ZigFi: Harnessing Channel State Information for Cross-Technology Communication

Xiuzhen Guo[1], Yuan He[1], Xiaolong Zheng[1], Liangcheng Yu[2], Omprakash Gnawali[3]
[1]School of Software, Tsinghua University & TNLIST, P.R. China
[2]KTH Royal Institute of Technology, Sweden
[3]University of Houston, USA
guoxz16@mails.tsinghua.edu.cn, {heyuan, zhengxiaolong}@mail.tsinghua.edu.cn,
liangcheng.yu46@gmail.com, gnawali@cs.uh.edu

***Abstract*—Cross-technology communication (CTC) is a technique that enables direct communication among different wireless technologies. Recent works in this area have made positive progress, but high-throughput CTC from ZigBee to WiFi remains an open problem. In this paper, we propose ZigFi, a novel CTC framework that enables direct communication from ZigBee to WiFi. Without impacting the ongoing WiFi transmissions, ZigFi carefully overlaps ZigBee packets with WiFi packets. Through experiments we show that Channel State Information (CSI) of the overlapped packets can be used to convey data from ZigBee to WiFi. Based on this finding, we propose a receiver-initiated protocol and translate the decoding problem into a problem of CSI classification with Support Vector Machine. We further build a generic model through experiments, which describes the relationship between the Signal to Interference and Noise Ratio (SINR) and the symbol error rate (SER). We implement ZigFi on commercial-off-the-shelf WiFi and ZigBee devices. We evaluate the performance of ZigFi under different experimental settings. The results demonstrate that ZigFi achieves a throughput of 215.9bps, which is 18X faster than the state-of-the-art.**

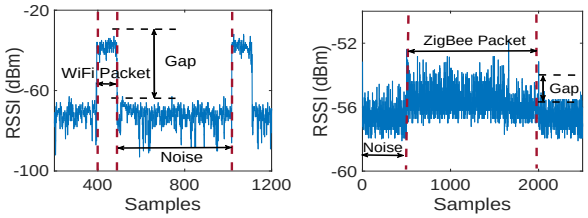
I. Introduction

Large-scale deployments of Internet of Things (IoT) have led to not only crowding of wireless spectrum but also heterogeneity in wireless technologies in devices and networks that are expected to work together. Devices that use different wireless technologies (e.g. WiFi, ZigBee, and BlueTooth) have to share the unlicensed spectrum (e.g. ISM bands) when they coexist in the common space. Traditional approaches to manage this crowding and heterogeneity try to avoid, mitigate, or tolerate the wireless interference, and use multi-radio gateway, whereas cross-technology communication (CTC) opens a new direction of direct communication among different wireless technologies [1] [2]. The ability to communicate across different technologies avoids the unnecessary hardware cost and communication delay, compared to the indirect solution based on a multi-radio gateway [3]. With CTC, it becomes easier to coordinate heterogeneous wireless devices even in a shared channel [4] [5]. CTC is also an enabling technology for emerging IoT applications (e.g. industrial surveillance and smart home), where seamless data collection and interoperation are desired [6] [7] [8].

In recent years, there has been some progress in CTC research. FreeBee [2] enables direct communication among different technologies by embedding symbols into beacons and shifting the beacon transmission timings. Esense [9] applies energy sampling to realize data transmission from a WiFi device to a ZigBee device. WiZig [10] employs energy modulation techniques in both the amplitude dimension and the temporal dimension to optimize the throughput from WiFi to ZigBee over a noisy channel. B^2W^2 [11] realizes data transmission from a Bluetooth Low Energy (BLE) device to a WiFi device, by leveraging the features of the overlapping channels.

Despite this progress, there is relatively little progress in CTC from ZigBee to WiFi. This problem is extremely challenging due to several asymmetries between the two technologies. First, there is a large difference in transmission power of ZigBee vs. WiFi. By default, the maximum transmission power of a WiFi device is 100 dBm, while the maximum transmission power of a ZigBee device is 0 dBm. Second, the bandwidths of ZigBee and WiFi channels have a large difference. The channel bandwidth of WiFi is 20 MHz, which is 10x of the channel bandwidth of ZigBee (2 MHz). The asymmetry in channel bandwidth also leads to apparent disharmony with regard to the encoding and decoding rates. As a result, from the view of a WiFi receiver, the ZigBee signals appear to be weak and susceptible to the noise. Simply increasing the transmission power of ZigBee will induce too much interference, not to mention the prohibitively high power consumption. As far as we know, FreeBee [2] and TCTC [12] are the only two existing proposals of CTC from ZigBee to WiFi. Their throughput, however, is limited by the inefficient encoding rate in the temporal dimension.

In this paper, we propose ZigFi, a receiver-initiated protocol for CTC from ZigBee to WiFi. The basic idea is to carefully piggy-back ZigBee packets over WiFi packets, without destroying or colliding with the ongoing WiFi transmissions. By tracking the PHY-layer features of the received packets, a WiFi receiver is able to decode not only the WiFi packets sent by the WiFi sender, but also the data sent by the ZigBee sender. By using a machine learning approach for decoding, ZigFi can efficiently convey data from a ZigBee device to a WiFi device, even in noisy environments. The main contributions of this work are summarized as follows.



(a) WiFi RSSI at ZigBee node (b) ZigBee RSSI at WiFi node

Fig. 1. The RSSI sequence with WiFi and ZigBee transmitters in the network.

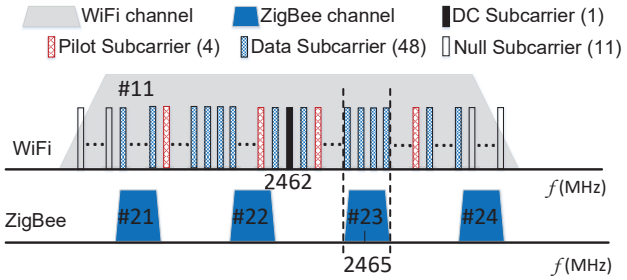


Fig. 2. The distribution of a WiFi channel and ZigBee channels

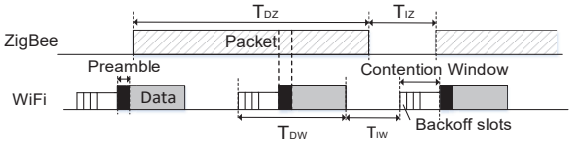


Fig. 3. Illustration of overlapping ZigBee and WiFi packets

- We study how ZigBee and WiFi packets transmissions interact with each other from both transmitter and receiver perspective. We find that it is feasible to use Channel State Information (CSI) of the overlapped packets to convey data from ZigBee to WiFi. Based on this finding, we propose ZigFi, a framework that translates the decoding problem into a problem of CSI classification with Support Vector Machine (SVM).
- We design a receiver-initiated protocol for practical application of ZigFi. Using this protocol, a WiFi receiver can coordinate the communication settings(e.g. packet length and transmission power) with both the ZigBee sender and the WiFi sender. In this way, ZigFi achieves efficient and robust CTC even in noisy environments, minimizing the impact to ongoing WiFi transmissions.
- We implement and evaluate ZigFi on commercial WiFi devices and ZigBee motes. The results demonstrate that ZigFi achieves a throughput of 215.9bps, which is 18X faster than the state-of-the-art.

The rest of this paper is organized as follows. Section II discusses the related work. In Section III, we verify the feasibility and challenges of ZigFi. Section IV presents the design of ZigFi. In Section IV-C, we evaluate the performance of ZigFi. We conclude this work in Section VI.

II. Related Work

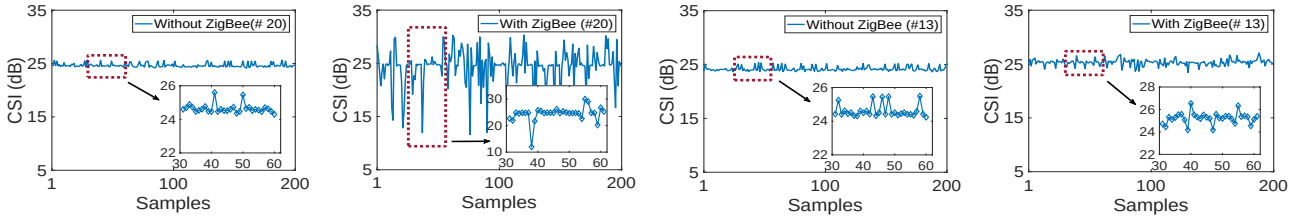
Co-existence of multiple wireless technologies has become a critical problem in IoT communication. For effective coordination and better spectrum utilization, early works on wireless coexistence mainly focus on managing collisions and wireless interference. Given that the new generation of IoT applications utilize devices with different wireless technologies, direct communication across wireless technologies can simplify many IoT deployments. Several existing studies have tried to address the challenges of CTC.

Collision avoidance and interference management. In the conventional studies, different wireless technologies deployed in range of each other are considered competitors and interferers of each other. Collision avoidance based approaches propose to separate competing devices in the temporal [13] or the frequency domain [14]. WISE [15] enhances ZigBee throughput by harnessing the white spaces between WiFi transmissions. ZIMO [1] proposes a MIMO design for harmony coexistence of ZigBee and WiFi networks with the goal of protecting ZigBee data packets.

Recent works show that obtaining cross-technology information can enhance the network performance. ZiFi [16] utilizes low power ZigBee radio to detect the existence of WiFi hotspots, so that the standby energy consumption of WiFi devices can be significantly reduced. ZiSense [17] identifies the received signal strength indicator (RSSI) signatures of different wireless technology, so as to protect duty-cycled ZigBee radios from false wake-up. Smoggy-Link [18] constructs fingerprints of the coexisting interferers, which are then utilized to exploit opportunities of concurrent transmissions under heterogeneous interference.

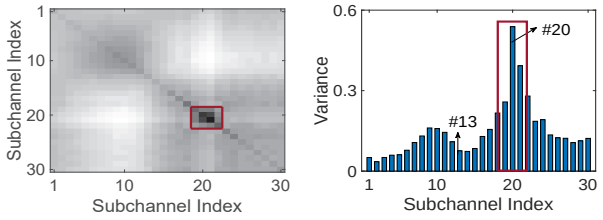
Cross-technology Communication. Most of the existing CTC works employ packet-level modulation. Data is encoded as modulated packets in either the temporal or the amplitude dimension. In the temporal dimension, FreeBee [2] embeds symbols into beacons by shifting their transmission timings. However, the throughput of FreeBee is bounded by the limited beacon frequency. TCTC [12] employs a similar technique with FreeBee while taking the application-layer data packets as targets to be shifted, which therefore has a similar limitation.

It is also possible to use energy as a side-channel for CTC. Esense [9] uses the power at which the packet is transmitted to encode data bits. HoWiEs [19] improves the Esense mechanism by modulating the packet length of WiFi. Gap Sense [20] leverages WiFi preamble to construct special energy pulses. The gap between the energy pulses is used to convey data. WiZig [10] employs modulation in both the amplitude and the temporal dimensions to optimize the throughput from WiFi to ZigBee. C-Morse [21] modulates the timing of packets that pass through WiFi APs, so as to construct recognizable radio energy patterns. B^2W^2 [11] builds a one-way communication from BLE to WiFi, by leveraging the feature of overlapped channels. WEBee [22] enables WiFi to ZigBee CTC by utilizing part of the payload in a WiFi packet to emulate



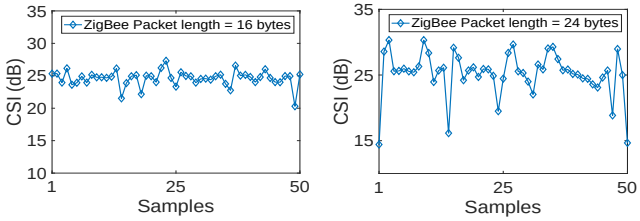
(a) Without ZigBee packets, subchannel=20 (b) With ZigBee packets, subchannel=20 (c) Without ZigBee packets, subchannel=13 (d) With ZigBee packets, subchannel=13

Fig. 4. The CSI sequences with/without ZigBee packets of different subchannels



(a) Covariance between subchannels (darker is higher correlation) (b) The histogram of subchannel variance

Fig. 5. The relationship among WiFi subchannels



(a) ZigBee length=16bytes (b) ZigBee length=24bytes

Fig. 6. The CSI sequences with different ZigBee packet lengths

a ZigBee packet at the physical-layer. WEBee significantly improves the CTC throughput from WiFi to ZigBee.

ZigFi differs from the existing works in the following aspects. First, ZigFi aims at CTC from ZigBee to WiFi, which is crucial, most challenging, but not well studied in the literature. We are aware of only two system, FreeBee and TCTC, that can be send data from ZigBee to WiFi. As we demonstrate in the evaluation, ZigFi enhances the CTC throughput by an order of magnitude, compared to FreeBee and TCTC. Second, instead of simple packet-level modulation, ZigFi proposes a machine learning approach to exploit fine-grained physical-layer features, which is more robust and efficient in noisy environments. Third, our work addresses practical challenges of CTC and makes ZigFi an integrated framework, which not only enables CTC from ZigBee to WiFi, but also includes design considerations in the reverse direction.

III. Observations

In this section, we study the feasibility of ZigBee to WiFi communication. We conduct several experiments to observe the impact of the ZigBee packets on RSSI and CSI amplitude sequence of ongoing WiFi packets.

A. Infeasibility of ZigBee to WiFi CTC using RSSI

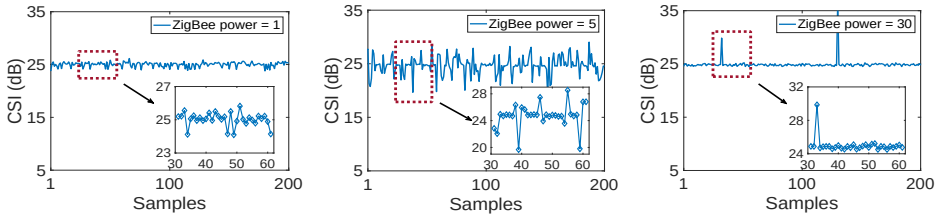
Although previous studies have shown the possibility of using RSSI to achieve WiFi to ZigBee communication, there is no study that examines the feasibility of using RSSI for ZigBee to WiFi communication. Here we fill that gap. We configure a WiFi sender to transmit 145-byte packets on channel 11 with a packet interval of 0.5ms and a TelosB node to transmit 28-byte ZigBee packets on channel 23 with a packet interval of 0.192 ms. Figure 1(a) shows the RSSI sequence of WiFi sampled by ZigBee. Figure 1(b) shows the RSSI sequence of ZigBee sampled by WiFi. Since the bandwidth of ZigBee channel is much narrower than the WiFi channel, it is difficult to detect ZigBee packets reliably using RSSI. In addition, the RSSI sequence is susceptible to noise and interference. As a result, it is impracticable to use the RSSI sequence for high-throughput ZigBee to WiFi CTC.

B. Feasibility of ZigBee to WiFi CTC using CSI

Next we explore the possibility of using CSI for ZigBee to WiFi CTC. First, the overlapping on the frequency domain provides a theoretical support for using the CSI amplitude sequence (CSI mentioned later refers to the amplitude of the complex value) for CTC. The distribution of WiFi and ZigBee channels is shown in Figure 2. A WiFi channel is divided into 64 different subcarriers and a ZigBee channel overlaps with several WiFi subcarriers. So the ZigBee signal mainly distributes in the overlapping subcarriers. Second, CSI can be used to describe the feature of each WiFi subcarrier. CSI is a simple version of channel frequency response and it is obtained by the predefined preamble [23] [24]. As shown in Figure 3, if there are ZigBee packets during the transmission of WiFi packets, the ZigBee transmission will interfere with the WiFi preamble and cause a change in the CSI amplitude.

C. CSI sequence on different subchannels

We conduct experiments to observe the CSI sequences on different subchannels with and without ZigBee transmissions. We configure a TelsoB node to transmit ZigBee packets at power level 13 (-9 dBm according to [25]). The CSI sequences with and without ZigBee transmissions are shown in Figure 4(a) and Figure 4(b). We find that the variation of CSI



(a) ZigBee power=1 (b) ZigBee power=5 (c) ZigBee power=30

Fig. 7. The CSI sequences with ZigBee packets of different transmission powers

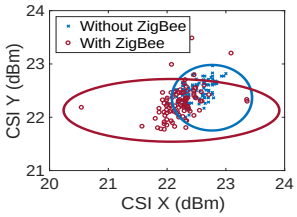


Fig. 8. The CSI pairs with/without ZigBee packets in two-dimension

sequences is distinct with and without ZigBee transmissions. Further, we compare the CSI sequences in different subchannels as shown in Figure 4(c) and Figure 4(d). We find that the CSI sequence of subchannel 20 has a larger variation range than the subchannel 13 when there are ZigBee packets.

To extract discriminative features of CSI sequence in the presence of ZigBee, we need to select the subchannel which is most affected by ZigBee. The covariance values of different suchannels are shown in Figure 5(a). We can find that the covariance within the subchannel 19-22 is distinct. We further explore the histogram of subchannel variance as shown in Figure 5(b). The subchannels 19-22 have the largest variance over all subchannels. In addition, the center frequency of subchannel 20 is nearly equal to the ZigBee. So the CSI sequence of subchannel 20 is mostly distinctive from other subchannels.

Hence, the transmission of ZigBee packets affects the CSI sequence of the WiFi receiver. In addition, the variation of the CSI sequence at each subcarrier is different.

D. CSI sequences with different ZigBee packet lengths

We redo the experiments on subchannel 20 with ZigBee packet lengths of 16 bytes and 24 bytes and show the resulting CSI sequences in Figure 6(a) and Figure 6(b). We find that if the packet length is short, the collision probability of ZigBee and WiFi is low and the variation of CSI sequence is small. The CSI sequence variations become more prominent with longer ZigBee packets. Due to the time asynchronization and asymmetry of data rates between the ZigBee and the WiFi, we need to transmit long enough ZigBee packets to guarantee that one ZigBee packet overlaps with at least one WiFi packet. Specifically, as shown in Figure 3, the length of the ZigBee packet must satisfy:

$$T_{DZ} \geq 2T_{DW} + T_{IW} \tag{1}$$

where T_{DZ} is the transmission time of the ZigBee packet. T_{DW} and T_{IW} are the transmission time of the WiFi packet and the transmission interval between two adjacent WiFi packets.

E. CSI sequences with different ZigBee transmit power

We redo the experiment on subchannel 20 with ZigBee transmission power at levels 1, 5, and 30 (corresponding power are -30 dBm, -20 dBm, and 0 dBm), packet length at 28 bytes, and plot the resulting CSI sequences in Figures 7(a), 7(b), and 7(c) respectively. We find that the CSI sequence with a transmit power of 1 is similar to the CSI sequence without any ZigBee transmission(Figure 4(a)). As ZigBee power increases, the CSI sequence is more distinct. When ZigBee power is too high, the CSI sequence includes only a few peaks as shown in Figure 7(c). With sufficiently high transmission power, the probability that the ZigBee packet collides with the WiFi preamble becomes low, because the WiFi sender can sense the interference from ZigBee transmission and backs off, resulting in sparse peaks and stable CSI in most cases and degraded WiFi performance.

Hence, the transmit power of ZigBee packets affects the degree of the CSI variation. If the ZigBee transmit power is low, the CSI variation will be too little to be detected at the WiFi receiver. Whereas, if the ZigBee transmit power is too high, the WiFi preamble will be subjected to strong interference impacting regular WiFi traffic. As a result, we need to choose an appropriate ZigBee power to make the CSI sequence more distinctive without impacting WiFi traffic.

F. On classifying CSI sequences

From the observations above, we conclude that the CSI sequence at the WiFi receiver varies if there are overlapping ZigBee packets. It is difficult to quantify the CSI variation because the channel is dynamic and noisy. As shown in the sub-figure of Figure 4(b) or other figures, there are no simple rules to describe the variation of CSI values.

If a ZigBee packet length satisfies Eq. (1), a ZigBee packet overlaps with at least one WiFi packet. We define a *CSI pair* as a pair of CSI values. The first value is interfered by ZigBee. The second value is obtained right after the first but is not necessarily interfered. We plot the CSI pairs obtained with and without ZigBee transmissions in two-dimension in Figure 8. No matter whether the experiment is in a controlled environment, CSI sequences with or without ZigBee transmission are obviously different. The two sets of CSI pair overlap with each other, making it difficult to identify the two cases (with and without ZigBee) with straightforward techniques (e.g. thresholding). As a result, we explore techniques that help us classify these two different clusters at higher dimensional space and achieve CTC transmission.

Summary: In order to use the CSI sequence to enable ZigBee to WiFi CTC, some conditions need to be satisfied.

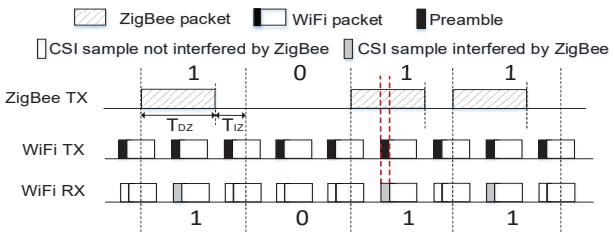


Fig. 9. The basic communication scheme of ZigFi

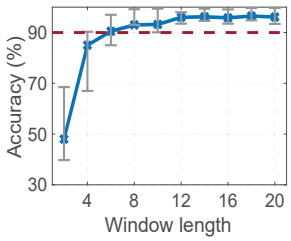


Fig. 10. The relationship between the accuracy and the window length

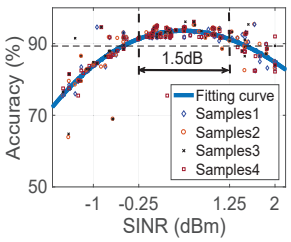


Fig. 11. The relationship between the accuracy and the SINR

- We should select an appropriate subchannel to make ZigBee and WiFi overlap in the frequency domain.
- The ZigBee packet length must be large enough make ZigBee packets overlap with WiFi packets in the time domain.
- We need to choose an appropriate ZigBee power to make the CSI sequence more distinctive.

After that, a classifier can be applied on appropriately mapped data to identify the CSI sequence.

IV. ZigFi Design

A. Overview

In this Section, we present the design of ZigFi, a novel CTC technique that enables direct communication from ZigBee to WiFi. At a high level, ZigFi leverages the variation of CSI sequence to encode and decode CTC information.

The ZigFi design utilizes existing packet transmissions by a WiFi node to encode information from ZigBee node. The ZigBee transmitter encodes the CTC symbols using presence or absence of ZigBee packets. Without modifying the WiFi or ZigBee physical layer, ZigBee packets are transmitted and piggy-backed to the existing the WiFi link. The WiFi receiver [1] then can receive two sets of information. It decodes packets transmitted by the WiFi sender as a regular WiFi packet. It also collects the CSI sequence and uses the SVM classifier to decode the CTC data. Thus, ZigFi achieves the CTC transmission from ZigBee to WiFi.

Figure 9 gives an overview of how encoding and decoding work with ZigFi. We now describe the steps in detail: (1) In our example, the ZigBee sender wants to send the bit sequence "1011" to the WiFi receiver. The ZigBee sender encodes the symbol "1" as the presence of the ZigBee packet and encodes the symbol "0" as the absence of the ZigBee packet. Then the Zigbee sender transmits these packets or remains silent. (2) There is an existing WiFi sender transmitting packets. ZigBee packets overlap with WiFi packets in the air. (3) The WiFi receiver uses the length of a ZigBee packet as the decoding window and collects the CSI sequence during this window. Then the receiver uses SVM to identify whether there is an interfered CSI value within a decoding window. If there is no interfered CSI value within a decoding window, the receiver decodes this window as a "0". Otherwise, the window is decoded as a "1".

[1]In this paper, the WiFi receiver refers to the WiFi node which has added functionality of decoding CTC symbols using CSI.

B. Decoding with a CSI classifier

In ZigFi, SVM is used to classify the received CSI sequence as "0" or "1". In this section, we discuss various aspects of ZigFi design that impact the accuracy of decoding.

1) The window length of the SVM classifier: The accuracy of the SVM classifier depends on the number of samples within a decoding window. If the decoding window is equal to the length of one ZigBee packet and the length of ZigBee packet satisfies the minimum requirement as shown in Eq. (1), one ZigBee packet collides with at most two WiFi packets. In other words, there are only two samples in a decoding window. So in this case, the performance of SVM is poor due to the limited CSI samples.

To improve the accuracy of SVM, we should increase the window length to allow more CSI samples within a decoding window. We conduct several experiments to find the appropriate window length. The experiment result is shown in Figure 10. We find that the accuracy of the SVM increases rapidly as we increase the window length initially. After a certain point, the increase in accuracy becomes marginal. We suggest a window size of 8, which achieves an accuracy above 0.9, as a good compromise between the classification accuracy and the data rate.

2) Training the SVM classifier: **DataSet** We conduct 120 experiments in four environments. We choose two different places, one is a 9m $\times$ 6m crowded office and another one is a 13m $\times$ 7m empty meeting room. We select the CSI sequences in these two places during the daytime and the night, respectively. In the daytime, there are twenty students in the office. These students can use WiFi to watch videos, surf the Internet, download files, and so on. There are five students in the conference room and they are talking with each other and only surf news site. Compared with the daytime environment, the office and the meeting room are relatively quiet at night. As a result, these four environments are different in terms of background noise, multipath fading, human mobility, and interference. In each environment, we collect CSI sequences and label them as -1 or $+1$. A CSI sequence with only the WiFi transmission (not overlapped with ZigBee) is labeled -1. A CSI sequence with the overlapped transmissions from WiFi and ZigBee is labeled $+1$. Each experiment lasts for 30 second

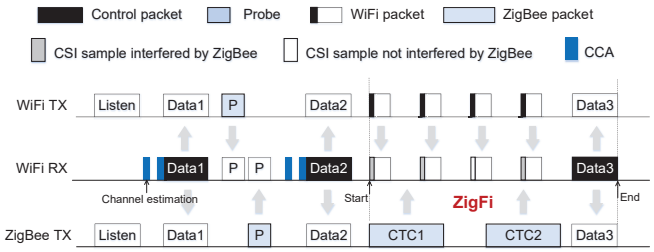


Fig. 12. The process of Receiver-initiated mechanism

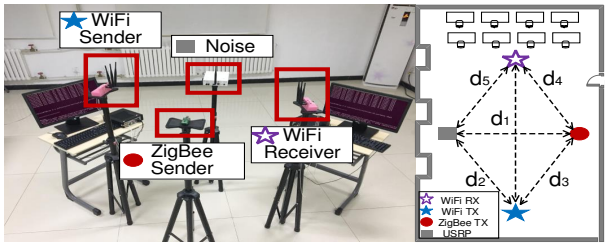
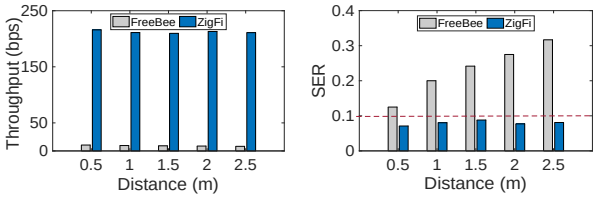


Fig. 13. The devices and the network used in the experimental evaluation



(a) Throughput (b) SER

Fig. 14. Performance comparison of ZigFi and FreeBee

and the CSI sampling rate is 2KHz. The total CSI sequences are randomly divided into training and test datasets.

Feature extraction We design light-weight features to allow fast decoding with SVM. We extract two features within a window: (1)the variance of CSI values and (2)peak-to-peak value in the time domain. These two features reveal the difference of CSI variation patterns when receiving packet-level information from a ZigBee sender.

3) The relationship between the Accuracy and the SINR: In Section III, we show that the ZigBee transmission power affects the degree of the CSI variation. If the ZigBee transmission power is low, the CSI variation will be too little to be detected at the WiFi receiver. Whereas, if the Zigbee power is too high, the WiFi to WiFi link will be affected. Hence, we define a new metric SINR in ZigFi, which can be calculated by

$$SINR = 10lg\frac{S_Z}{I_W + N} \tag{2}$$

where S_Z is the power of the received ZigBee packet, I_W is the power of the received WiFi packet, and N is the power of noise perceived by the WiFi receiver.

The SINR has a direct impact on the accuracy of decoding. In practice, it is difficult to quantify the CSI variation due to the channel dynamics. Thus we obtain an experimental model to describe the relationship between the SINR and the accuracy. We use the trained SVM to test the CSI sequences in the test dataset. Each CSI sequence corresponds to a certain SINR. Figure 11 shows the test results and the polynomial fitting curve:

$$f(x) = p_1 * x^3 + p_2 * x^2 + p_3 * x + p_4 \tag{3}$$

where $f(x)$ is the accuracy, x is the SINR, and the coefficients are $p_1 = 0.2159, p_2 = -4.44, p_3 = 4.094, p_4 = 91.13$.

As shown in Eq. (3), with an increase in SINR, the CSI sequence becomes more distinct and the accuracy accordingly increases. But the accuracy decreases when the SINR exceeds a certain value, because too strong ZigBee transmission will make the WiFi sender back-off. Figure 11 indicates that when the SINR is in the range $[-0.25, 1.25]$, the decoding accuracy is higher than 0.9.

Next, we analyze the energy cost on a ZigBee sender to participate in CTC using ZigFi. We assume that the energy cost of one ZigBee packet is E_T. there is a nonlinear relationship between E_T and the transmission power P_Z, describe by $E_T = G(P_Z)$. Meanwhile, S_Z=$P_Z \times \eta$, where η is the path loss factor and can be estimated online. We use E to denote the expected energy cost on the ZigBee sender to send one ZigFi packet successfully. We have

$$E = \frac{E_T}{1 - f(x)} \tag{4}$$

To minimize energy cost while achieving satisfactory decoding accuracy, we have:

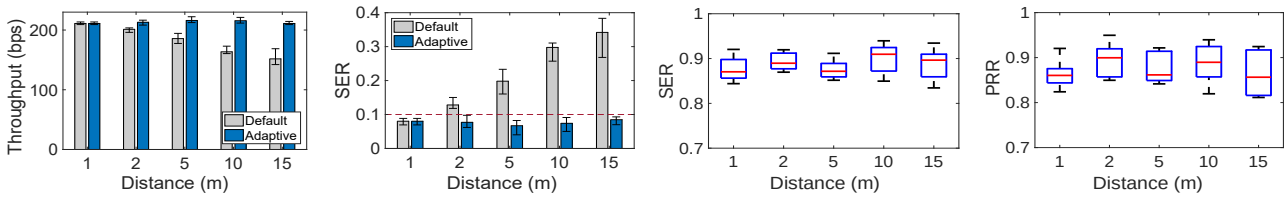
$$\min\ E$$
$$s.t. \begin{cases} E_t = G(P_Z) \\ th_l \le f(x) \le th_u, th_l = -0.25, th_u = 1.25 \\ f(x) = p_1 * x^3 + p_2 * x^2 + p_3 * x + p_4 \\ x = 10lg\frac{S_Z}{I_W + N} \\ S_Z = P_Z * \eta \end{cases} \tag{5}$$

The path loss can be measured by using the receiver-initiated mechanism, as described later. Solving the above equations yields the transmission power to be set at the ZigBee sender.

C. The Receiver-initiated mechanism

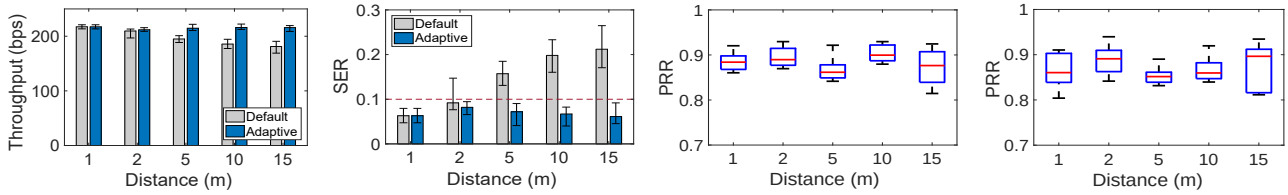
We design a receiver-initiated CTC mechanism to meet the following goals. First, the transmission of ZigFi relies on existing WiFi packet transmissions, so it is necessary to initiate or utilize packet transmissions between the WiFi sender and the WiFi receiver. Second, the transmission power of the ZigBee sender and the WiFi sender should be adjusted to achieve desired energy efficiency and decoding accuracy, as discussed in the previous subsection.

To send control information from the WiFi receiver to the ZigBee sender, multiple existing CTC techniques [10] [2] [9] can be used. In this work, we select the proposal in [10] to transmit control packets from WiFi to ZigBee. Specifically,



(a) ZigFi Throughput (b) ZigFi SER (c) WiFi PRR without ZigBee (d) WiFi PRR with ZigBee

Fig. 15. ZigFi under different distance between the ZigBee sender and the WiFi rceiver



(a) ZigFi Throughput (b) ZigFi SER (c) WiFi PRR without ZigBee (d) WiFi PRR with ZigBee

Fig. 16. ZigFi performance under different distance between the WiFi sender and the WiFi receiver

the payload of a WiFi packet includes data from the WiFi receiver to the WiFi sender, while the packet-level modulation carries the data from the WiFi receiver to the ZigBee sender. The receiver-initiated mechanism is shown in Figure 12. The specific process is as follows:

(1) The WiFi sender and the ZigBee sender listen to the channel. The following six parameters are determined by the WiFi receiver, before it sends the control packets: (a) the WiFi transmission power P_W, (b) the WiFi packet length T_{DW}, (c) the WiFi packet interval T_{IW}, (d) the ZigBee transmission power P_Z, (e) the ZigBee packet length T_{DZ}, and (f) the ZigBee packet interval T_{IZ}. These parameters have default values on the WiFi receiver. The default values are the parameters in the training state.

(2) The WiFi receiver conducts the channel estimation and detects whether there is an incoming WiFi packet. If there is no incoming WiFi packet, the WiFi receiver needs to trigger the packet transmissions from both the WiFi sender and ZigBee sender, using the default setting. Otherwise, the ZigFi transmission will piggy-back on an existing WiFi packet transmissions. The current setting on the WiFi sender is not updated. Only the parameters on the ZigBee sender need to be configured. The control packets are manipulated as *Data1* and sent simultaneously to the WiFi sender and the ZigBee sender, denoted by *Data1*.

(3) On receiving the control packets, the WiFi sender and the ZigBee sender respectively send a probe to the WiFi receiver, using the transmission powers specified in the control packets. For the WiFi sender, any normal packet is regarded as a probe. For the ZigBee sender, the probe is a preamble sequence.

(4) On receiving the probes, the WiFi receiver updates the parameters related to the channel condition (e.g. the path loss) and return them to the the WiFi sender and the ZigBee sender, denoted by *Data2*.

(5) On receiving *Data2*, the ZigFi transmission starts. Meanwhile, the WiFi receiver can send a control packet *Data3* to end the ZigFi transmission, when needed.

V. Evaluation

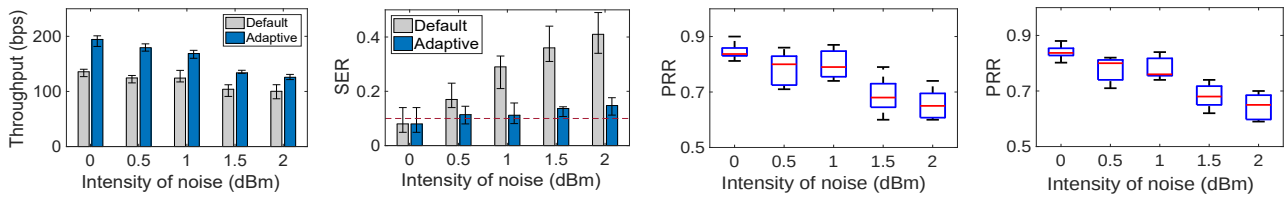
A. Implementation

We implement ZigFi on commercial off-the-shelf WiFi devices (Intel 5300) and TelosB motes, as shown in Figure 13. The CSITool software platform is installed on the WiFi devices and used to collect CSI readings. We configure the WiFi sender to transmit on channel 11 and ZigBee to transmit on channel 23, so that they overlap in the frequency domain. The sampling rate of CSITool is set at 2KHz. The WiFi packet interval is 0.5ms with the packet length of 145 bytes. The ZigBee packet interval is 0.192 ms with the packet length of 28 bytes. The experiment is carried out in a real office environment. Other than the naturally existing noise in the environment, we use a USRP to generate noise on demand. The performance metrics we use for evaluation mainly include throughput (measured by the successfully decoded bits per second) and SER.

B. Overall Performance Comparison

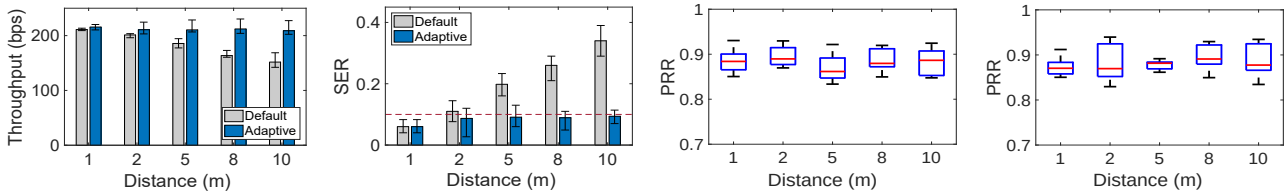
We compare ZigFi with FreeBee [2], a state-of-the art ZigBee to WiFi CTC work. FreeBee embeds the CTC symbols by shifting beacon transmission timings and decodes the message by detecting the RSSI variation. To make the RSSI distinctive, we let the TelosB mote transmit with the highest power (0 dBm). The parameters of FreeBee include the beacon interval and the folding time (for redundancy). We set the beacon interval at 100ms and the folding time at 5. The measured performance of FreeBee is close to the results reported in [2].

We change the distance between the ZigBee sender and the WiFi receiver. Figure 14(a) and Figure 14(b) plot the comparison results. ZigFi shows significant enhancement over FreeBee in terms of throughput and SER. For example, when



(a) ZigFi Throughput (b) ZigFi SER (c) WiFi PRR without ZigBee (d) WiFi PRR with ZigBee

Fig. 17. ZigFi performance under different intensity of noise



(a) ZigFi Throughput (b) ZigFi SER (c) WiFi PRR without ZigBee (d) WiFi PRR with ZigBee

Fig. 18. ZigFi performance under different distance between the ZigBee sender and the WiFi receiver in the NLOS scenario

the distance is 0.5m, the throughput of FreeBee and ZigFi are 10.5bps and 215.9bps, respectively. The SER of FreeBee and ZigFi are 0.125 and 0.071. The gap in SER widens when the distance increases. This is because FreeBee leverages the RSSI to identify beacons. RSSI usually decreases with the increasing of distance. The SER of Freebee therefore goes up and the SER of ZigFi stays stable, when the distance between ZigBee and WiFi increases.

C. Performance under different settings

Next we study ZigFi's performance under different settings in two different modes. The default mode doesn't have the power adjusting module, so the ZigBee sender transmits with a predefined minimum power. In the adaptive power mode, ZigBee sender transmits with an adaptive power, which satisfies the requirement of SINR and energy cost (explained in the previous section). In addition, we measure the potential negative impact of ZigFi to the existing WiFi traffic. By default there is no obstacle among the devices and no additional noise injected.

Performance with different distance between the ZigBee sender and the WiFi receiver. The distance between the ZigBee sender and the WiFi receiver affects SINR at the receiver. We change the distance between the ZigBee sender and the WiFi receiver d_4 from 1m to 15m. The distance between the WiFi sender and the WiFi receiver d_1 is 10m.

Figure 15(a) and Figure 15(b) plot the results. In the default mode, the SER increases and the throughput decreases with the increase of the distance d_4. Adaptive is better than default. In the adaptive mode, the performance of the ZigFi is consistently close to 210 bps accross all distances considered. When the distance is 15m, the default mode and the adaptive mode achieve a throughput of 151.7bps and 211.4bps, while the SER is 0.342 and 0.085, respectively.

Performance with different distance between the WiFi sender and the WiFi receiver. The distance between WiFi sender and WiFi receiver affects the WiFi signal strength at the receiver. Correspondingly, the ZigFi SINR will change, according to Eq. (2). We change the distance between the WiFi sender and the WiFi receiver d_1 from 1m to 15m. The distance between the ZigBee sender and the WiFi receiver d_4 is 5m.

Figure 16(a) and Figure 16(b) plot the results. We find that the performance gap between the default mode and the adaptive power mode widens, when the distance d_1 increases. When d_1 is 15m, the default mode and the adaptive mode achieve a throughput of 181.2bps and 215.9bps, while the SER is 0.212 and 0.061, respectively.

The performance of ZigFi under varied intensities of noise. The noise in the environment is another factor that impacts the ZigFi SINR. In this experiment, we use the USRP to generate gaussian noise with different powers. d_1 is 10m. d_4 is 5m. The distance between the noise source and the WiFi receiver d_5 is 5m.

Figure 17(a) and Figure 17(b) show that the throughput of ZigFi degrades when the noise intensity increases. When the noise is 2 dBm, which is much stronger than the transmission power ZigBee can use, the default mode and the adaptive mode achieve a throughput of 100.3bps and 125.7bps, while the SER is 0.411 and 0.147, respectively. ZigFi in the adaptive mode is more resilient to noise, with regard to the SER.

It is worth noting that stronger noise reduces the Packet Recept Ratio(PRR) of WiFi, as we will discuss in the next subsection. As a result, the WiFi receiver has a lower chance to retrieve CSI values from the received WiFi packets. That is the main cause of throughput degradation of ZigFi under strong noise.

The performance of ZigFi in the Non-Line-of-Sight (NLoS) Scenario. NLoS propagation of signals affect the

SINR of ZigFi at the receiver. In this experiment, we place an obstacle between the ZigBee sender and the WiFi receiver to block the line-of-sight transmission. Then we change the distance between the ZigBee sender and the WiFi receiver d_4 from 1m to 10m. As we can see from Figure 18(a) and Figure 18(b), ZigFi in the default mode is susceptible to signals' NLOS propagation. The throughput decreases and the SER increases sharply with the increase of the distances d_4. In comparison, ZigFi in the adaptive mode is robust under similar conditions, because the ZigBee sender can always find an appropriate power to transmit to the WiFi receiver. When d_4 is 10m, the throughput in the adaptive mode is 208.9bps and the SER is 0.094.

D. The impact on existing WiFi communication

Communication using ZigFi requires carefully interaction between signals from ZigBee and WiFi senders and could impact WiFi performance. In the previous four groups of experiments, we also measure the PRR of the WiFi link with/without ZigFi transmissions. Overall, ZigFi transmission has minimal impact on the PRR of the WiFi link. As an example, in the experiments corresponding to Figure 15(c) and Figure 15(d), we observe a WiFi PRR of 0.868 and 0.864 respectively. Under the same setting, we observe a decrease of 0.013-0.067 in WiFi PRR, due to ZigFi transmissions.

VI. Conclusion

CTC is a crucial technique for emerging IoT applications. In this paper, we tackle the problem of CTC from ZigBee to WiFi. Our study reveals that CSI of the overlapped packets can be utilized to convey data across different wireless technologies. Based on this finding, we design a receiver-initiated protocol and translate the decoding problem into a problem of CSI classification with SVM. The implementation and experiments demonstrate that ZigFi achieves high speed ZigBee to WiFi CTC with minimal impact on existing WiFi traffic in the network. In the future, we will try to extend the design ZigFi to comprehensively support multiple-to-one concurrent transmissions. We also plan to study the feasibility of CTC from ZigBee to BLE, using similar techniques.

Acknowledgment

This work was supported by National Key R&D Program of China 2017YFB1003000, National Basic Research Program (973 program) under Grant of 2014CB347800, National Natural Science Fund of China for Excellent Young Scientist No. 61422207, The NSFC No. 61772306 and No. 61672320, and China Postdoctoral Science Foundation No. 2016M601034.

References

[1] Y. Yan, P. Yang, X. Li, T. Yue, Z. Lan, and L. You, "Zimo: building cross-technology mimo to harmonize zigbee smog with wifi flash without intervention," in *Proceedings of ACM MobiCom*, 2013.

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

[3] T. Zachariah, N. Klugman, B. Campbell, J. Adkins, N. Jackson, and P. Dutta, "The internet of things has a gateway problem," in *Proceedings of ACM HotMobile*, 2015.

[4] B. Bloessl, S. Joerer, F. Mauroner, and F. Dressler, "Low-cost interferer detection and classification using telosb sensor motes," in *Proceedings of ACM MobiCom*, 2012.

[5] M. Doddavenkatappa, M. C. Chan, and B. Leong, "Splash: fast data dissemination with constructive interference in wireless sensor networks," in *Proceedings of USENIX NSDI*, 2013.

[6] J. Han, H. Ding, C. Qian, W. Xi, Z. Wang, Z. Jiang, L. Shangguan, and J. Zhao, "Cbid: A customer behavior identification system using passive tags," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2885–2898, 2016.

[7] Z. Chi, Z. Huang, Y. Yao, T. Xie, H. Sun, and T. Zhu, "Emf: Embedding multiple flows of information in existing traffic for concurren communication among heterogeneous iot devices," in *Proceedings of IEEE INFOCOM*, 2017.

[8] Y. Jiang, Z. Li, and J. Wang, "Ptrack: Enhancing the applicability of pedestrian tracking with wearables," in *Proceedings of IEEE ICDCS*, 2017.

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

[10] X. Guo, X. Zheng, and Y. He, "Wizig: Cross-technology energy communication over a noisy channel," in *Proceedings of IEEE INFOCOM*, 2017.

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

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

[13] S. Sen, R. R. Choudhury, and S. Nelakuditi, "Csma/cn: carrier sense multiple access with collision notification," in *Proceedings of ACM MobiCom*, 2010.

[14] A. Gonga, O. Landsiedel, P. Soldati, and M. Johansson, "Multi-channel communication vs. adaptive routing for reliable communication in wsns," in *Proceedings of ACM IPSN*, 2012.

[15] J. Huang, G. Xing, G. Zhou, and R. Zhou, "Beyond co-existence: Exploiting wifi white space for zigbee performance assurance," in *Proceedings of IEEE ICNP*, 2011.

[16] R. Zhou, Y. Xiong, G. Xing, L. Sun, and J. Ma, "Zifi:wireless lan discovery via zigbee interference signatures," in *Proceedings of ACM MobiCom*, 2010.

[17] X. Zheng, Z. Cao, J. Wang, Y. He, and Y. Liu, "Zisense: towards interference resilient duty cycling in wireless sensor networks," in *Proceedings of ACM SenSys*, 2014.

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

[19] Y. Zhang and Q. Li, "Howies: A holistic approach to zigbee assisted wifi energy savings in mobile devices," in *Proceedings of IEEE INFOCOM*, 2013.

[20] X. Zhang and K. G. Shin, "Gap sense: Lightweight coordination of heterogeneous wireless devices," in *Proceedings of IEEE INFOCOM*, 2013.

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

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

[23] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: gathering 802.11n traces with channel state information," *Proceedings of ACM Sigcomm Computer Communication Review*, vol. 41, no. 1, pp. 53–53, 2011.

[24] Z. Li, Y. Xie, M. Li, and K. Jamieson, "Recitation: Rehearsing wireless packet reception in software," in *Proceedings of ACM MobiCom*, 2015.

[25] C. Reports, "2.4 ghz ieee 802.15.4 / zigbee-ready rf transceiver," http://www.ti.com/lit/ds/symlink/cc2420.pdf.