Cross-Technology Communication for Heterogeneous Wireless Devices through Symbol-Level Energy Modulation

Junmei Yao, Xiaolong Zheng, Ruitao Xie and Kaishun Wu

Abstract—The coexistence of heterogeneous devices in wireless networks brings a new topic on cross-technology communication (CTC) to improve the coexistence efficiency and boost collaboration among these devices. Current advances on CTC mainly fall into two categories, *physical-layer CTC* and *packet-level energy modulation* (PLEM). The *physical-layer CTC* achieves a high CTC data rate, but with channel incompatible to commercial devices, making it hard to be deployed in current wireless networks. PLEM is channel and physical layer compatible, but with two main drawbacks of the low CTC data rate and MAC incompatibility, which will induce severe interference to the other devices' normal data transmissions. In this paper, we propose symbol-level energy modulation (SLEM), the first CTC method that is fully compatible with current devices in both channel and the physical/MAC layer processes, having the ability to be deployed in commercial wireless networks smoothly. SLEM inserts extra bits to WiFi data bits to generate the transmitting bits, so as to adjust the energy levels of WiFi symbols to deliver CTC information. We make theoretical analysis to figure out the performance of both CTC and WiFi transmissions. We also conduct experiments to demonstrate the feasibility of SLEM and its performance under different network situations.

Index Terms—Keywords: Wireless Networks; Cross-Technology Communication; WiFi; ZigBee.

1 Introduction

With the widespread proliferation of the Internet of Things (IoT), it is becoming a common phenomenon that numerous devices with different wireless technologies (e.g., WiFi, ZigBee and Bluetooth) share the unlicensed ISM spectrum. The coexistence of these devices brings a new topic on cross-technology communication (CTC), which establishes direct communication among heterogenous devices [1]–[3]. CTC has the potential to bring about quite a few benefits and applications [2], [4], [5], such as combating the cross-technology interference through exchanging coordination information among the devices [4], enabling the WiFi AP to directly control the Zigbee devices deployed for smart home [2], and etc.

Current works on WiFi to ZigBee CTC design are generally achieved through two methods: *physical-layer CTC* and *packet-level energy modulation* (PLEM). The *physical-layer CTC* makes a commercial WiFi device transmit ZigBee signals directly through signal emulation, such that this signal can be detected through ZigBee normal demodulation process [1]. It achieves the high CTC data rate comparable to a ZigBee radio. However, the main problem is that it is is hard to be deployed in commercial wireless networks due to the channel incompatible. According to WEBee [1] design, the pilot/null OFDM (orthogonal frequency division modulation) subcarriers should be avoided in the CTC transmission [1], but our investigation on the standard WiFi and ZigBee channels finds that no combination satisfies this requirement; CTC can only be achieved when the WiFi central frequency is adjusted to a non-standard one. Although some commercial chips surely have this ability, it is hardly permitted in commercial networks since all devices should comply with the standards.

The PLEM methods convey cross-technology information through employing the packet-level features, like packet transmission duration [6], [7], duration pattern [8], [9], and interval [2], [10], so that receivers can detect the information through energy sensing. This kind of methods are compatible with commercial devices in channel and the physical layer process. However, they have two main drawbacks. Besides with the low CTC data rate that can only be up to about $1Kbps$, they are incompatible with the commercial devices in the MAC (Medium Access Control) layer process. The commercial WiFi devices generally adopt CSMA/CA (carrier sense multiple access/collision avoidance) to access the channel and avoid interference through random backoff [11], while PLEM requires the devices to access the channel in encoded time patterns, which are usually in contradiction with random backoff. Thus, the CTC transmission will easily induce severe interference to the other devices' normal transmissions.

In this paper, we propose symbol-level energy modulation (SLEM), the first CTC method that is fully compatible with commercial devices in both channel and the physical/MAC layer processes, having the ability to be deployed in current wireless networks smoothly. SLEM delivers CTC information through exploiting known features of the WiFi signal. Since

- *Junmei Yao, Ruitao Xie and Kaishun Wu are with the College of Computer Science and Software Engineering, Shenzhen University, China. Email: {yaojunmei, xie, wu}@szu.edu.cn.*
- *Xiaolong Zheng is with the School of Computer Science, Beijing University of Posts and Telecommunications, China. Email: zhengxiaolong@bupt.edu.cn.*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3065998, IEEE Transactions on Mobile Computing

QAM (quadrature amplitude modulation) adopted in WiFi devices naturally has the feature of energy modulation, CTC can be achieved through adjusting the QAM points of the transmitting signal to make each symbol have distinguishable low or high power levels. Specifically, at the transmitter side, SLEM designs the transmitting bits (called SLEM bits) according to the WiFi data bits and CTC data bits. When these bits are passed through the standard WiFi transmission process, the transmitting signal exhibits the characteristic of energy modulation for CTC and can deliver both kinds of information concurrently. After receiving this signal, the ZigBee receiver decodes its data bits through energy sensing, while the WiFi receiver first decodes the SLEM bits and then recovers the original WiFi data bits.

Compared to PLEM, SLEM coincides with commercial devices in the MAC process, thus avoids unnecessary interference to current wireless networks. SLEM is more flexible than PLEM as the CTC bits can be delivered at any time when a WiFi packet is transmitting. This design also benefits SLEM with much higher CTC date rate than PLEM. In addition, it is worthy to note that SLEM has no channel incompatibility problem as *physical-layer CTC*, since a single pilot subcarrier has much less effect on the overall energy of multiple subcarriers.

The key contributions are summarized as follows:

- We design SLEM, the first CTC method that is fully compatible with commercial devices in both the channel usage and the physical/MAC layer processes. SLEM inserts extra bits to WiFi data bits to deliver both kinds of information concurrently.
- We give theoretical analysis for the SLEM performance in delivering both the CTC and WiFi data bits, compared to a typical PLEM method. The results demonstrate that SLEM can achieve much higher CTC data rate at the cost of requiring higher SNR, while the WiFi transmission has about 10% decrease on the data rate.
- We implement and evaluate SLEM on hardware testbed based on the USRP N210 and TelosB platforms. The experimental results reveal that SLEM can achieve a robust and fast concurrent transmissions of CTC and WiFi.

The rest of this paper is organized as follows. Section 2 gives the motivation of SLEM. Section 3 gives the overview of SLEM. Section 4 describes the SLEM design. Section 5 gives the design of SLEM bits generation when considering channel coding. Section 6 provides theoretical performance analysis of SLEM. Section 7 demonstrates the SLEM performance by hardware experiments. Section 8 introduces the related works. Section 9 concludes this paper.

2 Motivation

This section illustrates the motivation of SLEM through observing on both WiFi and ZigBee transmission processes.

2.1 Opportunity for CTC within One WiFi Packet

Some current packet-level energy modulation (PLEM) mechanisms have an assumption that the WiFi transmission duration τ_w is very small. Actually, from the point of view of WiFi protocol design, it is more efficient to have the data packet transmitted with larger τ_w, as this would induce smaller transmission overhead to the WiFi network, such as backoffs, control frame transmissions, etc.

Retrospecting the history of WiFi standards – the IEEE 802.11 family, we could see that they have made great efforts on avoiding extremely small value of transmission duration τ_w. As demonstrated in Table. 1, with the increase of the physical layer data rate R from $11Mbps$ in 802.11b and $54Mbps$ in 802.11a/g, to $600Mbps$ in 802.11n [11] and $>6Gbps$ in 802.11ac [12], the MAC layer is also revised to enlarge the maximum packet length L_w [1] to achieve comparable τ_w values among the standards, since τ_w is inversely proportional to the data rate R_w ($\tau_w = L_w/R_w$), as listed in Table 1. Especially, 802.11n and 802.11ac introduce A-MPDU (Aggregated - MAC Protocol Data Unit) to accomplish the super-length packet.

TABLE 1
Attribute comparisons of different 802.11.

Attribute	**802.11a/g**	**802.11n**	**802.11ac**
Maximum R_w	$54Mbps$	$600Mbps$	$6.9Gbps$
Maximum L_w	4095bytes	65535bytes	4,692,480bytes
Maximum τ_w	$5.46ms$	$5.484ms$	$5.484ms$

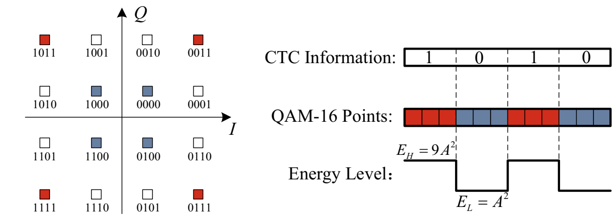
Compared to the WiFi packet transmission duration that is up to $5.48ms$, the RSSI (Received Signal Strength Indicator) sampling interval of ZigBee devices is extremely small, e.g., $32\mu s$ for TelosB [3]. Accordingly, if a WiFi packet contains a set of segments which have different energy, the ZigBee device is possible to obtain the energy changes through RSSI sampling. Thus, we have the opportunity to accomplish a CTC transmission within one WiFi packet through energy modulation.

2.2 Opportunity for Symbol-Level Energy Modulation

We then investigate the WiFi transmission process to answer the question about how to achieve energy modulation within one WiFi packet.

At the WiFi transmission side, the data bits will be mapped to constellation points after passing through the QAM (Quadrature Amplitude Modulation) module. QAM modulation can be regarded as a combination of both phase and amplitude modulations. Fig. 1(a) depicts the QAM-16 constellation points, each of which represents $M = log_2(16) = 4$ data bits. Among these 16 points, the four red points have 3× amplitude over the four blue points, corresponding to 9× energy difference. This characteristic provides us with an opportunity for symbol-level energy modulation within a single WiFi packet. For instance, if we let the blue points carry the CTC information ‘0’ and let the red points carry ‘1’, the two kinds of information will possess distinguishable

[1] Here the packet length indicates the PLCP (Physical Layer Convergence Procedure) payload size.



(a) QAM-16 constellation points (b) Energy modulation for CTC

Fig. 1. An example of symbol-level energy modulation.

energy levels and then have the possibility to be discerned at ZigBee, as shown in Fig. 1(b).

3 SLEM OVERVIEW

This section first introduces the CTC application scenario, then gives the overview of SLEM architecture accordingly.

3.1 CTC Application Scenario

CTC is a method to enhance performance and boost new applications under the coexistence of heterogeneous wireless networks. For example, in the WiFi and ZigBee networks, a WiFi device can transmit coordination information to ZigBee devices for interference management [4] to improve the network throughput, a WiFi device can directly control the ZigBee devices deployed for smart home [2], and ZigBee devices can even download the update files from a WiFi device. In these situations, CTC just works supplementarily while the normal WiFi and ZigBee data transmissions are still dominant in the wireless networks.

Therefore, each device may need to receive signals from heterogeneous devices. For example, in the coexistence of WiFi and ZigBee networks, as shown in Fig. 2, there exists the normal data exchange within both the WiFi and ZigBee devices; meanwhile, the WiFi AP or WiFi clients may need to transmit CTC information through the SLEM signal [2] to the ZigBee devices. Thus, a WiFi device may receive either a WiFi or a SLEM signal, and a ZigBee device may receive either a ZigBee or a SLEM signal. In this situation, a receiver needs to identify the signal type at first to decode the data bits correctly; especially, a ZigBee receiver should further classify the SLEM signal from other non-ZigBee signals at the beginning of the received signal, otherwise it will keep on decoding all the signals, this is obviously inappropriate for the low-cost and low-power ZigBee device. Therefore, it is a key issue to make a device quickly discern the incoming signal type.

3.2 SLEM Architecture

Fig. 3 depicts the architecture of SLEM under the scenario of Fig. 2. The white block represents that this process already exists in commercial devices based on standards, while the grey block represents new component of SLEM. The following figures are described in the same way.

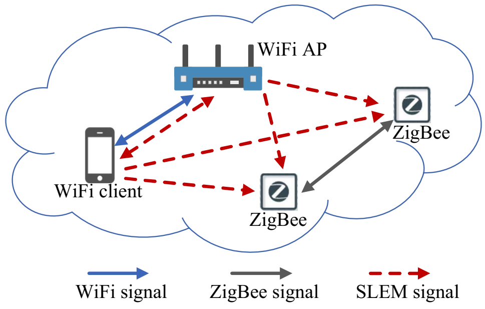


Fig. 2. The WiFi and ZigBee coexistence scenario.

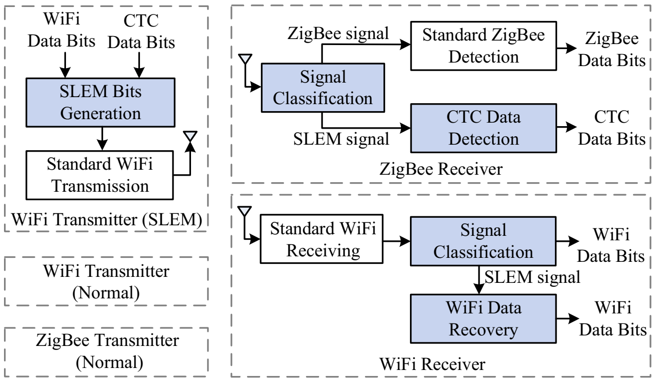


Fig. 3. SLEM Architecture.

There are three kinds of transmitted signals, including the SLEM signal from a WiFi transmitter for CTC transmission, the WiFi signal from a normal WiFi transmitter, and the ZigBee signal from a normal ZigBee transmitter. To generate a SLEM signal, the WiFi transmitter first generates the transmitting bits, which is called SLEM bits in this paper, according to both the WiFi data bits and CTC data bits. The SLEM bits are the payload of the WiFi packet, they will be passed through the standard WiFi transmission process and finally be transmitted after Radio Frequency (RF) front end.

When receiving a signal, the ZigBee receiver first determine whether it is a ZigBee or SLEM signal; for a ZigBee signal, the receiver conducts standard ZigBee detection process to obtain the data bits; for a SLEM signal, the receiver conducts CTC data detection to get the CTC data bits. For a WiFi receiver, it first conducts the standard WiFi receiving process to obtain the SLEM bits, then determine whether it is a WiFi or SLEM signal; for a WiFi signal, the SLEM bits are regarded as the data bits directly; for a SLEM signal, the receiver will conduct a recovery process to get the original WiFi data bits.

4 SLEM DESIGN

This section gives the detailed design of SLEM at the WiFi transmitter side, the ZigBee receiver side and the WiFi receiver side, respectively.

[2] For the ease of description, in this paper, we let the term 'SLEM signal' represent the signal of a WiFi data packet attached with CTC bits, and let the term 'WiFi signal' represent the signal of a normal WiFi data packet.

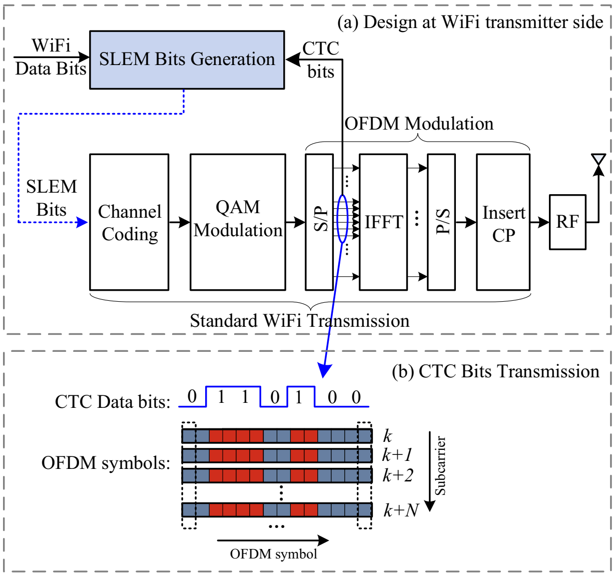


Fig. 4. Architecture of SLEM at the transmitter side. SLEM bits are first generated according to the WiFi and CTC data bits, then passed through the standard WiFi transmission process for signal transmission (a), the transmitted signal can deliver the CTC bits concurrently (c).

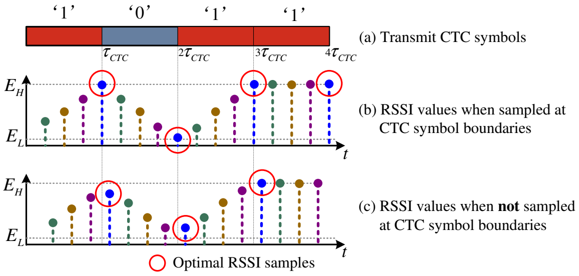


Fig. 5. An example of RSSI sampling at the receiver side.

4.1 SLEM Design at the Transmitter Side

4.1.1 Overview

As shown in Fig 4(a), the SLEM design at the transmitter side is to generate the SLEM bits according to the WiFi and CTC data bits. When the SLEM bits are passed through the standard WiFi transmission process, the transmitted SLEM signal contains both the desired energy modulated CTC signal and the WiFi signal, thus can deliver both kinds of data bits concurrently.

In the standard transmission process illustrated in Fig. 4(a), the data bits are first transformed to complex symbols after QAM modulation, and mapped into OFDM subcarriers after passing through the S/P (serial-to-parallel) module, then output as the time-domain OFDM symbols after IFFT (inverse fast fourier transform) and P/S (parallel-to-serial) processes; afterwards, each OFDM symbol is inserted with cyclic prefix (CP) to eliminate the inter-symbol interference; the signal will finally be transmitted after RF front end.

When the SLEM bits are passed through this standard process, the constellation points within the overlapped subcarriers will carry the CTC information through energy modulation, as shown in Fig. 4(b). For example, when QAM-16 is adopted, the points will be '$xx00$' if the OFDM symbol should have low power; otherwise, the points will be '$xx11$'. Here 'x' indicates the bit can either be '1' or '0'.

From Fig. 4(b) we see that, the energy levels of the OFDM symbols are determined by both the the CTC data bits and the CTC symbol duration τ_{CTC}, which finally determine the number of OFDM symbols required for one CTC bit transmission, denoted by N_{OFDM}. Since the OFDM symbol duration is fixed as $\tau_{OFDM} = 4\mu s$, we get $N_{OFDM} = \frac{\tau_{CTC}}{\tau_{OFDM}}$.

In the following parts, we will illustrate the impact of τ_{CTC} on the SLEM performance. We will also analyze the impact of cyclic prefix in the standard process. The SLEM bits generation is highly related with detailed processes in channel coding, we leave its design to Section 5.

4.1.2 CTC Symbol Duration Determination

The CTC symbol duration τ_{CTC} is decided by the characteristic of the ZigBee receiver, and in turn determines the parameter N_s at the transmitter side.

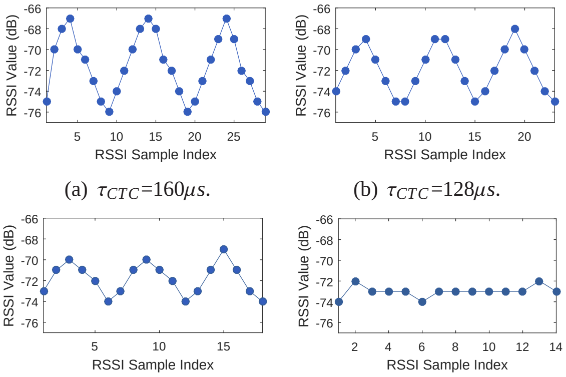
For the TelosB platform we use as ZigBee in this paper, the RSSI samples are generated every $32\mu s$, although the values are averaged over $128\mu s$. Under this circumstance, the CTC symbol duration τ_{CTC} and even the RSSI sample positions would affect the RSSI values at TelosB. Fig. 5 shows an example of the CTC bits $\{1, 0, 1, 1\}$ transmitted through a series of symbols with energy $\{E_H, E_L, E_H, E_H\}$, where E_H and E_L indicate the high and low energy levels, and $E_H = 9 \times E_L$ under QAM-16; the symbol duration is $128\mu s$. We see that the RSSI sample values are very different within one CTC symbol duration. When the RSSI values are not sampled at the CTC symbol boundaries, as shown in Fig. 5(c), the RSSI distance d_{RSSI}, which is the distance between the maximum and minimum RSSI values, will be much smaller than that in Fig. 5(b) when RSSI values are sampled at the CTC symbol boundaries, and the shorter distance will result in lower performance.

To demonstrate the effect of τ_{CTC}, we let USRP N210 transmit a set of CTC bits $\{1, 0, 1, 0, 1, 0\}$, and let τ_{CTC} equal to $160\mu s$, $128\mu s$, $96\mu s$ and $64\mu s$, respectively. The RSSI samples collected at TelosB under each situation are shown in Fig. 6. We see that the RSSI values demonstrate regular peaks and dips when $\tau_{CTC} \geq 96\mu s$. Specifically, the maximum d_{RSSI} is about $9dB$ when $\tau_{CTC} = 160\mu s$ (Fig. 6(a)), it has about $2dB$ and $5dB$ loss when τ_{CTC} is $128\mu s$ and $96\mu s$, respectively. As to $\tau_{CTC} = 64\mu s$ in Fig. 6(d), the peaks and dips disappear and the CTC bits can not be detected at all.

4.1.3 The Impact of Cyclic Prefixing

As shown in Fig. 4, in the WiFi transmission process, each OFDM symbol is inserted with cyclic prefixing (CP) to provide guard interval with the previous OFDM symbol, so as to eliminate the inter-symbol interference. Here we study the impact of CP on the SLEM performance.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3065998, IEEE Transactions on Mobile Computing



(a) τ_{CTC}=160μs.

(b) τ_{CTC}=128μs.

(c) τ_{CTC}=96μs.

(d) τ_{CTC}=64μs.

Fig. 6. The RSSI samples under different CTC symbol durations through experiments.

Each 20MHz WiFi channel contains 64 OFDM subcarriers, each of which is filled with a complex symbol after serial-to-parallel (S/P). We let $X(k)$ ($k \in [1, 64]$) indicate the complex symbol in the kth subcarrier. The frequency domain signal $\{X(k)\}$ will be transformed to the time domain signal $\{x(n)\}$ after 64-point IFFT, that is, $x(n) = \frac{1}{64}\sum_{k=1}^{64} X(k)e^{j\frac{2\pi kn}{64}}, n = 1 \sim 64$. The signal $\{x(n)\}$ lasts for 3.2μs and will be inserted a 0.8μs prefix $\{x'(l)\}$ ($l = 1 \sim 16$). $\{x'(l)\}$ is simply a copy of the end of $\{x(n)\}$, and $x'(l) = x(l + 48)$. For example, $x'(1) = x(49) = \frac{1}{64}\sum_{k=1}^{64} X(k)e^{j\frac{2\pi k\cdot 49}{64}}$. Thus, we see that the inserting of $\{x'(l)\}$ does not change the frequency components $\{X(k)\}$ in the signal. Since the energy of each SLEM symbol is determined by the energy of $X(k)$ in several adjacent subcarriers, it is obvious that CP will not change the energy in these subcarriers, thus have no impact on the performance of SLEM.

4.2 SLEM Design at ZigBee Receiver Side

In this part, we first introduce the signal classification process to distinguish the normal ZigBee signal and the SLEM signal at a ZigBee receiver, including the signal classification overview, CTC preamble design and CTC preamble detection design. After that, we describe the detailed design for CTC data detection.

4.2.1 Signal Classification Overview

After receiving a signal, the ZigBee receiver should first determine whether this is a normal ZigBee signal or a SLEM signal, then conduct the proper receiving process for data detection.

The normal ZigBee transmission utilizes a preamble field in the ZigBee frame to indicate the arrival of a ZigBee signal [13]; thus, the receiver can easily identify a ZigBee signal through this process. The key issue here is to make it identify a SLEM signal from other signals. In this paper, we use a CTC preamble to indicate the arrival of a SLEM signal at a Zigbee receiver. The detailed design of CTC preamble and its detection will be given in Section 4.2.2 and Section 4.2.3, respectively.

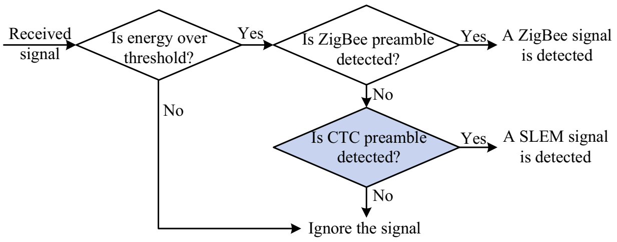

Received signal
Is energy over threshold?
Yes
No
Is ZigBee preamble detected?
Yes
A ZigBee signal is detected
No
Is CTC preamble detected?
Yes
A SLEM signal is detected
No
Ignore the signal

Fig. 7. The signal classification process at ZigBee receiver.

Fig. 7 briefly shows the signal classification process based on the ZigBee preamble and CTC preamble. When the energy of the received signal is over a threshold β_E, the ZigBee device first conducts ZigBee preamble detection. If the ZigBee preamble is detected, it determines a normal ZigBee signal is received and conducts the standard ZigBee detection to obtain the data bits, as shown in Fig. 3; otherwise, it conducts CTC preamble detection, and utilizes CTC data detection to obtain the data bits if the preamble is detected. If both kinds of preamble are not detected, the node will ignore this signal.

4.2.2 CTC Preamble Design

The CTC preamble is designed to indicate the arrival of a SLEM signal at a ZigBee device. The CTC frame format with the CTC preamble is shown in Fig. 8.

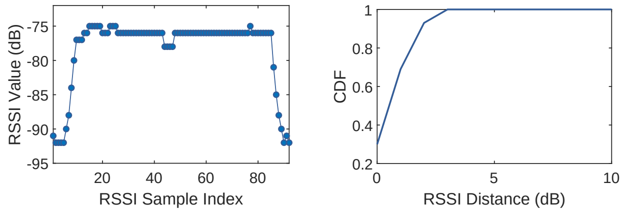
We let the CTC preamble be ‘0101’, which has the fixed energy pattern $\mathbf{E}_{pre} = \{E_L, E_H, E_L, E_H\}$. The most important issue here is to figure out whether the other signals also have this energy pattern, resulting in unnecessary process for CTC data detection. Thus, we analyze the ZigBee and WiFi signals separately.

CTC Preamble
CTC Data

Fig. 8. The CTC frame format.

At first, we note that a normal ZigBee signal cannot exhibit the characteristic of this energy pattern, as ZigBee utilizes OQPSK (Offset-QPSK) modulation and the amplitude of the transmitted signal remains constant. Then we want to figure out whether a normal WiFi signal have this energy pattern. We let USRP N210 transmit standard WiFi signals with QAM-64 and let TelosB collect the RSSI samples, while the WiFi channel is 13 with the central frequency of 2.472GHz, and the overlapped ZigBee channel is 25 with the central frequency of 2.475GHz. The RSSI values of one WiFi packet is depicted in Fig. 21(a). We see that the RSSI values increase at the beginning of the packet obviously, after that, the values remains nearly constant just with some slight variations. We consider that is because the RSSI samples at TelosB indicate the average energy of the WiFi signal within about seven subcarriers and τ_{CTC} time duration. With the random characteristic of the WiFi data bits, the QAM points are randomly distributed, thus the averaged energy levels have little changes. To investigate the RSSI variations within a WiFi packet except the beginning and the tail of it, we collects the RSSI samples of about 100 WiFi packets and calculate the cumulative distributed function (CDF) of the RSSI distance d_{RSSI} for the 100 sets of RSSI samples, the results are shown in Fig. 21(b). We see that d_{RSSI} is below $3dB$ with the probability of 80%, and it is below $5dB$ in nearly all the cases. The results demonstrate that the WiFi signal is hardly discerned as the CTC preamble.

Therefore, the energy pattern $\mathbf{E}_{pre}$ is the unique feature of the SLEM signal, and can be used as the CTC preamble.



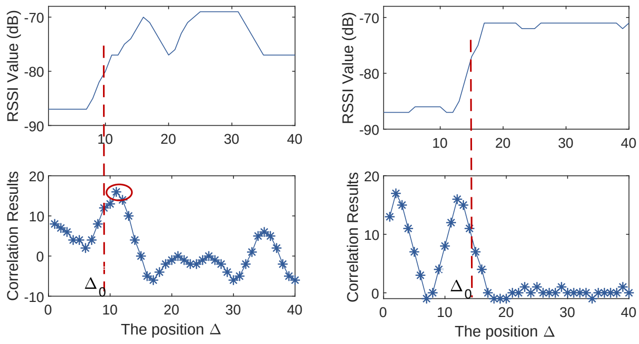
(a) RSSI samples for one WiFi packet.

(b) CDF of RSSI distances for WiFi packets.

Fig. 9. The characteristics of RSSI samples for normal WiFi signals.

4.2.3 CTC Preamble Detection

A ZigBee receiver should detect the CTC preamble to determine whether a SLEM signal is arrived. This process should only be conducted at the beginning of the incoming signal, so that the device can then decode CTC bits if the SLEM signal is received, or turn to energy-saving mode if not. Specifically, CTC preamble detection is to discerned the energy pattern $\mathbf{E}_{pre}$ from the received signal. Here we exploit the cross correlation technology for this process.



(a) SLEM signal.

(b) WiFi signal.

Fig. 10. The correlation results of $\{PRE_k\}$ with a SLEM or normal WiFi signal.

Cross correlation is always utilized to search for a known signal pattern in a long duration. In this context, it needs to be conducted between the RSSI samples $\{r_i\}$ and $\mathbf{E}_{pre}$. Since E_L and E_H varies with many parameters, such as the transmission power and the transmitter-receiver distance, we change $\mathbf{E}_{pre}$ to a fixed pattern $\{PRE_j\} = \{-1, 1, -1, 1\}(j = 1 \sim 4)$ in the calculation process.

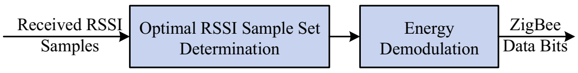


Fig. 11. The CTC receiving process.

Since the RSSI samples are generated every $32\mu s$ at ZigBee, the number of RSSI samples during one CTC symbol is calculated as $N_s = \frac{\tau_{CTC}}{32\mu s}$. Then, the CTC preamble '0101' corresponds to $4 \times N_s$ RSSI samples. At each position Δ in $\{r_i\}$, the receiver picks up the RSSI samples $\{rp_j^\Delta\} = \{r_i, i = \Delta + (j-1) \cdot N_s\}(j = 1 \sim 4)$ for cross correlation calculation, that is, $R_\Delta = \sum_{j=1}^{4} rp_j^\Delta \cdot PRE_j$. When CTC preamble is received and the energy pattern of $\{rp_j^\Delta\}$ matches $\mathbf{E}_{pre}$, the correlation result R_Δ will exhibit a peak value.

To demonstrate the feasibility of CTC preamble detection through this process, we conduct experiments based on the USRP N210 and TelosB. We let USRP N210 transmit normal WiFi and SLEM signals, respectively. The SLEM signals are beginning with the CTC preamble, and $\tau_{CTC} = 160\mu s$ in this situation. We then obtain the RSSI samples at TelosB, and conduct cross correlation between $\{PRE_j\}$ with the received signal at each position Δ. The results are shown in Fig. 10, we see that from the position Δ_0 where the averaged RSSI value of $\{rp_j^\Delta\}$ is over β_E, the correlation results for the SLEM signal have a peak value, while those for the normal WiFi signal remain small. We note that the correlation results before Δ_0 also have some peak values, that is because the RSSI sample set $\{rp_j^\Delta\}$ used for correlation contains the RSSI samples after Δ_0. Therefore, during the cross-correlation process, it is critical to first decide the averaged RSSI value is over β_E, which is set as $-80dB$ in this experiment. Through this way, a ZigBee receiver can determine with a high probability that whether the received signal is a SLEM signal or not.

In real networks, a peak value of R_Δ is determined through comparing R_Δ with a threshold β_{corr}, if R_Δ is over β_{corr}, it is a peak value and CTC preamble is detected; otherwise, it is not and the CTC preamble is not detected. Theoretically, the peak value at Δ is $R_\Delta = 2 \cdot (E_H - E_L) = 2 \cdot d_{RSSI}$. Thus, the value β_{corr} is highly related to d_{RSSI}, which varies with SNR, QAM modulation types and τ_{CTC}. We will discuss in Section 7.3 about the empirical values of β_{corr} through experiments.

4.2.4 CTC Data Detection

When the CTC preamble is detected, the ZigBee receiver begins to decode the CTC data bits through energy sensing. The receiving process is depicted in Fig. 11: the receiver first determines the optimal RSSI sample set from the received RSSI samples, then conducts energy demodulation to obtain the original ZigBee data bits.

1) Optimal RSSI Sample Set Determination

As shown in Fig. 5, the RSSI samples with red circles can represent the energy of the transmitted symbols best, they are regarded as the optimal RSSI sample set. The determination of this sample set contains two steps: (i) obtain the sample set candidates from the RSSI samples $\{r_i\}$, and (ii) determine the optimal one from the candidates.

The sample set candidates, denoted by $\{\hat{r}_i^k\}$ where k is the candidate index, can be easily obtained from $\{r_i\}$, simply through taking the samples from $\{r_i\}$ with the fixed interval N_s and different beginning positions, the number of candidates is N_s. For example, in Fig. 5, the samples with the same color belongs to a candidate; there are $N_s = \frac{128\mu s}{32\mu s} = 4$ candidates in this figure, corresponding to the four colors.

The optimal RSSI sample set is then determined from the sample set candidates. Our observation is that the optimal RSSI sample set has the largest RSSI distance d_{RSSI} compared to the other candidates. As shown in Fig. 5 and Fig. 6, the sample set candidates exhibit different characteristic of RSSI distance, and the one with the largest RSSI distance is obviously optimal. We present a simple method to obtain this optimal RSSI sample set. At first, the mean value of the received RSSI samples $\{r_i\}$ is calculated as $m_r = MEAN(\{r_i\})$. Please note that this calculation should be performed numerically while the RSSI samples obtained from TelosB are expressed in decibels. Then, for each candidate $\{\hat{r}_i^k\}$ ($k \in [1, N_s]$, $i \in [i, N]$ and N is the number of RSSI samples over β_E), the accumulated RSSI distance from m_r is calculated as $d_{RSSI}^k = \sum_{i=1}^{N} \mid \hat{r}_i^k - m_r \mid$. The kth candidate with the largest d_{RSSI}^k is discerned as the optimal sample set, which is denoted by $\{\overline{r_i}\}$.

2) Energy Demodulation

With the optimal RSSI samples $\{\overline{r_i}\}$, the ZigBee node will decode the CTC data bits through energy decoding. The process is pretty simple: if an RSSI value is over a threshold β_s, the corresponding bit is ’1’, otherwise the bit is ’0’.

The threshold β_s can not be fixed due to the varied RSSI values, which changes with the transmission power, the transmitter-receiver distance, etc. Here we simply use the mean value m_r of $\{r_i\}$ as the threshold. As $r_i = x_i + n_i$, where x_i indicates the transmitting signal and n_i is the noise with fixed mean value among the received samples, the value m_r can obviously vary adaptively with background noise.

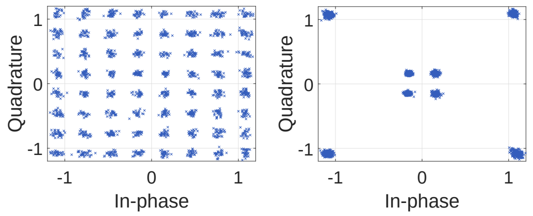
4.3 SLEM Design at WiFi Receiver Side

After receiving the transmitted signal, the WiFi receiver first conducts the 802.11 standard receiving process to obtain the SLEM bits, then determine whether it is a WiFi or SLEM signal. As shown in Fig. 3, for a WiFi signal, the SLEM bits are regarded as the data bits directly; for a SLEM signal, the receiver will conduct a recovery process to get the original WiFi data bits. Here we first introduce the signal classification process, then give the design for WiFi data bits recovery.

4.3.1 Signal Classification

The signal classification between the normal WiFi and SLEM signals can be achieved totally at the receiver side, without any changes at the transmitter side. The key insight of this design is based on the observation that, the SLEM signal within the ZigBee channel possesses much less constellation points compared to the normal WiFi signal. As shown in Fig. 12, when the signal is transmitted with QAM-64 modulation and the CTC bits are conveyed through the ideal low and high power points, the WiFi signal within 7 subcarriers overlapped with the ZigBee channel has 64 constellation points due to the random characteristic of WiFi data bits, while the SLEM signal has only four points. This feature can be exploited as a simple way to classify the two kinds of signals.

We note that signal classification here can also be achieved by simply adding a flag in the WiFi message. However, the flag should be added into the head of the WiFi message, which is always not controlled by the users. On the contrary, this design is fully compatible with commercial devices, as it does not require any modification on hardware.



(a) Normal WiFi signal. (b) SLEM signal.

Fig. 12. Constellation points of two kinds of signals with QAM-64 modulation at the receiver side.

4.3.2 WiFi Data Recovery

When the received signal is determined as a SLEM signal after the signal classification module, the process of WiFi data recovery is quite easy.

As will be discussed in Section 5, the WiFi transmitter will insert some bits to the WiFi data bits at fixed positions to generate the SLEM bits. The receiver just needs to first determine these positions according to the modulation type and overlapped channels, then eliminate bits in these positions from the SLEM bits to obtain the original WiFi data bits.

5 Design for SLEM Bits Generation

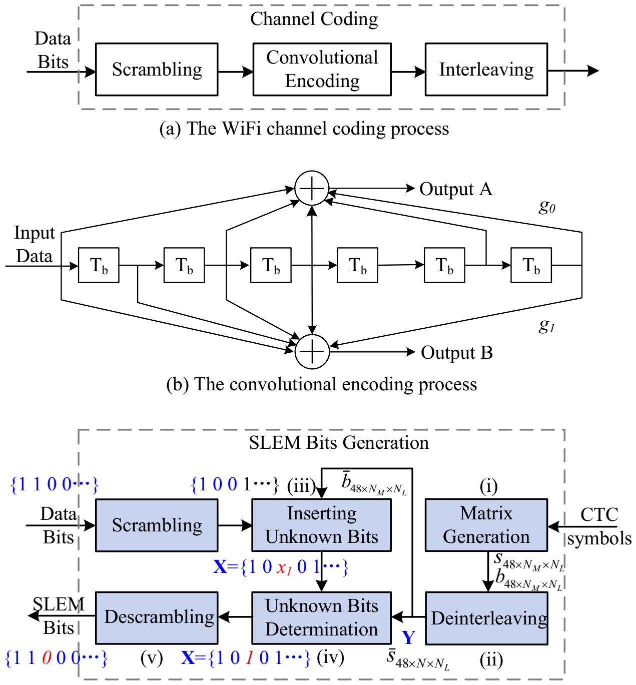
As described in Section 4.1, SLEM bits generation is the key issue in the protocol design. However, it is nontrivial when channel coding is considered. In this section, we first illustrate the standard channel coding, then describe detailed process of generating SLEM bits as well as its limitation.

5.1 Preliminary of Channel Coding

The standard WiFi channel coding process includes scrambling, convolutional encoding and interleaving, as shown in Fig. 13(a). Here we introduce the three parts respectively.

5.1.1 Scrambling

At the beginning of channel coding, WiFi data bits are scrambled to avoid long sequences of bits of the same value. Data scrambling is performed by XORing the data bits with a sequence of pseudo-random bits which have the fixed pattern. Thus, this process is a one-by-one mapping from data bits to scrambled bits. A receiver can recover the original data bits through XORing the receiving bits with the same sequence of pseudo-random bits.



(a) The WiFi channel coding process

(b) The convolutional encoding process

(c) The SLEM bits generation process

Fig. 13. SLEM bits generation with channel coding.

5.1.2 Convolutional Encoding

Convolutional encoding makes the WiFi transmissions more resilient to interference and noise through adding redundancy. The 802.11 standard recommends several coding rate, such as 1/2, 2/3 and 3/4, etc. The rate-1/2 convolutional encoding is shown in Fig. 13(b). It uses the generator polynomials $g_0 = (1011011)_2$ and $g_1 = (1111001)_2$. Here ‘A’ is the output of encoder after modulo-2 addition of the input data bit, 2^{th}, 3^{th}, 5^{th} and 6^{th} delay element based on g_0; ‘B’ is the output of encoder after modulo-2 addition of the input data bit, 1^{th}, 2^{th}, 3^{th} and 6^{th} delay element based on g_1. This process generates two encoded bits ‘A’ and ‘B’ for each input bit, while the bit ‘A’ shall be output before bit ‘B’. Other coding rates of 2/3 and 3/4 can be achieved by employing puncturing, which omits some of the encoded bits to reduce the number of transmitted bits and increase the coding rate.

This process can be formulated as a matrix $\mathbf{M}$ to indicate the relationship between the input bits $\mathbf{X}$ and the coded bits $\mathbf{Y}$ in the Galois Field GF(2) [1]. That is:

$$\mathbf{M} \times_{GF(2)} \mathbf{X} = \mathbf{Y}. \tag{1}$$

Due to page limit, we just analysis the SLEM bits generation under the 1/2 coding rate in this paper. The generation process under other rates are similar.

5.1.3 Interleaving

Interleaving is used to make error correction more robust with respect to burst errors. In the WiFi transmission process, interleaving is achieved by a two-step permutation. The first permutation make the adjacent coded bits mapped to nonadjacent subcarriers, and the second permutation makes the adjacent coded bits mapped alternately onto less and more significant bits of the signal constellations. Generally, this process is also a one-by-one mapping from input bits to interleaved bits. A receiver can recover the original input bits through conducting the demapping process.

5.2 SLEM Bits Generation

The SLEM bits will be generated through inserting extra bits to the original WiFi data bits, so as to adjust the constellation points in the overlapped subcarriers to deliver CTC information. As depicted in Fig. 4, the constellation points should have low power when this OFDM symbol is utilized to transmit CTC bit ‘0’, otherwise they should have high power. Each of the low power and high power point has two significance bits under QAM-16, as the shadowed ones shown in Table 2. Similarly, the point has four and six significance bits under QAM-64 and QAM-256 respectively. Thus, the key issue in SLEM bits generation is to insert extra bits to WiFi data bits at specific positions, so as to generate the significance bits in corresponding subcarriers and OFDM symbols according to the CTC information.

Here we first give an overview of this process, then analyze its limitation.

TABLE 2
An illustration of significance bits under QAM-16.

	Low Power				High Power			
Significance Bits in Points	0	0	0	0	0	0	1	1
	0	1	0	0	0	1	1	1
	1	0	0	0	1	0	1	1
	1	1	0	0	1	1	1	1
Masks	0	0	1	1	0	0	1	1

5.2.1 Overview

Fig. 13(c) shows the detailed processes of SLEM bits generation, from step (i) to step (vi).

At step (i), the WiFi AP first generates two matrices according to the CTC bits, including the bit significance matrix S_{48,N_M,N_L} and the bit mask matrix B_{48,N_M,N_L}, where 48 is the number of data subcarriers in each OFDM symbol, $N_M = log_2 M$ for M-level QAM modulation, N_L indicates the number of OFDM symbols. Specifically, $B_{i,j,k} = 1$ if the j-th bit of the point in i-th subcarrier is the significance bit, and the corresponding $S_{i,j,k}$ is determined by the required power level of the k-th OFDM symbol. Both matrices S_{48,N_M,N_L} and B_{48,N_M,N_L} will be transformed to one-dimensional arrays $s_{48\times N_M\times N_L}$ and $b_{48\times N_M\times N_L}$ respectively, and then passed through deinterleaving as $\bar{s}_{48\times N_M\times N_L}$ and $\bar{b}_{48\times N_M\times N_L}$ at step (ii). At step (iii), according to $\bar{b}_{48\times N_M\times N_L}$, the AP knows at which positions it should insert bits to generates the CTC information, it then inserts unknown bits $\{x_i\}$ to the scrambled data bits at these positions to generate the data stream $\mathbf{X}$. At step (iv), the unknown bits $\{x_i\}$ in $\mathbf{X}$ are determined according to the convolutional encoding process and the significance bits in $\mathbf{Y} = \bar{s}_{48\times N_M\times N_L}$. At step (v), the data stream $\mathbf{X}$ are descrambled to output the SLEM bits. When the SLEM bits are passed through the standard WiFi transmission process shown in Fig. 4, both the WiFi and CTC information can be delivered concurrently.

5.2.2 Limitation

The unknown bits $\{x_i\}$ in $\mathbf{X}$ should be determined according to the significance bits $\{y_k\}$ in $\mathbf{Y}$ and the convolutional encoding process, following Eq. 1. However, in some cases Eq. 1 can not completely hold.

We let $\overline{\mathbf{X}} = \{x_i\}(i \in [1,p])$ and $\overline{\mathbf{Y}} = \{y_k\}(k \in [1,q])$ for the easy of description, where p and q indicate the number of unknown bits and significance bits, respectively. The unknown bits $\overline{\mathbf{X}} = \{x_i\}$ shall be calculated through a simplified equation of Eq. 1, that is,

$$\widetilde{\mathbf{M}} \times_{GF(2)} \overline{\mathbf{X}} = \widetilde{\mathbf{Y}}, \tag{2}$$

where $\widetilde{\mathbf{M}}$ is a $q \times p$ matrix, $\widetilde{\mathbf{Y}} = \overline{\mathbf{Y}} + \mathbf{B}$ is a $q \times 1$ vector, $\mathbf{B}$ is determined by $\mathbf{M}$ and the known bits before $\{x_i\}$ in $\mathbf{X}$.

If all the significance bits are sparsely distributed in $\mathbf{Y}$, one bit x_i shall be inserted to determine the corresponding significance bit y_i; in this situation, $p = q$, $r(\widetilde{\mathbf{M}}) = r(\widetilde{\mathbf{M}}, \widetilde{\mathbf{Y}}) = q$ ($r(\cdot)$ indicates the rank of a matrix), it is easy to obtain $\overline{\mathbf{X}} = \{x_i\}$ through Eq. (2). However, in some cases when two significance bits come together in $\mathbf{Y}$, one bit x_i shall be inserted to determine two significance bits y_k and y_{k+1}, $p < q$ in this situation and there is a high probability of $r(\widetilde{\mathbf{M}}) < r(\widetilde{\mathbf{M}}, \widetilde{\mathbf{Y}})$, making Eq. (2) have no solution. Fig. 14 shows an example of this case. If two significance bits y_k and y_{k+1} come together in $\mathbf{Y}$ (y_k and y_{k+1} correspond to the two outputs bits 'A' and 'B' in Fig. 13(b)), and an input x_i should be inserted to generate y_k and y_{k+1} simultaneously, there will be no solution for x_i. When this situation occurs, the QAM points in the overlapped subcarriers may not be the designated ones, which will affect the performance of CTC transmission. Fortunately, although the significance bits come together in matrix $s_{48 \times N_M \times N_L}$, most of them are sparsely distributed in $\mathbf{Y} = \overline{s}_{48 \times N_M \times N_L}$ after deinterleaving, making SLEM still feasible. We note that in the situation of Fig. 14, Eq. (2) indeed has the possibility to have solution, that depends on the previous bits from a_1 to a_6.

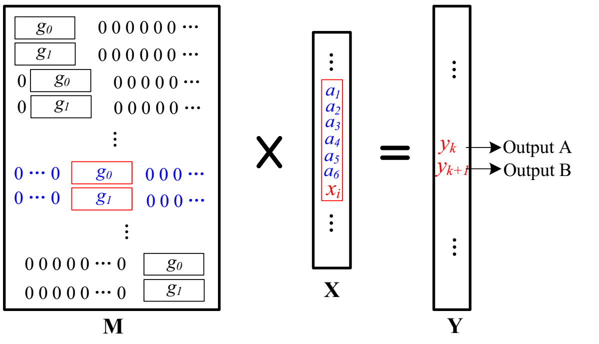


Fig. 14. A case in convolutional encoding when significance bits y_k and y_{k+1} come together. $a_1 \sim a_6$ are known bits before the inserted bit x_i.

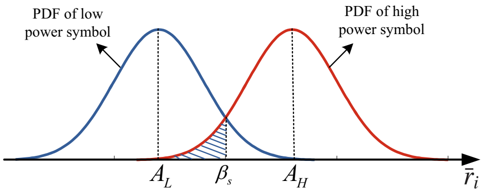


Fig. 15. The probability density function (PDF) of CTC symbols with low and high energy levels.

6 Theoretical Analysis

In this part, we try to theoretically figure out the SLEM performance of delivering both the WiFi and CTC data bits compared to PLEM.

6.1 Analysis for CTC Transmission

From the perspective of CTC transmission, the bits are transmitted by symbols with two energy levels E_H and E_L. We let $\{x_i\}$ indicate the transmitting signal, and let the CTC symbol duration τ_{CTC} be large enough such that the optimal RSSI sample sets $\{\overline{r_i}\}$ can represent the actual energy of the received signal.

We have:

$$\overline{r_i} = x_i + n_i, \tag{3}$$

where the noise n_i is the additive white gaussian noise (AWGN) and $n_i \sim \mathcal{N}(0, \sigma^2)$. Then, the received signal r_i also subjects to the normal distribution with the mean of x_i and the variance of σ^2. That means, for the symbol x_i with mean $A_L = \sqrt{E_L}$, the received signal $r_i \sim \mathcal{N}(A_L, \sigma^2)$; for the symbol x_i with mean $A_H = \sqrt{E_H}$, $r_i \sim \mathcal{N}(A_H, \sigma^2)$.

Fig. 15 depicts the relationship of the probability density function (PDF) of the two symbols. With $\{\overline{r_i}\}$, the symbol is determined to be '1' if $\overline{r_i} > \beta_s$, otherwise it represents '0'. Here β_s is set as $\frac{A_L + A_H}{2}$. In real network situations, the number of '1' or '0' is nearly equal within a packet, thus, β_s here is approximate to that in Section 4.2.4.

A symbol error occurs when '1'/'0' is transmitted but '0'/'1' is detected, as the shaded area shown in Fig. 15, then the symbol error probability is calculated as:

$$P_e = P(\overline{r_i} < \beta_s) = Q(\frac{A_H - A_L}{2\sigma}). \tag{4}$$

where $Q(x) = \int_x^{+\infty} \frac{1}{\sqrt{2\pi}} exp(-\frac{1}{2}t^2)dt$. Using this equation and $SNR = \frac{E_s}{\sigma^2} = \frac{E_L + E_H}{2\sigma^2}$, we can calculate the theoretical SER values of each kind of mechanisms.

6.1.1 PLEM

PLEM makes the symbol '0' or '1' transmitted through the absence or presence of a data packet. It can be regarded as a special case of the aforementioned situation where $E_L = 0$. As $SNR = \frac{E_H}{2\sigma^2}$, we have:

$$P_e(PLEM) = Q(\sqrt{\frac{SNR}{2}}).$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2021.3065998, IEEE Transactions on Mobile Computing

6.1.2 SLEM

With the aforementioned analysis, the subcarriers for CTC transmission may include pilot, which is not controlled and will obviously affect the SLEM performance. Here we first analyze the performance without pilot subcarrier, based on which we give the SLEM performance with pilot. For the ease of description, we let A_{SH} and A_{SL} be the amplitude of the high power and low power constellation points, respectively.

(i) SLEM without pilot: The SER of SLEM changes with the QAM modulation types. In this situation, $A_H = A_{SH}$, $A_L = A_{SL}$. When QAM-16 is adopted, as shown in Fig. 1, $A_H = 3A_L$ and we have:

$$P_e^{w/o}(SLEM\_16P) = Q(\sqrt{\frac{SNR}{5}}),$$

Similarly, for QAM-64, $A_H = 7A_L$ and:

$$P_e^{w/o}(SLEM\_64P) = Q(\frac{3}{5}\sqrt{SNR}).$$

For QAM-256, $A_H = 15A_L$ and:

$$P_e^{w/o}(SLEM\_256P) = Q(\sqrt{\frac{49}{113}SNR}).$$

(ii) SLEM with pilot: According to the standards, WiFi should utilize seven subcarriers to convey the CTC information, among which one is the pilot subcarrier and six are the data subcarriers. The pilots are with BPSK modulation and the pilot subcarriers are filled with $\{-1, 1\}$. As shown in Fig. 1, the amplitude of a pilot is $\frac{A_{SH}}{\sqrt{2}}$, which is also suitable for other modulation types.

Then, the amplitude of a high power or low power CTC symbol is the averaged value among these seven subcarriers, that is, $A_H = \frac{1}{7}(6 + \frac{1}{\sqrt{2}})A_{SH}$, $A_L = \frac{1}{7}(6 + \frac{1}{\sqrt{2}})A_{SL}$. We have:

$$P_e^{w/}(SLEM\_16P) = Q(\frac{6}{7}\sqrt{\frac{SNR}{5}}),$$

$$P_e^{w/}(SLEM\_64P) = Q(\frac{6}{7}\cdot\frac{3}{5}\sqrt{SNR}),$$

$$P_e^{w/}(SLEM\_256P) = Q(\frac{6}{7}\sqrt{\frac{49}{113}SNR}),$$

6.1.3 Summary

Fig. 16 depicts the theoretical SER of PLEM and SLEM. The results demonstrate that the SER of CTC transmission under QAM-16 is much lower than PLEM, while that of SLEM under QAM-256 highly approaches PLEM. In addition, we see that when the modulation type changes from QAM-16 to QAM-64, the CTC has remarkable performance improvement; however, when it changes from QAM-64 to QAM-256, the improvement is not so significant. We also see that pilot will largely affect the SLEM performance due to the decrease of RSSI distance; especially, with pilot, SER under QAM-256 is even higher than that under QAM-64 without pilot.

We should note that the performance shown in Fig. 16 is the theoretical upper bound for SLEM. When the WiFi and CTC information are transmitted concurrently through different subcarriers, the performance may surely decreased due to the limitation in in the SLEM bits generation. However, when only CTC information is transmitted, this performance can be achieved as we may design specific WiFi data bits which make the CTC information transmitted with designated constellation points.

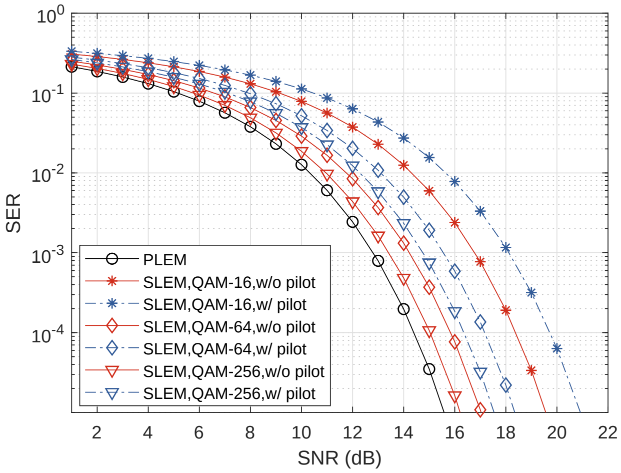
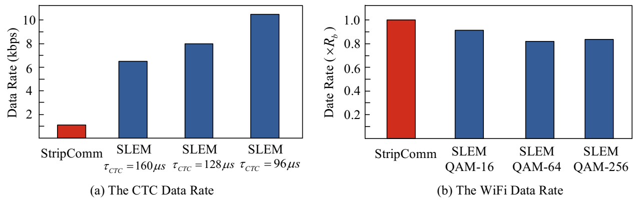


Fig. 16. SER of PLEM and SLEM in terms of SNR.



(a) The CTC Data Rate

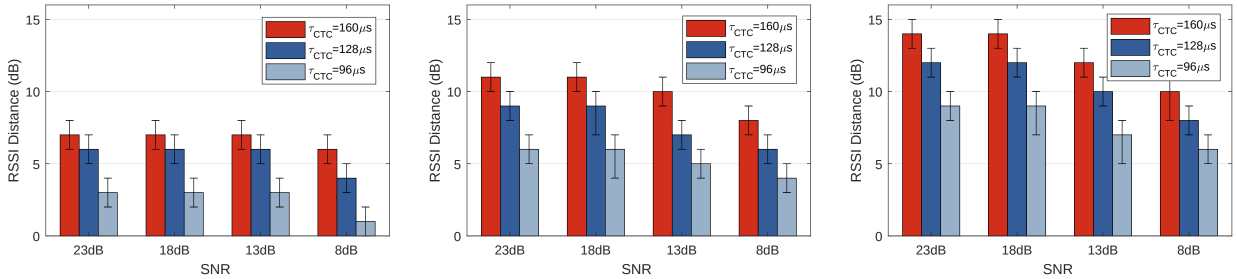
(b) The WiFi Data Rate

Fig. 17. Comparison of data rate for CTC transmission.

6.2 Analysis for Data Rate

The CTC data rate of SLEM is mainly determined by τ_{CTC}, and its theoretical value is $\frac{1}{\tau_{CTC}}$. Fig. 17(a) demonstrates the comparison of CTC data rate under different mechanisms. Since SLEM can deliver a set of CTC data bits through one WiFi packet, its data rate can outperform all the PLEM mechanisms. Compared to the state-of-art StripComm [3] which can achieve about $1.1kbps$ data rate, SLEM has at least $6kbps$ data rate. Especially, when $\tau_{CTC} = 96\mu s$, the data rate can be up be about $10kbps$.

For the WiFi data transmission, it is hard to give a specified data rate R_w as it is related to many factors except the QAM modulation types. Thus, we use StripComm [3] as the baseline, set its data rate as R_b to evaluate that of SLEM. Since the ZigBee channel is $2MHz$ and the bandwidth of each subcarrier is $312.5KHz$, besides pilot, six out of the 48 data subcarriers should be utilized for CTC data transmission. Since the SLEM bits generation is to insert extra bits to the original WiFi data bits to generate the energy-modulated CTC information, the WiFi data rate of SLEM depends on the ratio of the inserted bits, which varies with the modulation type. As shown in Fig. 17(b), WiFi data rates of SLEM under QAM-16, QAM-64 and QAM-256 are $90.6\% \times R_b$, $86.1\% \times R_b$ and $86.4\% \times R_b$, respectively.



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

Fig. 18. The RSSI distance in terms of SNR and τ_{CTC} under different QAM modulation types.

This analysis give the data rate at the situation of only one stream of CTC transmission. When parallel CTC streams are transmitted concurrently through one WiFi packet, the CTC data rate increases and WiFi data rate decreases accordingly.

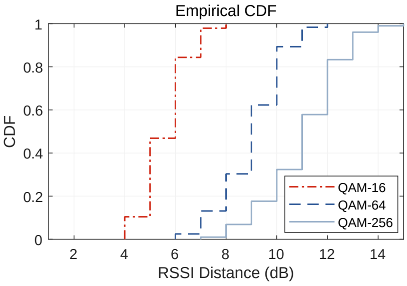


Fig. 19. The cumulative distribution function (CDF) of RSSI distance under different QAM modulations.

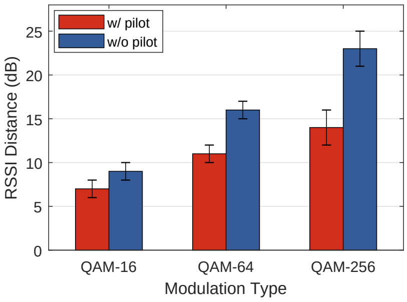


Fig. 20. The comparison of RSSI distance with or without pilot subcarrier under different QAM modulations.

7 Experimental Evaluation

7.1 Experimental Settings

We implement a prototype of SLEM containing the Universal Software Radio Peripheral (USRP) N210 and TelosB. We use USRP N210 to generate the WiFi signals following the IEEE 802.11 standard, while the SLEM bits are obtained through MATLAB based on the WiFi and CTC data bits, both of which are generated randomly. We use TelosB, a commercial ZigBee platform, to collect the RSSI samples of the CTC signal. For each WiFi data packet, the CTC bits required to transmit is first fixed and the WiFi data transmission duration is set accordingly. Other parameters such as τ_{CTC}, QAM modulation type and SNR vary as required. In addition, if not specified, the USRP N210 works at 2.472GHz, which is the 13*th* WiFi channel at 2.4GHz, and TelosB works at 2.470GHz, which is the 24*th* ZigBee channel. We also test other combinations of WiFi and ZigBee channels, the results have little change except when the ZigBee channel is overlapped with the WiFi null subcarriers. The experiments are conducted in a chamber where the required SNR is easy to get. Actually, we also conduct some experiments in other situations, and find that environments have little impact on RSSI distance under the same parameter settings, that is mainly due to the feature of energy modulation on CTC transmission.

7.2 RSSI Distance

The RSSI distance obviously affects the CTC performance. It varies with a set of parameters like τ_{CTC}, SNR, QAM modulation types, and even WiFi data bits which would affect the constellation points in overlapped subcarriers, as described in Section 5.2.2. To make thorough study on how RSSI distance are affected by these parameters, we adjust some parameters and fix the others in each experiment.

We first test the RSSI distance with the impact of τ_{CTC}, SNR and QAM modulation types. To eliminate the impact of the WiFi data bits, we let the low power and high power points for CTC transmissions be designated ones. In addition, we let the USRP N210 transmit WiFi signals with a fixed power, and adjust the distance between USRP N210 and TelosB to make the received CTC signals have required SNRs, since the background noise varies very slightly. Fig. 18 demonstrates that the RSSI distance increases obviously with the QAM level, and the adoption of QAM-256 results in the largest RSSI distance, as shown in Fig. 18(c). In the case of high SNR situations, such as 23dB, the RSSI distance is about 14dB with QAM-256, while the value is 11dB with QAM-64, and 7dB with QAM-16. In addition, the RSSI distance decreases significantly with the decrease of τ_{CTC} and SNR. For example, in Fig. 18(a), the RSSI distance with τ_{CTC} of 96μs has about 5dB decrease compared to τ_{CTC} of 160μs when SNR is as high as 23dB.

As analysis in Section 5.2.2, some significance bits may not be guaranteed and the situation depends on the WiFi data bits before the inserted bits. We then test the RSSI distance with different WiFi data bits. To eliminate the impact of other parameters, we set τ_{CTC} be $160\mu s$ and SNR be $23dB$. In the experiment, we fix the CTC data bits, then randomly generate the WiFi data bits, and obtain the SLEM bits according to the SLEM bits generation process; we finally feed the SLEM bits into USRP N210 for transmission. We repeat this process for one hundred times and make statistics of the RSSI distance, then show them through the cumulative distribution function (CDF) in Fig. 19. We see that the RSSI distance is over $6dB$ with about 80% probability under QAM-16; it is over $10dB$ in 61.5% cases under QAM-64, and over $12dB$ in 57.8% cases under QAM-256.

Finally, since pilot is within the overlapped subcarriers and will definitely affect the SLEM performance, as analyzed in Section 6, here we intend to further evaluate its impact through experiments. We set τ_{CTC} be $160\mu s$, SNR be $23dB$, the low power and high power symbols be the designated ones, to eliminate the impact of these parameters. For the situation without pilot, we let USRP N210 work at $2.474GHz$ and TelosB work at $24th$ channel. The comparison of RSSI distance with and without pilot subcarrier under the three modulation types is shown in Fig. 20. We see that the pilot subcarrier does affect the SLEM performance significantly. The RSSI distance has about $2dB$ decrease under QAM-16, $5dB$ decrease under QAM-64, and even $9dB$ decrease under QAM-256.

At last, it should be noted that pilot subcarrier has much higher impact on the RSSI distance compared to the unsatisfactory constellation points due to the limitation in SLEM bits generation. That is because the limitation in SLEM bits generation only makes the designated low or high power constellation points change to adjacent ones, while pilot induces dramatic shift to the points, thus largely affects the averaged RSSI distance.

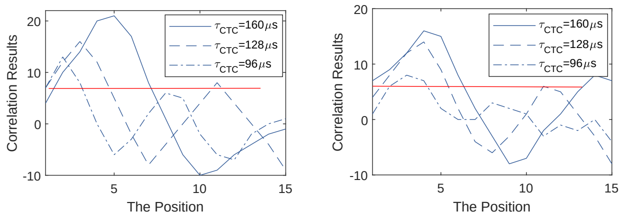
7.3 CTC Preamble Detection

Since CTC preamble detection is the key step to determine the arrival of a CTC packet, we then conduct experiments to measure its performance.

As described in Section 4.2.3, the CTC preamble detection process is to conduct cross correlation between $\{PRE_j\} = \{-1, 1, -1, 1\}$ with the received RSSI samples with interval N_s, when the average energy is over β_E. The CTC preamble is determined to be detected if the correlation result $R_\Delta > \beta_{corr}$. The main objective of this experiment is to obtain the typical values of β_{corr}, and measure the performance of CTC preamble detection.

We let USRP N210 transmit CTC packets beginning with the CTC preamble '0101' under different τ_{CTC}, SNR and QAM modulation types. Fig. 21 shows the correlation results of 15 continuous positions when the average energy is over $\beta_E = -80dB$, under QAM-64 with two SNR situations. The high SNR is $25dB$ and the low SNR is $8dB$. We see that the correlation results have higher values under higher QAM order, higher SNR and larger τ_{CTC}. Thus, we set the value of β_{corr} mainly based on the worst case. For QAM-64, we set $\beta_{corr} = 8$, as the red lines in Fig. 21(a) and Fig. 21(b). The results of QAM-16 and QAM-256 are not shown here, but the corresponding values of β_{corr} can be set in the same way.

We then test the performance of CTC preamble detection through the detection ratio under different situations, the results are shown in Fig. 22. We see that the CTC preamble can be detected with the probability of about 100% under high SNR situations, such as $18dB$ and $23dB$. Errors increase dramatically under low SNR situations. However, the detection ratio is still very high under low SNR and QAM-256, as the correlation results in this situation is still far higher than β_{corr}. We do not show the case of QAM-16 under $\tau = 96\mu s$ due to its bad performance.



(a) High SNR.

(b) Low SNR

Fig. 21. The correlation results of the CTC preamble under QAM-64.

7.4 Performance of CTC Transmission

We then intend to investigate the performance of CTC transmission under different parameter settings.

7.4.1 Single CTC Transmission

We first test the performance when only one CTC stream is transmitted. The factors which affect the RSSI distance finally affect the SER and PER significantly, such as the received SNR, τ_{CTC}, and QAM modulation types. The WiFi bits are generated randomly in this experiment. Fig. 23 depicts the SER of CTC transmission in terms of SNR under each QAM modulation type, while τ_{CTC} is set to be $160\mu s$ and $96\mu s$, respectively. We do not show the case of QAM-16 under $\tau = 96\mu s$ here due to its bad performance. We see that when SNR is above $20dB$, SER is approximate to zero nearly in all the situations; it has obvious increase when SNR decreases from $18dB$. We also see that although the smaller τ_{CTC}

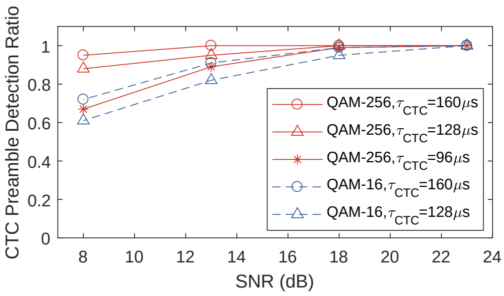


Fig. 22. The CTC preamble detection ratio under different situations.

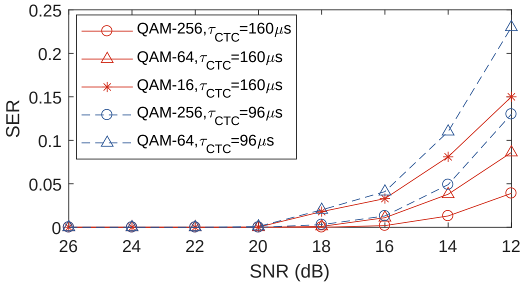


Fig. 23. SER of CTC transmission in terms of SNR under different modulation types and τ_{CTC}.

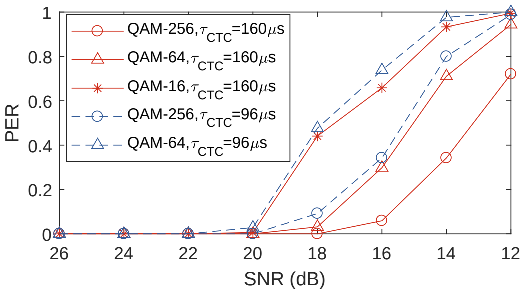


Fig. 24. PER of CTC transmission in terms of SNR under different modulation types and τ_{CTC}.

will inevitably result in more errors, it still exhibits a quite good performance when QAM-256 is adopted. Especially, the combination of QAM-256 and $\tau_{CTC} = 96\mu s$ results in a better performance than the combination of QAM-16 and $\tau_{CTC} = 160\mu s$.

Fig. 23 depicts the PER of CTC transmission in each situation of Fig. 23, and the CTC packet length is $32 bits$. We see that when QAM-256 is adopted and $\tau_{CTC} = 160\mu s$, PER is below 0.1 when SNR is above $16dB$. For the other situations, the same performance can be achieved only when SNR is above $20dB$.

The experimental performance is much lower than the theoretical counterpart, as we use the simplified model in the theoretical analysis. Actually, the results in Fig. 16 can be regarded as the upper bound of CTC transmission.

7.4.2 Parallel CTC Transmissions

The SLEM design naturally supports parallel CTC transmissions. Since the ZigBee channels are overlapped with different WiFi subcarriers, parallel CTC streams can be transmitted concurrently without mutual interference. The only issue that affects the performance is in the SLEM bits generation. After deinterleaving, two significance bits from different CTC streams may be together and fit the case shown in Fig. 14, making one of the bits unsatisfied. However, this case is relatively rare, and we only capture a very small amount of performance degradation. Fig. 25 shows PER of CTC under the situations of single stream and two parallel streams under QAM-64 and $\tau_{CTC} = 160\mu s$. We see that the performance degradation of parallel CTC transmission is negligible compared to the single stream.

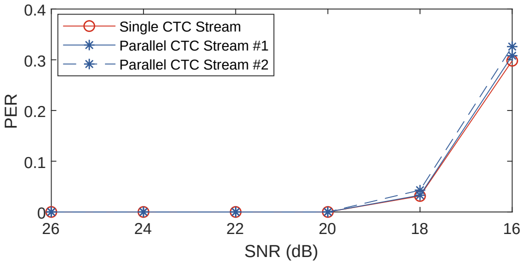
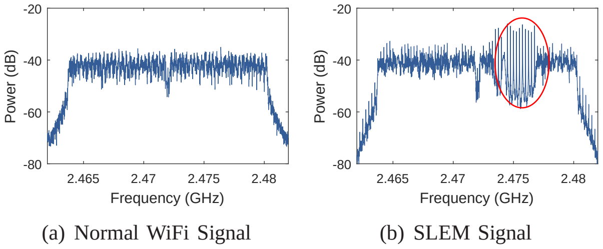


Fig. 25. SER of parallel CTC transmissions.

7.5 Analysis for WiFi Transmission

We finally intend to analyze whether the SLEM design affects the WiFi signal transmissions.

Fig. 26 depicts the spectrum density of both WiFi and SLEM signals under QAM-16, while both the WiFi and CTC data bits are generated randomly. The SLEM signal is a portion of a frame and contains both the low and high SLEM symbols. We see that the SLEM signal obviously exhibits much higher signal power variance within the ZigBee channel, while that value out of the ZigBee channel remains similar to the WiFi signal. The spectrum density is barely affected by the the value τ_{CTC}, and that under both QAM-64 and QAM-256 has the similar feature. Fig. 26 impels us to figure out whether the SLEM design affects the WiFi transmissions.



(a) Normal WiFi Signal

(b) SLEM Signal

Fig. 26. Spectrum density of two kinds of packets.

One key related characteristic which affects the WiFi performance is the peak-to-average-ratio (PAPR) of the time domain signal, as the higher PAPR results in lower performance due to degrading the efficiency of the power amplifier, thus may lead to lower transmission power with the same transmission gain. We get the cumulative distribution function (CDF) of PAPR for the two kinds of signals, the results are shown in Fig. 27. We see that the PAPR of SLEM signal looks similar with that of the WiFi signal. We also test the receiving power levels of the two kinds of signals under the same configurations, such as the transmission gain and transmitter-receiver distance, and find that they have no distinguishable difference. These results show that the SLEM design has little effect on the WiFi signal transmissions except the slightly decreased data rate, as analyzed in Section 6.2.

8 Related Work

Recent years have seen numerous research works on CTC between heterogenous devices, such as CTC between WiFi

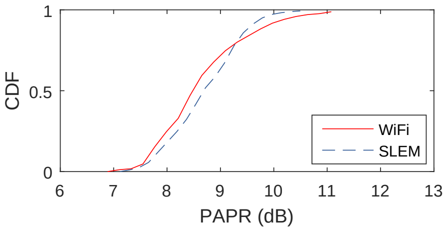


Fig. 27. The cumulative distribution function (CDF) of PAPR for both the normal WiFi and SLEM signals under QAM-16.

and ZigBee [1], [2], [8], [10], [14]–[20], between WiFi and Bluetooth [21], [22], between ZigBee and Bluetooth [23], [24], between LTE and ZigBee [25], between LoRa and ZigBee [26], between LoRa and Bluetooth [27], and between RFID and WiFi [28]. Some researchers have further analyzed other problems of the CTC system, such as attack [29], [30] and network throughput improvement [31], [32].

This paper focuses on WiFi to ZigBee CTC design, and the previous works mainly fall into two categories: *physical-layer CTC* and PLEM.

8.1 Physical-layer CTC

The *physical-layer CTC* was firstly proposed by WEBee [1] to make a commercial WiFi device elaborately construct the WiFi payload to transmit a ZigBee-compliant packet through signal emulation, which would then be detected by a ZigBee device directly. It has the high CTC rate comparable to a ZigBee radio. Since WEBee has a pretty high packet error rate due to the intrinsically distorted emulated signal, TwinBee [14] and LongBee [33] were further designed to improve its reliability and transmission range. PAR [34] establishes a feedback channel to improve the reliability of CTC. NetCTC [35] proposes upper layer design for *physical-layer* CTC to meet the requirements in heterogeneous unicast, multicast and broadcast. CRF [36] leverages *physical-layer* CTC for concurrently conducting routing within the WiFi network and flooding among ZigBee nodes using a single stream of WiFi packets. WIDE [37] utilizes digital emulation to achieve CTC from WiFi to ZigBee, it has no error induced by distorted signals.

The main problem of these mechanisms is that they can not work under the standard ZigBee and 20 *MHz* WiFi channels. As shown in the upper figure of Fig. 28, one standard 20*MHz* WiFi channel overlaps with four ZigBee channels, while three of them overlap with the pilot subcarriers, and the last one overlaps with the null subcarriers, all the situations are not permitted by the *physical-layer* CTC, and CTC can only be achieved when either the WiFi or ZigBee channel is changed to a non-standard one.

Since 802.11n [11] also recommends 40*MHz* WiFi channel at the 2.4 *GHz* band [3], we have further investigated the positions of pilot and null subcarriers for the 40 *MHz* channel. According to the 802.11n standard, each 40 *MHz* channel is composed of two 20 *MHz* channels, and is divided into 128 subcarriers, overlapping with eight ZigBee channels. As shown in the lower figure of Fig. 28, five ZigBee channels overlap with the pilot/null subcarriers, while the other three channels only overlap with data subcarriers. This analysis demonstrates that *physical-layer* CTC has the possibility to be applied to commercial WiFi networks when a 40*MHz* WiFi channel is adopted. However, we should note that using 40 *MHz* channels in the 2.4*GHz* band is very hard because two non-overlapped 20*MHz* channels must be clear in order to transmit. The author in [12] has emphasized this point when declaring why 802.11ac is kept from running in the 2.4*GHz* band. In addition, it is really cost-inefficient to let a 40*MHz* WiFi channel only transmit a 2*MHz* ZigBee signal.

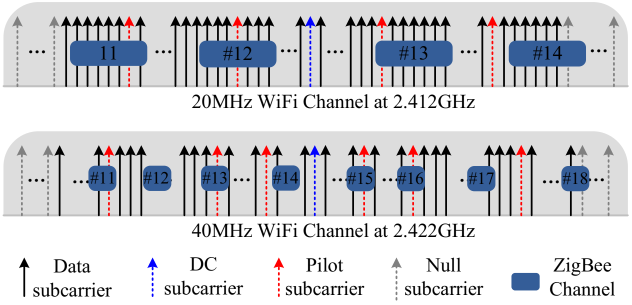


Fig. 28. Illustration of 20 *MHz* and 40 *MHz* WiFi channels overlapping with ZigBee channels.

8.2 Packet-Level Energy Modulation (PLEM)

Esense [6] is a pioneer in this area. It introduces extra signature packets with certain durations to represent an “alphabet set” for transmitting information between WiFi and ZigBee, but the injected extra packet transmissions will lead to a large amount of overhead to the wireless network. HoWiES [5] extends the basic idea of Esense for WiFi energy saving through using a low-power ZigBee radio to wake up the high-power WiFi interface. GSense [10] replaces the preamble of WiFi packets with a sequence of energy pulses, and uses the quiet period between pulses to convey the coordination information to heterogenous devices. FreeBee [2] shifts the beacons and utilizes the interval between beacons to represent the conveyed information, it suffers from low throughput due to the limited number of beacons (the average interval between beacons is approximate to 100*ms*). C-Morse [8] and DCTC [9] propose to exploit a set of WiFi packets with carefully designed transmission duration to convey ZigBee information. EMF [38] enables concurrent CTC transmissions between one WiFi and multiple ZigBee devices through packet reordering and transmission duration adjustment. Considering the variable interference and background noise in the networks, WiZig [7] proposes a rate adaptation algorithm according to the channel conditions to optimize the CTC throughput, through adjusting the number of energy levels and the length of receiving window; StripComm [3] introduces the concept of Manchester Coding to the packet level,

[3] The 80*MHz* channel recommended by 802.11n and 802.11ac works at 5 *GHz* band.

and modulates both presence and absence of a packet in a single CTC information to resist interference. Besides the low CTC data rate, all these mechanisms require fine time slot allocation, they are MAC incompatible with commercial devices as WiFi devices access the channel in a random way, inducing severe interference to current wireless networks.

It is worthy to note that another work OfdmFi [39] also adopts symbol-level energy modulation to achieve CTC between WiFi and LTE-U/LAA, and the basic idea at the transmitter side is quite similar with SLEM. The main difference between OfdmFi and SLEM on designing the transmitting bits is that, OfdmFi finds the solution based on heuristic observation on Viterbi decoding at the receiver side, but SLEM analyzes it in another way through formulating convolutional encoding as the matrix multiplexing, which provides a theoretical basis for the results.

9 Conclusion and Discussion

In this paper, we present the design and implementation of SLEM, a novel CTC method which delivers both the WiFi and CTC data bits concurrently through one standard WiFi packet. Beyond SLEM, two aspects of CTC are worthy of further study.

At first, current CTC mechanisms including SLEM mainly focus on the physical layer design, and simplify the integration of CTC at the upper layer. Since CTC varies the communication methods among wireless devices, we believe this will lead to significant changes in the upper layer design, which need further investigation.

In addition, the distinct physical layer technologies adopted by heterogeneous devices result in quite specific design of CTC. For example, the CTC mechanisms from ZigBee to WiFi, WiFi to ZigBee and Bluetooth to ZigBee may be totally different. In real networks, it is worthy to study which physical layer technology should be used by a device under certain situations.

Acknowledgments

This research was supported in part by the China NSFC Grant (62072317, 61672320, 61802263, U2001207, 61872248), Guangdong NSF 2017A030312008, Shenzhen Science and Technology Foundation (ZDSYS20190902092853047), Faculty Research Fund of Shenzhen University (2019052, 860/000002110322), the Project of DEGP (No.2019KCXTD005), the Guangdong "Pearl River Talent Recruitment Program" under Grant 2019ZT08X603, Guangdong Science and Technology Foundation (2019B111103001,2019B020209001). Kaishun Wu is the corresponding author.

References

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

[2] S. M. Kim and T. He, "FreeBee: cross-technology communication via free side-channel," in *Proc. of the ACM MobiCom*, 2015.

[3] X. Zheng, Y. He, and X. Guo, "StripComm: interference-resilient cross-technology communication in coexisting environments," in *Proc. of the IEEE INFOCOM*, 2018.

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

[5] Y. Zhang and Q. Li, "HoWiES: a holistic approach to ZigBee assisted WiFi energy savings in mobile devices," in *Proc. of the IEEE INFOCOM*, 2013.

[6] K. Chebrolu and A. Dhekne, "Esense: communication through energy sensing," in *Proc. of the ACM MobiCom*, 2009.

[7] X. Guo, X. Zheng, and Y. He, "WiZig: cross-technology energy communication over a noisy channel," in *Proc. of the IEEE INFOCOM*, 2017.

[8] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-Morse: cross-technology communication with transparent Morse coding," in *Proc. of the IEEE INFOCOM*, 2017.

[9] W. Jiang, Z. Yin, S. M. Kim, and T. He, "Transparent cross-technology communication over data traffic," in *Proc. of the IEEE INFOCOM*, 2017.

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

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

[12] M. S. Gast, "802.11ac: A survival guide," *O'Reilly Media*, 2013.

[13] IEEE Computer Society. 802.15.4, "IEEE Standard for Low-Rate Wireless Networks," 2015.

[14] Y. Chen, Z. Li, and T. He, "TwinBee: reliable physical-layer cross-technology communication with symbol-level coding," in *Proc. of the IEEE INFOCOM*, 2018.

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

[16] X. Guo, Y. He, X. Zheng, L. Yu, and O. Gnawali, "ZigFi: harnessing channel state information for cross-technology communication," in *Proc. of the IEEE INFOCOM*, 2018.

[17] X. Guo, Y. He, X. Zheng, Z. Yu, and Y. Liu, "LEGO-Fi: transmitter-transparent CTC with cross-demapping." in *Proc. of the IEEE INFOCOM*, 2019.

[18] 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.

[19] D. Xia, X. Zheng, L. Liu, C. Wang, and H. Ma, "c-Chirp: towards symmetric cross-technology communication over asymmetric channels," in *Proc. of the IEEE SECON*, 2020.

[20] W. Jeong, J. Jung, Y. Wang, S. Wang, S. Yang, Q. Yan, Y. Yi, and S. M. Kim, "SDR receiver using commodity wifi via physical-layer signal reconstruction," in *Proc. of the ACM MobiCom*, 2020.

[21] Z. Chi, Y. Li, H. Sun, Y. Yao, Z. Lu, and T. Zhu, "B^2W^2: N-way concurrent communication for IoT devices." in *Proc. of the ACM Sensys*, 2016.

[22] W. Wang, S. He, L. Sun, T. Jiang, and Q. Zhang, "Cross-technology communications for heterogeneous IoT devices through artificial doppler shifts," *IEEE Transactions on Wireless Communications*, vol. 18, no. 2, Feb. 2019.

[23] W. Jiang, Z. Yin, R. Liu, S. M. Kim, Z. Li, and T. He, "BlueBee: a 10,000x faster cross-technology communication via PHY emulation." in *Proc. of the ACM Sensys*, 2017.

[24] W. Jiang, S. M. Kim, Z. Li, and T. He, "Achieving receiver-side cross-technology communication with cross-decoding," in *Proc. of the ACM MobiCom*, 2018.

[25] R. Liu, Z. Yin, W. Jiang, and T. He, "LTE2B: time-domain cross-technology emulation under LTE constraints." in *Proc. of the ACM SenSys*, 2019.

[26] J. Shi, D. Mu, and M. Sha, "LoRaBee: cross-technology communication from LoRa to ZigBee via payload encoding." in *Proc. of the IEEE ICNP*, 2019.

[27] Z. Li and Y. Chen, "BLE2LoRa: cross-technology communication from bluetooth to LoRa via chirp emulation," in *Proc. of the IEEE SECON*, 2020.

[28] Z. An, Q. Lin, and L. Yang, "Cross-frequency communication: near-field identification of UHF RFIDs with WiFi." in *Proc. of the ACM MobiCom*, 2018.

[29] G. Chen and W. Dong, "Reactive jamming and attack mitigation over cross-technology communication links," *ACM Transactions on Sensor Networks*, vol. 17, no. 1, Nov. 2020.

[30] S. Yu, X. Zhang, P. Huang, L. Guo, L. Cheng, and K. Wang, "AuthCTC: defending against waveform emulation attack in heterogeneous IoT environments," in *Proc. of the ASIA CCS*, 2020.

[31] 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.

[32] J. Zhang, X. Guo, H. Jiang, X. Zheng, and Y. He, "Link Quality Estimation of Cross-Technology Communication," in *Proc. of the IEEE INFOCOM*, 2020.

[33] Z. Li and T. He, "LongBee: enabling long-range cross-technology communication," in *Proc. of the IEEE INFOCOM*, 2018.

[34] H. He, J. Su, Y. Chen, Z. Li, and L. Li, "Reliable cross-Technology communication with physical-layer acknowledgement," *IEEE Transactions on Communications*, vol. 68, no. 8, Aug. 2020.

[35] S. Wang, Z. Yin, Z. Li, and T. He, "Networking Support For Physical-Layer Cross-Technology Communication." in *Proc. of the IEEE ICNP*, 2018.

[36] W. Wang, X. Liu, Y. Yao, Y. Pan, Z. Chi, and T. Zhu, "CRF: coexistent routing and flooding using WiFi packets in heterogeneous IoT networks." in *Proc. of the IEEE INFOCOM*, 2019.

[37] X. Guo, Y. He, J. Zhang, and H. Jiang, "WIDE: physical-level CTC via digital emulation." in *Proc. of the ACM/IEEE IPSN*, 2019.

[38] Z. Chi, Z. Huang, Y. Yao, T. Xie, H. Sun, and T. Zhu, "EMF: embedding multiple flows of information in existing traffic for concurrent communication among heterogeneous IoT devices." in *Proc. of the IEEE INFOCOM*, 2017.

[39] 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.

Junmei Yao received the Ph.D. degree in Computer Science from the Hong Kong Polytechnic University in 2016, the M.E. degree in Communication and Information System from Harbin Institute of Technology, China in 2005, and the B.E. degree in Communication Engineering from Harbin Institute of Technology, China in 2003. She is currently an assistant professor in the College of Computer Science and Software Engineering, Shenzhen University, China. Her research interests include wireless networks, wireless communications and mobile computing.

Xiaolong Zheng is currently a research associate professor with the School of Computer Science and Beijing Key Laboratory of Intelligent Telecommunications Software and Multimedia, 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 Internet of Things, wireless networks, and ubiquitous computing.

Ruitao Xie received a PhD degree in Computer Science from City University of Hong Kong in 2014, and BEng degree from Beijing University of Posts and Telecommunications in 2008. She is currently an assistant professor in College of Computer Science and Software Engineering, Shenzhen University. Her research interests include AI networking and mobile computing, distributed systems and cloud computing.

Kaishun Wu received the Ph.D. degree in computer science and engineering from Hong Kong University of Science and Technology, Hong Kong, in 2011. After that, he worked as a Research Assistant Professor with the Hong Kong University of Science and Technology, Hong Kong. In 2013, he joined Shenzhen University, Shenzhen, China, as a Distinguished Professor. He has co-authored 2 books and published 80 refereed papers in international leading journals and primer conferences. He is the inventor of 6 U.S. and 43 Chinese pending patents (13 are issued). He was the recipient of the Best Paper Awards at IEEE Globecom 2012, IEEE ICPADS 2012, and IEEE MASS 2014, and the 2014 IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award and also selected as 1000 Talent Plan for Young Researchers.