



# Parallel Inclusive Communication for Connecting Heterogeneous IoT Devices at the Edge

Zicheng Chi\*

zicheng1@umbc.edu

Computer Science and Electrical Engineering, University of Maryland, Baltimore County

Yan Li\*

liy1@umbc.edu

Computer Science and Electrical Engineering, University of Maryland, Baltimore County

Xin Liu

xinliu1@umbc.edu

Computer Science and Electrical Engineering, University of Maryland, Baltimore County

Yao Yao

of90379@umbc.edu

Computer Science and Electrical Engineering, University of Maryland, Baltimore County

Yanchao Zhang

yczhang@asu.edu

School of Electrical, Computer and Energy Engineering, Arizona State University

Ting Zhu

zt@umbc.edu

Computer Science and Electrical Engineering, University of Maryland, Baltimore County

## ABSTRACT

WiFi and Bluetooth Low Energy (BLE) are widely used in Internet of Things (IoT) devices. Since WiFi and BLE work within the overlapped ISM 2.4 GHz band, they will interfere with each other. Existing approaches have demonstrated their effectiveness in mitigating the interference. However, further performance improvement has been hampered by the design goal of *exclusive* communication of WiFi or BLE, which only allows one WiFi or BLE device to transmit packets at any specific time slot on the overlapped channel within the communication range. In this paper, we explore a new communication method, called Parallel *Inclusive* Communication (PIC), which leverages the unique modulation schemes of WiFi and BLE for parallel *inclusive* bi-directional transmission of both WiFi and BLE data *at the same time* within the overlapped channel. In this communication system, the PIC gateway is designed upon the IEEE 802.11g and 802.15.1 frameworks while the WiFi and BLE clients are commercial off-the-shelf devices. PIC achieves similar data rates for these parallel WiFi and BLE communications as if WiFi and BLE are communicating separately. PIC's system architecture naturally fits at the edge of the Internet, which is an optimal site for concurrently collecting (or disseminating) data from (or to) an exponentially increasing number of IoT devices that are using WiFi or BLE. We conducted extensive evaluations under four real-world scenarios. Results show that compared with existing approaches, PIC can significantly i) increase the packet reception ratios by 183%; ii) reduce the round-trip delay time by 590 times and energy consumption by 50.5 times; and iii) improve the throughput under WiFi and BLE coexistence scenarios.

\* Authors contributed equally to the paper

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

*SensSys '19, November 10–13, 2019, New York, NY, USA*

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6950-3/19/11...\$15.00

<https://doi.org/10.1145/3356250.3360046>

## CCS CONCEPTS

• **Networks** → **Network protocol design; Home networks.**

## KEYWORDS

IoT, Heterogenous Networks, Parallel Communication

### ACM Reference Format:

Zicheng Chi, Yan Li, Xin Liu, Yao Yao, Yanchao Zhang, and Ting Zhu. 2019. Parallel Inclusive Communication for Connecting Heterogeneous IoT Devices at the Edge. In *The 17th ACM Conference on Embedded Networked Sensor Systems (SenSys '19), November 10–13, 2019, New York, NY, USA*. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/3356250.3360046>

## 1 INTRODUCTION

Bluetooth Low Energy (BLE) and WiFi are widely and increasingly used in Internet of Things (IoT) devices. Report [10] indicates that there will be 3.9 billion machine-to-machine connected devices and 1.1 billion wearable devices growing over two-fold in 2017. These devices mainly use WiFi or BLE for wireless communication. BLE and WiFi have their own unique features: BLE has a very narrow band, medium data-rate, low-power consumption, and frequency-hopping communication features, while WiFi has a much wider band, higher data-rate, and higher-power communication. Due to their unique features, BLE and WiFi have different strengths that are often the weakness of the other. For example, BLE's power consumption is more than 10 times lower than that of WiFi. On the other hand, WiFi's data rate can be more than 1,000 times higher than that of BLE. Due to their complementary strengths, BLE and WiFi radios coexist in IoT networks to support various applications and meet these applications' specific requirements.

Since a large number of WiFi and BLE devices operate on the overlapped 2.4 GHz frequency band, the coexistence of BLE and WiFi introduces severe cross-technology interferences, which cause high packet losses. This is because 39 out of 40 BLE channels are overlapped with WiFi channels (different countries have slightly different channel allocation methods [2]). For example, according to the analysis and experiments conducted by researchers at the National Institute of Standards and Technology [6, 27], the packet loss of WiFi can be 44% (see page 49 of [6]) even when one WiFi device coexists with only one Bluetooth sender-receiver pair. The situation is even worse in BLE and WiFi coexistence scenarios

because BLE's channel bandwidth (i.e., 2 MHz) is twice as wide as the one in classical Bluetooth (i.e., 1 MHz). One may argue that the WiFi transmission power in [6] is only 14 dBm. WiFi can improve its packet reception ratio by increasing its transmission power. However, increasing WiFi's transmission power will also introduce higher interference to neighboring BLE and WiFi devices.

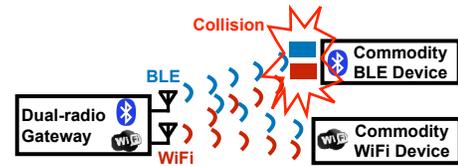
Existing approaches try to mitigate the interference. However, using time-division multiple access (TDMA) or carrier-sense multiple access (CSMA) is not efficient when the number of IoT devices increases. For interference cancellation [20, 26], or network coding [34] techniques, despite their effectiveness, further performance improvement, however, has been hampered by the implicit design goal of *exclusive* communication adopted in previous approaches. Due to the *exclusive* communication, the latency and energy consumption can be significantly increased while the throughput will be dramatically affected in WiFi and BLE coexisting networks as the number of IoT devices exponentially increases.

To address these issues, we introduce a new direction – Parallel *Inclusive* Communication (PIC) that allows parallel *inclusive* bi-directional transmission of both WiFi and BLE data *at the same time* within the overlapped channel. The main novelty of PIC is utilizing the unique features of WiFi and BLE protocols to conduct parallel communications while the performance is not necessarily sacrificed. Specifically, when the PIC device transmits packets to both WiFi and BLE devices in the downlink, PIC embeds four BLE symbols into one WiFi symbol within the overlapped WiFi subcarriers. In this way, the commodity BLE device is able to demodulate the signal at a fine-grained time scale (i.e., BLE symbol duration) while the commodity WiFi can demodulate the signal at a coarse-grained time scale (i.e., WiFi symbol duration). Since WiFi and BLE have different modulation schemes, PIC achieves similar data rates for both WiFi and BLE communications when the interference between WiFi and BLE is minimized.

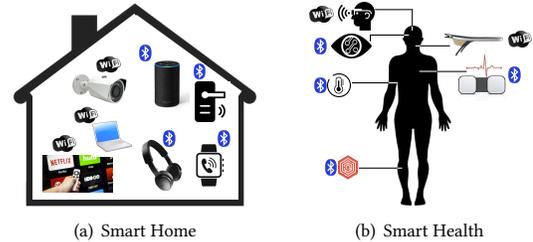
Our approach is theoretically backed up by advances in multiuser detection [3], which can demodulate mutually interfering digital streams of information in wireless communications without violating the Shannon-Hartley theorem on channel capacity. Different from traditional multiuser detection approaches [30] in which the users use the same wireless technology, in our problem the users use heterogeneous technologies (e.g., WiFi and BLE) that have different bandwidth (e.g., 20-40 MHz for WiFi and 2 MHz for BLE). Since BLE's data rate is more than 2 orders of magnitude lower than that of WiFi, concurrently transmitting or receiving WiFi and BLE data will not reach the channel capacity based on multiuser detection theories [57].

Figure 1 compares our downlink design with a commodity dual-radio gateway. The dual-radio gateway needs to avoid collision by allocating BLE and WiFi packets into different time or frequency slots, while our design can enable the concurrent bi-directional communication within the overlapped channel.

Our work is inspired by TIMO [23] which uses a special MIMO technique in the uplink to decode the overlapped signals. While our work enables both i) downlink (transmitting to WiFi and BLE devices) and ii) uplink (receiving from WiFi and BLE devices) concurrent communications with a *single antenna* (instead of MIMO). Our approach is different from [25, 38] that enable concurrent communication for performance improvement within the same technology



**Figure 1:** To avoid the collision of BLE and WiFi packets, the commodity dual-radio gateway needs to transmit BLE and WiFi packets separately over time/frequency in the downlink, while our design can enable the concurrent and parallel bi-directional communication within the overlapped channel.



**Figure 2:** IoT Applications with the support from Parallel Inclusive Communication (PIC)

(i.e., WiFi [25] or ZigBee [38]). Our approach is also different from physical-level [42] or packet-level [14, 37, 63, 64] cross-technology communication (CTC) techniques. The physical-level CTC only allows one technology (e.g., WiFi) to communicate to another technology (e.g., ZigBee) at any specific time slot. Though packet-level CTC provides concurrent transmission, the throughput is very low (e.g., one or a few bits per packet). Thus, the spectrum utilization is very low. The main design goal of CTC is to remove the gateway, while our design goal is to provide a better gateway for better spectrum utilization. On the other hand, our approach's BLE to gateway communication throughput<sup>1</sup> is more than 12,352 times higher than that of existing CTC solutions.

PIC's system architecture naturally fits at the edge of the Internet, which is an optimal site for concurrently collecting (or disseminating) data from (or to) IoT devices that use WiFi or BLE. One may argue that WiFi can move to 5 GHz to avoid its interference with BLE. However, WiFi will not abandon 2.4 GHz any time soon for three reasons. First, billions of WiFi devices with only 2.4 GHz radios have already been shipped. Many of these are mission-critical, such as those in medical and security applications, and may be too expensive to replace. Second, device vendors choose to use 2.4 GHz only radios to reduce the cost. For example, some home automation equipment only uses 2.4 GHz. Last, if all the WiFi devices use 5 GHz, it will make the 5 GHz band extremely crowded within a few years. Therefore, the existing 2.4 GHz WiFi infrastructure will relieve the pressure of the increasingly crowded 5 GHz band. Our PIC design can provide more efficient spectrum utilization and much higher throughput under WiFi and BLE coexistence scenarios.

There are a lot of IoT applications involving the persistent and concurrent WiFi and BLE communication that can benefit from PIC. For example, in a smart home scenario (shown in Figure 2(a)), the traffic of high-quality video streaming, surveillance monitoring, and Internet surfing uses the WiFi network. Nevertheless, there are

<sup>1</sup>For fair comparison, WiFi traffic is not counted

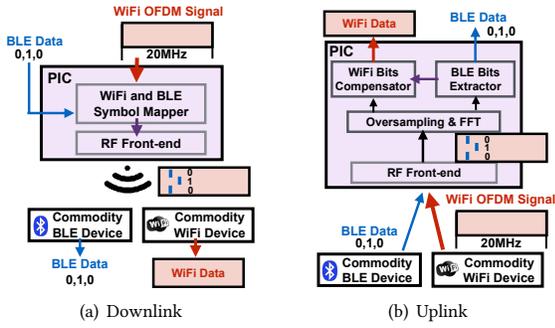


Figure 3: System architecture of PIC

a bunch of IoT devices (such as smart speakers, wireless headsets, smart watches, and smart door locks) using BLE communication. In smart health, Figure 2(b) shows an example of phobia treatment in which multiple wearable devices use WiFi and BLE techniques in parallel: i) 360-degree VR glasses (WiFi) places the patient in different levels of anxiety scenarios; ii) a brain sensing headband (WiFi) reduces symptoms associated with stress and anxiety, and improves focus; iii) the ECG sensor (BLE), smart contact (BLE), implanted thermometer (BLE), and smart tattoo (BLE) measure the heart beats glucose, temperature, and EMG, respectively. In both applications, without PIC, the WiFi and BLE traffics are likely: i) collided when using random access which may reduce the video/audio quality; or ii) delayed when using TDMA which may affect the real-time monitoring and sensing quality. With our PIC technique, the concurrent WiFi and BLE transmission can be smoother.

The major contributions of this paper are as follows:

- We design PIC, the first parallel inclusive communication technique that enables parallel bi-directional transmission of both WiFi and BLE data at the same time within the overlapped channel. The design does not require any modification to the hardware or the firmware of the WiFi and BLE peripheral devices.
- We provide solutions to address several unique challenges, including i) generating a frame that contains both WiFi and BLE data which can be demodulated by both commodity WiFi and commodity BLE devices; ii) extracting both WiFi and BLE data from the parallel and concurrent communication of commodity WiFi and BLE senders within the overlapped channel; and iii) supporting concurrent and parallel communication under frequency hopping of BLE. These solutions offer general insights for the design of other parallel inclusive communication schemes among heterogeneous wireless devices in the future.
- We implemented PIC on USRP and commodity WiFi and BLE devices. We also conducted extensive experiments under four real-world scenarios. Compared with existing approaches, PIC significantly i) increases the packet reception ratios by 183%; ii) reduces the round-trip delay time by 590 times and energy consumption by 50.5 times; and iii) improves the throughput under the WiFi and BLE coexistence scenarios.

## 2 SYSTEM OVERVIEW & CHALLENGES

The design goal of PIC is to enable the parallel inclusive communication to (and from) commodity WiFi and BLE devices at the same time within the overlapped channel and achieve dual-standard compliance (i.e., commodity WiFi and BLE devices can simultaneously receive/transmit the frame to/from the PIC device). For the sake of

clarity, the system architecture of PIC (shown in Figure 3) can be divided into downlink and uplink cases.

**Downlink:** The challenge is how to generate a frame that contains both WiFi and BLE data and can be demodulated by both commodity WiFi and commodity BLE devices that are operating at the overlapped channel. To address this challenge, we designed a novel WiFi and BLE symbol mapper, which combines BLE symbols and WiFi symbols together. Therefore, at the commodity WiFi and BLE receivers' sides, they can demodulate the respective WiFi and BLE data without any hardware modification. Specifically, as shown in Figure 3(a) we modulate the subcarriers of the OFDM based WiFi signal such that it contains BLE data at a small time scale (i.e., BLE symbol duration) as well as WiFi data at a large time scale (i.e., WiFi symbol duration) by using the WiFi and BLE symbol mapper. Then the combined signal is sent out by the RF Front-end and can be demodulated by commodity WiFi and BLE devices.

**Uplink:** The challenge is how to demodulate the parallel transmissions (overlapped in both time and frequency domains) from commodity WiFi and BLE senders. To overcome this challenge, our basic idea is to i) extract BLE signal from overlapped WiFi signal by oversampling and using a faster FFT module; and ii) compensate the WiFi signal by leveraging the protocol defined relationship between WiFi and BLE channels. Traditional approaches (e.g., dual-radio gateway) treat this type of communication as collision and require retransmissions, which introduces high latency and energy consumption. However, we discover that by oversampling the incoming signals, it is possible to recover BLE data. Moreover, by applying the signal compensation and utilizing the WiFi's interleaving and convolutional code, we can also recover the WiFi data. In other words, the PIC device can recover both BLE and WiFi data when WiFi and BLE senders are transmitting at the same time within the overlapped channel. As shown in Figure 3(b), the temporal and frequency-domain overlapped WiFi and BLE signals are received by the RF front-end. Since the two BLE GFSK frequencies (which are used to indicate "1" and "0") are overlapped with certain WiFi subcarriers, we can extract the BLE bits to recover the BLE data by using oversampling and FFT. Furthermore, after recognizing the BLE bits, we are able to apply signal compensation to WiFi bits in order to correctly demodulate the WiFi data.

## 3 DOWNLINK

In this section, we provide the background on WiFi and BLE modulation schemes, then introduce the downlink design.

### 3.1 Background: WiFi Modulation Scheme

A basic WiFi modulation scheme contains 3 steps:

**Step 1:** A WiFi transmitter maps the desired bits to sine waves. The WiFi standard uses Quadrature Amplitude Modulation (QAM) by mapping bits to phase ( $\theta = \arctan \frac{Q(k)}{I(k)}$ ) and amplitude ( $a = \sqrt{I^2(k) + Q^2(k)}$ ) in sine waves or equivalent In-phase  $I(k)$  and Quadrature  $Q(k)$  components:

$$S_w(k) = I(k) \cos(2\pi f_w t) - Q(k) \sin(2\pi f_w t) \quad (1)$$

where  $f_w$  is the subcarrier frequency.

**Step 2:** To efficiently utilize the spectrum, WiFi [1] adopts orthogonal frequency-division multiplexing (OFDM), which is a frequency-division multiplexing technique using multiple orthogonal carriers. In the transmitter, the components of the complex input of each

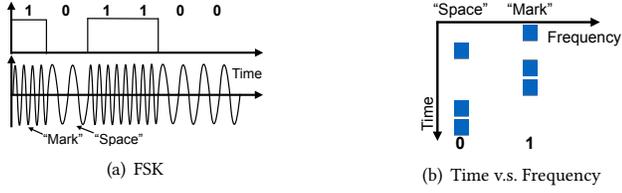


Figure 4: An example of FSK modulation

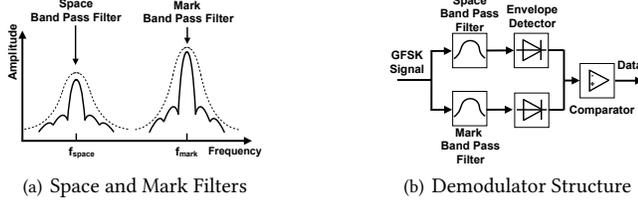


Figure 5: A Simplified BLE Demodulator

subcarrier are multiplied by cosine and sine carriers, and then aggregated together by using Inverse Fast Fourier Transform (IFFT):

$$W(t) = \sum S_w(k)e^{2\pi jkt} \quad (2)$$

**Step 3:** The WiFi sender up-converts the above baseband signal to the desired transmission frequency and sends out the WiFi frames.

At the receiver side, the QAM demodulator obtains In-phase ( $I(k)$ ) and Quadrature ( $Q(k)$ ) values by calculating the definite integral within a WiFi symbol's duration  $T_w$  for the received signal  $R(t)$ :

$$I(k) = \int_{kT_w}^{(k+1)T_w} R(t)(\cos(2\pi f_w t))dt \quad (3)$$

$$Q(k) = \int_{kT_w}^{(k+1)T_w} R(t)(-\sin(2\pi f_w t))dt \quad (4)$$

### 3.2 Background: BLE Modulation Scheme

BLE uses Gaussian frequency shift keying (GFSK), which has the same principle as frequency shift keying (FSK). Figure 4 shows an example of the FSK modulation scheme. As shown in Figure 4(a), when transmitting a data stream "101100", FSK uses the "mark" frequency and "space" frequency to represent data "1" or "0", respectively, where "mark" frequency is higher than the "space" frequency. Figure 4(b) shows the time v.s. frequency plot of the same data stream "101100". By applying the Gaussian filter, GFSK used by BLE can smoothly switch between two frequencies. Thus, the BLE's GFSK signal can be denoted as follows:

$$S_b(t) = A \cos[2\pi f_b t + D_f \int_{-\infty}^t d(\lambda)d\lambda] \quad (5)$$

where  $A$  and  $f_b$  are the amplitude and BLE's carrier frequency, respectively.  $d(t)$  is the baseband signal which is a square wave to indicate data "1" and "0".  $D_f = m/T_b$ , where  $m$  is the modulation index, and  $T_b$  is the symbol interval.

As shown in Figure 5, at the BLE receiver side, by filtering out the signal on space and mark frequencies and comparing the magnitudes between space and mark frequencies, the BLE demodulator is able to determine the transmitted bits. Note that some of the commodity BLE receivers are realized by using an IQ demodulator, which essentially still compares the magnitude difference between space and mark frequencies.

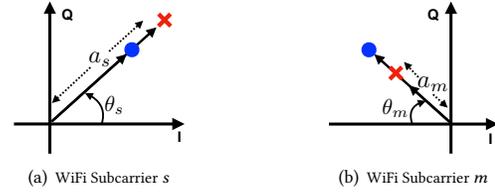


Figure 6: Mapping a BLE Symbol on WiFi Subcarriers

### 3.3 WiFi and BLE Symbol Mapper

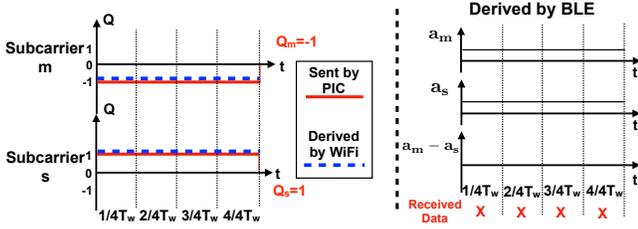
In our downlink design, the main challenge is how to generate a frame that contains both WiFi and BLE data at the PIC sender side. Meanwhile, this frame can be demodulated at commodity WiFi and BLE receiver sides. To address this challenge, we propose a novel WiFi and BLE symbol mapper. The basic idea is to utilize the distinct modulation dimensions (i.e., WiFi modulates on phase and amplitude while BLE modulates on frequency) and time scales (i.e., BLE is four times faster than WiFi) to convey WiFi and BLE data orthogonally.

**Embedding BLE data without changing WiFi's phase:** As described in Sections 3.1 and 3.2, WiFi uses QAM (phase and amplitude based modulation scheme) while BLE uses GFSK (frequency based modulation scheme). Potentially, it is possible to embed BLE symbols into WiFi symbols on the overlapped frequency. We first discuss how to embed BLE data without changing WiFi's phase. Without loss of generality, assuming BLE's "space" and "mark" frequencies are corresponding to WiFi's subcarriers  $s$  and  $m$ , respectively, where  $s$  (or  $m$ ) can be i) one subcarrier when the BLE's "space" (or "mark") frequency is aligned with one subcarrier; or ii) multiple subcarriers when the BLE's "space" (or "mark") frequency is not fully aligned with one subcarrier (it is worth noting that since the OFDM has very dense subcarriers, it is possible to use combined subcarriers to represent the BLE's "space" or "mark" frequency). We embed BLE symbols into WiFi symbols by mapping the original WiFi data on subcarrier  $s$  and subcarrier  $m$  to new values which satisfy one of the following two conditions:

$$\begin{cases} a_m(k) - a_s(k) < 0, & \text{if } d(k) = 0 \\ a_m(k) - a_s(k) > 0, & \text{if } d(k) = 1 \end{cases}$$

where  $d(k)$  is the BLE data.  $a_s(k)$  and  $a_m(k)$  are amplitude values of WiFi subcarriers  $s$  and  $m$ , respectively. These two conditions do not depend on the WiFi symbols' phases  $\theta_s$  and  $\theta_m$  on subcarriers  $s$  and  $m$ , respectively. This is because BLE's GFSK demodulator (introduced in Section 3.2) only cares about the amplitude difference between "space" and "mark" frequencies. Thus, by slightly changing the amplitude ( $a(k) = \sqrt{I^2(k) + Q^2(k)}$ ) of in-phase and quadrature values on WiFi subcarriers  $s$  and  $m$ , we are able to embed BLE data on top of WiFi data. Figure 6 shows an example, as long as the amplitude difference ( $a_m - a_s$ ) is positive, these two WiFi subcarriers ( $s$  and  $m$ ) contain a BLE bit "1", and vice versa. We note that we do not change the phase values  $\theta_s$  and  $\theta_m$  of WiFi symbols.

**Embedding BLE data without changing WiFi's amplitude:** One may argue that changing WiFi's amplitude at the symbol level may have the potential to cause demodulation errors at the WiFi receiver. To further reduce the impact on the original WiFi communication, we propose to change the WiFi's amplitude at the sub-symbol level based on the fact that **WiFi and BLE modulate data on different**



**Figure 7: The PIC sender conducts the original WiFi communication (i.e., does not change WiFi data within one WiFi symbol duration  $T_w$ ). The WiFi receiver gets  $Q_m = -1$  and  $Q_s = 1$  on subcarrier  $s$  and  $m$ , respectively. If we replace the y-axis by  $I$  at the WiFi receiver, we get  $I_s = 1$  and  $I_m = -1$ . In this case, no BLE data is embedded.**

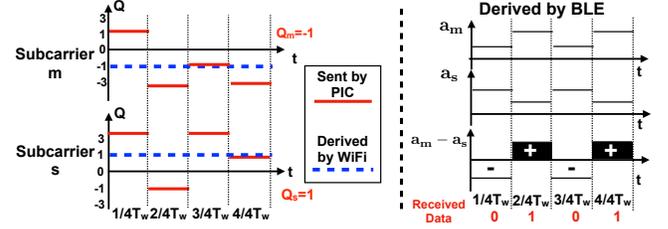
**time scales.** i.e., WiFi’s symbol rate is 250 Ksymbol/s, while BLE’s symbol rate is 1 Msymbol/s. In other words, the symbol duration of WiFi is four times longer than the one of BLE.

As described in Section 3.1, WiFi’s QAM demodulator calculates the definite integral of the input signals within a WiFi symbol duration on each subcarrier. Therefore, as long as the integration result does not change, there is flexibility to tune the WiFi symbol at the sub-symbol level (i.e., one fourth of the WiFi symbol duration) while embedding BLE symbols. To utilize this flexibility, we replace the in-phase components by sub-symbol level values  $I_s(k')$  and  $I_m(k')$ , respectively, where  $k' = 1/4k$ .  $Q_s(k')$  and  $Q_m(k')$  have similar representations. The optimal  $I_s(k')$ ,  $I_m(k')$ ,  $Q_s(k')$ , and  $Q_m(k')$  are obtained by solving the following minimization problem:

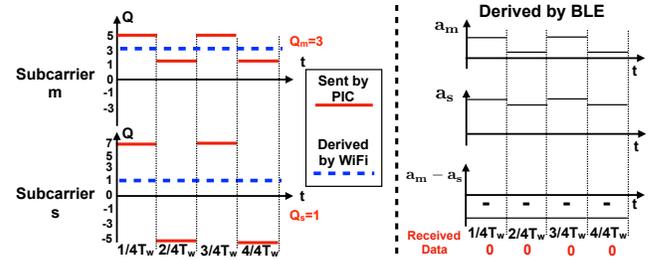
$$\begin{aligned} \min \quad & I_s(k) + I_m(k) + Q_s(k) + Q_m(k) - \\ & \sum_{k' \in [4n, 4n+3]} [I_s(k') + I_m(k') + Q_s(k') + Q_m(k')] \\ \text{s.t.} \quad & a_m(k') - a_s(k') - \epsilon = 0, \quad \text{if } d(k') = 0 \quad \text{or} \\ & a_m(k') - a_s(k') - \epsilon = 0, \quad \text{if } d(k') = 1 \end{aligned}$$

where  $a_m(k') = \sqrt{I_m^2(k') + Q_m^2(k')}$  and  $\epsilon$  is the difference that allows the success demodulation. Our goal is to minimize the difference between i) the aggregated sub-symbol level values  $I(k')$  and  $Q(k')$ ; and ii) the original  $I(k)$  and  $Q(k)$  values at the commodity WiFi receiver. Note that  $I_s(k')$ ,  $I_m(k')$ ,  $Q_s(k')$ , and  $Q_m(k')$  can be varying at the WiFi’s sub-symbol level (i.e., one fourth time duration of a WiFi symbol). The constraint ensures the embedded BLE symbols can be demodulated at the commodity BLE receiver side.

As an example, Figures 7 and 8 demonstrate that our approach can embed BLE data with minimal impact on data at the WiFi receiver side (for the sake of clarity, we only show the  $Q$  value). Figure 7 shows a typical WiFi symbol’s  $Q$  values  $Q_s = 1$  and  $Q_m = -1$  (red solid line) on subcarriers  $s$  and  $m$ , respectively, sent by the PIC sender. Since the WiFi symbol duration is  $T_w$ , the PIC sender does not change within this duration. At the WiFi receiver side, it uses Equations 3 and 4 to calculate the integral (blue dotted line) within  $T_w$  to demodulate the symbol values. At the BLE receiver, since the amplitudes  $a_m$  and  $a_s$  are the same, no valid BLE data is detected. In Figure 8, the PIC sender maps the original WiFi data in each  $1/4T_w$  (which is the duration of one BLE symbol). At the WiFi receiver, since it calculates the integral of the In-phase and Quadrature components, the result is still the same as shown in



**Figure 8: The PIC sender embeds 4 BLE symbols into a WiFi symbol by slightly changing the WiFi’s  $Q$  (or  $I$ ) values in every  $1/4T_w$ , which is equal to one BLE symbol duration. The WiFi receiver still gets  $Q_m = -1$  and  $Q_s = 1$  on subcarriers  $s$  and  $m$ , respectively. If we replace y-axis by  $I$  at the WiFi receiver, we get  $I_s = 1$  and  $I_m = -1$ . The commodity BLE receiver can decode the data  $\{0, 1, 0, 1\}$ , while the WiFi data is not affected at the WiFi receiver side.**



**Figure 9: By expanding the constellation diagram, the special case is resolved.**

Figure 7. However, at the BLE receiver, since the symbol duration is one fourth of that of WiFi, the demodulator outputs “0101”.

The first one fifth of a WiFi symbol is the repeat of the last one fifth, which is known as the cyclic prefix for elimination of inter-symbol interference in OFDM. In the PIC system, instead of directly copying the last one fifth of a WiFi symbol, we solve the aforementioned minimization problem, in which the  $I$  and  $Q$  values are the repeat of the last one fifth but the  $a_m$  and  $a_s$  are specific during the cyclic prefix. After doing this, the results of the minimization problem are appended to the first one fifth of the WiFi symbol.

In our empirical study, the optimization function works well in more than 95% different combinations of WiFi and BLE data. However, there is a special case (e.g., the probability is  $2/16 \times 4/16 \times 12/16 \approx 2\%$  when WiFi uses 16QAM), in which the four bits of BLE data within one WiFi symbol are the same and in total contradiction to the WiFi data on subcarriers  $s$  and  $m$ . For example, assuming that WiFi uses 16QAM to transmit “1000” and “1011” on subcarriers  $s$  and  $m$ , respectively. This results in the amplitude difference  $a_m(k) - a_s(k) > 0$  ( $a_m(k) = 3\sqrt{2}$  and  $a_s(k) = \sqrt{2}$ ). For the same symbol, if the BLE data to be transmitted is “0000” which requires amplitude difference  $a_m(k) - a_s(k) < 0$ , these two conditions contradict each other.

To deal with this special case, we propose to expand the constellation diagram at the sender (e.g., expand 16QAM to a larger space), but have the commodity WiFi receiver still use an unexpanded constellation diagram (e.g., 16QAM) to demodulate.

Take the  $Q$  component of the special case as an example. To send WiFi symbols “1000” and “1011”, the  $Q$  values on subcarriers  $s$  and  $m$  are  $Q_s = 1$  and  $Q_m = 3$ . Though it is impossible to transmit BLE

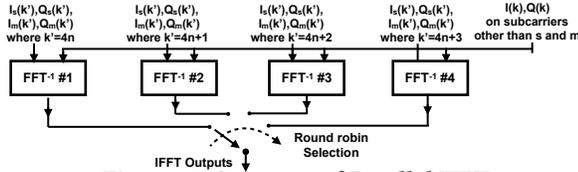


Figure 10: Structure of Parallel IFFT

data “0000” along with these specific WiFi symbols, it is possible to send both the WiFi and BLE data in this special case by expanding the constellation diagram beyond 3 (the maximum value of 16QAM). As shown in Figure 9, by expanding and transmitting  $Q$  values of  $\{5, 1, 5, 1\}$  and  $\{7, -5, 7, -5\}$  on subcarriers  $s$  and  $m$ , respectively, the overall values are still  $Q_s = 1$  and  $Q_m = 3$  at the WiFi receiver. However, PIC successfully transmits “0000” to the BLE receiver because  $a_m$  is less than  $a_s$  in every  $1/4T_w$  time slot.

Our evaluation results also demonstrate that our approach has negligible impact on the original WiFi and BLE communications under various scenarios (detailed in Section 6.4).

### 3.4 Structure of Parallel IFFT

As we described in Section 3.3, to embed the BLE data into a WiFi symbol, we need to chop one WiFi symbol into four pieces so that the BLE data can be accommodated. Since the OFDM signal uses IFFT to aggregate the QAM signals on different subcarriers (introduced in Section 3.1), we have to “speed up” the IFFT module by four times. However, a faster IFFT module may destroy the parameters of a well-defined OFDM signal.

To address this problem, we design a novel IFFT module (shown in Figure 10). In this module, we use four parallel IFFTs (from #1 to #4). Each one is a 64-point IFFT (exactly the same as the original WiFi modulator) to ensure that the whole WiFi channel is divided into 64 subcarriers with each occupying 312.5 KHz. For the input of these four IFFT, the I and Q values on subcarriers other than  $s$  and  $m$  are the same. For the I and Q values on subcarriers  $s$  and  $m$ , since we change the I and Q values in every  $1/4T_w$ , we have the four pairs of I and Q values feed into the four IFFT (e.g.,  $1/4T_w$  corresponded I and Q feeds into IFFT #1). Thus, we will have four different signals at the outputs of the IFFTs. Then a round-robin selector alternately outputs the results from each IFFT during the corresponding time slot (e.g., the selector outputs the results from IFFT #1 during  $1/4T_w$ ).

## 4 UPLINK

In Section 3.3 we introduced the possibility to transmit a WiFi and BLE combined signal at the PIC side by utilizing the difference of modulation schemes as well as time scales. In this section, we discuss how to design the receiver of the PIC device to disentangle WiFi and BLE overlapped signals. In a nutshell, we oversample the incoming combined signal to extract BLE data from WiFi signals and recover the WiFi data by using signal compensation.

### 4.1 BLE Bits

As described in Section 3.3, WiFi and BLE have different symbol rates (i.e., BLE’s symbol rate is as fast as four times of WiFi’s symbol rate). Thus, we oversample the incoming WiFi and BLE combined signals by four times and leverage a fine resolution FFT to extract BLE bits from the I/Q magnitude matrix (each value can be obtained from Equations 3 and 4). Figure 11 shows an example of the FFT

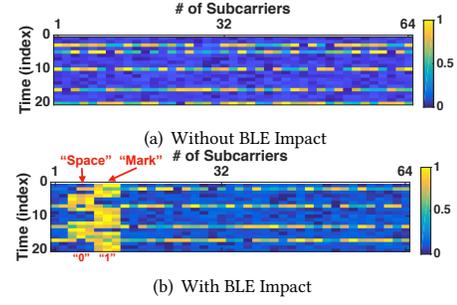


Figure 11: Visualized I/Q magnitude matrices. Compared with no BLE impact (shown in (a)), when the BLE signal exists, it introduces distinct magnitudes (see the yellow blocks on “Space” and “Mark” in (b)).

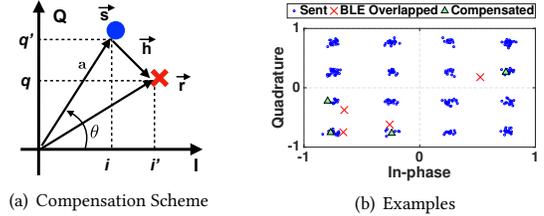


Figure 12: Compensation of WiFi data in BLE overlapped subcarriers: the blue dots indicate the expected bits; the red crosses are the bits affected by the BLE signal; the green triangles are WiFi values after our compensation scheme.

results of the oversampled signals where each column indicates the value of each subcarrier (in a total of 64) and each row is the time index. In Figure 11(a), since there is no BLE signal, the matrix is not affected. In Figure 11(b), with the present of BLE bits at “Space” and “Mark” frequencies during different BLE symbol, we can observe significant magnitude changes (i.e., the yellow blocks) comparing with the remaining area. We denote one element in the matrix as  $y_{k,n} \in \mathbb{Y}^{K \times N}$ , where  $\mathbb{Y}$  is a matrix with  $K$  subcarriers and  $N$  time slots. Each  $y_{k,n}$  can be calculated using the following equation:

$$y_{k,n} = \sqrt{I_{k,n}^2 + Q_{k,n}^2} \quad (6)$$

where  $I_{k,n}$  and  $Q_{k,n}$  are the in-phase and quadrature values on subcarrier  $k$ , respectively. Note that the center frequencies of BLE and WiFi channels are well defined in BLE and WiFi specifications [4, 5], respectively. This means the “Space” and “Mark” frequencies of a BLE channel only affect specific WiFi subcarriers. For example, subcarriers  $m$  and  $s$  are overlapped with “Space” and “Mark” frequencies, respectively. Thus, during a BLE symbol duration (which equals to  $1\mu s$ ), only one value in either vector  $\vec{y}_s$  or  $\vec{y}_m$  is impacted by the BLE “Space” or “Mark”. By knowing the current BLE channel (we will discuss the association issue in Section 5), we monitor the corresponding elements  $y_{s,n} \in \vec{y}_s$  and  $y_{m,n} \in \vec{y}_m$  and compare them with  $\bar{y}_n$  which is the mean across all the unaffected values. By finding the BLE bits present, the BLE bitstream  $b_n$  can be determined by using the following equation:

$$b_n = \begin{cases} 1 & \text{if } y_{m,n} > \bar{y}_n, \\ 0 & \text{if } y_{m,n} \leq \bar{y}_n \end{cases} \quad (7)$$

### 4.2 WiFi Bits

Compared with BLE signals, WiFi signals are much wider (in terms of bandwidth) and much longer (in terms of time duration). Thus,

the majority of the incoming combined signal is an unaffected WiFi signal. However, to correctly demodulate all the WiFi bits, we need to deal with the part affected by the BLE signal. We handle the BLE impact by dividing the WiFi signals into two categories: i) pilot signal and ii) data signal.

**Pilot signal:** The pilot signal (which usually appears in the preamble) is predefined and used to estimate the channel. By having the estimated channel, the demodulator compensates the data signal to correctly demodulate. Therefore, if the BLE signal is overlapped with the pilot signal, the wrongly estimated channel causes errors in compensation which yields an incorrect demodulation of the WiFi data. To solve this problem, we drop the BLE affected pilot signal at the PIC receiver to prevent bad channel estimation. Specifically, after extracting the BLE bits (as described in Section 4.1), we know which subcarriers are overlapped with BLE signals at a specific time. During this time, if the pilot signal is incoming, we drop the pilot signal on these subcarriers and interpolate the value from a neighboring (in terms of frequency) pilot signal. By doing this, the impact of the BLE signal can be minimized that the channel estimation is valid for data signal demodulation.

**Data signal:** The data signal is used for carrying the real user data. If BLE data is transmitted in “space” and “mark” frequencies that are overlapped with a small portion of WiFi data subcarriers, it may cause distortions on the concurrently transmitted WiFi data at the PIC receiver side. To address this issue, we utilize WiFi properties of signal compensation and interleaving/convolutional code.

As shown in Figure 12(a),  $\vec{s}$  is the expected data signal which carries data on the in-phase component  $I$  and quadrature component  $Q$ .  $\vec{r}$  is the vector that is affected by concurrently transmitted BLE signal.  $\vec{h}$  (the difference between  $\vec{s}$  and  $\vec{r}$ ) is the distortion caused by BLE data. The receiver estimates the channel ( $\vec{h}$ ) by using the pilot signal (as we described before, the invalid pilot signal has been dropped and interpolated). The compensated values  $i$  and  $q$  can be calculated using the following equations:

$$i = h_i[i' \cos(\theta) + q' \sin(\theta)]; \quad q = h_q[q' \cos(\theta) - i' \sin(\theta)] \quad (8)$$

where  $i'$  and  $q'$  are the computed pilot signal's I/Q values.  $h_i$  and  $h_q$  are the in-phase and quadrature components of vector  $\vec{h}$ , respectively. These values are sent to the WiFi OFDM demodulator and the original WiFi data can be recovered.

Figure 12(b) shows an example of the compensation scheme. The blue dots are expected values. When the BLE signal is present, the received WiFi signal is shifted (red crosses). By using compensation, the WiFi signal can be recovered (green triangles).

The impact of the BLE signal can be further reduced by WiFi's inherent interleaving and convolutional code. These two techniques are used to provide redundancy which can suppress burst errors (i.e., several bit errors which are likely to be seen close to each other). As we stated above, the BLE signal affects a small portion (in terms of frequency and time) of the whole WiFi signal. Therefore, the WiFi signal is likely to be corrected by interleaving and convolutional code if the signal is not well compensated by the channel estimation.

### 4.3 Synchrony between WiFi and BLE

In the concurrent downlink (transmitting to both WiFi and BLE), synchrony is not an issue, because the PIC gateway controls the

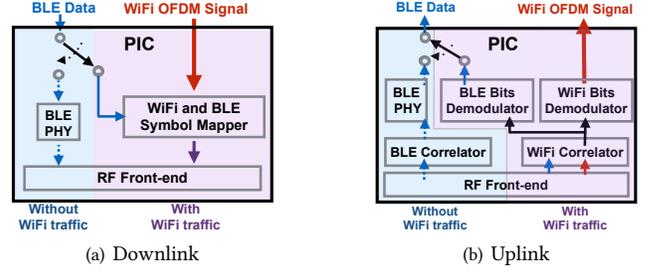


Figure 13: Synchrony between WiFi and BLE

physical layers of both WiFi and BLE. In the concurrent uplink, it is possible that the WiFi and BLE signals do not arrive or end at the same time. To address this issue, we use two correlators to decide whether to launch the BLE bits demodulator because the properties of WiFi signals are very different from those of BLE signals in terms of bandwidth (20MHz v.s. 2MHz) and modulation dimensions (amplitude and phase v.s. frequency). The correlators can recognize the WiFi or BLE signals easily by matched filters. The two correlators (i.e., WiFi and BLE) are used to compare the similarity between the received signal and a normal WiFi or BLE signal by using the following equations:

$$C_w[k] = \sum_{n=0}^{\infty} x_w[n-k]y[n]; \quad C_b[k] = \sum_{n=0}^{\infty} x_b[n-k]y[n] \quad (9)$$

where  $y[n]$  is the received signal,  $x_w[n]$  and  $x_b[n]$  are the normal WiFi and BLE signals, respectively.  $C_w[k]$  (or  $C_b[k]$ ) is the result that indicates the similarity between  $y[n]$  and  $x_w[n]$  (or between  $y[n]$  and  $x_b[n]$ ). By comparing  $\max_{k \in [0, n]} C_w[k]$  and  $\max_{k \in [0, n]} C_b[k]$ , the

PIC receiver can identify whether there exists WiFi traffic in the received signal. For example, if  $\max_{k \in [0, n]} C_w[k] > \max_{k \in [0, n]} C_b[k]$ , it

indicates that WiFi traffic exists. Then, the received signal goes to the right-hand side (i.e., purple colored part) of Figure 13(b). To figure out whether BLE signals are embedded in the WiFi signal and extract the BLE bits, we can use the techniques introduced in Section 4.1). Otherwise, the signal goes to the left-hand side (i.e., cyan colored part), and the BLE signal is demodulated by using the standard BLE PHY layer.

We note that the concurrent uplink does not require symbol level synchronization between BLE and WiFi because the oversampled signals and fine resolution FFT can extract the BLE bits wherever the BLE signals are. For WiFi bits, although partially overlapped BLE signals may impact the symbol on corresponding subcarriers, WiFi's compensation scheme can help mitigate the impact. Moreover, the redundancy provided by interleaving/convolutional code can also help mitigate the impact. Our single antenna-based PIC design is generic and complementary to other wireless techniques (such as full-duplex). With a single antenna, the gateway cannot handle the following two scenarios: i) Wi-Fi downlink coexisting with BLE uplink; and ii) Wi-Fi uplink coexisting with BLE downlink. However, by combining our single antenna-based PIC design with the latest full-duplex techniques (such as [11, 13, 36]), we can easily enable the gateway to handle the above two scenarios.

## 5 ASSOCIATION BETWEEN PIC AND BLE

After we enable the uplink and downlink communications between the PIC gateway and BLE devices, we introduce how to enable the

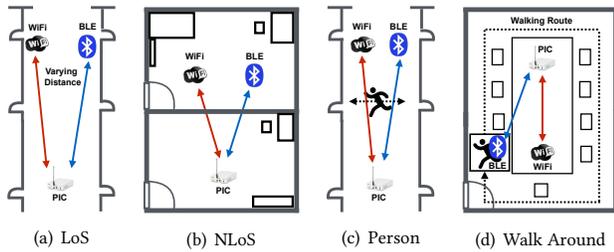


Figure 14: Experimental Setup

association (i.e., pairing) between the PIC gateway and BLE devices in this section. A typical BLE network (i.e., piconet) contains a master device and multiple slave devices. During a pairing process, the master device shares a unique pseudorandom sequence (which is known as the hopping sequence) and time synchronization information with the slave devices. After the pairing is accomplished, the master and slave devices rapidly hop up to 1,600 times per second among BLE channels on the 2.4 GHz ISM band. Since Bluetooth 1.2 [8], the adaptive frequency hopping (AFH) is introduced. In AFH, the BLE device labels each channel as an *available* channel or an *unavailable* channel. By having all the channels labeled as *available* channels or *unavailable* channels, a mapping from *unavailable* channels to *available* channels is formed to reallocate the BLE device such that the BLE device always hops within the set of *available* channels.

In our design, the PIC gateway serves as the master device and conducts pairing with other BLE devices (i.e., slave device) by exchanging the frequency hopping sequence information. By using the `HCI_SET_AFH_CHANNEL_CLASSIFICATION` function (defined in the BLE standard [5]), we label the WiFi overlapped BLE channels as *available* channels and the non-overlapped channels as *unavailable* channels. By doing this, we can enable BLE devices to hop within the WiFi overlapped channel.

The advantage of our design is that a PIC gateway can communicate with multiple BLE devices and a WiFi device in parallel within a single time slot. This is because one WiFi channel is much wider than a BLE channel. By providing different hopping sequences within the overlapped WiFi channel to different slave devices at the same time, our design can support parallel communication between the PIC gateway and multiple BLE devices within a single time slot. While a traditional BLE master device can only communicate with one slave device at a specific time slot, even when the master device is associated with multiple slave BLE devices. The number of concurrently communicating BLE devices can be further increased if we allow the PIC gateway to concurrently work under three non-overlapped WiFi channels (e.g., 1, 6, and 11), which is increasingly popular in new buildings with the pressing demand of improving the spectrum utilization and connecting more WiFi devices.

## 6 EVALUATION

### 6.1 Implementation

The PIC gateway design is fully implemented on a single antenna USRP. We used off-the-shelf BLE TI Sensor Tags [7] as BLE receivers and senders. We also use a Dell XPS 9550 laptop as a WiFi sender (by using `iperf` to transmit data) and receiver (by using `wireshark` to capture data). Because the WiFi protocol includes different modulation schemes, to show the generality of PIC, we implemented

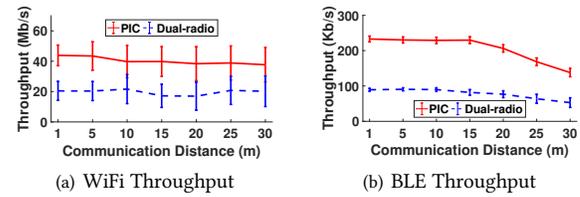


Figure 15: Overall throughput v.s. distance. PIC's throughput is more than 2 times better than the dual-radio gateway's throughput at different communication distance.

two major WiFi modulation schemes: i) 16-QAM with coding rate 3/4 and ii) 64-QAM with coding rate 3/4 under two different bandwidth setups: i) 20MHz and ii) 40MHz specified in IEEE 802.11n. To ensure the reproducibility of our experimental results, on each data point, we continually transmitted and received packets among PIC, WiFi, and BLE for 10 minutes and repeated for 10 times. The error bars in the figures represent standard deviation. The packet sizes of WiFi and BLE are 16~2320 bytes and 11~262 bytes, respectively. To compare with PIC, we also implemented the following strategies:

**W2W:** Pure WiFi to WiFi communication without interference from BLE. This serves as the upper bound when comparing the communications i) from the PIC gateway to the WiFi device (P2W) or ii) from the WiFi device to the PIC gateway (W2P).

**B2B:** Pure BLE to BLE communication without interference from WiFi. This serves as the upper bound when comparing the communications i) from the PIC gateway to the BLE device (P2B) or ii) from the BLE device to the PIC gateway (B2P).

**Dual-radio:** The dual-radio gateway communicates to both the WiFi and the BLE device. This serves as the baseline.

**Oracle:** The theoretical upper bound of fully concurrent transmissions of BLE and the WiFi, in which all the concurrently transmitted WiFi and BLE bits can be resolved.

### 6.2 Experimental Scenarios

To evaluate the robustness of PIC, we conducted experiments for both the uplink and downlink in two stationary scenarios and two mobile scenarios:

**S1. Line-of-sight (LoS):** There is no obstacle or moving object between the sender and the receivers. The distance between the WiFi and PIC gateway is varying while the BLE is fixed. The varying distance is also applied to BLE while WiFi is fixed (Fig. 14(a)).

**S2. None-line-of-sight (NLoS):** The sender and the receivers are in different rooms with a multi path rich channel (Fig. 14(b)).

**M1. Person moving between sender and receiver:** The physical channel between senders and receivers (10m apart) is impacted by a walking person in the middle (Fig. 14(c)).

**M2. Person moving around with BLE attached:** The physical channel is dynamically changing that the BLE device is attached to the wrist of a person who is walking around the room (Fig. 14(d)).

### 6.3 Overall Performance

In this section, we jointly evaluate PIC's performance in terms of overall throughput, packet loss, round trip delay time, energy consumption, and generality.

**6.3.1 Throughput over Distance.** In this experiment, we compare the overall throughput (including both of the downlink and uplink transmissions) of PIC with the traditional dual-radio gateway over

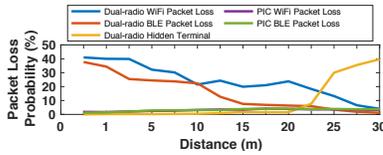


Figure 16: Packet Loss Probability v.s. Distance

distance. As shown in Figure 15(a), from 1 meter to 30 meters, the throughput of WiFi to PIC communication is up to 2.5 times higher than the throughput of the dual-radio gateway. A similar trend holds for the BLE to PIC communication ( Figure 15(b)). The main reason is at downlink, PIC gateway can concurrently transmit to both the WiFi and BLE receivers while dual-radio needs to use TDMA to allocate WiFi and BLE packets into different time slots. At uplink, the collisions at dual-radio side reduced the packet reception ratio. However, the PIC gateway is able to resolve the collisions.

6.3.2 *Packet Loss.* In this experiment, the PIC gateway (or dual-radio gateway) communicates (including uplink and downlink) with both BLE and WiFi devices in the NLoS scenario (shown in Figure 14(b)). We measured packet loss, which is caused by collisions and delays by using an RF streaming network monitor, under different distances between the WiFi and BLE devices. We captured the collisions by filtering out abnormal received power (a collided packet has a higher power due to the simultaneous transmissions). We then measured the delays between the packets after the collision. Figure 16 shows the results. For the dual-radio approach, packet loss may be caused by the following reasons:

**Packet loss due to collisions:** Since BLE does not use CSMA, when WiFi is transmitting, a BLE transmission will corrupt both the WiFi and BLE packets. When the distance is increasing, the collision is less because the signal strength is lower.

**Packet loss due to delay:** A WiFi transmitter senses the channel before transmission. If the channel is busy, it will back-off (i.e., CSMA). Therefore, the WiFi transmission has a less probability to corrupt an on-going BLE transmissions. However, the exponential back-off would cause the packet to be delayed. A long delay causes a packet loss. Overall, WiFi packets have a higher packet loss probability than that of BLE.

**Packet loss due to a hidden terminal:** If the WiFi devices are far away from the BLE devices, the WiFi transmitter cannot sense the transmission of BLE packets (i.e., hidden terminal problem). However, both the transmissions (from WiFi and BLE) will collide at the gateway side. Our results show that this collision probability increases when the distance between WiFi and BLE increases.

For PIC, since we can i) resolve the collided packet at the PIC gateway; and ii) concurrently transmit different packets to WiFi and BLE devices to avoid the exponential back-off time, the packet loss probabilities for both WiFi and BLE are very low. In Figure 16, their curves are overlapped with each other.

6.3.3 *Round Trip Time.* Low energy devices have the trade-off between delay and energy consumption. To achieve shorter delay, the devices need to be awake most of the time. BLE standard [5] defines the Connection Event (each instance of communication between two devices) and the Connection Interval (the period of communication). The Connection Interval ( $CI$ ) has a minimum of  $7.5ms$  and maximum of  $4s$  with a step of  $1.25ms$ . For different

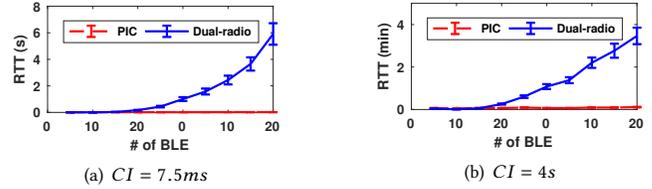


Figure 17: Round Trip Time (RTT). Due to the collisions of BLE and WiFi packets, dual-radio gateway’s RTT is 590 times higher than PIC’s RTT.

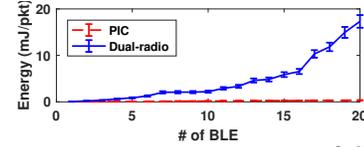


Figure 18: Energy consumption per successfully transmitted and acknowledged packet at the commodity BLE sender side. Since PIC can recover the overlapped WiFi and BLE packets, the energy consumption is more than 50 times lower than traditional dual-radio gateway.

applications and requirements, the devices use a different Connection Interval ( $CI$ ) to balance connectivity and energy consumption. Figures 17(a) and 17(b) show the Round Trip Time (RTT) results under  $CI = 7.5ms$  and  $CI = 4s$ , respectively. When the Connection Interval  $CI = 7.5ms$  and  $CI = 4s$ , compared with dual-radio, PIC shows an extremely low RTT around  $0.01s$  and  $0.1m$ , respectively. RTT is a critical metric to some applications. Due to the packet loss (collided with WiFi packet), the dual-radio gateway has high RTT on BLE communications (especially in a WiFi and BLE coexistence environment). On the contrary, since PIC can demodulate the WiFi and BLE packets concurrently, the BLE sender does not need to retransmit even when the previous packet is overlapped with WiFi packet. Thus, the large number of retransmissions are dismissed

6.3.4 *BLE Energy Consumption.* Figure 18 shows how much energy is needed for the BLE sender to successfully transmit a packet to PIC (or the dual-radio gateway) and receive the acknowledgement. We observe that an exponential increase of dual-radio’s energy consumption happens when the number of BLE devices increases. For example, the energy per packet reaches  $17mJ$  when the number of BLE devices reaches 20. This is because the BLE protocol uses frequency hopping and lacks channel sensing techniques. When the number of BLE devices increases, BLE packets have a higher probability to collide with WiFi packets if using traditional dual-radio gateway side. The retransmission requested by the dual-radio gateway increases the energy consumption per packet. On the contrary, PIC can successfully recover both WiFi and BLE packets that are overlapped with each other so that the number of retransmissions can be reduced. With less retransmissions, the energy consumption for BLE devices and gateway drops significantly. As shown in the figure, the energy consumption per successfully transmitted and acknowledged packet is as low as  $0.35mJ$ .

6.3.5 *Support of Different WiFi Modulation Schemes and Bandwidth.* In Figure 19, we compare PIC with four different modulation and bandwidth combinations: i)  $20MHz/16QAM$ ; ii)  $20MHz/64QAM$ ; iii)  $40MHz/16QAM$ ; and iv)  $40MHz/64QAM$ . The results show that PIC can achieve similar performance in both downlink and uplink.

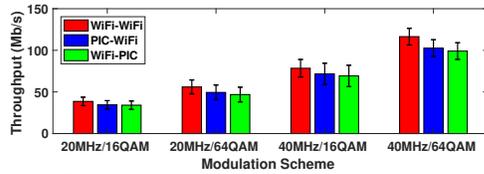
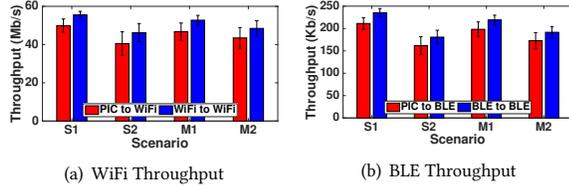


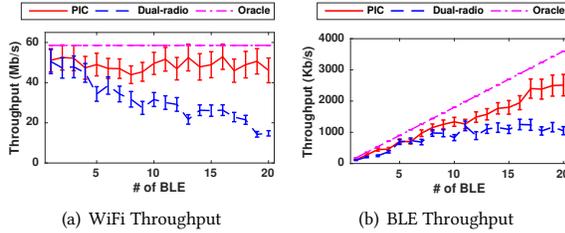
Figure 19: PIC can support different WiFi modulation schemes and bandwidth.



(a) WiFi Throughput

(b) BLE Throughput

Figure 20: Comparing PIC with a single WiFi or BLE traffic in different scenarios, PIC achieves similar throughput as if WiFi and BLE traffic are transmitted separately.



(a) WiFi Throughput

(b) BLE Throughput

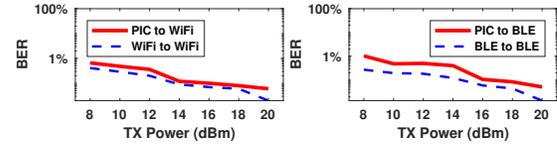
Figure 21: Downlink throughput v.s. # of BLE receivers. When the # of BLE receivers increases, the throughput of PIC to WiFi transmission is stable and the throughput of PIC to BLE transmission linearly increases.

We conclude that the design of PIC is generic to support different iterations of the WiFi standard.

## 6.4 Downlink Performance

**6.4.1 Throughput. i) Impact on the original WiFi or BLE traffic.** In our experiment, the PIC sender is transmitting data to both a WiFi and a BLE receiver. We compare the throughput of PIC to WiFi (or PIC to BLE) with a *single* WiFi to WiFi (or BLE to BLE) communication. In a single WiFi to WiFi communication, there is no BLE interference. Similarly, in a single BLE to BLE communication, there is no WiFi interference. Since the *single* WiFi to WiFi (or BLE to BLE) communication does not have interference between WiFi and BLE, the experimental results serve as the real-world upper bounds for PIC. Figure 20(a) compares the throughput of PIC to WiFi with WiFi to WiFi under four scenarios: normal LoS (S1), multipath rich NLoS (S2), people moving (M1), and wearing the BLE (M2). We conclude that the throughput of the PIC to WiFi communication is very close to a single WiFi to WiFi communication (the difference is within 10%). Similarly, there is no obvious throughput difference between an inclusive PIC to BLE communication and a single BLE to BLE communication (the difference is within 7%). This is because PIC can embed four BLE symbols in one WiFi symbol with negligible impact to WiFi communication.

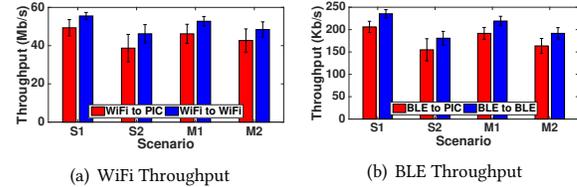
**ii) Impact of the number of BLE receivers.** In this setting, we want to explore PIC's performance under a large number of BLE receivers. Figure 21(a) shows the downlink throughput for the WiFi communication. The results show that PIC maintains a constant



(a) WiFi

(b) BLE

Figure 22: Downlink BER v.s. TX Power (y-axis is in log scale). PIC downlink can achieve a BER as low as WiFi to WiFi and BLE to BLE communications.



(a) WiFi Throughput

(b) BLE Throughput

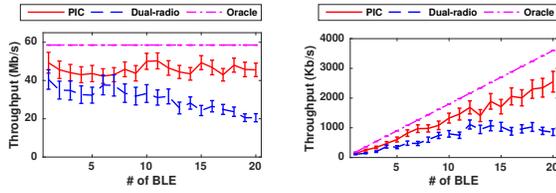
Figure 23: Comparing PIC uplink with a single WiFi or BLE.

speed of around 50.9 Mbit/s while the dual-radio keeps reducing to 14.7 Mbit/s when the number of BLE reaches 20. When comparing with oracle (the theoretical upper bound), PIC is slightly lower than it because of the noise, imperfection of the hardware, channel fading, and etc. The aggregated throughput of the BLE communication is shown in Figure 21(b). We can observe that both dual-radio and PIC are increasing when the number of BLEs is less than and around 10. However, the increasing trend on dual-radio slows at around 15 BLE devices and then starts to drop, while the PIC keeps increasing from 1 to 20 BLE devices. Compared with oracle, PIC follows the linearly increasing trend. This is because the dual-radio gateway needs to allocate the WiFi and BLE packets at different time slots to avoid a collision. When the number of BLE is small, the dual-radio gateway can handle it without much degradation but the situation is worsening when the number of BLE grows while PIC allows concurrent and parallel transmissions of WiFi and BLE packets. We also note that when the number of WiFi devices increases, we will see similar trend that the performance of dual-radio degrades while that of PIC keeps linearly increasing.

**6.4.2 Bit Error Rate.** Figure 22(a) and 22(b) show the Bit error rate (BER) from PIC to WiFi and BLE devices, respectively, while the TX Power is increasing. By comparing with normal WiFi to WiFi and BLE to BLE communications, we observe that the PIC downlink can achieve similar BER under different TX powers (less than 1%). The reason is that our WiFi and BLE symbol mapper (introduced in Section 3.3) can efficiently embed BLE bits into WiFi symbol and transmit to BLE and WiFi receivers concurrently.

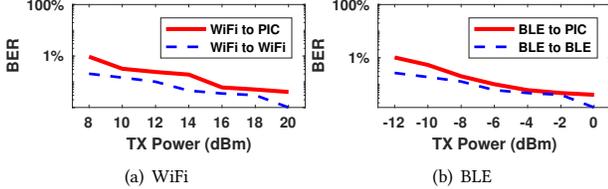
## 6.5 Uplink Performance

**6.5.1 Throughput. i) Impact to the original WiFi or BLE traffic.** Figure 23 compares the uplink performance of PIC with that of a single WiFi traffic (i.e., no BLE traffic) or a single BLE traffic (i.e., no WiFi traffic). Since the *single* WiFi to WiFi (or BLE to BLE) communication does not have interference between WiFi and BLE, the experimental results serve as the real-world upper bounds for PIC. Figure 23(a) compares the WiFi throughput under four scenarios: S1, S2, M1, and M2. We can observe that the throughput of PIC is very close to that of a single WiFi to WiFi communication because we used the compensation technique to recover the BLE overlapped



(a) WiFi throughput v.s. # of BLE (b) BLE throughput v.s. # of BLE

Figure 24: Uplink throughput v.s. # of BLE



(a) WiFi (b) BLE

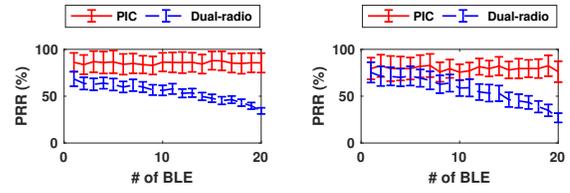
Figure 25: Uplink BER v.s. TX Power (y-axis is in log scale).

subcarriers. Similarly, the throughput difference between PIC to BLE and a single BLE to BLE communication is small.

**ii) Impact of the number of BLE receivers.** Figure 24(a) shows the comparison between commodity WiFi transmitting to PIC and dual-radio gateway when the number of BLE devices is increasing. Different from downlink, the throughput has a difference between dual-radio and PIC when the number is small (i.e., less than 5) because the BLE devices do not have a CSMA type technique to avoid collision. When the number of BLE devices increases, the throughput difference between dual-radio and PIC increases. Finally, the throughput of PIC reaches 3.15 times that of dual-radio when there are 20 BLE devices. For the communication with BLE device (results shown in Figure 24(b)), the aggregated throughput of PIC is 2.4 times that of dual-radio when the number of BLE devices reaches 20. This is because PIC can resolve the collisions between WiFi and BLE packets while dual-radio treats the packets as corrupted. For both the WiFi and BLE throughput, PIC follows the trend of oracle while the throughput significantly drops in dual-radio approach. We note there is a gap between PIC and oracle because of the noise, imperfection of the hardware, channel fading, and etc. We also note that with the increasing number of WiFi devices, the result shows similar trend because more WiFi traffic causes more collisions.

**6.5.2 Bit Error Rate.** To further characterize PIC’s uplink capability, we compared the scheme to traditional WiFi and BLE communications’ Bit Error Rate (BER) with different TX powers. We performed this experiment with a Rohde and Schwarz Vector Signal Analyzer. In the WiFi to PIC communication, the results (Figure 25(a)) demonstrate that PIC’s uplink design can achieve similar performance to the traditional WiFi to WiFi communication such that the BER is lower than 1% when the TX Power is greater than 20 dBm. The results for the BLE to PIC communication are shown in Figure 25(b), PIC also shows trend similar to a normal BLE to BLE transmission.

**6.5.3 Packet Reception Ratio.** Figure 26 shows the results of packet reception ratio. For the PIC to WiFi link (Figure 26(a)), the dual-radio shows an exponential decrease when the number of BLE devices increases. The PRR reduces to 28% when the number of BLE devices reaches 20. In our approach, the PRR is stable across different numbers of BLE devices. For the PIC to BLE link, we aggregated the received packets across all the BLE devices (the results are shown in Figure 26(b)). It shows a similar trend to the



(a) WiFi PRR v.s. # of BLE (b) BLE PRR v.s. # of BLE

Figure 26: Uplink packet reception ratio v.s. # of BLE.

PIC to WiFi link. The reason is that the WiFi and BLE devices adopt different interference avoidance techniques. The collision cannot be always avoided, especially when the traffic is busy. For the dual-radio, the WiFi and BLE packets collide more. Thus, the PRRs of both WiFi and BLE drop. For our approach, because we can demodulate the WiFi and BLE packets concurrently, the packet loss due to collision is negligible.

## 7 DISCUSSION

**Generality:** Given the exponentially increasing number of IoT devices and limited wireless spectrum, there is a pressing need for more efficient communication methods. The parallel inclusive communication (PIC) concept introduced in this paper is very generic and opens a new door for more efficient communication within the shared wireless spectrum. In this paper, we already demonstrated that using PIC is possible to achieve efficient communication among WiFi and BLE. Without changing the existing standards, the PIC concept can also be applied to different wireless communication protocols, such as i) ZigBee and BLE; ii) LoRa, LTE, and ZigBee in the shared ISM 900 MHz band; and etc. We note that protocol diversities (including the difference of modulation dimensions and symbol rates) and receiving tolerance are two main factors for applying PIC technique. However, these two factors need to be treated systematically. For example, although both WiFi and ZigBee use phase-based modulation scheme, we believe that PIC can be applied to achieve inclusive communication for WiFi and ZigBee even without extra FFTs and oversampling. The reasons are i) ZigBee’s symbol duration is  $16\mu s$  which is four times longer than WiFi; ii) ZigBee adopts DSSS which provides high tolerance on interfered signals at the receiver side. On the other hand, PIC requires these protocols have overlapped channels in the frequency domain. For example, PIC cannot be applied to BLE and LTE because these protocols do not have an overlapped channel.

**Portability:** It is easy to transfer PIC to different versions of WiFi and Bluetooth protocols. Since PIC utilizes the OFDM signals’ property (i.e., different frequency components can be treated separately), PIC technique can be extended to any OFDM-based WiFi protocol. For Bluetooth protocols, we mainly rely on the fact that Bluetooth’s symbol rate is 4x faster than that of WiFi. Recently released Bluetooth 5.0 provides 2 times higher throughput (i.e., 2 Mbps) than the one (i.e., 1 Mbps) in Bluetooth 4.0 used in PIC by doubling the symbol rate. Therefore, PIC technique can provide a better performance when it is used in Bluetooth 5.0.

**Limitation & Opportunity:** In this paper, we introduce a new direction for more efficient communication among IoT devices within the shared wireless spectrum. Due to the hardware constraint, currently PIC does not support the uplink and downlink transmissions in full-duplex mode (i.e., concurrent uplink and downlink transmissions). This is because existing off-the-shelf WiFi and BLE

devices do not support full-duplex. Our future plan is to leverage the advanced multi-antenna or full-duplex techniques to realize concurrent uplink and downlink transmissions by using customized devices. The experience gained in this direction can also provide some guidance to the industry for designing more spectrum efficient communication standards in increasingly crowded environments.

**Overhead:** The main computational overhead comes from the fine resolution FFT. Since the most commonly used FFT algorithm has a computational complexity of  $O(N \log N)$  [19], where  $N = 64$  is the number of data points to be processed by the FFT algorithm. A WiFi device needs to conduct one FFT for one WiFi symbol. To extract BLE bits, PIC needs to conduct four FFT within the duration of a WiFi symbol. Thus, the maximum computational overhead is only four times of a WiFi device, which is not much for an edge device. For example, many WiFi chips have already implemented higher resolution FFT and sampling rate for supporting wider bandwidth (i.e., 80 MHz and 160 MHz). The power consumption overhead mainly comes from the parallel IFFT and oversampling. The parallel IFFT requires four 64-point IFFTs to run at the same time, which introduces four times of power consumption. However, since the IFFT is one part of a synthesizer which costs around 17% of a transmitter (according to the analysis in [40]), the overall power consumption will not significantly increase. For the oversampling part, the analog to digital converter (which conducts the sampling) only costs 1.6% of the total power consumption of a receiver [40]. Since the power consumption scales linearly with the sampling rate [9], PIC only increases the sampling rate by 4 times, the increased power consumption overhead is only 6.4% which is negligible. On the other hand, PIC can reduce the BLE senders' energy consumption by more than 50 times (see Figure 18). Therefore, the advantages (e.g., low energy consumption, high PRR, and low delay) provided by PIC technique outweigh the introduced overheads.

## 8 RELATED WORK

The related work can be divided into two categories:

**Interference from the same technology:** To mitigate the interference from the same technology, researchers proposed various techniques, such as MIMO [39, 53], parallel processing [32], AP selection [62], spectrum adaptation [60], channel selection [43], network selection [47], user selection [56], advanced antenna [44, 59], scheduling [50], reducing redundant content [21], and reducing loss by wired network [12]. When a collision happens, retransmission [35] and interference cancellation [52] can be applied to reduce the impact to the network performance. mZig [38] resolves up to 4 ZigBee packets by exploring physical layer feature. Glossy [22] exploits constructive interference for rapid reliable flooding. Unlike the above approaches considers the performance improvement of a single wireless technology, the PIC works on another dimension, in which PIC explores the diversities among heterogeneous wireless technologies (i.e., wider-band WiFi and narrower-band BLE) for more efficient communication for IoT networks.

**Interference across technologies:** Researchers proposed different approaches to mitigate the interference among heterogeneous wireless devices in cellular networks [48] and multi-hop wireless networks [33]. The first approach is to eliminate the interference by isolating the signals in space, frequency (e.g., OFDM subcarrier suppression [31, 45, 49]), or time (e.g., CSMA [29]). To further improve

the performance of CSMA, researchers introduced more advanced design such as compressing backoff [28], improving the fairness of the backoff [46, 55], and adding collision notification [51] or synchronization schemes [54]. The second approach tries to mitigate the cross-technology interference by using interference cancellation [24, 61] or utilizing coexistent features to do cross-technology communication [14–17, 37, 58, 63, 64]. The most related approaches are  $B^2W^2$  [17], WEBee [42], PMC [18], and Chiron [41] that enable communications between WiFi and ZigBee (or BLE). The throughput of  $B^2W^2$  is as low as 850 bps, because it does not leverage the unique modulation schemes of WiFi and BLE. The spectrum utilization of WEBee is low because the whole 20MHz WiFi channel can only emulate up to two ZigBee channels (i.e., 2 MHz) with upto 250 Kbps throughput, which result in 0.025 bit/s/Hz spectrum utilization. Since ZigBee's throughput (i.e., 250 Kbps) is much lower than the original WiFi's throughput (e.g., 288.8 Mbps for 802.11n), WEBee has 3 orders of magnitude lower spectrum utilization than the original WiFi. Therefore, the emulation method proposed by WEBee is not a spectrum efficient solution. PMC enables parallel communications to WiFi and ZigBee but the communication is asymmetric because it does not support the uplink communication. Moreover, PMC cannot concurrently transmit both WiFi and ZigBee data within the same frequency. Chiron only enables the concurrent communication of WiFi and ZigBee. The techniques introduced in Chiron cannot directly be applied to solve the problems in a WiFi and BLE coexistence environment because ZigBee and BLE use totally different protocols. Furthermore, PIC supports up to 20 BLE devices while Chiron only supports four. Different from above approaches, PIC leverages the unique modulation schemes of WiFi and BLE for concurrent and parallel communication within the overlapped channel. Our evaluation results demonstrate that PIC achieves similar throughput as if the WiFi and BLE traffics are transmitted separately, which indicates that PIC can increase the spectrum efficiency. Moreover, PIC significantly decreases the communication delay and power consumption.

## 9 CONCLUSION

In this work, we present a new direction for efficient communication – the Parallel Inclusive Communication (PIC) technique that leverages the unique modulation schemes of WiFi and BLE for parallel bi-directional transmission of both WiFi and BLE data at the same time within the overlapped channel. The transmitted WiFi and BLE data can be modulated/demodulated by commercial off-the-shelf devices. PIC's system architecture i) naturally fits at the edge of the Internet; ii) supports edge computing; and iii) provides efficient, concurrent, and parallel communications among the ever-increasing number of IoT devices. We extensively evaluated our system under one ideal setting and five real-world scenarios. Results show that compared with existing approaches, PIC can significantly i) increase the packet reception ratios by 183%; ii) reduce the round-trip delay time by 590 times and energy consumption by 50.5 times; and iii) improve the throughput under the WiFi and BLE coexisting scenarios.

## ACKNOWLEDGMENTS

This work is supported in part by NSF grants CNS-1824491, CNS-1824355, and CNS-1652669. We also thank anonymous shepherd and reviewers for their valuable comments.

## REFERENCES

- [1] [https://en.wikipedia.org/wiki/IEEE\\_802.11](https://en.wikipedia.org/wiki/IEEE_802.11).
- [2] [https://en.wikipedia.org/wiki/List\\_of\\_WLAN\\_channels](https://en.wikipedia.org/wiki/List_of_WLAN_channels).
- [3] [https://en.wikipedia.org/wiki/Multiuser\\_detection](https://en.wikipedia.org/wiki/Multiuser_detection).
- [4] <http://standards.ieee.org/findstds/standard/802.11-2016.html>.
- [5] <https://www.bluetooth.com/specifications/bluetooth-core-specification>.
- [6] [http://www.ieee802.org/802\\_tutorials/04-July/802CoexistenceTutorialJuly04a.pdf](http://www.ieee802.org/802_tutorials/04-July/802CoexistenceTutorialJuly04a.pdf).
- [7] [http://www.ti.com/ww/en/wireless\\_connectivity/sensortag/](http://www.ti.com/ww/en/wireless_connectivity/sensortag/).
- [8] 2005. IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 15.1a: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Wireless Personal Area Networks (WPAN). *IEEE Std 802.15.1-2005 (Revision of IEEE Std 802.15.1-2002)* (June 2005), 1–700. DOI: <https://doi.org/10.1109/IEEESTD.2005.96290>
- [9] 2016. How Low Can You Go? How to Reduce Power Consumption in Data-acquisition Systems. (2016). [https://e2e.ti.com/blogs\\_/archives/b/precisionhub/archive/2016/05/27/is-it-possible-to-reduce-power-consumption-by-adding-components-to-a-board](https://e2e.ti.com/blogs_/archives/b/precisionhub/archive/2016/05/27/is-it-possible-to-reduce-power-consumption-by-adding-components-to-a-board)
- [10] 2019. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper. <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html>
- [11] Ehsan Aryafar, Mohammad Amir Khojastepour, Karthikeyan Sundaresan, Sampath Rangarajan, and Mung Chiang. 2012. MIDU: Enabling MIMO Full Duplex. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (Mobicom '12)*. ACM, New York, NY, USA, 257–268. DOI: <https://doi.org/10.1145/2348543.2348576>
- [12] Anirudh Badam, Michael Kaminsky, Dongsu Han, Konstantina Papagiannaki, David G. Andersen, and Srinivasan Seshan. 2011. The Hare and the Tortoise: Taming Wireless Losses by Exploiting Wired Reliability. In *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '11)*. ACM, New York, NY, USA, Article 7, 11 pages. DOI: <https://doi.org/10.1145/2107502.2107511>
- [13] Dinesh Bharadia, Emily McMilin, and Sachin Katti. 2013. Full Duplex Radios. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM (SIGCOMM '13)*. ACM, New York, NY, USA, 375–386. DOI: <https://doi.org/10.1145/2486001.2486033>
- [14] Kameswari Chebrolu and Ashutosh Dhekne. 2009. Esense: Communication Through Energy Sensing. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking (MobiCom '09)*. ACM, New York, NY, USA, 85–96. DOI: <https://doi.org/10.1145/1614320.1614330>
- [15] Z. Chi, Z. Huang, Y. Yao, T. Xie, H. Sun, and T. Zhu. 2017. EMF: Embedding Multiple Flows of Information in Existing Traffic for Concurrent Communication among Heterogeneous IoT Devices. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*. 1–9.
- [16] Z. Chi, Y. Li, Z. Huang, H. Sun, and T. Zhu. 2019. Simultaneous Bi-directional Communications and Data Forwarding using a Single ZigBee Data Stream. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. 577–585. DOI: <https://doi.org/10.1109/INFOCOM.2019.8737555>
- [17] Zicheng Chi, Yan Li, Hongyu Sun, Yao Yao, Zheng Lu, and Ting Zhu. 2016. B2W2: N-Way Concurrent Communication for IoT Devices. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM (SenSys '16)*. ACM, New York, NY, USA, 245–258. DOI: <https://doi.org/10.1145/2994551.2994561>
- [18] Z. Chi, Y. Li, Y. Yao, and T. Zhu. 2017. PMC: Parallel Multi-protocol Communication to Heterogeneous IoT Radios within A Single WiFi Channel. In *2017 IEEE 25th International Conference on Network Protocols (ICNP)*. 1–10. DOI: <https://doi.org/10.1109/ICNP.2017.8117550>
- [19] J. Cooley, P. Lewis, and P. Welch. 1967. Historical Notes on the Fast Fourier Transform. *IEEE Transactions on Audio and Electroacoustics* 15, 2 (June 1967), 76–79. DOI: <https://doi.org/10.1109/TAU.1967.1161903>
- [20] Halperin Daniel, Ammer Josephine, Anderson Thomas, and David Wetherall. Interference Cancellation: Better Receivers for a New Wireless MAC. In *ACM HotNets, 2007*.
- [21] T. Dao, A. K. Roy-Chowdhury, H. V. Madhyastha, S. V. Krishnamurthy, and T. La Porta. 2017. Managing Redundant Content in Bandwidth Constrained Wireless Networks. *IEEE/ACM Transactions on Networking* 25, 2 (April 2017), 988–1003. DOI: <https://doi.org/10.1109/TNET.2016.2616901>
- [22] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh. 2011. Efficient Network Flooding and Time Synchronization with Glossy. In *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*. 73–84.
- [23] Shyamnath Gollakota, Fadel Adib, Dina Katabi, and Srinivasan Seshan. 2011. Clearing the RF Smog: Making 802.11N Robust to Cross-technology Interference. In *Proceedings of the ACM SIGCOMM 2011 Conference (SIGCOMM '11)*. ACM, New York, NY, USA, 170–181. DOI: <https://doi.org/10.1145/2018436.2018456>
- [24] Shyamnath Gollakota, Fadel Adib, Dina Katabi, and Srinivasan Seshan. 2015. Cross Technology Interference Cancellation. (March 17 2015). US Patent 8,983,011.
- [25] Shyamnath Gollakota and Dina Katabi. 2008. Zigzag Decoding: Combating Hidden Terminals in Wireless Networks. In *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication (SIGCOMM '08)*. ACM, New York, NY, USA, 159–170. DOI: <https://doi.org/10.1145/1402958.1402977>
- [26] Shyamnath Gollakota, Samuel David Perli, and Dina Katabi. 2009. Interference Alignment and Cancellation. In *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication (SIGCOMM '09)*. ACM, New York, NY, USA, 159–170. DOI: <https://doi.org/10.1145/1592568.1592588>
- [27] N. Golmie, N. Chevrollier, and O. Rebal. 2003. Bluetooth and WLAN Coexistence: Challenges and Solutions. *IEEE Wireless Communications* 10, 6 (Dec 2003), 22–29. DOI: <https://doi.org/10.1109/MWC.2003.1265849>
- [28] M. Gowda, N. Roy, R. R. Choudhury, and S. Nelakuditi. 2016. Compressing Backoff in CSMA Networks. In *2016 IEEE 24th International Conference on Network Protocols (ICNP)*. 1–10. DOI: <https://doi.org/10.1109/ICNP.2016.7784437>
- [29] Ramakrishna Gummadi, David Wetherall, Ben Greenstein, and Srinivasan Seshan. 2007. Understanding and Mitigating the Impact of RF Interference on 802.11 Networks. In *Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '07)*. ACM, New York, NY, USA, 385–396. DOI: <https://doi.org/10.1145/1282380.1282424>
- [30] D. Guo and C. Wang. 2008. Multiuser Detection of Sparsely Spread CDMA. *IEEE Journal on Selected Areas in Communications* 26, 3 (April 2008), 421–431. DOI: <https://doi.org/10.1109/JSAC.2008.080402>
- [31] Yong He, Ji Fang, Jiansong Zhang, Haichen Shen, Kun Tan, and Yongguang Zhang. 2010. MPAP: Virtualization Architecture for Heterogeneous Wireless APs. In *Proceedings of the ACM SIGCOMM 2010 Conference (SIGCOMM '10)*. ACM, New York, NY, USA, 475–476. DOI: <https://doi.org/10.1145/1851182.1851271>
- [32] Christopher Husmann, Georgios Georgis, Konstantinos Nikitopoulos, and Kyle Jamieson. 2017. FlexCore: Massively Parallel and Flexible Processing for Large MIMO Access Points. In *14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17)*. USENIX Association, Boston, MA, 197–211. <https://www.usenix.org/conference/nsdi17/technical-sessions/presentation/husmann>
- [33] A. B. M. Alim Al Islam and Vijay Raghunathan. 2015. SiAc: Simultaneous Activation of Heterogeneous Radios in High Data Rate Multi-Hop Wireless Networks. *Wireless Networks* 21 (10 2015). DOI: <https://doi.org/10.1007/s11276-015-0923-2>
- [34] Sachin Katti, Shyamnath Gollakota, and Dina Katabi. 2007. Embracing Wireless Interference: Analog Network Coding. In *Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '07)*. ACM, New York, NY, USA, 397–408. DOI: <https://doi.org/10.1145/1282380.1282425>
- [35] M. O. Khan, L. Qiu, A. Bhartia, and K. C. Lin. 2015. Smart Retransmission and Rate Adaptation in WiFi. In *2015 IEEE 23rd International Conference on Network Protocols (ICNP)*. 54–65. DOI: <https://doi.org/10.1109/ICNP.2015.48>
- [36] Mohammad A. Khojastepour, Karthik Sundaresan, Sampath Rangarajan, Xinyu Zhang, and Sanaz Barghi. 2011. The Case for Antenna Cancellation for Scalable Full-duplex Wireless Communications. In *Proceedings of the 10th ACM Workshop on Hot Topics in Networks (HotNets-X)*. ACM, New York, NY, USA, Article 17, 6 pages. DOI: <https://doi.org/10.1145/2070562.2070579>
- [37] Song Min Kim and Tian He. 2015. FreeBee: Cross-technology Communication via Free Side-channel. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*. ACM, New York, NY, USA, 317–330. DOI: <https://doi.org/10.1145/2789168.2790098>
- [38] Linghe Kong and Xue Liu. 2015. mZig: Enabling Multi-Packet Reception in ZigBee. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*. ACM, New York, NY, USA, 552–565. DOI: <https://doi.org/10.1145/2789168.2790104>
- [39] Swarn Kumar, Diego Cifuentes, Shyamnath Gollakota, and Dina Katabi. 2013. Bringing Cross-layer MIMO to Today's Wireless LANs. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM (SIGCOMM '13)*. ACM, New York, NY, USA, 387–398. DOI: <https://doi.org/10.1145/2486001.2486034>
- [40] Y. Li, B. Bakalloglu, and C. Chakrabarti. 2005. A Comprehensive Energy Model and Energy-quality Evaluation of Wireless Transceiver Front-ends. In *IEEE Workshop on Signal Processing Systems Design and Implementation, 2005*. 262–267. DOI: <https://doi.org/10.1109/SIPS.2005.1579876>
- [41] Yan Li, Zicheng Chi, Xin Liu, and Ting Zhu. 2018. Chiron: Concurrent High Throughput Communication for IoT Devices. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '18)*. ACM, New York, NY, USA, 204–216. DOI: <https://doi.org/10.1145/3210240.3210346>
- [42] Zhijun Li and Tian He. 2017. WEBee: Physical-Layer Cross-Technology Communication via Emulation. In *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking (MobiCom '17)*. ACM, New York, NY, USA, 2–14. DOI: <https://doi.org/10.1145/3117811.3117816>
- [43] Changbin Liu, Xiaozhou Li, Shiv Muthukumar, Harjot Gill, Taher Saeed, Boon Thau Loo, and Prithwish Basu. 2010. A Policy-based Constraint-solving Platform Towards Extensible Wireless Channel Selection and Routing. In *Proceedings of the Workshop on Programmable Routers for Extensible Services of Tomorrow (PRESTO '10)*. ACM, New York, NY, USA, Article 9, 6 pages. DOI: <https://doi.org/10.1145/1921151.1921164>

- [44] Xi Liu, Anmol Sheth, Michael Kaminsky, Konstantina Papagiannaki, Srinivasan Seshan, and Peter Steenkiste. 2009. DIRC: Increasing Indoor Wireless Capacity Using Directional Antennas. In *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication (SIGCOMM '09)*. ACM, New York, NY, USA, 171–182. DOI : <https://doi.org/10.1145/1592568.1592589>
- [45] S. M. Mishra, R. W. Brodersen, S. t. Brink, and R. Mahadevappa. 2007. Detect and avoid: An Ultra-wideband/WiMAX Coexistence Mechanism [Topics in Radio Communications]. *IEEE Communications Magazine* 45, 6 (June 2007), 68–75. DOI : <https://doi.org/10.1109/MCOM.2007.374435>
- [46] Thyagarajan Nandagopal, Tae-Eun Kim, Xia Gao, and Vaduvur Bharghavan. 2000. Achieving MAC Layer Fairness in Wireless Packet Networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom '00)*. ACM, New York, NY, USA, 87–98. DOI : <https://doi.org/10.1145/345910.345925>
- [47] J. Pang, B. Greenstein, M. Kaminsky, D. McCoy, and S. Seshan. 2010. Wifi-Reports: Improving Wireless Network Selection with Collaboration. *IEEE Transactions on Mobile Computing* 9, 12 (Dec 2010), 1713–1731. DOI : <https://doi.org/10.1109/TMC.2010.151>
- [48] N. Prasad, M. Arslan, and S. Rangarajan. 2014. Enhanced Interference Management in Heterogeneous Cellular Networks. In *2014 IEEE International Symposium on Information Theory*. 1603–1607. DOI : <https://doi.org/10.1109/ISIT.2014.6875104>
- [49] Hariharan Rahul, Nate Kushman, Dina Katabi, Charles Sodini, and Farinaz Edalat. 2008. Learning to Share: Narrowband-friendly Wideband Networks. In *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication (SIGCOMM '08)*. ACM, New York, NY, USA, 147–158. DOI : <https://doi.org/10.1145/1402958.1402976>
- [50] I. Ramani, R. R. Kompella, S. Ramabhadran, and A. C. Snoeren. 2010. Covenant: An Architecture for Cooperative Scheduling in 802.11 Wireless Networks. *IEEE Transactions on Wireless Communications* 9, 1 (January 2010), 175–185. DOI : <https://doi.org/10.1109/TWC.2010.01.081272>
- [51] Souvik Sen, Romit Roy Choudhury, and Srihari Nelakuditi. 2012. CSMA/CN: Carrier Sense Multiple Access with Collision Notification. *IEEE/ACM Trans. Netw.* 20, 2 (April 2012), 544–556. DOI : <https://doi.org/10.1109/TNET.2011.2174461>
- [52] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi. 2013. Successive Interference Cancellation: Carving Out MAC Layer Opportunities. *IEEE Transactions on Mobile Computing* 12, 2 (Feb 2013), 346–357. DOI : <https://doi.org/10.1109/TMC.2012.17>
- [53] Clayton Shepard, Abeer Javed, and Lin Zhong. 2015. Control Channel Design for Many-Antenna MU-MIMO. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*. ACM, New York, NY, USA, 578–591. DOI : <https://doi.org/10.1145/2789168.2790120>
- [54] J. Shi, E. Aryafar, T. Salonidis, and E. W. Knightly. 2009. Synchronized CSMA Contention: Model, Implementation and Evaluation. In *IEEE INFOCOM 2009*. 2052–2060. DOI : <https://doi.org/10.1109/INFCOM.2009.5062128>
- [55] N. Singh, D. Gunawardena, A. Proutiere, B. Radunovi, H. V. Balan, and P. Key. 2011. Efficient and Fair MAC for Wireless Networks with Self-interference Cancellation. In *2011 International Symposium of Modeling and Optimization of Mobile, Ad Hoc, and Wireless Networks*. 94–101. DOI : <https://doi.org/10.1109/WIOPT.2011.5930070>
- [56] Sanjib Sur, Ioannis Pefkianakis, Xinyu Zhang, and Kyu-Han Kim. 2016. Practical MU-MIMO User Selection on 802.11Ac Commodity Networks. In *Proceedings of the 22Nd Annual International Conference on Mobile Computing and Networking (MobiCom '16)*. ACM, New York, NY, USA, 122–134. DOI : <https://doi.org/10.1145/2973750.2973758>
- [57] Sergio Verdu. 1998. *Multuser Detection* (1st ed.). Cambridge University Press, New York, NY, USA.
- [58] W. Wang, X. Liu, Y. Yao, Y. Pan, Z. Chi, and T. Zhu. 2019. CRF: Coexistent Routing and Flooding using WiFi Packets in Heterogeneous IoT Networks. In *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. 19–27. DOI : <https://doi.org/10.1109/INFCOM.2019.8737525>
- [59] Jie Xiong, Karthikeyan Sundaresan, Kyle Jamieson, Mohammad A. Khojastepour, and Sampath Rangarajan. 2014. MIDAS: Empowering 802.11Ac Networks with Multiple-Input Distributed Antenna Systems. In *the 10th ACM International on Conference on Emerging Networking Experiments and Technologies (CoNEXT '14)*. ACM, New York, NY, USA, 29–40. DOI : <https://doi.org/10.1145/2674005.2675014>
- [60] Sangki Yun, Daehyeok Kim, and Lili Qiu. 2013. Fine-grained Spectrum Adaptation in WiFi Networks. In *Proceedings of the 19th Annual International Conference on Mobile Computing and Networking (MobiCom '13)*. ACM, New York, NY, USA, 327–338. DOI : <https://doi.org/10.1145/2500423.2500442>
- [61] S. Yun and L. Qiu. 2015. Supporting WiFi and LTE Co-existence. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. 810–818. DOI : <https://doi.org/10.1109/INFCOM.2015.7218451>
- [62] Yunze Zeng, Ioannis Pefkianakis, Kyu-Han Kim, and Prasant Mohapatra. 2017. MU-MIMO-Aware AP Selection for 802.11Ac Networks. In *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '17)*. ACM, New York, NY, USA, Article 19, 10 pages. DOI : <https://doi.org/10.1145/3084041.3084057>
- [63] X. Zhang and K. G. Shin. 2013. Gap Sense: Lightweight Coordination of Heterogeneous Wireless Devices. In *2013 Proceedings IEEE INFOCOM*. 3094–3101. DOI : <https://doi.org/10.1109/INFCOM.2013.6567122>
- [64] Y. Zhang and Q. Li. 2013. HoWiES: A Holistic Approach to ZigBee Assisted WiFi Energy Savings in Mobile Devices. In *2013 Proceedings IEEE INFOCOM*. 1366–1374. DOI : <https://doi.org/10.1109/INFCOM.2013.6566930>