Enabling Direct Messaging from LoRa to ZigBee in the 2.4 GHz Band for Industrial Wireless Networks

Junyang Shi, Xingjian Chen, Mo Sha
Department of Computer Science
State University of New York at Binghamton
{jshi28, xchen218, msha}@binghamton.edu

***Abstract*—IEEE 802.15.4-based wireless sensor-actuator networks (WSANs) have been quickly adopted by process industries in recent years because of their significant role in improving industrial efficiency and reducing operating cost. Battery-powered wireless modules can be used to easily and inexpensively retrofit existing sensors and actuators in industrial facilities without running cabling for communication and power. Wireless-enabled sensors, actuators, and controllers form a low-power multi-hop mesh network to exchange sensing data and control commands. Today, industrial WSANs are becoming tremendously larger and more complex than before. A large and complex mesh network is hard to manage and inelastic to change once the network is deployed. Besides, flooding-based time synchronization and information dissemination introduce significant communication overhead to the network. More importantly, the deliveries of urgent and critical information such as emergency alarms suffer long delay, because those messages must go through the hop-by-hop transport. A promising solution to overcome those limitations is to enable the direct messaging from a long-range radio to an IEEE 802.15.4 radio. Then messages can be delivered to all field devices in a single-hop fashion. This paper presents our study on enabling the cross-technology communication (CTC) from LoRa to ZigBee (IEEE 802.15.4) using the energy emission of the LoRa radio in the 2.4 GHz band as the carrier to deliver information. Experimental results show that our CTC approach provides reliable communication from LoRa to ZigBee with the throughput of up to 576.80bps and the bit error rate (BER) of up to 5.23% in the 2.4 GHz band.**

***Index Terms*—Industrial Wireless Sensor-Actuator Networks, Cross-Technology Communication, LoRa, ZigBee**

I. Introduction

Industrial networks have developed alongside the Internet. While the Internet is built to interconnect billions of heterogeneous devices communicating globally large amounts of data, industrial networks typically connect hundreds or thousands of sensors and actuators in industrial facilities, such as steel mills, oil refineries, chemical plants, and infrastructures implementing complex monitoring and control processes. Although the typical process applications have low data rates, they pose unique challenges because of their critical demands for *reliable* and *real-time* communication in harsh industrial environments. Failing to achieve such performance can lead to production inefficiency, safety threats, and financial loss. Those demands have been traditionally met by specifically chosen wired solutions, e.g., the Highway Addressable Remote Transducer (HART) communication protocol [1], where cables connect sensors and forward sensor readings to a control room where a controller sends commands to actuators. However, wired networks are often costly to deploy and maintain in industrial environments and difficult to reconfigure to accommodate new production requirements.

Wireless sensor-actuator network (WSAN) technology is appealing for use in industrial applications because it does not require wired infrastructure. Battery-powered wireless modules easily and inexpensively retrofit existing sensors and actuators in industrial facilities without running cabling for communication and power. IEEE 802.15.4-based WSANs operate at low-power and can be manufactured inexpensively, which make them ideal where battery lifetime and costs are important. The leading industrial WSAN standards (WirelessHART [2] and ISA100 [3]) have adopted the IEEE 802.15.4-based WSANs.

The current approach to implementing industrial WSANs relies on a multi-hop mesh network to deliver sensing data and control commands. Today, industrial WSANs are becoming tremendously larger and more complex than before. A large and complex mesh network is hard to manage and inelastic to change once the network is deployed. Besides, flooding-based time synchronization and information dissemination introduce significant communication overhead to the network. More importantly, the deliveries of urgent and critical information such as emergency alarms suffer long delay, because those messages must go through the hop-by-hop transport.

Low-power wide-area networks (LPWANs) are emerging as a promising technology, which provides long-distance connections to a large number of devices [4]. Recent years have witnessed rapid real-world adoption of LPWAN for various Internet of Things (IoT) applications. The limitations of multi-hop mesh networks can be overcome by enabling the direct messaging from a long-range LPWAN radio to an IEEE 802.15.4 (ZigBee) radio. Leveraging the large coverage, a LPWAN-enabled base station can disseminate the network management messages, time synchronization beacons, and urgent information to WSAN devices in a single-hop fashion. Semtech's recently announced LoRa SX1280/SX1281 wireless RF chips [5], operating in the 2.4 GHz industrial, scientific and medical (ISM) band, open new opportunities for the direct messaging from LoRa to ZigBee. This paper presents a direct messaging solution from LoRa to ZigBee, leveraging the recent advancements on the cross-technology communication (CTC) technologies. The CTC from LoRa to ZigBee is achieved by putting specific bytes in the payload of

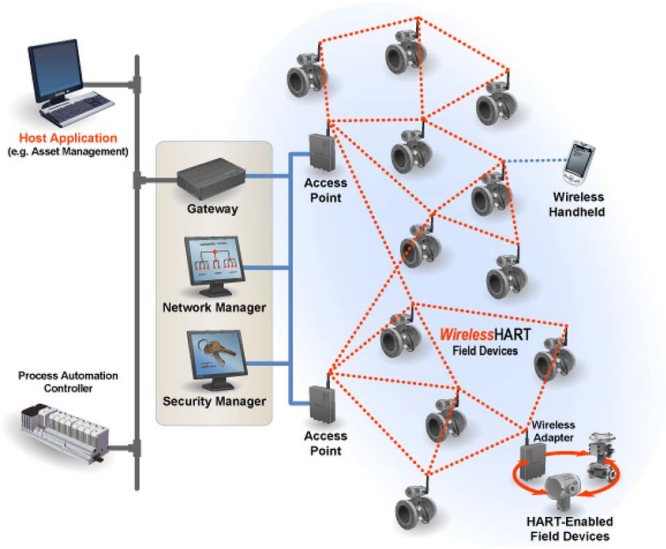


Fig. 1. Architecture of a WirelessHART network (Credit: HART Communication Foundation [2]).

legitimate LoRa packets. The bytes are selected such that the corresponding information can be understood by the ZigBee devices through sampling the received signal strength (RSS). Our LoRa to ZigBee CTC solution does not require any hardware modification to the existing WSAN field devices. Specifically, this paper makes the following contributions:

- To our knowledge, this is the first paper to investigate the CTC from LoRa to ZigBee in the 2.4 GHz band, distinguished with previous work pertaining to the CTC among WiFi, ZigBee, and Bluetooth devices.
- This paper performs an empirical study that investigates the characteristics of LoRa in the 2.4 GHz band from a CTC's point of view and provides a set of new observations.
- This paper introduces a novel LoRa to ZigBee CTC approach. By elaborately tuning the LoRa's packet payload, a ZigBee device is capable of decoding the information carried by the LoRa packet by purely sampling the RSS.
- Our proposed CTC approach has been implemented and tested on real hardware. Experimental results show that our approach provides reliable communication from LoRa to ZigBee with the throughput of up to $576.80bps$.

The remainder of the paper is organized as follows. Section II discusses the background of IEEE 802.15.4-based industrial WSANs and LoRa technology. Section III introduces our empirical study and Sections IV presents the design of our CTC approach. Section V shows our evaluation. Section VI reviews the related work and Section VII concludes the paper.

II. Background

In this section, we provide a brief introduction to the IEEE 802.15.4-based industrial WSANs and LoRa technology.

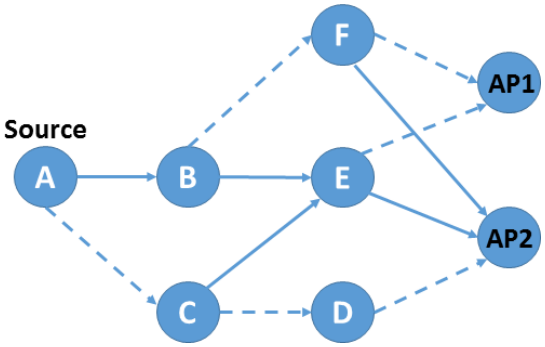


Fig. 2. An example of graph routing. The solid lines represent the primary routing paths and the dash lines denote the backup routes.

A. IEEE 802.15.4-Based Industrial WSANs

To meet the stringent reliability and real-time requirements, industrial WSAN standards make a set of unique network design choices that distinguish industrial WSANs from traditional wireless sensor networks (WSNs) designed for best effort services [6]. For instance, WirelessHART [2] and ISA100 [3], the leading industrial WSAN standards, specify a centralized network management architecture that enhances the timing predictability of packet deliveries and visibility of network operations. Figure 1 shows the architecture of a WirelessHART network. A WirelessHART network consists of a gateway, multiple access points, and a set of field devices (sensors and actuators). The access points and field devices are equipped with half-duplex omnidirectional radio transceivers, which are compatible with the IEEE 802.15.4 physical layer, and form a multi-hop wireless mesh network. The access points are connected with the gateway device through wired links and serve as bridges between the gateway and field devices. The network manager, a software module running on the gateway, is responsible for managing the entire wireless network. The network manager collects the link traces and network topology information from the field devices, and determines the routes between itself and all devices.

To enhance the reliability of packet deliveries, WirelessHART supports source routing and graph routing. Source routing provides a single routing path for each data flow (from sensors to actuators), whereas graph routing first generates a reliable graph in which each device should have at least two neighbors to which they may send packets and then provides multiple redundant routes based on the graph. Figure 2 shows a graph routing example. To send a packet to access points, Device A may transmit the packet to Device B by using the main routing path or Device C through the backup route. From those devices, the packet may take several alternate routes to reach the access points. Graph routing is designed to enhance the network reliability through route diversity and redundancy.

To enhance the timing predictability of packet deliveries, WirelessHART adopts the time-slotted channel hopping (TSCH) technology in the medium access control (MAC) layer. As Figure 3 shows, all devices clocks are synchronized, and time is divided into time slots with a fixed length. To

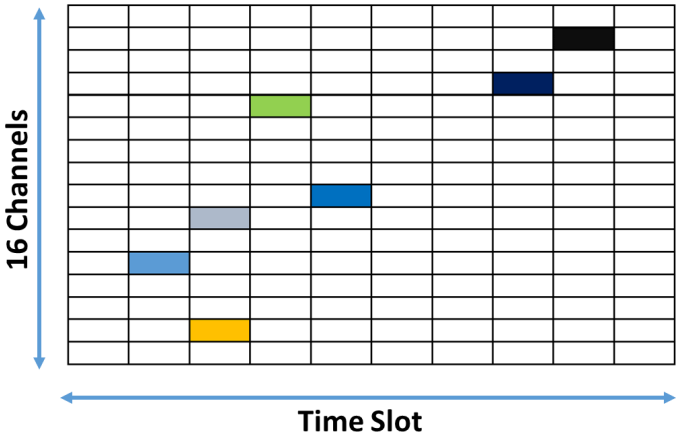


Fig. 3. TSCH technology.

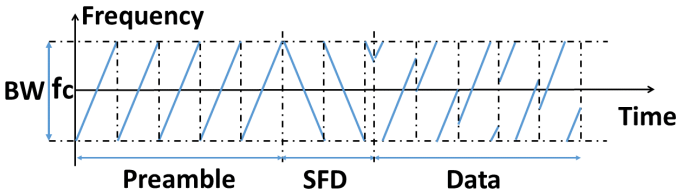


Fig. 4. An example of LoRa transmission with upchirps, downchirps and data chirps.

combat narrow-band interference and multi-path fading, TSCH uses up to 16 channels operating in the 2.4 GHz band, and each device switches its channel in every slot. Channel blacklisting is an optional feature that allows the network operator to restrict the channel hopping of field devices network-wide to selected channels in the wireless band.

B. LoRa Overview

LPWAN is emerging as a promising wireless technology to provide long-distance connections with a greater than one-kilometer range, covering a large number of IoT devices [4]. LoRa, which is short for "Long Range", is an industry LPWAN technology, initiated by Semtech [7] and promoted by the LoRa Alliance [8] to build scalable wireless networks. LoRa leverages the chirp spread spectrum (CSS) to modulate data in the physical layer and operates in the unlicensed 915MHz (in the United States, Canada and South America) and 2.4 GHz bands (globe). In this paper, we focus on the LoRa technology, which operates in the 2.4 GHz band, specifically using the Semtech's new SX1280/SX1281 wireless RF chips [5].

TABLE I
KEY LORA PHYSICAL-LAYER PARAMETERS IN THE 2.4 GHZ BAND.

Parameter	Options
f_c	between 2400 MHz to 2482 Mhz
SF	5, 7, 8, 9, 10, 11, 12
BW(KHz)	203, 406, 812, 1625
CR	4/5, 4/6, 4/7, 4/8

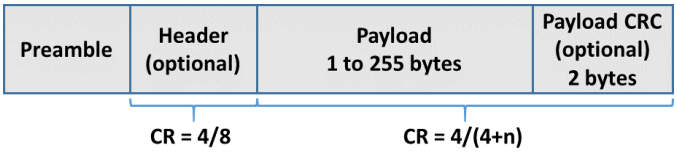


Fig. 5. LoRa variable-length packet format ($n \in [1, 4]$).

Physical-Layer Characteristics: LoRa employs the CSS modulation, which leverages frequency chirps with a constantly increasing or decreasing frequency sweeping through a predefined bandwidth. Figure 4 shows an example of LoRa transmission with upchirps, downchirps, and data chirps in the frequency variation over time. The first several upchirps, which are configurable from 2 to 65535, are preambles. Each chirp's frequency sweeps from the minimum frequency (f_{min}) to the maximum frequency (f_{max}). The following 2.25 downchirps are Start Frame Delimiter (SFD), whose frequency goes from f_{max} to f_{min}. The rest chirps are data chirps. The position of frequency discontinuity (a sudden change from f_{max} to f_{min}) of data chirps represents different encoded data bits.

The key LoRa physical-layer parameters, which are configurable by the user, include the frequency bandwidth (BW), central carrier frequency (f_c), spreading factor (SF), and coding rate (CR). Table I lists the possible values for each parameter. The time duration of transmitting a single LoRa chirp (T_s) is:

$$T_s = \frac{2^{SF}}{BW} \tag{1}$$

and each LoRa chirp can convey SF bits of information. Thus, the physical-layer data transmission bit rate of LoRa (R_b) is:

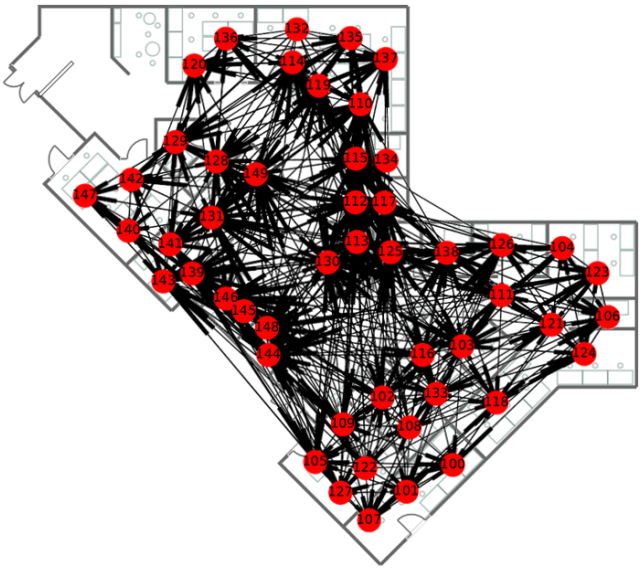
$$R_b = \frac{SF * CR}{T_s} = SF * \frac{BW}{2^{SF}} * CR \tag{2}$$

The selection of those parameters makes significant impacts on the LoRa decoding sensitivity and transmission range. For instance, either an increase in SF or a decrease in BW enlarges the transmission range.

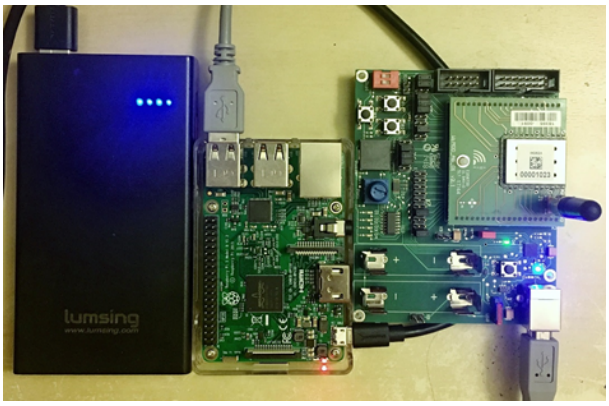
Physical Frame Format: Semtech specifies the physical frame format of LoRa packets. As Figure 5 shows, a LoRa frame starts with a preamble followed by an optional header using a coding rate of $4/8$. The payload size (PL) of each LoRa packet ranges from 1 to 255 bytes. LoRa uses one byte to store the payload size. CRC check is optional and uses a configurable coding rate.

The number of LoRa data chirps (N_{chirp}) for transmitting a packet with PL bytes payload can be calculated by Eq. 3, where PL is the LoRa payload size in bytes, CRC is 16 if the CRC check is enabled or 0 otherwise, H is the size of LoRa packet header, and DE is either 2 if the low data rate optimization is enabled or 0 otherwise.

$$N_{chirp} = 8 + max(\lceil \frac{8PL - 4SF + 8 + CRC + H}{4(SF - DE)} \rceil * \frac{4}{CR}, 0) \tag{3}$$



(a) Testbed deployment: Red circles are TelosB motes and black lines are wireless links when devices transmit at 0dBm.



(b) Raspberry Pi Model B and iM282A.

Fig. 6. Testbed deployment and LoRa device.

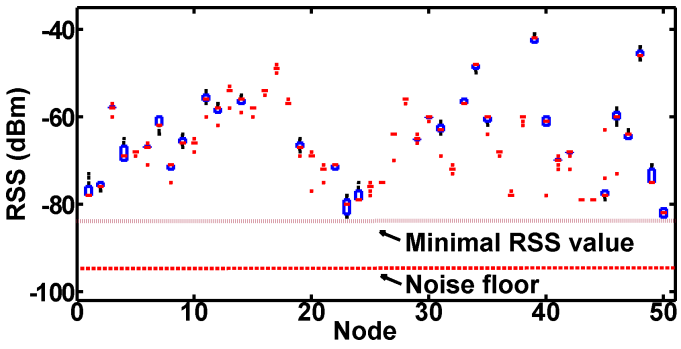
With Eq. 1 and 3, the on-air time of a LoRa packet can be calculated as:

$$T_o = (N_{chirp} + N_{preamble}) * \frac{2^{SF}}{BW} \quad (4)$$

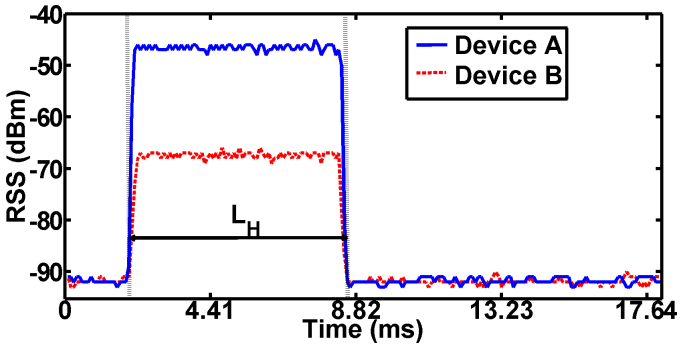
where $N_{preamble}$ denotes the number of preamble chirps and $N_{chirp} + N_{preamble}$ represents the total number of chirps used to carry the LoRa packet.

III. Empirical Study

In our empirical study, we first examine the detectability of LoRa signals on ZigBee devices and then explore the RSS features, which can be used for the CTC from LoRa to ZigBee. The empirical study is performed on our testbed, which consists of 50 TelosB motes [9] (ZigBee devices) placed throughout 22 student offices, lounge, labs and conference rooms [10]. Figure 6(a) shows the device placement on our testbed. The wireless network has up to 4 hops when the testbed devices transmit at $0dBm$. A Raspberry Pi Model B [11] integrated with a WiMOD iM282A LoRa



(a) Boxplot of RSS measurements.



(b) Example RSS signatures measured by two ZigBee devices located at different places when the LoRa device transmits a packet.

Fig. 7. Detectability of LoRa signals on ZigBee devices.

transceiver [12] (with a Semtech SX1280 LoRa chip [5]) is used as the LoRa transmitter, as Figure 6(b) shows. We configure the LoRa transceiver to transmit at $15dBm$.

A. Detectability of LoRa Signals on ZigBee

We first perform experiments to examine the detectability of LoRa signals on ZigBee devices in the 2.4 GHz band. We configure the LoRa transmitter placed in the center of our testbed to broadcast packets and control the 50 ZigBee devices on our testbed to sample the RSS. The ZigBee and LoRa channels are configured to overlap with each other. Figure 7(a) shows the Bloxplot of RSS measurements. All ZigBee devices on our testbed can detect the ongoing LoRa transmissions if they set the RSS threshold between the minimal RSS value ($-83dBm$) and the noise floor ($-92dBm$). As a comparison, the transmissions generated by any ZigBee device can reach up to 66.0% of devices on the testbed.

Observation 1*: ZigBee devices can detect ongoing LoRa transmissions through sampling the RSS when the ZigBee and LoRa channels overlap.*

Figure 7(b) shows the example RSS signatures measured by two ZigBee devices located at different places when LoRa transmits a packet. We define the sequence of RSS values measured by ZigBee when LoRa transmits a packet as a RSS signature. The naive approach would be to using different RSS values to encode different information, but this would require each device to generate its own mapping between the RSS values and encoded information since the RSS values

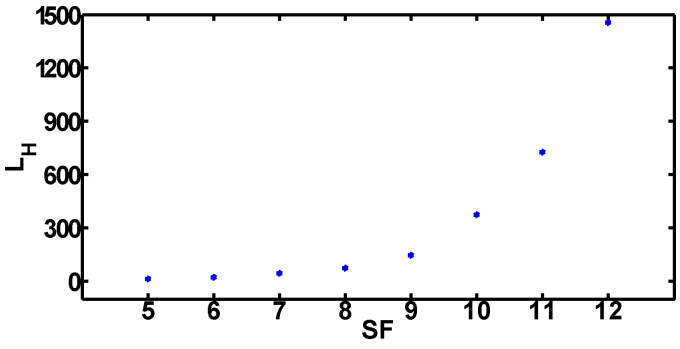


Fig. 8. L_H values captured by ZigBee when LoRa transmits the same packet with using different SF.

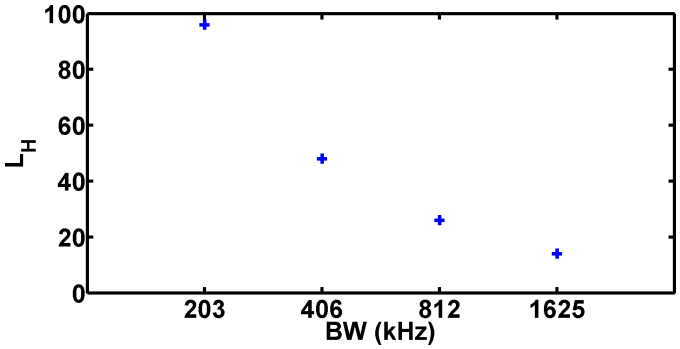


Fig. 9. L_H values captured by ZigBee when LoRa transmits the same packet with using different BW.

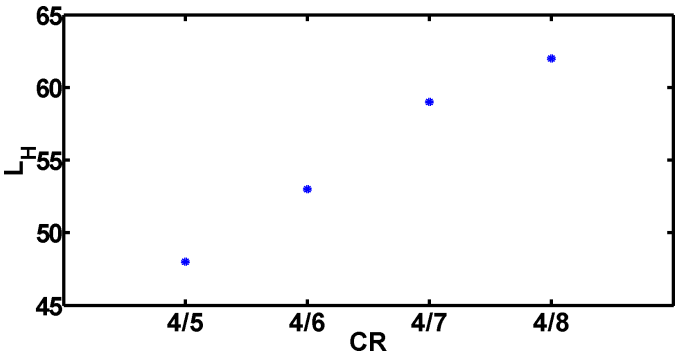


Fig. 10. L_H values captured by ZigBee when LoRa transmits the same packet with using different CR.

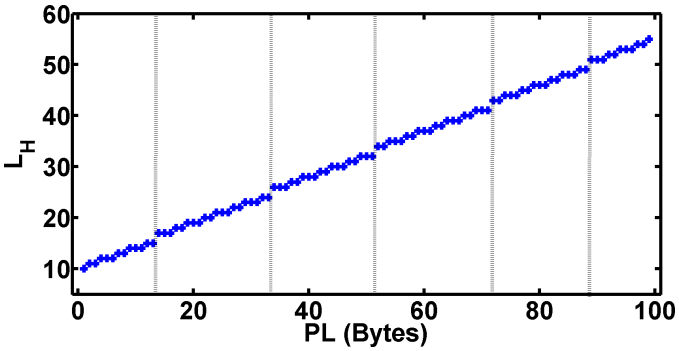


Fig. 11. L_H values captured by ZigBee when LoRa transmits packets carrying different payload. Payload size is PL.

depend on the link distance. An alternative approach is to use the number of consecutive RSS values higher than the threshold (L_H) to encode information. As Figure 7(b) shows, both Device A and B get $L_H = 73$ when the RSS threshold is $-85dBm$, and L_H is independent of link distance. Using L_H instead of the absolute RSS values, the network only needs one device to generate the mapping between RSS values and encoded information and share it with the rest. This significantly reduces the device setup and calibration overhead.

Observation 2*: The number of consecutive RSS values higher than the threshold (L_H) can be used to encode information.*

B. Creation of RSS Signatures with Different L_H

As discussed in Section II-B, using different physical-layer parameters (i.e., SF, BW, CR, and PL) can create the RSS signatures with different L_H. Our goal is to maximize the number of distinguishable RSS signatures (with different L_H). We next run experiments to study the impact of tuning those physical-layer parameters on the number of distinguishable RSS signatures.

We first set BW to 1625 MHz, CR to $4/5$, and PL to 5 bytes, vary SF from 5 to 12, and then measure the L_H captured by the ZigBee device. Figure 8 plots L_H under different SF. The L_H is 13, 24, 46, 74, 147, 374, 728, and 1457 for SF from 5 to 12. Tuning SF can generate eight distinguishable RSS signatures.

We then fix SF to 5, CR to $4/5$, and PL to 5, vary BW from $203kHz$ to $1625kHz$, and measure L_H. As Figure 9 shows, every time BW doubles, L_H roughly reduces to a half. Tuning BW can generate four distinguishable RSS signatures.

We also repeat the experiments under different CR when $SF = 5$, $BW = 406$, and $PL = 5$. Tuning CR can generate four distinguishable RSS signatures, as Figure 10 shows.

Finally, we run the experiments when LoRa transmits packets with different payload sizes PL. Figure 11 shows L_H when the LoRa payload size increases from 1 byte to 99 bytes when $SF = 5$, $BW = 1625$, and $CR = 4/5$. From the results, we can see that through changing PL, the ZigBee device obtains a large number of distinguishable RSS signatures with different L_H.

Observation 3*: Tuning the payload size PL is the most effective way to generate a large number of distinguishable RSS signatures.*

IV. CTC Design

In this section, we present the design of our CTC approach from LoRa to ZigBee in the 2.4 GHz band based on the observations presented in Section III.

We assume that the time slot used in WSANs has the length of $15ms$ and define the total time (T_t) for the LoRa device to transmit a packet as:

$$T_t = T_o + T_r \tag{5}$$

TABLE II
PL, T_o, T_r, T_t, AND L_H OF EACH DISTINGUISHABLE RSS SIGNATURE.

Index	PL(byte)	T_o(ms)	T_r(us)	T_t(ms)	L_H
1	1	0.905	2800	3.705	10, 11, 12
2	9	1.305	3100	4.405	13, 14, 15, 16
3	17	1.605	3400	5.005	17, 18, 19
4	23	1.805	3700	5.505	20, 21, 22
5	32	2.205	4000	6.205	23, 24, 25, 26
6	39	2.505	4300	6.805	27, 28, 29
7	47	2.805	4600	7.405	30, 31, 32
8	54	3.105	4900	8.005	33, 34, 35, 36
9	62	3.405	5200	8.605	37, 38, 39
10	69	3.705	5500	9.205	40, 41, 42
11	77	4.005	6800	10.805	43, 44, 45, 46
12	84	4.305	7100	11.405	47, 48, 49
13	92	4.605	8000	12.605	50, 51, 52
14	99	4.905	8800	13.705	53, 54, 55, 56
15	107	5.205	9200	14.405	57, 58, 59

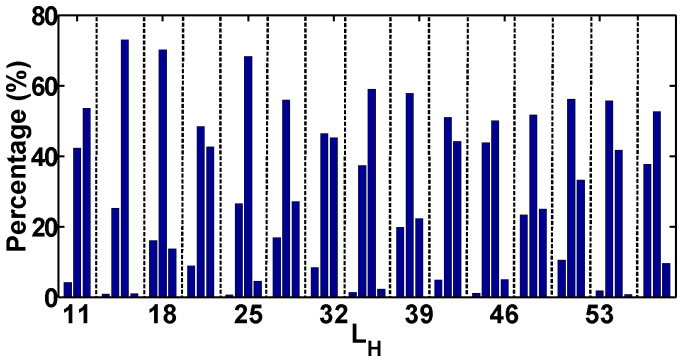


Fig. 12. Percentage histogram of each distinguishable RSS signature (L_H).

where T_o denotes the on-air time of a LoRa packet and T_r denotes the software delay of packet transmission. Our ZigBee device has a RSS sampling rate of 11.33KHz, providing 170 samples in every time slot.

Because of measurement inaccuracy, the ZigBee device may produce multiple L_H values when the LoRa device transmits the same payload. Thus, we must identify a set of payload sizes, which can be used to reliably generate RSS signatures with distinct L_H. There are three requirements for the payload size selection: (i) Different LoRa payload sizes must provide distinct L_H, which can be captured by the ZigBee device; (ii) T_t must not exceed $15ms$; (iii) The other three physical-layer parameters (SF, BW, and CR) must be determined before selecting the payload sizes. When we set $SF = 5$, $BW = 1625$, and $CR = 4/5$, we get 15 payload sizes, which meet the above requirements. Table II lists the payload size (PL), LoRa packet on-air time (T_o), software delay (T_r), total time (T_t), and possible L_H values of each distinguishable RSS signature. Figure 12 shows the percentage histogram of each distinguishable RSS signature when we configure LoRa to transmit 5000 packets using each PL and control ZigBee to measure L_H. As Figure 12 shows, the L_H values belonging to any two distinguishable RSS signatures are complete different. For example, when PL is one byte, 4.18% of the L_H values

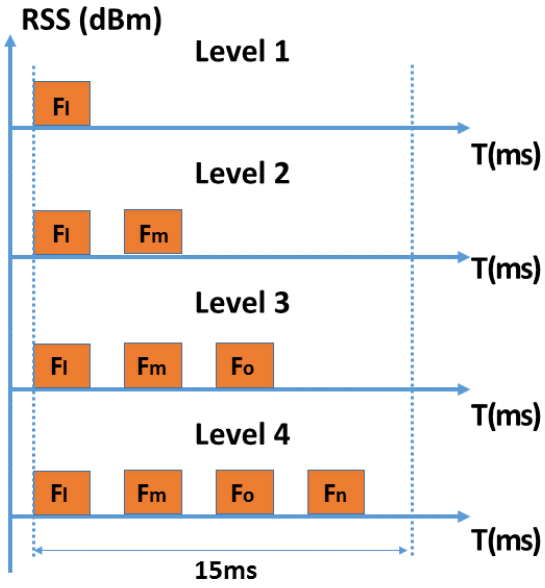


Fig. 13. RSS patterns for different levels.

captured by ZigBee is 10, 42.26% is 11, and 53.56% is 12. When PL is nine bytes, 0.83% of the L_H values captured by ZigBee is 13, 25.21% is 14, 72.95% is 15, and 1.01% is 16.

After identifying the set of payload sizes, the next step is to determine how to use those distinguishable RSS signatures to encode information. As Table II shows, each RSS signature $F_k = (k, PL_k, T_{o_k}, T_{r_k}, T_{t_k}, L_{H_k})$ $\{1 \leq k \leq 15\}$ is denoted by an index k, a payload size PL_k, a packet on-air time T_{o_k}, a software delay T_{r_k}, a total time T_{t_k}, and a set of L_{H_k}. We can put multiple distinguishable RSS signatures in a single time slot to get a set of distinguishable RSS patterns to encode information. Multiple RSS signatures $\{F_l, F_m, ..., F_n\}$ are combined to form the RSS pattern P_z in a single time slot. We follow the below four steps to get the total number of distinguishable RSS patterns.

Step I: Identify the maximum distinguishable RSS pattern level n_{level}: n_{level} determines the maximum number of distinguishable RSS signatures, which can be put in a single time slot. n_{level} is computed as:

$$n_{level} = \lfloor \frac{15}{T_{t_0}} \rfloor \tag{6}$$

where 15 is the time slot length and T_{t_0} is the smallest time duration of our distinguishable RSS signatures. According to Table II, T_{t_0} is 3.705ms. Then $n_{level} = 4$. In each time slot, we can put (i) one RSS signature $\{F_l\}$, (ii) two RSS signatures $\{F_l, F_m\}$, (iii) three RSS signatures $\{F_l, F_m, F_o\}$, or (iiii) four RSS signatures $\{F_l, F_m, F_o, F_n\}$ to form the distinguishable RSS pattern P_z $(1 \leq l, m, o, n \leq 15)$. Figure 13 shows the example RSS distinguishable patterns from level 1 to 4.

Step II: Combine the distinguishable RSS signatures: Multiple distinguishable RSS signatures (F_k $\{1 \leq k \leq 15\}$) can be combined to form different distinguishable RSS patterns. Algorithm 1 shows the algorithm, which computes the number of distinguishable RSS patterns. $count_1$, $count_2$, $count_3$, and

$count_4$ store the number of distinguishable RSS patterns in level 1 to 4, respectively. Algorithm 1 first initializes all $count_j$ to zero (Line 1). There are four nested loops (Line 2-19) and each loop iterates over the 15 distinguishable RSS signatures. Algorithm 1 uses four nested loops because at most four distinguishable RSS signatures can be put in a single time slot. The counter $count_1$ increases by one in Line 3 because it only considers the distinguishable RSS patterns in level 1 ($\{F_l\}$), which uses one RSS signature to form the distinguishable RSS pattern. Because T_{t_k} is not longer than $15ms$, a single distinguishable RSS signature can always be directly put into the distinguishable RSS pattern set. Similarly, line 6, 10, and 14 increase the counters by one for level 2, 3, and 4 distinguishable RSS patterns, respectively. Please note that the sum of T_{t_k} should be not longer than $15ms$ for level 2 (Line 5), 3 (Line 9), and 4 (Line 13). The output $count$ denotes the total number of distinguishable RSS patterns. By running Algorithm 1, we get 15, 81, 80, and 1 for $count_1$, $count_2$, $count_3$, and $count_4$, respectively. The total number of distinguishable RSS patterns ($count$) is $15+81+80+1 = 177$.

Algorithm 1: Algorithm to compute the number of distinguishable RSS patterns

```
   Input  : T_{t_k}
   Output: count
1  count_1 = 0, count_2 = 0, count_3 = 0, count_4 = 0,
     count = 0;
2  for l = 1; l ≤ 15; l++ do
3      count_1++;
4      for m = 1; m ≤ 15; m++ do
5          if T_{t_l} + T_{t_m} ≤ 15 then
6              count_2++;
7          end
8          for o = 1; o ≤ 15; o++ do
9              if T_{t_l} + T_{t_m} + T_{t_o} ≤ 15 then
10                 count_3++;
11             end
12             for n = 1; n ≤ 15; n++ do
13                 if T_{t_l} + T_{t_m} + T_{t_o} + T_{t_n} ≤ 15 then
14                     count_4++;
15                 end
16             end
17         end
18     end
19 end
20 count = count_1 + count_2 + count_3 + count_4;
```

Step III: Add the empty signature: More distinguishable RSS patterns can be created by adding the empty RSS signature (F_{kX}) into the time slot. F_{kX} represents the RSS signature captured by ZigBee when LoRa does not transmit any packet for the time duration T_{t_k}. Figure 14 shows two example distinguishable RSS patterns ($\{F_1, F_2, F_1\}$ and $\{F_1, F_{2X}, F_1\}$). Please note that we do not add the distinguishable RSS pattern which only has empty signatures into the distinguishable RSS pattern set, because using only empty RSS signatures to deliver a message can easily be interfered by external interference. For each level j, Algorithm 1 computes the number of combined signatures ($count_j$) without considering the empty signature. To prevent repetition, we fix the tail RSS signature in each RSS pattern. Then each RSS signature (F_k) except the tail one can be substituted by an empty RSS signature (F_{kX}). The number of distinguishable RSS patterns in each level after adding the empty signature is:

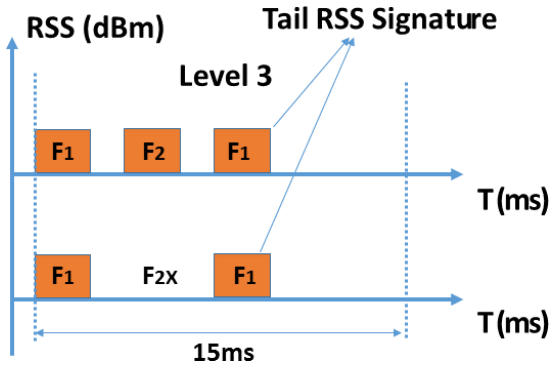


Fig. 14. Two example distinguishable RSS patterns in level 3.

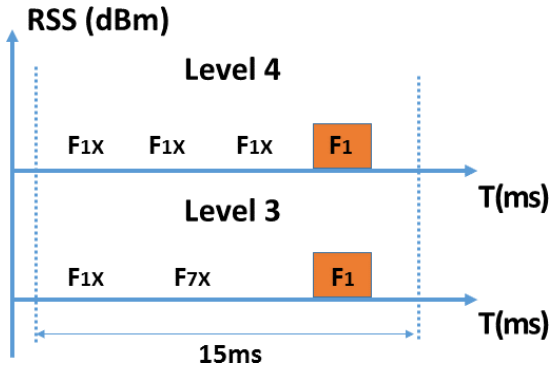


Fig. 15. An example cross-level duplication.

$$e_j = count_j * 2^{(j-1)} \tag{7}$$

Then the total number of distinguishable RSS patterns after adding the empty signature is:

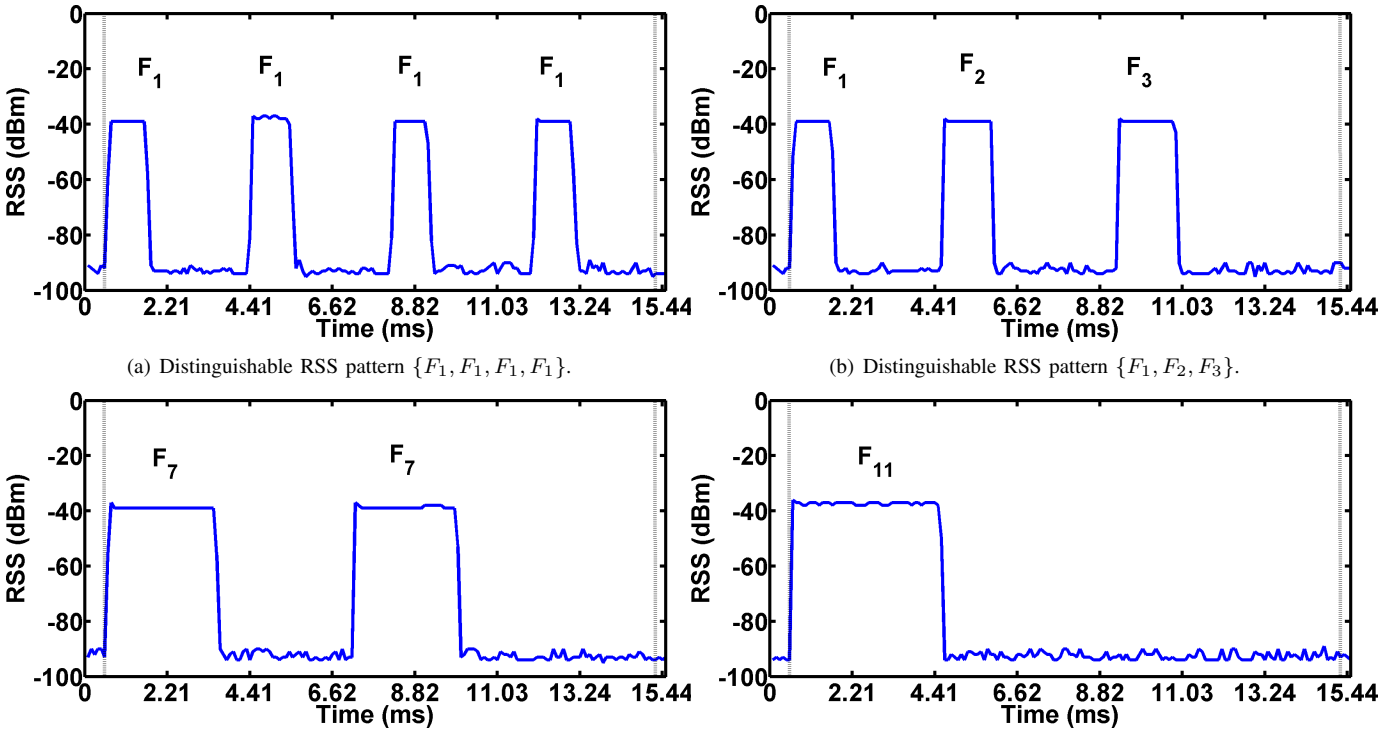
$$sum = \sum_{i=1}^{n_{level}} e_j \tag{8}$$

With Eq. 7 and 8, we get $sum = 1 * 2^3 + 80 * 2^2 + 81 * 2^1 + 15 * 2^0 = 505$.

Step IV: Remove the duplication: Because the distinguishable RSS patterns have the empty signature, there may exist multiple RSS patterns, which cannot be distinguished by the ZigBee device. We first remove the duplication at the same level. For instance, if there exist two distinguishable RSS patterns $\{F_1, F_2, F_1\}$ and $\{F_2, F_1, F_1\}$ in level 3 and we substitute F_1 and F_2 with the empty signatures. Then $\{F_{1X}, F_{2X}, F_1\}$ and $\{F_{2X}, F_{1X}, F_1\}$ are same. We use F_{kX} to denote the empty signature, which locates at the kth

TABLE III
30 DISTINGUISHABLE RSS PATTERNS.

	RSS patterns		RSS patterns		RSS patterns		RSS patterns		RSS patterns
1	$\{F_1, F_1, F_1, F_1\}$	2	$\{F_1, F_1, F_{1X}, F_1\}$	3	$\{F_1, F_{1X}, F_1, F_1\}$	4	$\{F_{1X}, F_1, F_1, F_1\}$	5	$\{F_1, F_{1X}, F_{1X}, F_1\}$
6	$\{F_{1X}, F_{1X}, F_1, F_1\}$	7	$\{F_{1X}, F_1, F_{1X}, F_1\}$	8	$\{F_{1X}, F_{1X}, F_{1X}, F_1\}$	9	$\{F_1, F_1, F_1\}$	10	$\{F_1, F_1, F_2\}$
11	$\{F_1, F_1, F_3\}$	12	$\{F_1, F_1, F_4\}$	13	$\{F_1, F_1, F_5\}$	14	$\{F_1, F_1, F_6\}$	15	$\{F_1, F_1, F_7\}$
16	$\{F_1, F_2, F_1\}$	17	$\{F_1, F_2, F_2\}$	18	$\{F_1, F_2, F_3\}$	19	$\{F_1, F_2, F_4\}$	20	$\{F_1, F_2, F_5\}$
21	$\{F_1, F_1\}$	22	$\{F_1, F_2\}$	23	$\{F_1, F_3\}$	24	$\{F_1, F_4\}$	25	$\{F_1, F_5\}$
26	$\{F_1\}$	27	$\{F_2\}$	28	$\{F_3\}$	29	$\{F_4\}$	30	$\{F_5\}$



(a) Distinguishable RSS pattern $\{F_1, F_1, F_1, F_1\}$.

(b) Distinguishable RSS pattern $\{F_1, F_2, F_3\}$.

(c) Distinguishable RSS pattern $\{F_7, F_7\}$.

(d) Distinguishable RSS pattern $\{F_{11}\}$.

Fig. 16. Four example RSS traces. Each trace lasts 15ms.

position in the distinguishable RSS pattern. The time duration of $F_{1X} + F_{2X}$ is equal to the one of $F_{2X} + F_{1X}$. The ZigBee device cannot distinguish them by sampling the RSS. Therefore, we must remove the duplication from sum, which is computed in Step III. Duplication also happens between different levels. Thus, the cross-level time equivalent test is required. The ZigBee device can generate 170 RSS samples in each time slot. The sampling interval is $0.088ms$. For two different distinguishable RSS patterns in different levels, if the time difference between two consecutive empty RSS signatures are less $0.088ms$, the ZigBee device cannot distinguish them. Figure 15 shows an example duplication. Because $|(T_{t_1} + T_{t_1} + T_{t_1}) - (T_{t_1} + T_{t_7})| = 0.005ms < 0.088ms$, the ZigBee device cannot distinguish $\{F_{1X}, F_{1X}, F_{1X}, F_1\}$ and $\{F_{1X}, F_{7X}, F_1\}$ by sampling the RSS. We use the brute-force method to compare any two distinguishable RSS patterns with each other in the same level and identify 33 duplicated distinguishable RSS patterns. We also find 18 duplicated distinguishable RSS patterns in the cross-level time equivalent test. After removing the duplication, the total number of distinguishable RSS patterns, which can be used to encode information, is $505 - 33 - 18 = 454$.

To maximize the throughput, the LoRa device can convert its binary data $i_{(2)}$ to 454-nary data $i_{(454)}$, while the ZigBee device can reverse the process to get the original binary data.

V. EVALUATION

In evaluation, we first run microbenchmark experiments to examine whether the ZigBee device can correctly capture every distinguishable RSS pattern in a single time slot and then measure the throughput and bit error rate (BER) of our CTC approach.

A. Microbenchmark Experiments

As presented in Section IV, we combine 15 distinguishable RSS signatures in different ways to generate 454 distinguishable RSS patterns, which are used to encode information.

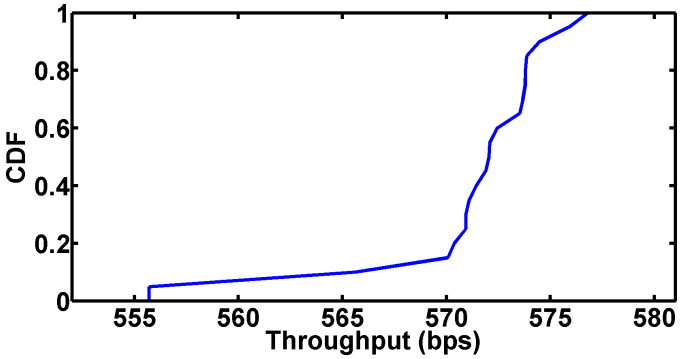


Fig. 17. CDF of CTC throughput.

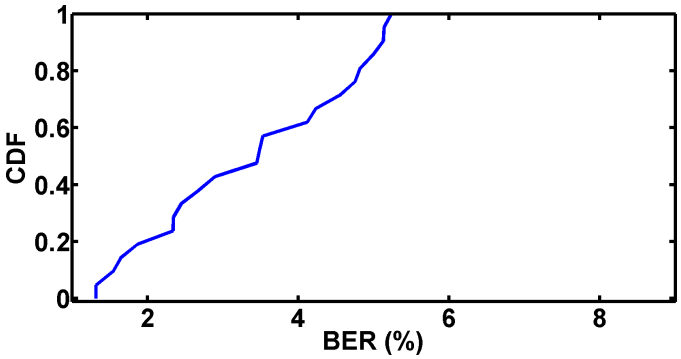


Fig. 18. CDF of CTC BER.

Table III lists 30 examples for RSS patterns in different levels. In this set of experiments, we control the LoRa device to generate all patterns in a round robin fashion by putting different payloads in the LoRa packets and observe that the RSS measurements captured by the ZigBee device always match the design. Figure 16 shows four example RSS traces following the the distinguishable RSS patterns, $\{F_1, F_1, F_1, F_1\}$, $\{F_1, F_2, F_3\}$, $\{F_7, F_7\}$, and $\{F_{11}\}$, respectively. By comparing the RSS measurements and pattern design, we confirm that all distinguishable RSS patterns generated by the LoRa device can be effectively identified by the ZigBee device.

B. Throughput and BER

In this set of experiments, we measure the throughput and BER of our CTC approach. We randomly generate 550 bytes, control the LoRa device to encode and transmit them, and measure the throughput and BER on the ZigBee device after it decodes them. We repeat the experiments for 20 times. Figure 17 plots the cumulative distribution function (CDF) of the measured throughput. The measured throughput ranges from 555.71bps to 576.80bps with the mean value of 571.52bps. The measured throughput values are very close to our theoretical maximum CTC throughput of $log_2^{454} * \frac{1000}{15} = 588.44bps$. The averaged throughput is 2.87% less than the theoretical value. The results show the efficiency of the encoding and decoding processes of our CTC approach. Figure 18 shows the CDF of BER. The BER ranges from 1.32% to 5.23% and the average value is 3.45%. The low BER values demonstrate the high reliability of our CTC approach.

VI. Related Works

With the unprecedented proliferation of heterogeneous wireless technologies and wireless devices, there exist severe wireless coexistence and management problems with the devices sharing the same unlicensed ISM bands. Early studies show that enabling CTC among heterogeneous devices can effectively address those problems and significantly improve the network performance. For instance, Zhou et al. developed ZiFi, which allows an embedded device to reduce its power consumption by using a low-power ZigBee radio to detect nearby WiFi APs [13]. Hao et al. and Yu et al. used WiFi signals to achieve time synchronization among ZigBee devices [14], [15]. Gawlowicz et al. leveraged CTC to coordinate the coexistence between LTE-U and WiFi devices in the 5 GHz band [16]. CTC has seen appreciable advancement in recent years. Significant efforts have been made to enable the CTC among ZigBee, WiFi, and Bluetooth devices in the 2.4 GHz band [15], [17]–[34]. Among those solutions, the wireless devices' capability of sensing the RSS in the air has been used to enable CTC between heterogeneous devices and the most widely used CTC scheme is to encode information on the temporal or amplitude dimension. For instance, Kim et al. enabled the CTC from WiFi to ZigBee by shifting the appearance of WiFi beacons in a temporal dimension to embed different symbols [21], [35]. Chebrolu et al. achieved the same goal by building an alphabet set and using different energy profile lengths to deliver messages [29]. Yin et al. designed C-Morse, which modulates the timing of WiFi packets to construct special energy patterns [22]. Guo et al. proposed a method to optimize the CTC throughput over a noisy channel [21], [35]. More recently, Guo et al. [19] developed the cross-demapping technique, which achieves the physical-level CTC from ZigBee to WiFi and leaves the computation overhead to the receiver. Chen et al. proposed to reserve part of the spectrum for narrow-band devices to perform concurrent transmissions and allowed a WiFi device to detect ZigBee signals without introducing extra traffic [34]. Li et al. developed WEBee, which emulates the ZigBee signals in the physical layer on commercial off-the-shelf (COTS) WiFi devices [23] and Jiang et al. proposed SymBee, which achieves symbol-level CTC from ZigBee to WiFi [24]. Jiang et al. developed XBee, which interprets a ZigBee frame by observing the bit patterns obtained at the Bluetooth receiver [25]. Chi et al. [31] proposed a communication framework that enables multiple concurrent communication among WiFi and Bluetooth devices. Unfortunately, those solutions are not directly applicable to send messages from a long-range LoRa radio to a ZigBee device because of the unique characteristics of LoRa radios operating in the 2.4 GHz ISM band. In contrast to previous studies among ZigBee, WiFi, and Bluetooth, this paper investigates the CTC from LoRa to ZigBee; to our knowledge, it represents the first systematic study of the characteristics of LoRa in the 2.4 GHz ISM band from a CTC's point of view. Our work is therefore orthogonal and complementary.

VII. Conclusions

IEEE 802.15.4-based WSANs operate at low-power and can be manufactured inexpensively and have been adopted by the leading industrial WSAN standards (WirelessHART and ISA100). The current approach to implementing industrial WSANs relies on a multi-hop mesh network to deliver sensing data and control commands. However, a large and complex mesh network is hard to manage and inelastic to change once the network is deployed. Besides, flooding-based time synchronization and information dissemination introduce significant communication overhead to the network. More importantly, the deliveries of urgent and critical information such as emergency alarms suffer long delay, because those messages must go through the hop-by-hop transport. A promising solution to overcome the limitations of using multi-hop mesh networks for industrial WSANs is to enable the direct messaging from a long-range radio to an IEEE 802.15.4 radio. Then messages can be delivered to field devices in a single-hop fashion. This paper presents our study on enabling the CTC from LoRa to ZigBee using the energy emission of the LoRa radio in the 2.4 GHz band as the carrier to deliver information. Our CTC approach puts specific bytes in the payload of legitimate LoRa packets. The bytes are selected such that the corresponding information can be understood by the ZigBee devices through sampling the RSS. Experimental results show that our CTC approach provides reliable communication from LoRa to ZigBee with the throughput of up to 576.80bps and the BER of up to 5.23% in the 2.4 GHz band.

Acknowledgment

This work was supported by the NSF through grant CRII-1657275 (NeTS).

References

[1] HART. (1986) Hart communication protocol and foundation (now the fieldcomm group). [Online]. Available: https://fieldcommgroup.org/

[2] WirelessHART. [Online]. Available: https://fieldcommgroup.org/technologies/hart/hart-technology

[3] ISA 100. [Online]. Available: http://www.isa100wci.org/

[4] A. Augustin, J. Yi, T. H. Clausen, and W. Townsley, "A study of lora: Long range and low power networks for the internet of things," *Sensors*, vol. 16, p. 1466, 10 2016.

[5] Semtech's new SX1280/SX1281 wireless RF chips. [Online]. Available: https://www.digikey.com/catalog/en/partgroup/sx1280-and-sx1281/68471

[6] C. Lu, A. Saifullah, B. Li, M. Sha, H. Gonzalez, D. Gunatilaka, C. Wu, L. Nie, and Y. Chen, "Real-Time Wireless Sensor-Actuator Networks for Industrial Cyber-Physical Systems," *Proceedings of the IEEE, Special Issue on Industrial Cyber Physical Systems*, vol. 104, no. 5, 2016.

[7] Semtech. [Online]. Available: https://www.semtech.com/

[8] LoRa. [Online]. Available: https://lora-alliance.org/

[9] TelosB: Telosb Mote Platform, Datasheet Provided by MEMSIC Inc. [Online]. Available: http://www.memsic.com/userfiles/files/Datasheets/WSN/telosb%5fdatasheet.pdf

[10] "Testbed at the State University of New York at Binghamton." [Online]. Available: http://www.cs.binghamton.edu/\%7emsha/testbed

[11] Raspberry Pi 3 Model B. [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/

[12] SK-iM282A. [Online]. Available: https://wireless-solutions.de/products/starterkits/sk-im282a.html

[13] R. Zhou, Y. Xiong, G. Xing, L. Sun, and J. ma, "Zifi: Wireless lan discovery via zigbee interference signatures," in *International Conference on Mobile Computing and Networking (MobiCom)*, 01 2010, pp. 49–60.

[14] T. Hao, R. Zhou, G. Xing, M. W. Mutka, and J. Chen, "WizSync: Exploiting Wi-Fi Infrastructure for Clock Synchronization in Wireless Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 13, no. 6, pp. 1379–1392, June 2014.

[15] Z. Yu, C. Jiang, Y. He, X. Zheng, and X. Guo, "Crocs: Cross-Technology Clock Synchronization for WiFi and ZigBee," in *International Conference on Embedded Wireless Systems and Networks (EWSN)*, 2018.

[16] P. Gawlowicz, A. Zubow, and A. Wolisz, "Enabling cross-technology communication between lte unlicensed and wifi," *IEEE International Conference on Computer Communications (INFOCOM)*, pp. 144–152, 2018.

[17] X. Guo, X. Zheng, and Y. He, "WiZig: Cross-Technology Energy Communication over a Noisy Channel," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2017.

[18] X. Guo, Y. He, X. Zheng, L. Yu, and O. Gnawali, "ZIGFI: Harnessing Channel State Information for Cross-Technology Communication," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2018.

[19] X. Guo, Y. He, X. Zheng, Z. Yu, and Y. Liu, "LEGO-Fi: Transmitter-Transparent CTC withCross-Demapping," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2019.

[20] X. Zheng, Y. He, and X. Guo, "StripComm: Interference-Resilient Cross-Technology Communication in Coexisting Environments," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2018.

[21] S. M. Kim and T. He, "Freebee: Cross-Technology Communication via Free Side-Channel," in *Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2015.

[22] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-Morse: Cross-Technology Communication with Transparent Morse Coding," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2017.

[23] Z. Li and T. He, "WEBee: Physical-Layer Cross-Technology Communication via Emulation," in *Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2017.

[24] S. Wang, S. M. Kim, and T. He, "Symbol-Level Cross-Technology Communication via Payload Encoding," in *IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2018.

[25] W. Jiang, S. M. Kim, Z. Li, and T. He, "Achieving Receiver-Side Cross-Technology Communication with Cross-Decoding," in *Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2018.

[26] X. Zhang and K. G. Shin, "Gap Sense: Lightweight Coordination of Heterogeneous Wireless Devices," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2013.

[27] Y. Zhang and Q. Li, "HoWiES: A Holistic Approach to ZigBee Assisted WiFi Energy Savings in Mobile Devices," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2013.

[28] X. Zhang and K. G. Shin, "Cooperative Carrier Signaling: Harmonizing Coexisting WPAN and WLAN Devices," *IEEE/ACM Transactions on Networking*, vol. 21, no. 2, 2013.

[29] K. Chebrolu and A. Dhekne, "Esense: Communication through Energy Sensing," in *Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2009.

[30] S. Yin, Q. Li, and O. Gnawali, "Interconnecting WiFi Devices with IEEE 802.15.4 Devices without Using a Gateway," in *International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2015.

[31] Z. Chi, Y. Li, H. Sun, Y. Yao, Z. Lu, and T. Zhu, "B2W2: N-Way Concurrent Communication for IoT Devices," in *ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2016.

[32] S. Wang, Z. Yin, S. M. Kim, and T. He, "Achieving Spectrum Efficient Communication under Cross-Technology Interference," in *International Conference on Computer Communication and Networks (ICCCN)*, 2017.

[33] W. Wang, X. Liu, Y. Yao, Y. Pan, Z. Chi, and T. Zhu, "Crf: Coexistent routing and flooding using wifi packets in heterogeneous iot networks," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2019.

[34] R. Chen and W. Gao, "Enabling cross-technology coexistence for extremely weak wireless devices," in *IEEE International Conference on Computer Communications (INFOCOM)*, 2019.

[35] S. M. Kim, S. Ishida, S. Wang, and T. He, "Free Side-Channel Cross-Technology Communication in Wireless Networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2974–2987, 2017.