Parallel Cross-technology Transmission from IEEE 802.11ax to Heterogeneous IoT Devices

Dan Xia, Xiaolong Zheng*, Liang Liu, Huadong Ma
Beijing Key Laboratory of Intelligent Telecommunication Software and Multimedia
Beijing University of Posts and Telecommunications, Beijing, China
{xiadan, zhengxiaolong, liangliu, mhd}@bupt.edu.cn

Abstract—Cross-Technology Communication (CTC) is an emerging technique that enables direct interconnection among incompatible wireless technologies. However, for the downlink from WiFi to multiple IoT technologies, serially emulating and transmitting the data of each IoT technology has extremely low spectrum efficiency. Recent parallel CTC uses IEEE 802.11g to send emulated ZigBee signal and let the BLE receiver decodes its data from the emulated ZigBee signal with a dedicated codebook. It still has a low spectrum efficiency because IEEE 802.11g exclusively uses the whole channel. Besides, the codebook design hinders the reception on commodity BLE devices. In this paper, we propose *WiCast*, a parallel CTC that uses IEEE 802.11ax to emulate a composite signal that can be received by commodity BLE, ZigBee, and LoRa devices. By taking advantage of OFDMA in 802.11ax, *WiCast* uses a single Resource Unit (RU) for parallel CTC and sets other RUs free for high-rate WiFi users. But such a sophisticated composite signal is very easily distorted by emulation imperfections, dynamic channel noises, cyclic prefix, and center frequency offset. We propose a CTC link model that jointly models the emulation errors and channel distortions. Then we carve the emulated signal with elaborate compensations in both time and frequency domains to solve the above distortion problem. We implement a prototype of *WiCast* on the USRP platform and commodity devices. The extensive experiments demonstrate *WiCast* can achieve an efficient parallel transmission with the aggregated goodput up to $390.24kbps$.

I. INTRODUCTION

To interconnect heterogeneous Internet of Things (IoT) devices, Cross-technology Communication (CTC) is proposed to enable direct transmissions among incompatible wireless technologies by establishing side channels with distinguishable transmission patterns or directly emulating the waveform of target signal. With the ability of direct interconnection, CTC can provide new technical routes for the coexistence of multi-technology transmissions. For example, in smart home scenario, WiFi traffic of various Internet applications coexists with traffic from IoT devices, such as BLE smart wearable, LoRa smart meters, and ZigBee monitors. Instead of using multiple gateways, CTC enables a commodity WiFi AP to act as the multi-technology gateway for the transmissions of multiple wireless technologies. In addition to the benefit of hardware cost reduction, CTC based multi-technology gateway can avoid the channel access conflict, and even further improve the channel utilization by encoding and transmitting the data from multiple technologies in parallel.

Though promising, parallel CTC transmission of the multi-technology gateway is not easy to obtain by existing methods. Most of the existing methods focus on bi-technology CTC that enables CTC between two kinds of technologies. A straightforward idea is combining several bi-technology CTC methods to achieve a multi-technology gateway. However, such a gateway still only emulates one technology at a time and has channel access conflict for multi-technology downlink transmissions. Round-robin scheduling the downlink transmissions of multiple technologies will cause a long service interruption of WiFi because of the long IoT frames, e.g., tens to hundreds of milliseconds of a LoRa frame.

Recent researches propose parallel CTC to enable the gateway directly transmits multi-technology data without switching among different CTC methods. A concurrent downlink that emulates ZigBee, BLE, and WiFi (QPSK) with the 64-QAM waveform of IEEE 802.11g is proposed in [1]. However, it requires changes of the commodity BLE receiver by using a dedicated codebook to decode BLE data from the emulated ZigBee phase shifts. What's worse, such a parallel CTC downlink is still of low spectrum efficiency because using IEEE 802.11g to emulate the long IoT frame exclusively occupies the channel and hinders the high-rate WiFi traffic. In a nutshell, an efficient parallel CTC downlink for the multi-technology gateway is still missing.

To improve the spectrum efficiency, we desire that the multi-technology gateway downlink can transmit data from WiFi to BLE, ZigBee, and LoRa in parallel and have minimized influence on the legacy high-rate WiFi traffic. Hence, we turn our attention to IEEE 802.11ax [2] which uses orthogonal frequency domain multiple access (OFDMA) and divides the spectrum into resource units (RUs) of various sizes to enable multi-user usage at the same time. We can allocate a WiFi RU to allow the parallel transmission of BLE, ZigBee, and LoRa data rather than the whole spectrum. Then WiFi users can still use other RUs to keep high-rate WiFi transmissions. By this way, the spectrum efficiency can be retained.

However, it is non-trivial to accomplish the parallel CTC downlink from WiFi to BLE, ZigBee, and LoRa using IEEE 802.11ax. First, simultaneously emulating three kinds of signals is difficult. On the one hand, the emulated waveform desired by different technologies is different. When combining them into one composite waveform, conflicts inevitably

happens, leading to emulation errors. On the other hand, the elaborate composite signal carries too much information and is easily distorted by random channel noise, leading to serious demodulation errors. Second, Cyclic Prefix (CP) of WiFi symbols will cause serious errors for the emulated signal containing three kinds of technologies. A possible solution is shortening the guard interval of WiFi to minimize the CP influence time. But it will cause cross-symbol CP errors because the WiFi symbol length ($13.6\mu s$) is no longer a multiple or submultiple of other three technologies when using the shortest guard interval. Dealing with the CP errors is challenging. Third, parallel multi-technology transmission inherently suffers from phase shifts and phase drifts, which are hard to completely compensate in time domain. Different from previous work [1] that requires different technologies use the same center frequency, we allow heterogeneous commodity devices to operate in their own standard channels. Then the emulated signal suffers from a fixed phase shift caused by center frequency offset (CFO) and a varying phase drifts caused by CFO and CP together. How to alleviate the distortions of phase shift and phase drift is challenging.

By addressing the above challenges, we propose *WiCast*, a novel parallel CTC method that uses a small IEEE 802.11ax RU to emulate a composite signal that can be simultaneously received by commodity BLE, ZigBee, and LoRa devices. To construct a parallel signal that combines three technologies, we first combine the emulated phase sequences of BLE and ZigBee that have similar chip/symbol rates and then add the low rate LoRa into the composite signal. To cope with channel distortions on the sophisticated parallel signal, we propose a CTC link model to jointly quantify the emulation errors and channel distortions. Based on the link model, we propose an optimization function to calculate a composite phase sequence that dynamically compensates the phase distortions and correctly indicates the symbols of BLE and ZigBee. Then we deal with the CP errors by elaborately combining the mode flipping and phase correction. We further alleviate the distortions of phase drift through the CFO compensation in both the time and frequency domain, based on our theoretical analysis of the influence of CFO on decoding. The main contributions of this work are summarized as follows.

- We propose a novel CTC method that enables parallel transmission CTC from IEEE 802.11ax to BLE, ZigBee, and LoRa, for establishing an efficient downlink of the CTC-based multi-technology gateway.
- We devise *WiCast* that solves several challenges to simultaneously emulate three kinds of signals, including the dynamic channel distortions, cross-symbol CP errors, and phase drifts. With *WiCast*, heterogeneous IoT commodity devices demodulate their own data from a single composite signal emulated by a small IEEE 802.11ax RU.
- We implement a prototype of *WiCast* with USRP N210 platform and commodity devices. Our extensive experiments demonstrate that *WiCast* can achieve an efficient parallel transmission with the aggregated goodput of
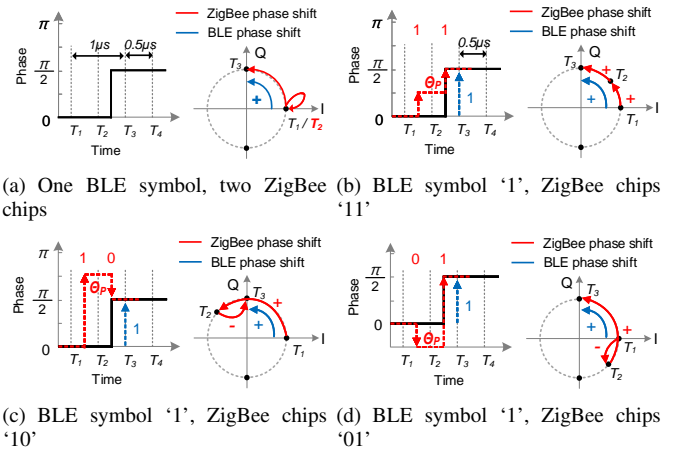

(a) One BLE symbol, two ZigBee chips

(b) BLE symbol '1', ZigBee chips '11'

(c) BLE symbol '1', ZigBee chips '10'

(d) BLE symbol '1', ZigBee chips '01'

Fig. 1. Representing BLE symbol and ZigBee chips with a phase sequence.

$390.24kbps$.

The rest of this paper is organized as follows. We discuss the related work in Section II. We theoretically analyze the feasibility of emulating three kinds of signal by a single composite phase sequence in Section III, and introduce the design details in Section IV. We present the evaluation of *WiCast* in Section V and conclude our work in Section VI.

II. RELATED WORK

CTC is a promising technique that enables direct transmissions among incompatible wireless technologies. Packet-level CTC manipulates packet transmissions such as transmission timing [3], [4], signal strength [5]–[10] and channel state variations [11]–[13] to establish side channels. Physical-level CTC [14]–[24] emulates the target signals directly to generate a heterogeneous receiver compliant packet. WEBee [14] is the first work that emulates the ZigBee signal by specific WiFi payloads. FLEW [25] uses a single FSK chip to fully emulate both transmission and reception of WiFi signals. Then researchers further propose emulation based CTC for other technologies, such as Bluebee from BLE to ZigBee [16] and BLE2LoRa [26] from BLE to LoRa.

To solve the serious spectrum inefficiency problem, parallel communication attracts increasing research interests [27]–[37]. PMC [27] enables the parallel communication from one WiFi device to another WiFi and multiple ZigBee receivers. Chiron [28] achieves the concurrent communication for a gateway to (or from) commodity WiFi and ZigBee devices. PIC [29] proposes a parallel communication scheme that enables concurrent transmission of both WiFi and BLE data at the same time. However, these works only achieve the parallel transmission between WiFi and one IoT technology.

Interconnecting WiFi and multiple IoT technologies also attract researchers' interests. XFi [38] enables the uplink reception of LoRa and ZigBee data using the WiFi radio. In [1], a parallel CTC downlink is proposed to enable the parallel transmission from one 64-QAM WiFi to QPSK-WiFi, ZigBee, and BLE. Though effective, it stops the downlink traffic of all the other WiFi users and therefore suffers serious
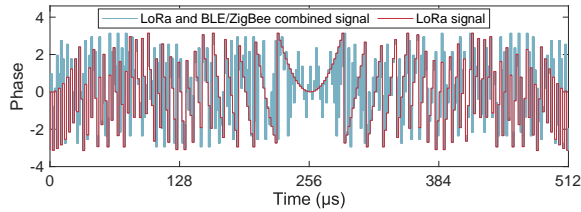
Fig. 2. Combined BLE, ZigBee and LoRa signals.

Fig. 3. The framework of *WiCast*.

spectrum inefficiency problem. Besides, it requires the changes of commodity BLE receiver to decode the BLE data from the parallel signal. Different from [1], our work allocates a single IEEE 802.11ax RU to establish the parallel downlink from WiFi to ZigBee, BLE, and LoRa rather than the whole spectrum to improve the spectrum efficiency.

III. EMULATING THREE KINDS OF SIGNALS BY A SINGLE COMPOSITE PHASE SEQUENCE

Simultaneous emulation of BLE, ZigBee and LoRa signals is the core of our parallel CTC downlink design. We propose emulating the target signal by phase emulation, which is first proposed in [15]. But how to simultaneously emulate the phases of three technologies that have emulation conflicts is still an open problem. In this section, we introduce our design for obtaining the composite phase sequence that includes three technologies in parallel. We first combine the emulated phase sequences of BLE and ZigBee that have similar chip/symbol rates and then add the low rate LoRa into the composite signal.

Both of BLE's GFSK and ZigBee's OQPSK utilize the sign (+ or −) of phase difference between consecutive samples to indicate symbols (chips for ZigBee). Since BLE's rate of phase shifts is half of ZigBee, a single phase shift in BLE can be interpreted as two phase shifts in ZigBee, as per bandwidth difference.Then we can leverage the rate difference and elaborately insert a phase shift in the middle of two BLE phase samples to construct the desired ZigBee phase sequence.

Take the phase sequence in Fig. 1(a) as an example. The BLE receiver samples at T_1 and T_3 and the ZigBee receiver samples at T_1, T_2 and T_3. To generate a phase sequence that can be decoded to a BLE symbol '1' and ZigBee chips '11', we intentionally introduce an extra phase value θ_P in T_2 and $0 < \theta_P < \pi/2$, as shown in Fig. 1(b). Then the phase shift between T_1 and T_2 will be $\theta_P > 0$ and the phase shift between T_2 and T_3 will be $\pi/2 - \theta_P > 0$, which yields chips '11'. The BLE symbol is still decoded as '1', because the phase shift between T_1 and T_3 is $\pi/2$. To represent a BLE symbol '1' and ZigBee chips '10', we let $\pi/2 < \theta_P < \pi$, then the phase shift between T_2 and T_3 becomes smaller than zero, as shown in Fig. 1(c). The same applies to a BLE symbol '1' and ZigBee chips '01'. We let $-\pi/2 < \theta_P < 0$ and the phase shift between T_1 and T_2 becomes smaller than zero, as Fig. 1(d) depicts. Different from other three cases, there is no valid phase sequence for BLE symbol '1' and ZigBee chips '00'. Hence, we replace ZigBee chips '00' with '10' or '01', incurring 1 chip error in either cases. While such a chip error
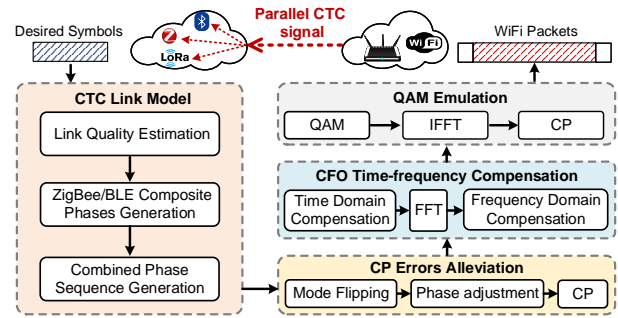
is inevitable, ZigBee maps a 32bit chip sequence to the 4bit symbol, leaving tolerance for chip errors. In this way, we can combine the emulated phase sequences of BLE and ZigBee.

For LoRa phase emulation, we find that LoRa phase changes much slower than the phase of ZigBee and BLE. Then we can add the desired LoRa phases into BLE/ZigBee composite phase sequence, as shown in Fig. 2. Though it is feasible to simultaneously emulate three signals in theory, such a sophisticated composite phase sequence is very easy to be corrupted by channel distortions and emulations errors caused by CPs and CFO. Achieving the parallel CTC transmission from WiFi to BLE, ZigBee and LoRa still needs elaborate designs.

IV. SYSTEM DESIGN

In this section, we first present an overview of *WiCast* and then introduce the key designs that cope with the distortions of the parallel CTC signal.

A. Overview

WiCast uses a IEEE 802.11ax RU to emulate a composite signal that can be recognized and decoded by commodity BLE, ZigBee, and LoRa devices. As shown in Fig. 3, *WiCast* first estimates the channel distortion based on our CTC link model and the feedback from receivers. Then *WiCast* considers both channel distortion and emulation errors to calculate a composite phase sequence that correctly indicates the symbols of BLE and ZigBee. According to the digital emulation principle, *WiCast* leverages WiFi high sampling rate to construct a combined phase sequence that carries both the BLE/ZigBee phases and LoRa phases in parallel. *WiCast* also reduces the CP errors by elaborately combining the CP mode flipping and phase correction. *WiCast* further designs a time-frequency CFO compensation method to alleviate the distortions of phase drift caused by WiFi CP and the CFO of multiple technologies.

B. CTC Link Model

Basic link model that models emulation and channel errors. QAM emulation is the core of signal emulation, where the target signals are fed into the FFT of WiFi to find the QAM constellation points nearest to the desired signals. However, the number of WiFi's predefined QAM points is limited. The
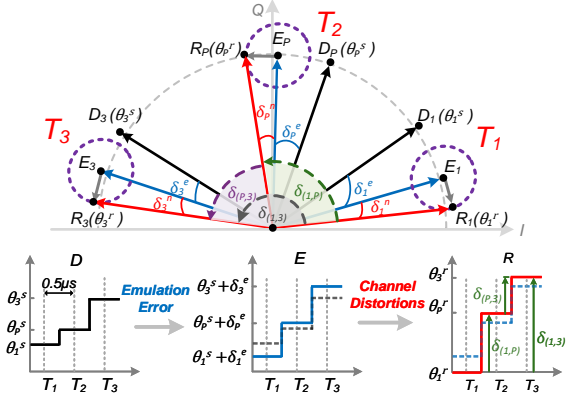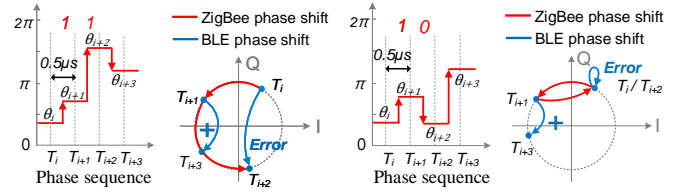
Fig. 4. The CTC link model that jointly quantifies emulation errors and channel distortions.

(a) BLE's phase shift larger than π (b) BLE's phase shift equal to 0

Fig. 5. Impact of clock drift on BLE's phase shift.

Fig. 6. Illustration of combined phase sequence generation.

desired points of target signal cannot perfectly match to the discrete QAM points, leading to inevitable emulation errors. Besides, the wireless channel also introduces random distortion to each sample in the sophisticated emulated signal, which further increases the BLE and ZigBee decoding difficulty. This is because that the intrinsically dynamic wireless channel leads to dynamic distortions, which is hard to compensate.

To find the composite phase sequence correctly indicating the symbols of BLE and ZigBee in a dynamic channel, we first derive a CTC link model that jointly quantifies the channel distortions and emulation errors. As Fig. 4 depicts, the original phases of two consecutive BLE samples at T_1 and T_3 are θ_1^s and θ_3^s, and the corresponding phase drifts caused by emulation error are δ_1^e and δ_3^e. Then the random noise vector $\vec{E_i R_i}$ with a deflection angle varying from 0 to 2π can be regarded on the circle centered on E_i, $i \in \{1, P, 3\}$. Suppose the phase drift caused by random channel distortion as δ_1^n and δ_3^n, the actually received phases at T_1 and T_3 will be $\theta_1^r = \theta_1^s + \delta_1^e + \delta_1^n$ and $\theta_3^r = \theta_3^s + \delta_3^e + \delta_3^n$. Then the phase shift between these two BLE samples is $\delta_{(1,3)} = (\theta_3^s + \delta_3^e + \delta_3^n) - (\theta_1^s + \delta_1^e + \delta_1^n)$. To represent the BLE symbol and the ZigBee chip with a single phase sequence, we intentionally introduce an extra phase value θ_P^s at T_2. Denote the phase drift caused by emulation error and channel distortion as δ_P^e and δ_P^n respectively. Then, the phase shift between T_1 and T_2 is $\delta_{(1,P)} = \theta_P^r - \theta_1^r$, and the phase shift between T_2 and T_3 is $\delta_{(P,3)} = \theta_3^r - \theta_P^r$, where $\theta_P^r = \theta_P^s + \delta_P^e + \delta_P^n$ is the actually received phase at T_2. By this model, we capture the distortions caused by channel noise and emulation errors.

Based on the link model, we design an optimization function to determine the value of θ_P^s that can correctly insert the phase shift desired by ZigBee chips into phase sequence of BLE symbols. Suppose the required SER is P_e, the optimization function is:

$$\theta_P^s = \arg\max(\min(|\delta_{(1,P)}|, |\delta_{(P,3)}|, |\delta_{(1,3)}|))$$
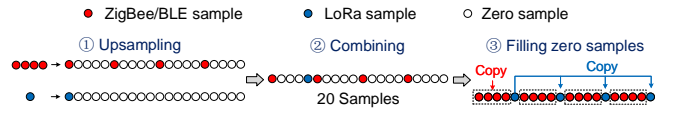
$$\begin{cases} \delta_{(1,P)} = (\theta_P^s + \delta_P^e + \delta_P^n) - (\theta_1^s + \delta_1^e + \delta_1^n) \\ \delta_{(P,3)} = (\theta_3^s + \delta_3^e + \delta_3^n) - (\theta_P^s + \delta_P^e + \delta_P^n) \\ \delta_{(1,3)} = \delta_{(1,P)} + \delta_{(P,3)} \\ -\pi < \delta_{(1,3)} \cdot (-1)^b < 0 \\ -\pi < \delta_{(1,P)} \cdot (-1)^{z_1} < 0 \\ -\pi < \delta_{(P,3)} \cdot (-1)^{z_2} < 0 \end{cases} \quad (1)$$

$$s.t. : -\arcsin \frac{\sin(\frac{\pi(1-P_e)}{2})}{x} \leq \delta_{i=1,P,3}^n \leq \arcsin \frac{\sin(\frac{\pi(1-P_e)}{2})}{x}$$

where b, z_1, and z_2 are the desired BLE chip, the 1st and 2nd ZigBee chip, respectively. In this way, we can calculate the value of θ_P^s as long as the channel parameter x is known. x is the ratio of received signal to noise and can be obtained from the data packets for most of the commodity ZigBee and BLE devices. Hence, the sender just sends the data packets and the channel parameters (e.g., RSSI and SNR) can be piggybacked in the ACK and sent back to the sender.

Enhanced link model that measures the errors caused by asynchronous sampling. So far, we assume the clock of sender and receivers are perfectly synchronized. However, restricted by the hardware, they have clock drift. Fig. 5(a) is an example of the composite sequence $\{\theta_i, \theta_{i+1}, ..., \theta_{i+3}\}$ representing a BLE symbol '1' and two ZigBee chips '11'. When BLE samples at T_{i+1} and T_{i+3}, the phase difference $\theta_{i+3} - \theta_{i+1}$ lies in a range from 0 to π, which yields symbol '1' as expected. However, due to the asynchronous sampling problem, BLE's samples may appear in $[T_i - 0.5, T_{i+1} + 0.5]$ and $[T_{i+2} - 0.5, T_{i+3} + 0.5]$, not necessarily at T_{i+1} and T_{i+3}. When BLE samples at T_i and T_{i+2}, the phase difference $\theta_{i+2} - \theta_i$ will be larger than π, incurring an emulation error. Fig. 5(b) is another example of one BLE symbol '1' and two ZigBee chips '10'. When BLE samples at T_{i+1} and T_{i+3}, the phase difference $\theta_{i+3} - \theta_{i+1}$ lies in a range from 0 to π, which is correct. But when BLE samples at T_i and T_{i+2}, the phase difference $\theta_{i+2} - \theta_i$ will be zero, incurring the emulation error.

From these two examples, we can find that the phase value in a correct composite sequence is related to the phase of its preceding two samples, but not just the phase of its preceding sample. That is, the phase value θ_{i+2} depends on θ_i and θ_{i+1}. However, the previous basic model only
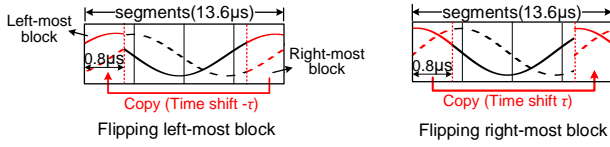
Fig. 7. Illustration of two flipping modes.

Fig. 8. Illustration of CP errors.

Fig. 9. Two flipping modes leads to different errors.
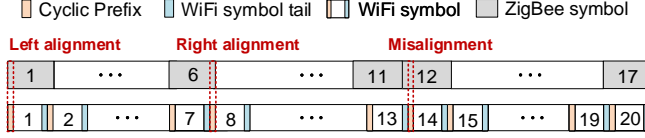

Fig. 10. Left alignment.

considers its preceding one sample, leading to BLE symbol errors. To eliminate these errors, we further add two additional constraints to enhance the basic model and give high priority to the correctness of BLE phase shifts because BLE lacks the error tolerance. Suppose the phase value at T_i and T_{i+1} is θ_i and θ_{i+1}, then the phase value θ_{i+2} at T_{i+2} must satisfy the following two constraints.

$$\begin{cases} |\theta_{i+2} - \theta_i| < \pi & if \ var_{xor} = 0 \\ |\theta_{i+2} - \theta_{i+1}| < |\theta_{i+1} - \theta_i| & if \ var_{xor} = 1 \end{cases} \quad (2)$$

where var_{xor} represents $(\theta_{i+2} - \theta_{i+1}) \oplus (\theta_{i+1} - \theta_i)$. Based on Eq. (1) and Eq. (2), we can obtain a composite phase sequence to represent the BLE's and ZigBee's symbols simultaneously.

Next, we add the emulated LoRa phases into the composite phase sequence. Fig. 6 illustrates the whole process with an example of LoRa phase sequence with a sampling rate of $1MHz$. *WiCast* first upsamples the LoRa and BLE/ZigBee phase sequences from $1MHz$ and $4MHz$ to $20MHz$. Then, *WiCast* combines two phase sequences in dislocation. Since zero-valued samples are vulnerable to noise, we copy the phases of BLE/ZigBee and LoRa to the remaining zero-valued samples based on the ratio of their bandwidths to the WiFi bandwidth. Since LoRa typically has a longer transmitting time than BLE and ZigBee, we combine the phase sequence of one LoRa packet with the phase sequences of multiple BLE and ZigBee packets to generate efficient parallel signals.

After obtaining the composite phase sequence, we can generate the parallel signal containing three technologies. The waveform with the composite phase sequence is fed into FFT (Fast Fourier Transform) module and then we select the nearest QAM constellation points to construct the WiFi payloads. For the RU selection, according to [39], the WiFi RU should satisfy $m \geq \frac{s}{c}$ for successful emulation, where $s = \lceil \frac{BW}{78.125kHz} \rceil$ is the number of subcarriers in a WiFi RU, BW is the spectrum width of combined signal, and c is code rate of WiFi.

C. CP Errors Alleviation

WiFi uses Cyclic Prefixing (CP) to overcome inter-symbol interference. CP has two flipping modes, the left-most and the right-most flipping mode, as shown in Fig. 7. The left-most mode that flips 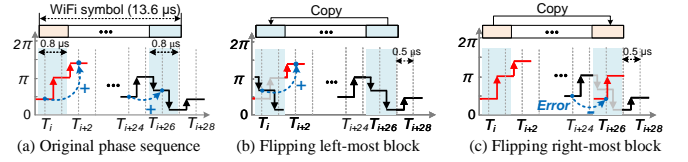the left-most block by overwrit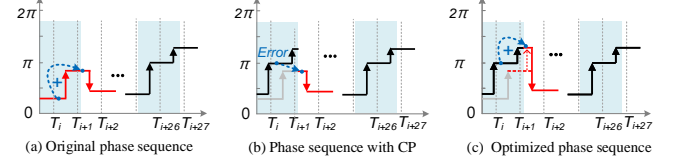ing it with the right-most block. The right-most mode is just in opposite. Then a small portion of the target signal is overwritten and distorted by the CP insertion. With the shortest guard interval ($0.8\mu s$ in IEEE 802.11ax), the length of CP can be reduced from 64 samples to 16 samples, and less impairment is introduced by CP. But the reduction of CP will cause cross-symbol CP errors because the WiFi symbol length ($13.6\mu s$) is no longer a multiple or submultiple of other three technologies when using the shortest guard interval, which makes it hard to correct the phase changes caused by CPs. Fig. 8 shows an example that 20 WiFi symbols emulate 17 ZigBee symbols. WiFi CP and symbol tail can appear in two different ZigBee symbols and WiFi CP occurs with different alignments to ZigBee symbol. As shown in Fig. 8, the beginning time of WiFi CP and ZigBee symbol is same in left alignment, the ending time of WiFi CP and ZigBee symbol is aligned in right alignment, and neither the beginning and ending time is aligned in misalignment case.

To cope with the CP error, we first analyze its influence on the BLE/ZigBee composite phase sequence and find that two flipping modes have different influence on emulation. Fig. 9(a) shows a desired BLE/ZigBee composite phase sequence. The left-most and right-most blocks are marked in red and black lines. If we use left-most flipping mode, the resulted phase sequence is shown in Fig. 9(b). We can find that though CP distorts the ZigBee phase shift in T_i, but the BLE phase shift is still positive, which does not impair the desired phase shift. However, if using the right-most flipping mode, the right-most block will be overwritten by the left-most block, as shown in Fig. 9(c). We can find that it not only leads to the ZigBee's phase shift, but also results in BLE phase errors due to the negative BLE phase shift. Hence, we can alleviate the CP distortion on BLE's phase shifts by elaborately selecting the more harmless flipping mode.

Though selecting the flipping mode can reduce CP errors, it cannot solve all the cases and we need to correct the resulting phase to further alleviate the CP distortions. Note that we cannot modify the CP insertion process but only control the emulated phases outside the CP duration. But fortunately, the non-multiple relationship between CP duration and ZigBee
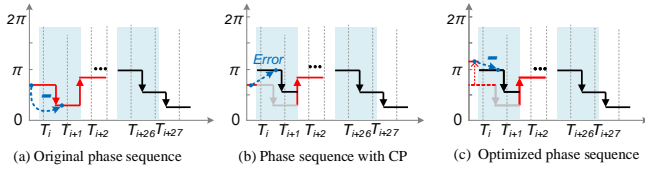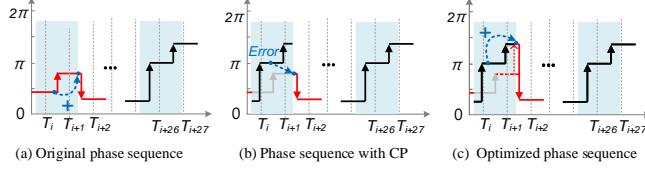
Fig. 11. Right alignment.

Fig. 12. Misalignment.

chip duration leaves us a residual part to correct the phase shift distorted by CP. After deciding the flipping mode, we still face three different alignment cases, as shown in Fig. 8.

Case 1: Left alignment. When we insert CP into the BLE/ZigBee composite phase sequence in Fig. 10(a), we get a new phase sequence in Fig. 10(b). The resulted phase has a negative phase shift from T_i to T_{i+1}, which is desired as a positive shift in original phase sequence. Hence, we will have a ZigBee chip error after adding CP. In this case, we can adjust the phase value of original phase sequence at T_{i+1} to be larger than the phase in T_{i+26}. Then after CP insertion, we can get an optimized phase sequence without ZigBee chip error as shown in Fig. 10(c).

Case 2: Right alignment. Fig. 11(a) shows a desired BLE/ZigBee composite phase sequence, and Fig. 11(b) shows the resulted phase sequence after adding CP. The phase at T_{i+1} is designed to be smaller than T_i in the original phase sequence, but the phase at T_{i+26} is larger than T_i, which causes a ZigBee chip error after adding CP. Hence, we correct the phase value of original phase sequence in T_i to be larger than the phase in T_{i+26}. Then after CP insertion, the remaining part in T_i can be larger than the resulted phase in T_{i+1}, as shown in Fig. 11(c).

Case 3: Misalignment. Suppose we desire a BLE/ZigBee composite phase sequence in Fig. 12(a), the CP insertion distorts it into the phase in Fig. 12(b). The desired positive phase shift from T_i to T_{i+1} is changed to a negative shift because the phase at T_{i+26} is larger than T_{i+1}. To correct this error, we modify the phase in T_{i+1} in original phase sequence to be larger than the phase in T_{i+26}. Then we get a phase sequence without ZigBee chip error after CP insertion, as shown in Fig. 12(c).

In this way, the harmful impact of CPs can be effectively alleviated. Note that the condition of flipping right-most block is similar and we omit the analysis due to the limited space.

Due to the low rate of LoRa chirp, the cross-symbol CP errors have limited impact on the decoding because it occurs only once at the tail of LoRa chirp. For the CP distortions during a LoRa chirp, we can use the mode flipping method proposed in [39] to alleviate the harmful influence. As shown

in Fig. 7, the time offsets in the left-most and right-most flipping mode, $-\tau$ and τ, can be translated into different phase rotations in the frequency domain. Then the peaks in the frequency domain have opposite phase shifts, $e^{-j2\pi f\tau}$ and $e^{j2\pi f\tau}$, leading to opposite destructive effects on FFT peaks during LoRa decoding. Then, we can estimate the CP errors of two flipping modes for each symbol in advance and then select the corresponding mode for the symbols to emulate.

D. Center Frequency Offset Compensation

When different commodity IoT receivers operate in their own standard channel, emulating the parallel signal has to deal with the Center Frequency Offset (CFO) problem. CFO among multiple technologies can lead to the varying phase drift of the emulated signals, leading to the increase of symbol error rate and even packet reception failure. We first analyze the influence of CFO on decoding and then reduce decoding errors by CFO compensation in both time and frequency domain.

We first consider the case where the center frequencies of ZigBee, BLE, and LoRa are perfectly aligned, but there is a CFO between them and WiFi. The k-th sample of the s-th WiFi symbol can be expressed as $x_{s,k} = \Sigma_{i \in D} c_{s,i} \cdot e^{\frac{-j2\pi i(sm+k)}{m}}$, $k \in [0, m]$, where D is the set of data subcarriers in a RU and m is total number of WiFi subcarriers. CP copies the last n_{cp} samples of a WiFi symbol as the first n_{cp} samples to avoid inter-symbol interference. Then, the n-th sample of the s-th WiFi symbol after CP can be calculated as follows.

$$y_{s,n} = \begin{cases} x_{s,m-n_{cp}+n} & 0 \le n \le n_{cp}-1 \\ x_{s,n-n_{cp}} & n_{cp} \le n \le m+n_{cp}-1 \end{cases} \quad (3)$$

Suppose the center frequencies of WiFi and other three technologies are f_w and f_c, then the CFO is $\Delta f = f_w - f_c$. When $n_{cp} \le n \le m + n_{cp} - 1$, the received samples $r_{s,n}(x_{s,k}, \Delta f)$ can be calculated as

$$\begin{aligned} & y_{s,n} \cdot e^{-j2\pi\Delta f \cdot [s \cdot (m+n_{cp})+n] \cdot T_s} \\ = & x_{s,k} \cdot e^{-j2\pi\Delta f \cdot (sm+k) \cdot T_s} \cdot e^{-j2\pi\Delta f \cdot (s+1)n_{cp} \cdot T_s} \\ = & \Sigma_{i \in D} c_{s,(i-\Delta f)} \cdot e^{\frac{-j2\pi ik}{m}} \cdot e^{\frac{-j2\pi\Delta f \cdot (s+1)n_{cp}}{m}} \end{aligned} \quad (4)$$

where $k = n - n_{cp}$, $k \in [0, m]$ and T_s is the inverse of WiFi's sampling rate. As for $0 \le n \le n_{cp} - 1$, the received samples $r_{s,n}(x_{s,k}, \Delta f)$ can be obtained and expressed as follows.

$$\begin{aligned} & y_{s,n} \cdot e^{-j2\pi\Delta f \cdot [s \cdot (m+n_{cp})+n] \cdot T_s} \\ = & x_{s,k} \cdot e^{-j2\pi\Delta f \cdot (sm+k) \cdot T_s} \cdot e^{-j2\pi\Delta f \cdot [(s+1)n_{cp}-m] \cdot T_s} \\ = & \Sigma_{i \in D} c_{s,(i-\Delta f)} \cdot e^{\frac{-j2\pi ik}{m}} \cdot e^{\frac{-j2\pi\Delta f \cdot (s+1)n_{cp}}{m}} \end{aligned} \quad (5)$$

where $k = m - n_{cp} + n$, $k \in [m - n_{cp} + 1, m]$. Compared to the desired signal $x_{s,k}$, the actual received signal suffers from a fixed phase shift caused by CFO, $\frac{-j2\pi ik}{m}$ and a varying phase drifts caused by CFO and CP together, $\frac{-j2\pi\Delta f \cdot (s+1)n_{cp}}{m}$. Due to the phase drift has a extra phase difference $e^{j2\pi\Delta f \cdot m \cdot T_s}$ in WiFi CP, compensating it in time domain is incomplete. Hence, we propose a time-frequency compensation method. We compensate the fixed phase shift in the time domain by the dot-product of each sample with $e^{j2\pi\Delta f \cdot (sm+k) \cdot T_s}$ and the
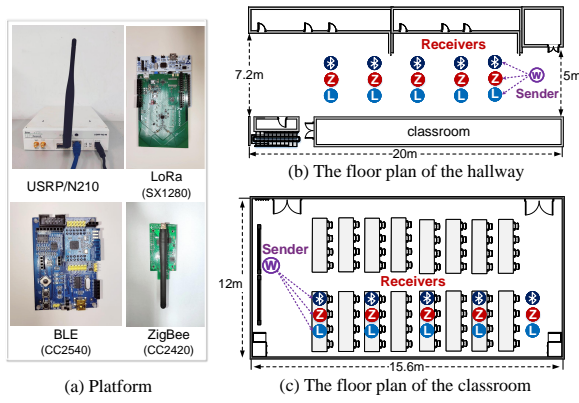
Fig. 13. The hardware platforms and two experiment environments.

Fig. 14. The format of composite packet.

phase drifts in the frequency domain by the dot-product of each corresponding subcarrier $c_{s,(i-\Delta f)}$ with $e^{\frac{j2\pi\Delta f\cdot(s+1)n_{cp}}{m}}$.

Then we consider the center frequencies of ZigBee, BLE and LoRa are also misaligned. Considering a simple case where the center frequencies of ZigBee and BLE are perfectly aligned and different from that of LoRa. Denote the center frequencies of WiFi, LoRa and ZigBee/BLE are f_w, f_l and f_c. The received samples at the LoRa and BLE/ZigBee can be expressed as $r^l_{s,n} = r_{s,n}(x^l_{s,k},(f_w - f_l))$ and $r^c_{s,n} = r_{s,n}(x^c_{s,k},(f_w - f_c))$. Since we use different WiFi samples to carry the BLE/ZigBee phases and LoRa phases in parallel, we can compensate their phases separately. We first shift the BLE/ZigBee signal in the frequency domain by $(f_w - f_c)$, which is done by the dot-product of each BLE/ZigBee sample with $e^{j2\pi(f_w-f_c)\cdot(sm+k)\cdot T_s}$ in the time domain. Then we compensate the phase drifts of BLE/ZigBee signal by the dot-product of each WiFi data subcarrier with a carrier $e^{\frac{j2\pi(f_w-f_c)\cdot(s+1)n_{cp}}{m}}$ in frequency domain.

Due to the compensation for BLE/ZigBee in frequency domain, LoRa samples change to $r^l_{s,n} \cdot e^{j2\pi(f_w-f_c)\cdot(s+1)n_{cp}\cdot T_s}$ when $n_{cp} \le n \le m + n_{cp} - 1$, and change to $r^l_{s,n} \cdot e^{j2\pi(f_w-f_c)\cdot[(s+1)n_{cp}-m]\cdot T_s}$ when $0 \le n \le n_{cp} - 1$. To compensate LoRa's phase shifts and drifts, we multiply a carrier $(e^{j2\pi(f_w-f_l)(sm+k)\cdot T_s} \cdot e^{j2\pi(f_w-f_l)(s+1)n_{cp}\cdot T_s} \cdot e^{-j2\pi(f_w-f_c)(s+1)n_{cp}\cdot T_s})$ for each LoRa's sample in time domain. Note that the emulated phases of BLE/ZigBee and LoRa are staggered in time domain, such a two-step compensation does not influence the above compensated BLE/ZigBee phase. But further compensating in frequency domain will in turn influence the BLE/ZigBee phase that has already been compensated. Hence, we don't further compensate the LoRa phase in frequency domain but rely on the CP flipping mode selection to correct the CP errors for LoRa.

We further consider a more complicated case where center frequencies of all technologies are misaligned. We first discuss the CFO compensation for BLE/ZigBee composite signal. Suppose the center frequencies of WiFi, ZigBee and BLE are f_w, f_z and f_b, then the received samples at the BLE and ZigBee can be expressed as $r^z_{s,n} = r_{s,n}(x^c_{s,k},(f_w - f_z))$ and $r^b_{s,n} = r_{s,n}(x^c_{s,k},(f_w - f_b))$ because WiCast uses a single
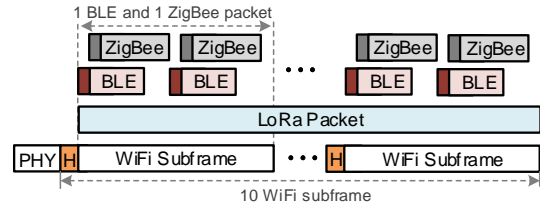
phase sequence to represent the phase shifts of BLE and ZigBee. We multiply a carrier $e^{j2\pi(f_w-f_z)\cdot(sm+k)\cdot T_s}$ for each sample of the BLE/ZigBee composite signal in time domain, and do dot-product of each data subcarrier with a carrier $e^{\frac{j2\pi(f_w-f_z)\cdot(s+1)n_{cp}}{m}}$ in frequency domain. Compared to the desired BLE phase shifts, the phase difference of two consecutive BLE samples within the same WiFi symbol increases by $2\pi(f_b - f_z)T_b$, where T_b is the inverse of BLE's sampling rate. The phase difference between two consecutive samples in two WiFi symbols increases by $2\pi(f_b - f_z)T_b \cdot (1 + n_{cp} - m)$. While such a phase drift is inevitable due to a center frequency offset $f_b - f_z$ ($\pm 1MHz$) between ZigBee and BLE, the phase drift Interestingly happened to be a constant $\pm\pi$ ($2\pi \cdot (\pm 1) \cdot \frac{1}{2} \cdot (1 + 16 - 256)$). That is, by representing BLE symbol '1' to negative phase shift and BLE symbol '0' to positive phase shift, we are able to decode the received phase sequence to correct BLE symbols.

V. EVALUATION

We implement a prototype of *WiCast* using USRP N210 software radios and commodity devices, as shown in Fig. 13. We implement the *WiCast* ZigBee receiver on both USRP N210 with ZigBee PHY and TelosB [40], the *WiCast* BLE receiver using USRP N210 with BLE PHY and commodity BLE CC2540 [41], *WiCast* LoRa receiver on both USRP N210 with LoRa PHY and the commercial LoRa platform equipped with Semtech SX1280 chip [42]. The *WiCast* WiFi sender is implemented on USRP N210 with IEEE 802.11ax OFDMA. *WiCast* WiFi sender can be implemented on commodity WiFi devices because we do not change any hardware or firmware.

We use a WiFi aggregated frame including 10 subframes to emulate a composite packet containing LoRa, ZigBee and BLE data, as shown in Fig. 14. Each emulated ZigBee packet consists of four bytes of preamble(0x00000000), a byte of start of frame delimiter (SFD) (0xA7), two bytes of packet length, and 8 bytes payload. Each emulated BLE packet consists of one byte of preamble (0x55), four bytes of Access Address (0x8E89BED6), and 31 bytes payload. We stagger the preamble and header of BLE and ZigBee in the composite parallel signal. Each emulated LoRa packet consists of 8 up-chirps as preamble, 2-symbol sync word (0x18), 2.25 down-chirps as SFD, and a variable number of payloads. The spreading factor, coding rate, and bandwidth of LoRa are generally set to 7, 4/5 and $400kHz$. By default, we set the WiFi channel to 13 and set the central frequency of the LoRa, ZigBee and BLE devices to $2480MHz$. To ensure statistical validity, we
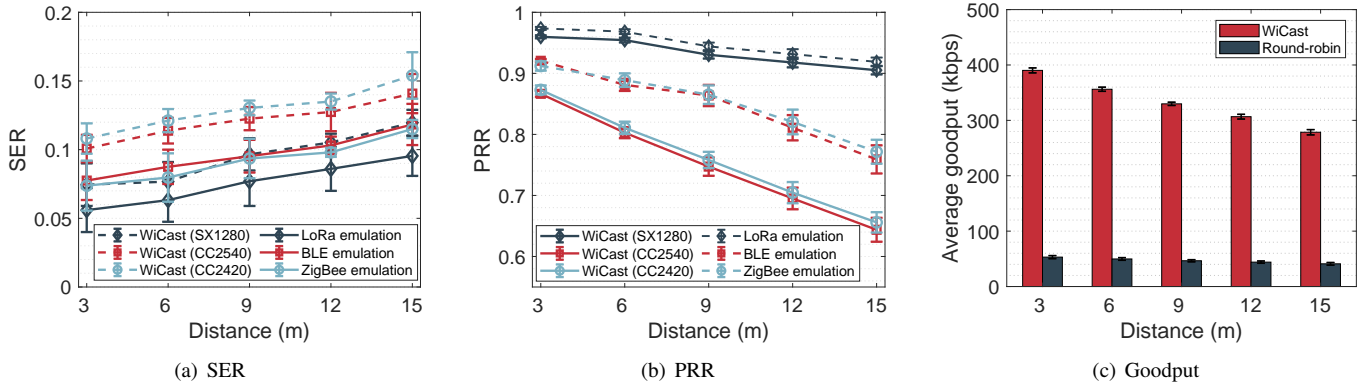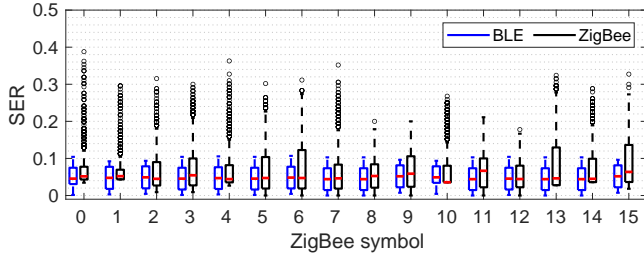
Fig. 15. Overall performance comparison.

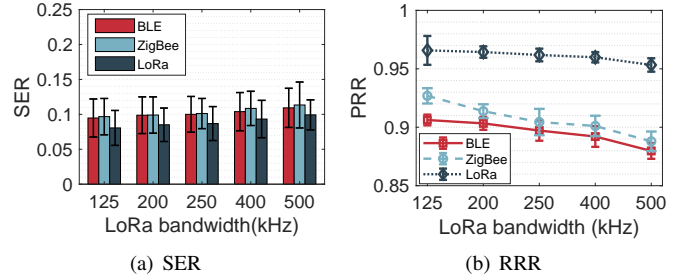Fig. 16. SER of ZigBee/BLE Composite Phases.

Fig. 17. Impact of LoRa bandwidth.

obtain the average result of 10 experiments, each of which sends 1000 composite packets.

A. Overall Performance Comparison

We conduct experiments in the classroom environment (Fig. 13(c)) to evaluate the overall performance of *WiCast*. For comparison, we also implement bi-technology CTC from WiFi to ZigBee, WiFi to BLE, and WiFi to LoRa based on the design principle of state-of-the-art method using phase shift emulation [15] and phase emulation [39]. We measure the SER and goodput of *WiCast* when varying the distance between *WiCast* sender and receiver from $3m$ to $15m$. The transmission power of *WiCast* sender is $20dBm$.

The results are shown Fig. 15. When increasing the distance from $3m$ to $15m$, the SER of *WiCast* for BLE, ZigBee, and LoRa receiver increases from 0.101 to 0.141, 0.108 to 0.154 and 0.075 to 0.119, which is 26.28%, 41.9% and 25.51% higher than the bi-technology emulation of BLE, ZigBee and LoRa, respectively. This is because *WiCast* introduces more emulation errors for emulating a composite phase sequence that carries three heterogeneous signal simultaneously. Similar results are observed in PRR. When increasing the distance from $3m$ to $15m$, the PRR of *WiCast* BLE, ZigBee, and LoRa receivers decreases from 0.862 to 0.624, 0.866 to 0.64 and 0.96 to 0.905, which is 11.55%, 10.91% and 1.45% lower than corresponding bi-technology emulation methods.

Though having a higher SER, *WiCast* achieves the parallel transmission of three technologies, which can significantly improve the aggregated goodput and the channel utilization. When the distance is $3m$, the goodput rate of *WiCast* is $390.24kbps$, while the aggregated goodput rate when serially transmitting the emulated data of three technologies in a round-robin manner is only $53.11kbps$. When the distance increase to $15m$, the goodput rate of *WiCast* slowly drops to $278.57kbps$, which is still $5.05\times$ higher than the round-robin transmission. Due to the parallel CTC transmission of *WiCast*, the channel utilization can be significantly improved.

B. Effectiveness of the Composite Phase Sequence

WiCast improves the channel utilization by achieving parallel CTC transmission with a single composite phase sequence. The effectiveness of the composite phase sequence is vital. To verify the effectiveness of using a single phase sequence to simultaneously represent the phase shifts of ZigBee and BLE, We study the SER for different ZigBee/BLE composite phase sequences. For each of the 16 ZigBee symbols, we combine it with 30 different BLE phase shift sequences, including one best embedding sequence, one worst embedding sequence and 28 random sequences. The experiment is conducted in a classroom and the distance between sender and receiver is $5m$. We repeat 1000 times for each composite phases. The results are shown in Fig. 16. The average SER of BLE in different ZigBee/BLE composite phase sequences varies from 0.044 to 0.052. There is no specific phase sequence having obviously higher SER. We can find the SER of ZigBee occurs a few outliers due to *WiCast* gives priority to the correctness of BLE phase shifts, but ZigBee still has a satisfying performance and the average SER of ZigBee is smaller than 0.067.

For LoRa, we directly add its desired phase into the composite phase sequence of BLE and ZigBee because LoRa has a

(a) SER of ZigBee receiver

(b) SER of LoRa receiver
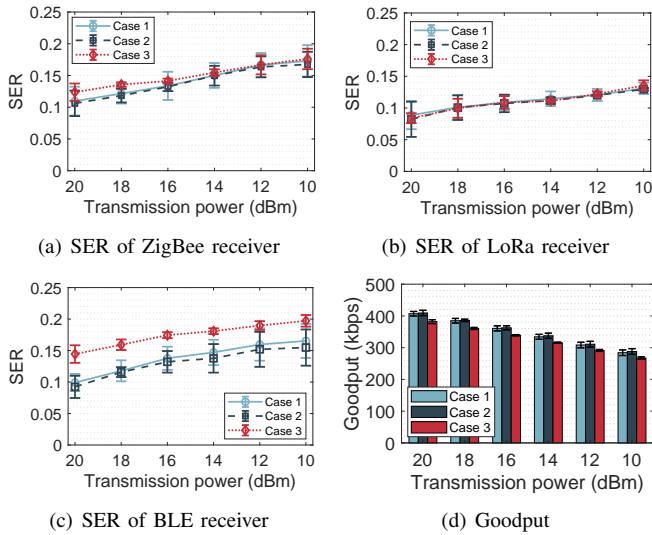
(c) SER of BLE receiver

(d) Goodput

Fig. 18. Impact of the center frequency offset.

significantly lower symbol rate. However, when the bandwidth of LoRa increases, the symbol rate also increases and LoRa may influence the composite phase sequence. Hence, we evaluate the performance of *WiCast* when LoRa has different bandwidths. We conduct the experiments in the classroom where the *WiCast* sender and receiver are $5m$ apart. The experimental results are shown in Fig. 17. For the typical LoRa bandwidths from $125kHz$ to $500kHz$, *WiCast* has similar SER and PRR. When integrating LoRa signal with different bandwidths, the SER of *WiCast* BLE, ZigBee, and LoRa receivers is lower than 0.109, 0.113 and 0.093, and the corresponding PRR is larger than 0.880, 0.888 and 0.953, respectively. The results demonstrate the effectiveness of *WiCast* to achieve parallel CTC transmission of three different technologies by a single composite phase sequence.

C. Performance under Different Settings

1) Impact of Center Frequency Offset: In this subsection, we evaluate the performance of our CFO compensation method. We set the WiFi channel to 13 ($2472MHz$) and consider three different cases, including i) Case 1: the center frequencies of ZigBee, BLE and LoRa are same at $2480MHz$, ii) Case 2: the center frequencies of ZigBee and BLE are same at $2480MHz$ but different from the center frequencies of LoRa at $2479MHz$, iii) Case 3: the center frequencies of ZigBee, BLE and LoRa are all different, which are $2480MHz$, $2481MHz$, and $2479MHz$ respectively.

The experimental results are shown in Fig. 18. We find the *WiCast* LoRa and ZigBee receivers achieve similar SER in three different cases. However, the performance of *WiCast* BLE receivers in case 3 is obviously worse than that in the other two cases. This is because the clock offset between *WiCast* WiFi sender and BLE receiver causes the phase drift is no longer equal to $\pm\pi$, which is hard to compensate and results in more decoding errors. The CFO among ZigBee and BLE indeed has an influence on the performance but within an acceptable range. Even with a relatively low transmission
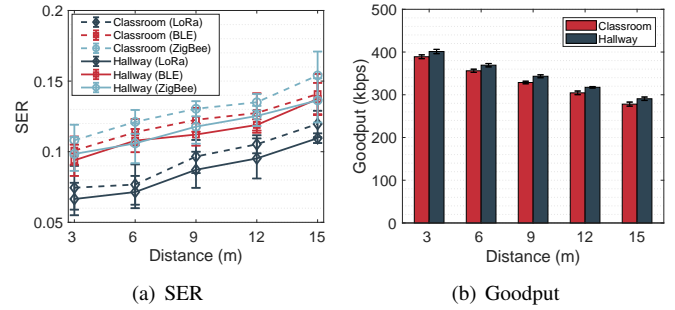

(a) SER

(b) Goodput

Fig. 19. Performance of *WiCast* in different environments.

power $10dBm$, *WiCast* can still achieve a goodput higher than $267.50kbps$ for all the cases, and SER of *WiCast* BLE, ZigBee, and LoRa receiver is lower than 0.197, 0.176, and 0.136.

2) Impact of environment: We also evaluate *WiCast* in different environments, including the classroom and the hallway in Fig. 13. Fig. 19 presents the experimental results. We can find that *WiCast* has slightly better performance in the hallway. This is because the environment of classroom is more complicated, which leads to more significant influences on the received signals and results in more decoding errors. But all the *WiCast* receivers in two environments have achieved a SER lower than 0.2 and a goodput larger than $275kbps$. When the distance is $15m$, the SER of *WiCast* BLE, ZigBee, and LoRa receiver in the classroom is 0.141, 0.154, and 0.120, which is 6.36%, 11.13%, and 9.99% higher than that in the hallway. The corresponding goodput of *WiCast* in the classroom and hallway is $278.09kbps$ and $290.65kbps$.

VI. Conclusion

In this paper, we propose *WiCast*, a novel CTC method that enables parallel CTC transmission from IEEE 802.11ax to the commodity BLE, ZigBee and LoRa devices. We propose a CTC link model to jointly quantify the emulation errors and channel distortions. Based on the link model, we compensate the dynamic phase distortions and construct a single composite phase sequence containing three kinds of signals. We deal with CP errors by elaborately combining the mode flipping and phase correction. We also propose a time-frequency CFO compensation method to alleviate the emulation distortions. We implement a prototype of *WiCast* on USRP N210 and commodity devices. The extensive experimental results have shown that the commodity receivers of all the three technologies can efficiently decode their own data from the parallel CTC signal. The aggregated goodput of *WiCast* can be up to $390.24kbps$, which significantly improves the channel utilization.

REFERENCES

[1] W. Chen, Z. Yin, and T. He, "Global cooperation for heterogeneous networks," in *Proceedings of IEEE INFOCOM*, 2020.

[2] W. L. W. Group, "Ieee draft standard for information technology – telecommunications and information exchange between systems local and metropolitan area networks – specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment enhancements for high efficiency wlan," *IEEE P802.11ax/D6.0*, 2019.

[3] S. M. Kim and T. He, "Freebee: Cross-technology communication via free side-channel," in *Proceedings of ACM MobiCom*, 2015.

[4] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-morse: Cross-technology communication with transparent morse coding," in *Proceedings of IEEE INFOCOM*, 2017.

[5] X. Guo, Y. He, and X. Zheng, "Wizig: Cross-technology energy communication over a noisy channel," *IEEE/ACM Transactions on Networking*, vol. 28, no. 6, pp. 2449–2460, 2020.

[6] X. Zheng, Y. He, and X. Guo, "Stripcomm: Interference-resilient cross-technology communication in coexisting environments," in *Proceedings of IEEE INFOCOM*, 2018.

[7] J. Shi, D. Mu, and M. Sha, "Lorabee: Cross-technology communication from lora to zigbee via payload encoding," in *Proceedings of IEEE ICNP*, 2019.

[8] K. Chebrolu and A. Dhekne, "Esense: Communication through energy sensing," in *Proceedings of ACM MobiCom*, 2009.

[9] S. Tong, Y. He, Y. Liu, and J. Wang, "De-spreading over the air: long-range ctc for diverse receivers with lora," in *Proceedings of ACM MobiCom*, 2022.

[10] 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*, vol. 21, no. 11, pp. 3926–3940, 2021.

[11] X. Guo, Y. He, X. Zheng, L. Yu, and O. Gnawali, "Zigfi: Harnessing channel state information for cross-technology communication," *IEEE/ACM Transactions on Networking*, vol. 28, no. 1, pp. 301–311, 2020.

[12] W. Wang, X. Zheng, Y. He, and X. Guo, "Adacomm: Tracing channel dynamics for reliable cross-technology communication," in *Proceedings of IEEE SECON*, 2019.

[13] D. Xia, X. Zheng, L. Liu, C. Wang, and H. Ma, "c-chirp: Towards symmetric cross-technology communication over asymmetric channels," *IEEE/ACM Transactions on Networking*, vol. 29, no. 3, pp. 1169–1182, 2021.

[14] Z. Li and T. He, "Webee: Physical-layer cross-technology communication via emulation," in *Proceedings of ACM MobiCom*, 2017.

[15] X. Guo, Y. He, J. Zhang, and H. Jiang, "Wide: physical-level ctc via digital emulation," in *Proceedings of ACM/IEEE IPSN*, 2019.

[16] W. Jiang, Z. Yin, R. Liu, Z. Li, S. M. Kim, and T. He, "Bluebee: a 10,000 x faster cross-technology communication via phy emulation," in *Proceedings of ACM SenSys*, 2017.

[17] H.-W. Cho and K. G. Shin, "Bluefi: bluetooth over wifi," in *Proceedings of ACM SIGCOMM*, 2021.

[18] 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 *Proceedings of ACM MobiCom*, 2020.

[19] Z. Li and Y. Chen, "Bluefi: Physical-layer cross-technology communication from bluetooth to wifi," in *Proceedings of IEEE ICDCS*, 2020.

[20] Z. Li and T. He, "Longbee: Enabling long-range cross-technology communication," in *Proceedings of IEEE INFOCOM*, 2018.

[21] Y. Chen, Z. Li, and T. He, "Twinbee: Reliable physical-layer cross-technology communication with symbol-level coding," in *Proceedings of IEEE INFOCOM*, 2018.

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

[23] J. Zhang, X. Guo, H. Jiang, X. Zheng, and Y. He, "Link quality estimation of cross-technology communication," in *Proceedings of IEEE INFOCOM*, 2020.

[24] Y. He, X. Guo, X. Zheng, Z. Yu, J. Zhang, H. Jiang, X. Na, and J. Zhang, "Cross-technology communication for the internet of things: A survey," *ACM Computing Surveys*, vol. 55, no. 5, pp. 1–29, 2022.

[25] H.-W. Cho and K. G. Shin, "Flew: fully emulated wifi," in *Proceedings of ACM MobiCom*, 2022.

[26] Z. Li and Y. Chen, "Ble2lora: cross-technology communication from bluetooth to lora via chirp emulation," in *Proceedings of IEEE SECON*, 2020.

[27] Z. Chi, Y. Li, Y. Yao, and T. Zhu, "Pmc: Parallel multi-protocol communication to heterogeneous iot radios within a single wifi channel," in *Proceedings of IEEE ICNP*, 2017.

[28] Y. Li, Z. Chi, X. Liu, and T. Zhu, "Chiron: Concurrent high throughput communication for iot devices," in *Proceedings of ACM MobiSys*, 2018.

[29] Z. Chi, Y. Li, X. Liu, Y. Yao, Y. Zhang, and T. Zhu, "Parallel inclusive communication for connecting heterogeneous iot devices at the edge," in *Proceedings of ACM Sensys*, 2019.

[30] Z. Li and Y. Chen, "Achieving universal low-power wide-area networks on existing wireless devices," in *Proceedings of IEEE ICNP*, 2019.

[31] W. Wang, T. Xie, X. Liu, and T. Zhu, "Ect: Exploiting cross-technology concurrent transmission for reducing packet delivery delay in iot networks," in *Proceedings of IEEE INFOCOM*, 2018.

[32] Z. Chi, Y. Li, H. Sun, Y. Yao, Z. Lu, and T. Zhu, "B2w2: N-way concurrent communication for iot devices," in *Proceedings of ACM SenSys*, 2016.

[33] Z. Chi, Y. Li, H. Sun, Z. Huang, and T. Zhu, "Simultaneous bidirectional communications and data forwarding using a single zigbee data stream," *IEEE/ACM Transactions on Networking*, vol. 29, no. 2, pp. 821–833, 2021.

[34] M. Jin, Y. He, C. Jiang, and Y. Liu, "Parallel backscatter: Channel estimation and beyond," *IEEE/ACM Transactions on Networking*, vol. 29, no. 3, pp. 1128–1140, 2021.

[35] M. Jin, Y. He, X. Meng, D. Fang, and X. Chen, "Parallel backscatter in the wild: When burstiness and randomness play with you," in *Proceedings of ACM MobiCom*, 2018, pp. 471–485.

[36] M. Jin, Y. He, X. Zheng, D. Fang, D. Xu, T. Xing, and X. Chen, "Smoggy-link: Fingerprinting interference for predictable wireless concurrency," in *Proceedings of IEEE ICNP*, 2016.

[37] X. Xia, N. Hou, Y. Zheng, and T. Gu, "Pcube: scaling lora concurrent transmissions with reception diversities," in *Proceedings of ACM MobiCom*, 2021.

[38] R. Liu, Z. Yin, W. Jiang, and T. He, "Xfi: Cross-technology iot data collection via commodity wifi," in *Proceedings of IEEE ICNP*, 2020.

[39] D. Xia, X. Zheng, F. Yu, L. Liu, and H. Ma, "Wira: Enabling cross-technology communication from wifi to lora with ieee 802.11 ax," in *Proceedings of IEEE INFOCOM*, 2022.

[40] "Cc2420: 2.4 ghz ieee 802.15.4 / zigbee-ready rf transceiver," Available: https://www.ti.com/lit/ds/symlink/cc2420.pdf.

[41] "Cc2540 simplelink bluetooth low energy wireless mcu," Available: https://www.ti.com/lit/ds/symlink/cc2540.pdf.

[42] Smetech, "Semtech sx1280: Long range, low power 2.4 ghz wireless rf transceiver with ranging capability," Available: https://www.semtech.com/products/wireless-rf/lora-24ghz/sx1280.