S. M. Kim, and T. He, “Explicit channel coordination via cross-technology communication,” in *Proc. of the ACM MobiSys*, 2018.

[19] W. Wang, T. Xie, X. Liu, and T. Zhu, “ECT: Exploiting Cross-Technology Transmission for Reducing Packet Delivery Delay in IoT Networks,” *ACM Transactions on Sensor Networks*, vol. 15, no. 2, Feb. 2019.

[20] Y. Li, Z. Chi, X. Liu, and T. Zhu, “Chiron: concurrent high throughput communication for iot devices.” in *Proc. of the ACM MobiSys*, 2018.

[21] Z. Chi, Y. Li, Z. Huang, H. Sun, and T. Zhu, “Simultaneous bi-directional communications and data forwarding using a single ZigBee data stream,” in *Proc. of the IEEE INFOCOM*, 2019.

[22] W. Chen, Z. Yin, and T. He, “Global cooperation for heterogeneous networks,” in *Proc. of the IEEE INFOCOM*, 2020.

[23] Z. Yu, P. Li, C. A. Boano, Y. He, M. Jin, X. Guo, and X. Zheng, “Bi-Cord: bidirectional coordination among coexisting wireless devices,” in *Proc. of the IEEE ICDCS*, 2021.

[24] Y. Wang, X. Zheng, L. Liu, and H. Ma, “CoHop: quantitative correlation-based channel hopping for low-power wireless networks,” *ACM Transactions on Sensor Networks*, vol. 17, no. 2, pp. 1–29, Jun. 2021.

[25] Y. Chae, S. Wang, and S. M. Kim, “Exploiting WiFi guard band for safeguarded ZigBee,” in *Proc. of the ACM SenSys*, 2018.

[26] R. Chen and W. Gao, “Enabling cross-technology coexistence for extremely weak wireless devices,” in *Proc. of the IEEE INFOCOM*, 2019.

[27] F. Hermans, O. Rensfelt, T. Voigt, E. Ngai, L. Nordn, and P. Gunningberg, “SoNIC: classifying interference in 802.15.4 sensor networks,” in *Proc. of the IEEE IPSN*, 2013.

[28] A. Hithnawi, H. Shafagh, and S. Duquennoy, “TIIM: technology-independent interference mitigation for low-power wireless networks,” in *Proc. of the IEEE IPSN*, 2015.

[29] J. Meng, Y. He, X. Zheng, D. Fang, D. Xu, T. Xing, and X. Chen, “Smoggy-Link: fingerprinting interference for predictable wireless concurrency,” in *Proc. of the IEEE ICNP*, 2016.

[30] W. Zheng, J. Yao, and K. Wu, “Mitigating cross-technology interference in heterogeneous wireless networks based on deep learning,” in *Proc. of the IEEE ICPADS*, 2020.

[31] G. Chen, W. Dong, and J. Lv, “LoFi: enabling 2.4GHz LoRa and WiFi coexistence by detecting extremely weak signals,” in *Proc. of the IEEE INFOCOM*, 2021.

[32] J. Yao, X. Zheng, R. Xie, and K. Wu, “Cross-technology communication for heterogeneous wireless devices through symbol-level energy modulation,” *IEEE Transactions on Mobile Computing*, Mar. 2021. DOI: 10.1109/TMC.2021.3065998.

[33] P. Gawowicz, A. Zubow, S. Bayhan, and A. Wolisz, “Punched cards over the air: cross-technology communication between LTE-U/LAA and WiFi,” in *Proc. of the IEEE WoWMoM*, 2020.

[34] S. Wang, S. M. Kim, and T. He, “Symbol-level cross-technology communication via payload encoding,” in *Proc. of the IEEE ICDCS*, 2018.