Forward the Collision Decomposition in ZigBee

Yifeng Cao*†, Zhe Wang*†, Linghe Kong†, Guihai Chen†, Jiadi Yu†, Shaojie Tang‡, Yingying Chen§
†Shanghai Jiao Tong University
‡University of Texas at Dallas
§Rutgers University
Corresponding Email: linghe.kong@sju.edu.cn

***Abstract*—As wireless communication is tailored for low-power devices while the number of Internet of Things is growing exponentially, the collision problem in ZigBee is worsen. The classical approaches of solving collision problems lie in collision avoidance and packet retransmission, which could incur considerable overhead. The new trend is to decompose multi-packet collision directly, however, the high bit error rate limits its practical applications. Toward this end, we observe three major issues in the existing solutions: 1) all existing solutions adopt the priori-chip-dependent decomposition pattern, leading to the error propagation; 2) the available samples for chip decoding can be scarce, resulting in severe scarce-sample errors; 3) existing solutions assume the consistent frequency offset for consecutive packets, leading to inaccurate frequency offset estimation. To solve the issues of collision decomposition in ZigBee, we propose FORWARD, a novel physical layer design to enable highly accurate collision decomposition in ZigBee. The key idea is to generate all possible collided combinations as reference waveforms. The decomposition is determined by comparing the collided signal with the reference waveforms. Such a priori-chip-independent design has the advantages to eliminate the cumulative errors incurred from error propagation. When decoding, FORWARD always choose the longest segment to ensure sufficient samples for decoding. Furthermore, the recursive calibration design is approaching the real-time frequency offset and dynamically compensates the reference waveform. We implement FORWARD on USRP based testbed and evaluate its performance. Experimental results demonstrate that FORWARD reduces bit error rate by 4.96× and increases throughput 1.46∼2.8× compared with the state-of-the-art mZig.**

I. Introduction

ZigBee is a lower-power, low-rate and short-range wireless protocol based on IEEE 802.15.4 [1]. With the proliferation of *Internet of Things* (IoTs) [2], [3], [4], the ever-increasing deployments of ZigBee devices incur severe collision problem. Conventionally, ZigBee relies on *carrier sense multiple access with collision avoidance* (CSMA/CA) in the *medium access control* (MAC) layer to avoid these collisions. However, CSMA/CA introduces additional overhead of sensing and backoff, causing conspicuous throughput degradation.

Yifeng Cao* and **Zhe Wang*** are co-first authors.
Yifeng Cao†, Zhe Wang†, Linghe Kong†, Guihai Chen† and Jiadi Yu† are with Shanghai Jiao Tong University, China. (e-mail: yyshxyh2013, wang-zhe,linghe.kong@sjtu.edu.cn, gchen, jdyu@cs.sjtu.edu.cn)
Shaojie Tang‡ is with University of Texas at Dallas, USA. (e-mail: shaojie.tang@utdallas.edu)
Yingying Chen§ is with Rutgers University, USA. (e-mail: yingche@scarletmail.rutgers.edu)

Collision decomposition is an alternative approach to mitigate the overhead via resolving a multi-packet collision directly. For instance, ZigZag [5] utilizes the different time offsets between consecutive collided retransmission packets to decode sliding chunks from head to tail. However, the requirement of multiple retransmissions constrains the throughput improvement. mZig [6], the state-of-the-art decomposition approach leveraging the features of pulse shape, extrapolates samples in overlapped parts and resolves the collision for each chip in sequence. The bit error rates of these solutions are still high. Firstly, they exploit the decoded parts to extrapolate overlapping parts, thereby incur severe errors due to dependency between adjacent chunks or chips. Error in current chip has an influence on the following chips, which may cause consecutive errors, we name this pattern error propagation. This kind of consecutive errors leads to higher bit error rate when recovering original symbols from chip sequences in ZigBee even under the same chip error rate, for which existing works cannot achieve very low bit error rate. Secondly, the number of available samples may be scarce when alternatively decoding overlapped packets due to the randomness of time offsets, which causes scarce-sample error. Errors are prone to occur in the side that has few available samples for decoding. Lastly, they work based on the assumption that *frequency offset* (FO) remains constant in short time, which is usually impractical. Our field test shows that the FO deviation can achieve up to 9.2% in every 1000 packets, which degrades the decomposition accuracy dramatically.

The aforementioned drawbacks of existing solutions pose three major challenges to the design of practical multi-packet decomposition. Firstly, to mitigate the effect of dependent decoding, the decomposition cannot draw any more support from last chunk/chip/sample, which is the basis of all existing solutions. This requirement largely increases the difficulty since the 'independent' decomposition should be developed from a brand new direction. Secondly, to resolve the scarce-sample error, our method should be able to maintain sufficient samples for decoding under various time offsets. It's possible that the number of available samples is uneven when decoding multi-packet collisions, our design should be resistant to the randomness of time offsets. Lastly, considering the existence of FO deviation, achieving fine-grained FO calibration on overlapping signals is challenging. It is normal that FO deviation is well compensated in a one-transmitter one-receiver

communication since FO can be computed directly through the preamble at the header of ZigBee packet. However, the collided packets provide limited information in the header and the accurate FO cannot be obtained easily.

In this paper, we propose *Frequency-Offset Reference-WAveform Recursive Decomposition* (FORWARD), a novel ZigBee physical-layer design that enables the practical and accurate decomposition of multi-packet collision. FORWARD has three major components: *reference waveform generation, frequency offset calibration*, and *recursive operation* between calibration and generation. First, the core idea of the reference waveform is to generate the baseband waveforms of all possible overlapping cases as reference, with which the receiver compares the received signals to decompose the collision directly. Specifically, we use both the real and imaginary part of the signal sample to measure the similarity between received signals and reference waveforms. This design is entirely free of the priori-chip-decoding and makes sure that there always exists sufficient samples for decoding under whatever time offsets, which improves decoding accuracy significantly. Second, to tackle the dynamic FO deviation, we design a multi-dimension calibration mechanism, which includes *last FO* obtained in the last decomposed packet and *direct FO* extracted from non-overlapped parts of received packet. Third, the calibrated FO can help to generate a more accurate reference waveform, and the generated reference waveform can help to estimate a more accurate FO after decomposing the collided packet correctly. Hence, we design a recursive scheme to operate these two components iteratively to gradually approach the truth.

We implement FORWARD on a USRP N210 based testbed, in which multiple transmitters can concurrently send packets to one receiver. Based on this testbed, we conduct extensive experiments to evaluate FORWARD under various channel conditions. The results show that FORWARD outperforms the state-of-the-art mZig by 4.96× in bit error rate. In terms of the throughput, FORWARD achieves an improvement of 1.46∼2.8× compared with mZig.

The main contributions of this paper are summarized as follows:

- We propose FORWARD, a practical ZigBee physical design, to decompose a multi-packet collision via reference waveform and frequency offset calibration. We believe this design could forward the collision decomposition technology in ZigBee because FORWARD removes the error propagation, scarce-sample error and overcomes the dynamic FO deviation from existing approaches.
- We implement FORWARD on USRP and conduct extensive experiments to evaluate its performance. Experiment results show that FORWARD significantly outperforms existing collision decomposition approaches on the bit error rate and throughput.

The paper is organized as follows. Section II reviews background and presents our motivation. Design details are presented in Section III. Implementation and evaluation are introduced in Section IV and V, respectively. Section VI presents the related works and Section VII concludes this paper.

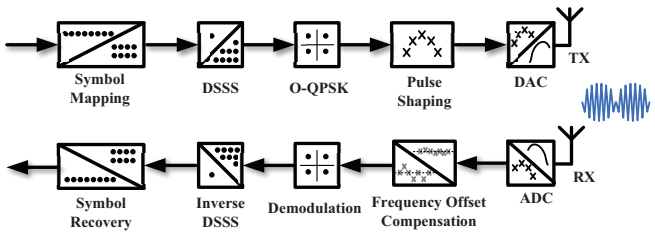


Fig. 1: The block diagram of standard ZigBee TX/RX.

II. Preliminary

In this section, we review the physical layer of standard ZigBee, state the multi-packet collision problem, and present the motivation of this work.

A. ZigBee Primer

Before introducing the proposed FORWARD, we firstly introduce the physical (PHY) layer of standard ZigBee. The TX receives the bitstream from the upper-layer and encapsulates them into a PHY protocol data unit (PPDU). PPDU contains three parts: the *synchronization header* (SHR), the *physical header* (PHR) and the payload. Fig. 1 shows how PPDU is sent out in the physical layer of TX, from step (i) to (v). (i) The TX firstly splits each octet in the byte stream to form a four-bit-symbol stream. (ii) These symbols are extended to a 32-chip pseudo-random sequence in *direct sequence spread spectrum* (DSSS). (iii) Then each chip is modulated with O-QPSK modulation: the even-index chips ($c_0c_2\cdots$) are mapped to I-phase while the odd-index chips ($c_1c_3\cdots$) are mapped to Q-phase. (iv) After that, the TX shapes pulses as follows: chip '1' is shaped to positive half sine and '0' takes the inverse. (v) The samples are then sent into *digital-to-analog converting* (DAC) and up-converting to the carrier frequency.

When the RX receives the packet, it down-converts the signal to the baseband and obtains the digital samples through *analog-to-digital conversion* (ADC). Since the samples are distorted by FO, the RX needs to compensate the FO in baseband before demodulation. After that, signals are demodulated into I/Q-phase chip sequence. Then, the chip sequence is transformed into symbols by inverse DSSS.

B. Multi-Packet Collision Problem

Conventional One-TX-One-RX Communication. Wireless signals can be represented as a stream of complex symbols in the baseband [7]. After over-sampling on these complex symbols in DAC, the TX is able to transmit packets in the form of analog signals over wireless links. In the receiver, the RX down-samples the signal and obtains a stream of complex samples. With the existence of channel fading and noises, the received signals are distorted in the amplitude and phase. Let the samples in the TX/RX be $\mathbf{x}[n]$/$\mathbf{y}[n]$ respectively, their relationship is approximated as:

$$\mathbf{y}[n] = \mathbf{H}\mathbf{x}[n] + \mathbf{w}[n], \tag{1}$$

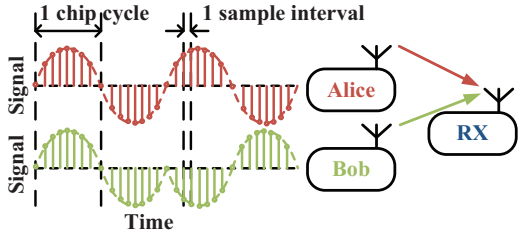


Fig. 2: Alice and Bob transmits packets to an RX concurrently.

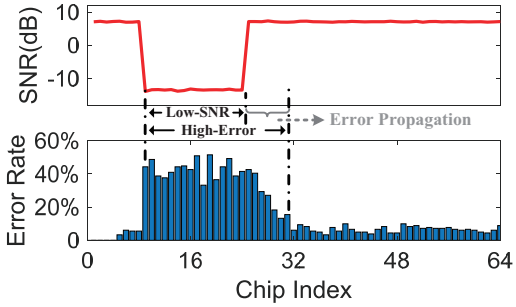


Fig. 3: When SNR drops sharply, mZig suffers consecutive chip errors beyond the low-SNR periods because of error propagation.

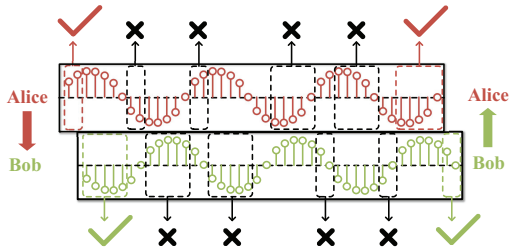


Fig. 4: When only few samples are available for dependent decoding, mZig is vulnerable to noise.

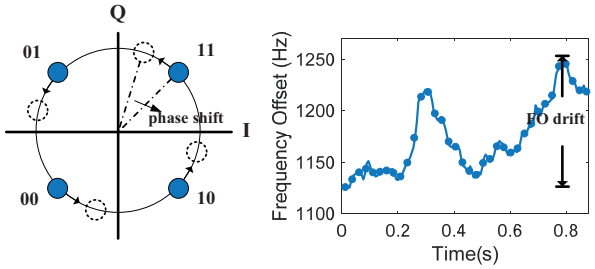


Fig. 5: Frequency offset does not change significantly, but may have deviation over a not long period.

where $\mathbf{H} = |\mathbf{H}|e^{j\phi}$ is the channel parameter and $\mathbf{w}[n]$ is the noise. However, Eq. 1 is a usual but simplified expression of received signal since FO exists widely in real links. The FO is caused by a hardware imperfection where the TX and RX are not exactly at the same center frequency. Taking the FO into consideration, the received signal is:

$$\mathbf{y}[n] = \mathbf{H}\mathbf{x}[n]e^{j2\pi\delta f Tn} + \mathbf{w}[n], \quad (2)$$

where T is one sample interval as shown in Fig. 2 and δf is the frequency offset. In the One-TX-One-RX scenario, FO can be computed from phase shift in the preamble. After FO compensation, the RX can decode the packet from the raw samples.

Multi-Packet Collision by Many-TX-One-RX Communication. We consider a simple collision case with two concurrent transmissions from Alice (A) and Bob (B) as shown in Fig. 2. With FO, the received signal can be expressed as:

$$\mathbf{y}[n] = \mathbf{H}_A\mathbf{x}_A[n]e^{2\pi\delta f_A Tn} + \mathbf{H}_B\mathbf{x}_B[n]e^{2\pi\delta f_B Tn} + \mathbf{w}[n]. \quad (3)$$

Conventional wireless RXs fail to resolve $\mathbf{x}_A$ and $\mathbf{x}_B$ with two unknown variables in only one equation. Thus, how to accurately decompose an m-packet collision remains a challenge in wireless area.

Note that the concurrent transmissions from multiple TXs may not be aligned at RX. They usually have a time offset, which has been validated by field test [6].

C. Motivation

Existing approaches proposed to resolve a ZigBee collision directly in the physical layer, e.g., ZigZag [5] and mZig [6]. The throughput of the state-of-the-art mZig [6] is still far from the theory upper limits in concurrent transmissions. With extensive field tests, we observe that previous techniques have still high BER and low throughput because of suffering three major drawbacks: (i) error propagation, (ii) scarce-sample error and (iii) inaccurate FO estimation.

Error propagation. Priori-chip dependent schemes suffers from error propagation. We take mZig as an example to illustrate this problem. We purposely manipulate the signal by adding the ultra noise to certain original samples. Note that scenarios with such in-packet ultra noise/interference are common in industry (e.g, interfered by bluetooth frequency hopping). We repeat the simulation to observe the overall error rate of each chip in the packet. As shown in Fig. 3, when SNR drops sharply, chip errors increase drastically not only during the low-SNR period, but also during the period right behind the low-SNR area. The successive chips after the low-SNR are also highly-likely to be decoded incorrectly. This is because RX tries to subtract the decoded data from the raw samples, thereby the initial deviation rolls repeatedly as the decomposition proceeds. ZigZag has the similar problem since the decoding of the current chunk also relies on the preceding chunk.

Scarce-sample error. The performance of previous works degrades sharply when the number of available samples for decoding is small. It is intuitive since the digital signal process in RX is not able to calibrate the effect of outlier samples when scarce samples are available. As shown in Fig. 4, when the RX decodes the collided packet from head to tail, all the segments with scarce samples are extremely likely to be incorrectly decoded. The reverse case (from tail to head) is all the same. Furthermore, in m-packet collisions, the occurrence of scarce-sample error increases drastically. The reason is that in a m-packet collision, the smallest overlapped segment in one chip is never larger than $\frac{\lambda}{m}$, where λ is the number of samples in

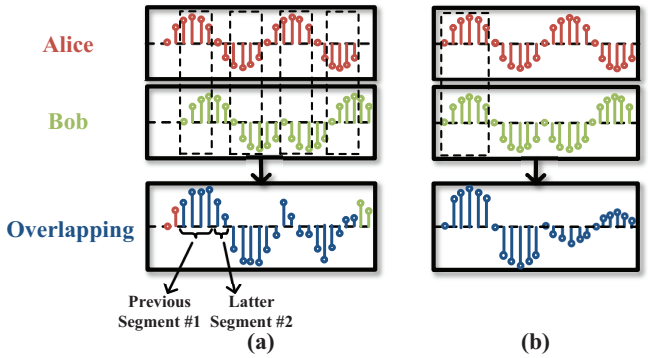


Fig. 6: Overlapping combinations in two-packet collision.

one chip. Previous works fail to approach this problem. When only few samples are available for decoding, existing works are vulnerable to noise.

Inaccurate FO estimation. Frequency offset widely exists in the wireless communication when the central frequency of the carrier in the TX is not exactly the same as that in RX. FO produces a linearly-changed phase offset in the constellation between the TX and the RX shown in Fig. 5, which requires to be compensated before decomposition. In One-TX-One-RX communication, FO is computed accurately from the preamble. But in Many-TX-One-RX communication, FO fails to be estimated because the preambles may be overlapped, which severely degrades the decomposition accuracy. Previous works utilize the initial FO for signal compensation. Although the FO does not change significantly, the initial coarse estimation is not sufficient for long-period FO compensation, i.e, the existing FO drift will incur a non-negligible error into decomposition, which has also been mentioned in [5], [6]. To validate this, we use multiple USRPs to collect large amounts of consecutive packets. We observe that deviation exists in FO as shown in Fig. 5. Although coarse calibration is exploited in existing works, it is far from enough in point of that the minor deviation may incur unaffordable BER.

These drawbacks motivate us to design a brand new technique to forward the collision decomposition.

III. Design of FORWARD

Considering the aforementioned problems, we propose the *Frequency-Offset Reference- WAveform Recursive Decomposition* (FORWARD) to achieve a highly accurate collision decomposition in ZigBee.

A. Design Overview

The core idea of FORWARD is to generate reference waveforms of the received signals based on standard half-sine pulse $\mathbf{x}$ and the channel parameter $\mathbf{H}$. Since each single chip in ZigBee is shaped as half-sine pulse with positive or negative amplitude, the potential overlapping combinations are finite. Taking Fig. 6a as an example, each chip of B overlaps with two consecutive chips of A, named the basic decoding chip, and we split each basic decoding chip into the previous segment and the latter segment. In a two-packet collision, there are

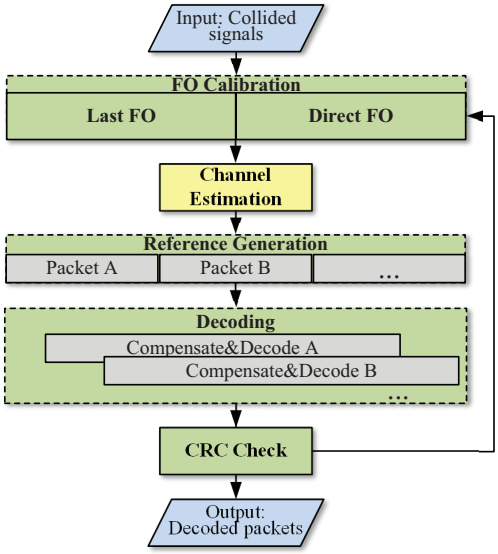


Fig. 7: Overview of FORWARD design.

only four overlapping combinations, 11, 00, 10, 01, for each segment. We propose an *always-the-longest* scheme which means that FORWARD always choose the segment with more samples as *basic decoding segment*(BDS). This ensures that FORWARD has a strong capability to calibrate outlier samples. A special case is when two packets are aligned chip-by-chip as shown in Fig. 6b. This case also has four overlapping combinations and we choose the entire chip as our BDS.

Note that even when packets are overlapped, the preambles can still be detected by correlation, which will help to obtain the sample offset among collided packets. According to the sample offset, the overlapped signals are divided into segments. By comparing reference waveforms with the BDSs chip-by-chip, we can find the *maximum-likelihood* (ML) reference waveform. As the chip combinations of reference waveforms are already known, we can extract chips from overlapped signals directly.

To generate the reference waveforms, we have to estimate a list of parameters, frequency offset, sample offset and channel parameter. FOs occur with hardware imperfection and is not stabilized across the transmissions, which will distort the signals and further destroy the comparison of waveforms. FORWARD proposes a novel scheme to estimate and compensate for the FOs before generating reference waveforms. The details are specified in Section III-B. On the basis of well estimated frequency offset, the channel parameters of multiple channels in a collision can be calculated accurately as shown in Section III-C. After that, the reference waveforms are generated and updated according to estimated parameters and the whole collided packets are decoded directly by comparing with reference waveforms. The detailed decoding procedures are shown in Section III-D.

In summary, to decompose multiple packets in a collision, FORWARD is divided into four procedures.

- Estimating and calibrating frequency offsets.
- Estimating channel parameters $\mathbf{H}$.

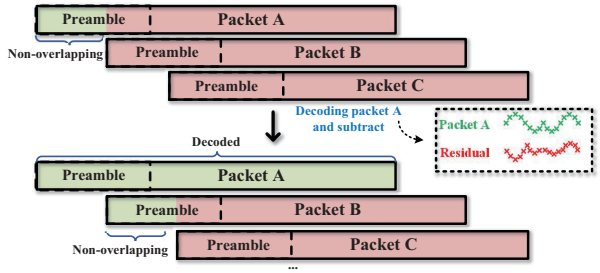


Fig. 8: Non-overlapping segments can be used to calibrate the frequency offset recursively.

TABLE I: Weight settings for frequency offset calibration.

Non-overlapping Length	w_1	w_2
>160 chips	0.3	**0.7**
≤160 chips	**0.9**	0.1

- Generating reference waveforms and decoding.
- Recursively decoding to enhance the reliability.

The diagram of FOWARD is shown in Fig. 7. Next, we specify each step in detail.

B. Frequency Offset Calibration

ZigZag [5] tackles FO deviation by 'calibrating while decoding', which uses decoded chunks to calibrate the phase shift caused by FO deviation. FORWARD calibrates FO in a more fine-grained scheme. In all, we leverage two accessible hints for effective calibration: (i) decomposed packets and (ii) non-overlapping segments.

- **Last FO**. Decomposed packets are validated through *cyclic redundancy check* (CRC). FORWARD re-computes and caches the frequency offset when a decoded packet passes the CRC validation. Since the $\mathbf{x}_A$, $\mathbf{x}_B$ in Eq. 3 are known a priori once correctly decoded, the FO can be obtained directly.
- **Direct FO**. Typically, collided packets are not aligned exactly from the first chip, thus there are often non-overlapping segments exposed to the RX The exposed parts can be used for the FO calibration. As shown in Fig. 8, when the non-overlapping preamble is sufficiently long, FORWARD can obtain the exact frequency offset. Then FORWARD can decode packet A from the collision by compensating A's frequency offset, with the method in Sec. III-D. After that, the packet A can be subtracted from the raw signal and thus part of packet B is non-overlapping. The above steps proceed recursively so that the FO for each packet can be calibrated.

Intuitively, Direct FO is closer to the Last FO when the non-overlapping length is long enough. In contrast, Last FO performs much better when the non-overlapping part is not sufficiently long. To make them robust to different overlapping cases, we simply combine them through weighted average to obtain fine-grained FOs. Weights of our design in some common scenarios are given in Table. I.

C. Channel Estimation

To generate reference waveforms for overlapped signal comparison, the RX exploits correlation, a generic packet synchronization method to estimate channel parameters.

Let p be the preamble length and N_t be the sample offset between two collided packets. The known preamble is denoted as $\mathbf{x}[n], 1 \leq n \leq p$ and $\mathbf{x}^*[n]$ is the conjugate of the known preamble. Mathematically, the correlation of B is computed as:

$$\Gamma(\Delta) = \sum_{n=1}^{p} \mathbf{x}^*[n]\mathbf{y}[n+\Delta], \tag{4}$$

where Δ slides through the received signal in one-sample stride. When $\Delta = N_t$, the correlation spikes because the preamble aligns with the beginning of B's packet. Since the preamble is independent of Alice's data and noise [5], the correlation at $\Delta = N_t$ after compensation is approximated with:

$$\begin{aligned}\Gamma(N_t) &= \sum_{n=1}^{p} \mathbf{x}^*[n]\mathbf{y}_B[n]e^{j2\pi\delta f_B T n} \\ &= \sum_{n=1}^{p} \mathbf{x}^*[n]\mathbf{H}_B\mathbf{x}[n]e^{j2\pi\delta f_B T n}e^{-j2\pi\delta f_B T n} \\ &= \mathbf{H}_B \sum_{n=1}^{p} |\mathbf{x}[n]|^2.\end{aligned} \tag{5}$$

Thus $\mathbf{H}_B$ can be estimated given the correlation and $\mathbf{x}[n]$.

D. Decomposed by Reference Waveform

After frequency offset and channel parameter are estimated, we try to decode the overlapping signals directly by comparing with all combinations of reference waveforms. In this part, we will show details about how to generate reference waveforms and how to decode the overlapped signals based on reference waveforms. Basically, decoding with reference waveforms outperforms existing approaches because of two reasons:

(i) The decoding scheme is priori-chip-independent. FORWARD generates reference waveforms for each BDS separately, thus the decoding of each chip is entirely free of the previous chips.

(ii) FORWARD can minimize the effect of random time offset. As illustrated in Section II, insufficient-sample errors occur when only few samples are available. To solve this problem, FORWARD proposes an *always-the-longest* scheme. In an overlapped-chip, FORWARD always uses the segment with more samples to generate the reference waveforms and decode. Given the sample number λ in one chip in an $\mathbf{m}-$packet collision, the samples exploited for decoding is always $> \frac{\lambda}{\mathbf{m}}$. The *always-the-longest* scheme is feasible and effective for two reasons. On the one hand, the overlapped chip can be decoded from segments with either more or less samples since both segments are parts of this chip. On the other hand, longer segments have stronger anti-noise and fault-tolerant ability. By comparing reference waveforms with more samples, the decoding result is more robust and accurate.

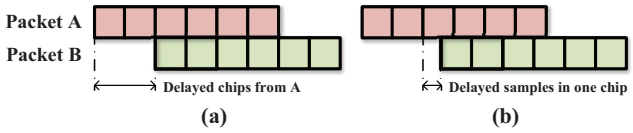


Fig. 9: Delayed chips and delayed samples in one chip.

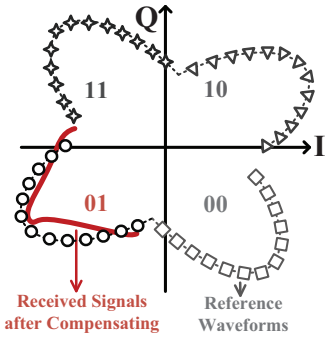


Fig. 10: An example of two-packet collision decomposition: 4 possible reference waveforms can be generated for a basic decoding segment (BDS). We plot constellations of samples in the complex plane based on received signals and generated reference waveforms. After FO calibration and compensating, the received signal can find the most similar reference waveform whose distance is the smallest, as shown in bottom-left, which is the correct '01'.

According to the sample offset between A and B, the delayed samples N_t can be divided into a number of delayed whole chips (C_t) and several delayed samples (S_t) in one chip. Denoting the number of samples in one chip as λ, N_t is the sum of $C_t\lambda$ and S_t. Generally, both C_t and S_t are not zeros as shown in Fig. 9b but sometimes the two packets are aligned chip-by-chip as shown in Fig. 9a.

In Fig. 6, we give an example of two-packet collision when S_t is not zero. The number of chips in packet A and B are denoted as L_A and L_B respectively. Considering the k-th overlapped chip of B, $1 \leq k \leq \min(L_B, L_A - C_t)$, the previous segment is overlapping with the $(k + C_t)$-th chip of A and the latter segment is overlapping with the $(k+C_t+1)$-th chip of A. The signal of the k-th chip of B after overlapping is:

$$\begin{aligned}\mathbf{y}[i] &= \mathbf{y}_A[i + N_t] + \mathbf{y}_B[i] + \mathbf{w}[i] \\ &= \mathbf{H}_A\mathbf{x}_A[i + N_t]e^{j2\pi\delta f_A T(i+N_t)} + \\ &\quad \mathbf{H}_B\mathbf{x}_B[i]e^{j2\pi\delta f_B T i} + \mathbf{w}[i],\end{aligned} \tag{6}$$

where $((k-1)\lambda+1 \leq i \leq k\lambda - S_t)$ for the previous segment and $(k\lambda - S_t + 1 \leq i \leq k\lambda)$ for the latter segment.

For a BDS, there are only four overlapping combinations when there exits S_t. When S_t is equal to 0, i.e., the two collided packet is aligned chip-by-chip as shown in Fig. 9a, we choose the whole overlapped chip as our BDS and there are also four overlapping combinations of chips. Thus, it is possible for us to generate all kinds of overlapping possibilities as reference waveforms. By comparing all kinds of reference waveforms, we can find the most similar one with the overlapped signal and extract the corresponding chips.

We use the decoding procedure of B's chip as an example in the following explanation for that the A's decoding is similar. For a specific BDS, we compensate frequency offset of A and B respectively to get compensated signal $\mathbf{y}^A$ and $\mathbf{y}^B$. After compensating, we get:

$$\mathbf{y}^B[i] = \mathbf{H}_A\mathbf{x}_A[i + N_t]e^{j2\pi T(\delta f_A(i+N_t) - \delta f_B i)} + \mathbf{H}_B\mathbf{x}_B[i]. \tag{7}$$

Considering $\mathbf{x}_A$ and $\mathbf{x}_B$ in Eq. 7, their waveforms are half-sine with positive or negative amplitudes according to the modulation and pulse shaping scheme of ZigBee standard. Denoting the sign of chips as $\mathbf{a}_A$ and $\mathbf{a}_B$, we can get possible $\mathbf{x}_B$ by:

$$\mathbf{x}_B[j] = \mathbf{a}_B \sin(\frac{(j-1)\pi}{\lambda}), \tag{8}$$

where $\mathbf{a}_B = \pm 1$ and $1 \leq j \leq \lambda$. Next we can recover the signal of A, B and generate the reference signals $\mathbf{r}^A$ and $\mathbf{r}^B$ as follows:

$$\begin{aligned}\mathbf{r}^B[i] =& \mathbf{H}_A\mathbf{a}_A \sin(\frac{(a-1)\pi}{\lambda})e^{j2\pi T(\delta f_A(i+N_t) - \delta f_B i)} + \\ & \mathbf{H}_B\mathbf{a}_B \sin(\frac{(b-1)\pi}{\lambda}),\end{aligned} \tag{9}$$

where $\mathbf{a}_A = \pm 1, \mathbf{a}_B = \pm 1, 1 \leq b \leq (\lambda - S_t), (S_t + 1) \leq a \leq \lambda$ when BDS is the previous segment, $(\lambda - S_t + 1) \leq b \leq \lambda, 1 \leq a \leq S_t$ when BDS is the latter segment of B's chip.

We generate four kinds of reference waveforms for a specific basic decoding segment of A or B based on Eq. 9. After that, we use distance to measure the similarity between the compensated signal and reference signal. The distance vector is a complex vector which is calculated as the difference between the compensated signal and reference signal. The distance $\mathbf{d}$ is defined as the L1 norm of the distance vector which can be formulated as:

$$\mathbf{d}[k] = \sum |(\mathbf{r}^B[i] - \mathbf{y}^B[i])|, \tag{10}$$

where the range of i is decided by the choice of basic decoding segment and k means distance with the *k-th* reference waveform. As shown in Fig. 10, FORWARD compares the received signal after compensation with four reference wavefroms, and select the one which has the least $DIST$ with the received signal.

The decoding procedure can be summarized as follows:

- Compensating frequency for A and B respectively.
- Generating reference waveforms of A and B.
- Calculating distance between compensated signals and reference waveforms.
- Choosing the most similar waveform as the final result and the chip values of reference waveforms are obtained.

It is worth noting that we compensate and decode the overlapped chips of A and B respectively. After compensating based on frequency offset of packet A or B, samples of A or B will be the major reference for decoding since the uncompensated signals of chip '0' and '1' may be very similar on account of frequency offset. High decoding accuracy can be achieved on the basis of A's and B's compensated signals when

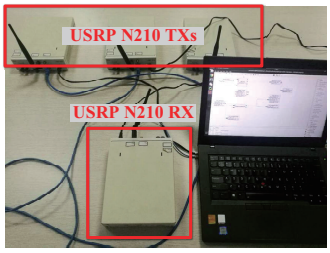


Fig. 11: The four-USRP testbed.

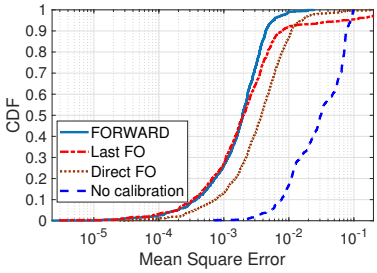


Fig. 12: Accuracy of frequency offset calibration.

decoding their overlapped chips. To further increase accuracy of reference waveforms, we will do FO calibration based on existing information after each packet is decoded.

Our method can be extended to multi-packet collision. For an **m**-packet collision, we can always find a BDS that are formed by the overlap of **m** chips according to sample offsets between them. There are at most $2^{\mathbf{m}}$ combinations of overlapping and we can generate all kinds of reference waveforms according to the estimated frequency offsets and channel parameters.

E. Recursive Operation

Although decomposing by referenced waveforms can resolve a collision correctly in most cases, we observe some packets fail CRC validation due to minor errors in the payload. To improve the accuracy, we design a recursive scheme to iteratively decompose the packet.

Consider a packet with minor errors after the first decomposition. Based on bits resolved correctly, FORWARD can re-compute the FO and make the decomposition repeatedly. Note that the influence of error bits is negligible because the correctly-resolved bits are in the majority. In each iteration, the FO will be calibrated closer to the ground truth, thus the reference waveform is also reformed, which is used to correct error bits in the payload.

IV. Implementation

We build the testbed of FORWARD with multiple USRP N210 devices and GNURadio, in which the standard ZigBee adopts the open source code [8]. The testbed is shown in Fig. 11. The daughterboard of our USRPs is XCVR2450, covering the frequency of 2.4-2.5GHz and 4.9-6.0GHz. In our experiments, FORWARD works on the 2.48GHz frequency band.

The TXs run the standard 802.15.4 PHY-layer to generate successive ZigBee packets and the CSMA/CA in MAC is disabled in order to allow concurrent transmission. In the RX, we modify the original decoding process and incorporate FORWARD to cope with collisions. Recall that FORWARD includes three main components: (i) FO calibration, (ii) reference waveform generation, and (iii) recursive operation. Note that historical FOs are available in the RX, cached for fast initialization, in our implementation.

In the FO calibration, FORWARD synthetically exploits the last FO and direct FO to improve the accuracy of FO estimation in diverse scenarios. For calibration, the RX stores 10 historical FOs for each TX with timestamps, TX IDs, and a flag indicating whether it is decoded without error. The RX firstly uses the FO of the last successfully-decoded packet in the cache to extract the preamble of the first packet. After that, the RX can re-calibrate the FO with the decoded preamble and extract the preamble of the second packet. This process proceeds recursively so that the FO of each overlapped packet can be calibrated. The recursively-calibrated FO is stored as the direct FO so that the RX can make a weighted-FO-calibration formula to update the cache in the RX.

The generation of reference waveforms proceeds as illustrated in Section III. However, two problems exist when $m \geq 3$. (i) Firstly, the increase of reference waveform combinations (2^m) makes the received signal less distinguished, degrading the accuracy in decoding. (ii) When decoding segment-by-segment, there are at most m chips skipped because FORWARD only select segment that has maximum samples of a chip for decomposition. To eliminate them, FORWARD exploits another segment, which has the second most samples in basic decoding chip, to decode as a complement.

In the recursive operation, the RX re-computes the FO with the decoded data to calibrate minor errors in the payload. Once the packet passes the CRC validation or repeated loops is larger than 3, the iteration terminates.

V. Experiment

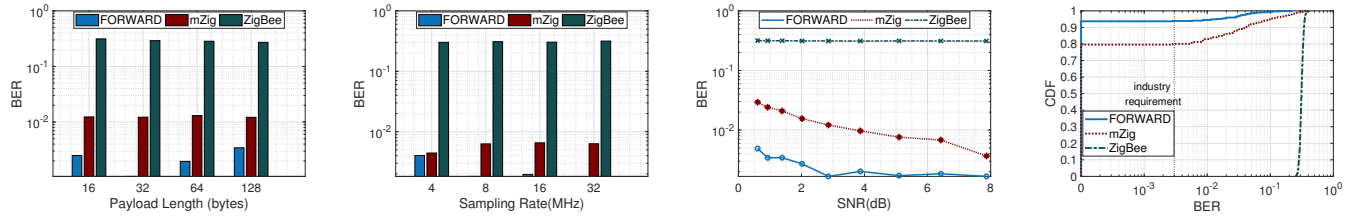
Based on our testbed, we conduct extensive experiments to evaluate the performance of FORWARD, which is compared with existing decomposition approaches.

A. Experiment Configuration

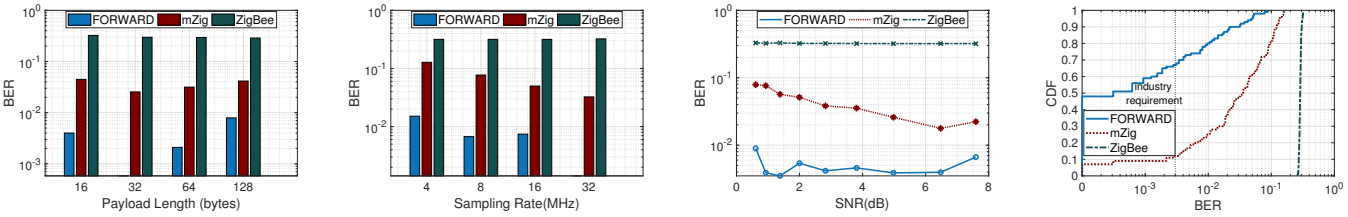
Our experiments are conducted in a $2 \times 3\text{m}^2$ laboratory environment with varying packet lengths, channel conditions, and *signal-to-noise ratios* (SNR). We set the sampling rate of USRP as 32MHz, where one half-sine pulse is shaped by 32 samples. In addition, both the TX and RX gains are set as 0.75, so the transmission power is nearly 2dB, which is sufficient in a laboratory environment. To eliminate the interference of WiFi, we select Channel 26 of ZigBee, which is a non-overlapping channel with WiFi. Without special illustration, the payload length is 128 bytes.

In our experiments, we compare the proposed FORWARD with two existing approaches:

- ZigBee [1]: is the 802.15.4 based standard ZigBee protocol. ZigBee adopts CSMA/CA in MAC layer to avoid



(a) Impact of payload length on BER when $m = 2$.
(b) Impact of sampling rate on BER when $m = 2$.
(c) Impact of SNR on BER when $m = 2$.
(d) Bit error distribution when $m = 2$.



(e) Impact of payload length on BER when $m = 3$.
(f) Impact of sampling rate on BER when $m = 3$.
(g) Impact of SNR on BER when $m = 3$.
(h) Bit error distribution when $m = 3$.

Fig. 13: Impacts of payload length, sampling rate, and signal-noise-ratio (SNR) on bit error rate (BER).

collisions. Retransmission mechanisms are used to guarantee the successful packet reception.

- mZig [6]: is the state-of-the-art collision decomposition method in ZigBee communications. Its key idea is leveraging the collision-free samples and known pulse shaping to estimate the next samples, and then decompose the collision chip by chip.

B. Frequency Offset

We firstly validate the effectiveness of FO calibration, which is one of the core designs in FORWARD. FORWARD calculates the FO from two dimensions to realize FO calibration over various conditions. Coarsely-estimated FO leads to decomposition failure and triggers retransmission, which will reduce the overall throughput dramatically. For comparison, we test the performance when each calibration approach is applied independently as the benchmarks. The *cumulative distribution function* (CDF) is plotted in Fig. 12, in which M1, M2 represent the case when only last FO or direct FO is used.

Results show that the multi-dimension calibration in FORWARD achieves the *mean square error* (MSE) less than 0.5% for 90% packets, which is an acceptable error variance for correlation, channel estimation, and reference waveform generation. In contrast, any individual calibration is not robust: (i) With no FO calibration, the offset would drift since the hardware imperfection, leading an accumulating error of more than 10%. (ii) The last FO performs the best when the FO deviation is slow and smooth, but it degrades steeply when a channel 'jitter' occurs. (iii) The direct FO can achieve high accuracy if non-overlapped preamble is sufficient enough. However, its performance drops significantly as the non-overlapped length decreases.

C. Bit Error Rate

In this subsection, we evaluate the bit error rate (BER) of FORWARD, verifying its feasibility across extensive environment settings. In the experiment, we control m TXs ($m \leq 3$) to transmit concurrently to one RX and measure the BER, by comparing the ground-truth packets in the TXs with the decomposed results in RX.

BER is tightly related to packet loss rate, which directly influences the throughput. Higher BER means the frequent retransmission, which introduces dramatic overhead. The industrial standard for ZigBee requires the BER should be $< 3 \times 10^{-3}$. Hence, we also use BER as the metric to testify whether FORWARD is an industry-applicable design. Typically, BER is usually lower than CER due to ZigBee's DSSS allows minor chip errors when recovering raw samples to the correct symbols.

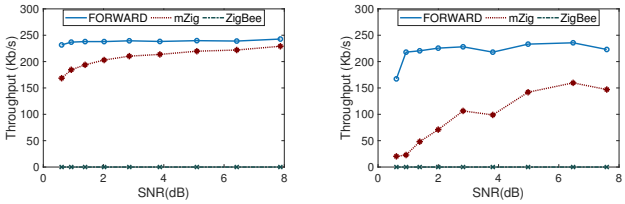
We compare the performance of FORWARD with mZig and ZigBee. mZig extrapolates the next samples in a chip with the decoded samples in the same chip, and then subtracts them from the overlapping signal. The dependency on decoded parts will cause consecutive errors when error happens in the current chip. The performance of mZig also relates to the time offsets which determine how many available samples when decomposing a specific packet from the overlapped signal. We also test the BER when ZigBee attempts to decode a multi-packet collision directly without any modification. We amend environmental settings include the payload length, the sampling rate and SNR to observe the performance of FORWARD in different settings.

Impact of payload length. In this experiment, we evaluate BER with the varying payload length of 16, 32, 64 and 128 bytes. Note that the payload length is not allowed to exceed 128 bytes according to the 802.15.4 protocol. The results are shown in Fig. 13a and 13e. In a two- or three-packet collision,

FORWARD improves the average BER by $4.96\times$ compared with mZig. In a two-packet collision, mZig estimates the mean amplitude α, β for each packet, and slides from head to tail to extrapolate the overlapped samples based on the decoded samples. Thus consecutive errors occur under burst noise. Besides, when the number of delayed samples in one chip is close to 0 or λ, the number of available samples are imbalance among the collided packets, leading to large amount of errors in the side has less samples. FORWARD, however, eliminates the error accumulation and dependent error problems by performing context-free BDS comparison between the received and reference waveforms. FORWARD will always choose BDS that has maximum samples as decoding reference, which removes the effect of imbalanced samples distribution and improve robustness simultaneously. Additionally, while mZig tracks the initial phase shift and compensates the FO chip-by-chip, it is not sufficient in the existence of FO variance. Comparatively, FORWARD is more robust in performance. There is also a slight degradation when the payload is too short or long. We analyze that the small payload is more noise-sensitive when bit errors occur. For long-payload packets, the channel parameters are slightly varying in a packet. Thus the estimation error in $\mathbf{H}$ increases since FORWARD uses the channel parameters calculated from the preamble all along. Actually, FORWARD is totally free of consecutive error and dependent error. From the figures, FORWARD keeps the BER below 3×10^{-3} even in the worst case.

Impact of sampling rate. The sampling rate determines the samples taken in one half-sine waveform, i.e., a chip period. We test FORWARD in the sampling rate of 4MHz, 8MHz, 16MHz and 32MHz. Although USRP N210 supports the sampling rate up to 61.44MHz, we do not test any $>$ 32MHz implementation, where the computational delay will be non-negligible. Results in Fig. 13b and 13f show that the decoding performance is slightly influenced by the sampling rate when m increases. It is intuitive because the RX needs to select the ML one from more reference waveforms, in which case signals with sufficient samples in one pulse are more distinguished. In addition, to apply FORWARD when $m \geq 3$, the sampling rate should not be lower than 32MHz.

Impact of SNR. As the real-link noise is out of control, we emulate different SNR by mixing the original signal with a noise generator in the GNURadio. We control the SNR ranges from 0dB to 8dB to evaluate FORWARD's robustness to noise. From the Fig. 13c and 13g, FORWARD outperforms mZig over all the SNRs in bit error rate, up to $5.97\times$. Although FORWARD degrades slightly in performance when SNR decreases, the BER is controlled in 3×10^{-3} when SNR$\geq$ 0 dB. It can be observed that FORWARD is more robust to the effect of noise since we always choose basic decoding segment which has largest number of samples. However, there always exists a packet which has less available samples when dependently decoding in mZig due to the random distribution of sample offsets. Thus, mZig is vulnerable to the effect of noise while our method retains relatively stable even under low SNRs.



(a) Throughput comparison of 3 decomposition methods when $m = 2$.

(b) Throughput comparison of 3 decomposition methods when $m = 3$.

Fig. 14: Evaluation of throughput.

Error distribution. To make the variance between FORWARD and mZig clear, we plot the CDF to show the BER distribution of all packets. In the experiment, the SNR is set to about 2.8dB and the sample rate is 32MHz. When $m = 2$, about 87% packets have a BER lower than 3×10^{-3} while mZig is only 73%. When $m = 3$, the gap between two methods is even clearer. With FORWARD, 70% packets sustain BER below 3×10^{-3} while only 10% packets with mZig can still have such low BER. The performance of mZig degrades a lot in three-packet collision decoding under low SNRs. The number of available samples for chip estimation in one-chip is prone to be insufficient when $m = 3$. Thus, mZig is sensitive to the effect of noise while the performance of FORWARD keeps relatively stable even under strong noise.

D. Throughput

We take trace-driven simulation for throughput evaluation. Specifically, we collect 1000 packets through USRP Tx/Rx and then let them pass through the FORWARD module. FORWARD determines whether a packet is effective by comparing BER with 3×10^{-3}. Each data packet has a 128-byte payload. The SNR is controlled ranging from 0dB to 8dB and the RX samples at a rate of 32MHz. The round-trip time (RTT) for single TX-RX transmission and time delay for *acknowledgement* is computed through extensive packets in the USRP testbed, which is 420.5μs and 82.84μs.

To collect collided packets, CSMA/CA in standard ZigBee is disabled. Standard ZigBee only receives the collided signals, however, cannot decode the collided packets correctly. As a result, its throughput is near 0. For FORWARD and mZig, they are able to decode the collisions directly and we compare their throughput by calculating the overall packet reception ratio in concurrent transmissions.

Results are shown in Fig. 14a, 14b. The metric is the average throughout for each TX. When $m = 2$, FORWARD achieves an improvement up to 43.8% compared to mZig. Even when the SNR is low, FORWARD sustains the 90.4% performance of the theoretical throughput upper bound (254.3KB/s). When $m = 3$, FORWARD achieves a better performance, which is up to $8\sim10\times$ compared to mZig. In the general industrial environment where the SNR is 2$\sim$8 dB, FORWARD achieves the $1.46\sim2.8\times$ performance compared with mZig.

FORWARD increases the overall throughput of concurrent transmissions significantly, especially in three-packet collision scenarios. It can be well explained that FORWARD is almost impervious to the randomness of sample offsets since we have

at least $\frac{\lambda}{3}$ samples for decoding in the worst cases. While it is of high probability that there are insufficient samples when dependently decoding packet A, B or C in mZig, the performance of mZig is unstable and more susceptible to the effect of noise.

This result demonstrates that FORWARD is a generical, robust and effective applicable approach to multi-packet decomposition.

VI. Related Work

Collisions cause severe throughput degradation, especially in industrial wireless communication with frequent interference. Extensive researches have been proposed to tackle this classic problem. Generally, they can be classified into two categories based on layers: i) MAC-layer based collision avoidance and (ii) PHY-layer based collision resolution.

MAC-layer based collision avoidance. Early work advocates the modification of MAC-layer protocols to bypass collisions. Studies in [9], [10], [11] apply CSMA-based random access to multi-hop networks, which reduce collisions by distributing random backoff time with fairness to each node. However, these protocols fail when there are hidden terminals. Moreover, it is time-consuming for the RX to notify each TX about its backoff time, which also degrades the overall throughput. An alternative method is the Request to Send/Clear to Send (RTS/CTS) protocol [12]. In RTS/CTS, TX/RX uses a hand-shaking protocol to ensure that the transmission is not interfered by hidden terminals. But the additional overhead is introduced by the hand-shaking.

Since MAC-layer based methods cannot obtain fine-grained physical information, the extra overhead is almost unavoidable. Unlike MAC-layer based methods, FORWARD decomposes a packet collision efficiently by operating on the samples in the physical layer.

PHY-layer based collision resolution. Physical-layer based methods exploits physical features (e.g., signal strength) for collision resolution. Successive interference cancellation (SIC) [13], [14] enables the RX to decode multiple packets concurrently according to different power levels. SIC firstly decodes the strongest signal and then decode others in the amplitude order. SIC requires prior scheduling and known uses, which is hard to obtain in the distributed network system. Moreover, in densely-deployed wireless networks, SIC fails since the differences among amplitudes are not distinguishable.

Different from SIC, [15], [16], [17] encourage certain TXs with prior information to support the on-going transmission. With the prior knowledge of interference, the RX can resolve collisions by subtraction operation in baseband signal. [18] extends the network coding to full-duplex scenarios to further improve the decoding efficiency. However, they still fail to resolve the collisions with hidden terminals, where information of TXs cannot be obtained in advance. Constructive interference [19], [20], [21] leverages multiple transmissions of the same packet to resolve a collision, but it cannot be applied when multiple senders transmit different packets.

State-of-the-art approaches focus on developing a generic method to decompose packets independent of signal strength and prior knowledge. ZigZag [5] exploits different time offset of two successive retransmissions to resolve a collision chunk by chunk. ZigZag is robust to hidden terminals but the retransmission incurs additional overhead. To remove the retransmission delay, mZig [6] decomposes multiple packets on real-time chip by chip. Meanwhile, mZig achieves higher throughput when a collision involves m packets ($m > 2$). However, our experiments show that mZig exists the consecutive error and dependent error problem. CrossZig [22] uses an adaptive forward error (FEC) coding and a packet-merging policy to recovery packets adaptively from a cross-technology collision. SoNIC [23] proposes a classification method for efficient corrupted bits detection and recover packets correspondingly. Other physical-layer solutions, such as [24], [25], exploit special physical hints to extrapolate and decode the collided packet but fail to sustain their performance in all scenarios.

Compared with existing approaches, FORWARD also operates in the physical layer to resolve a collision directly. Similar to ZigZag and mZig, it requires no prior knowledge and works on real-time. Furthermore, FORWARD introduces nearly no transmission overhead and decodes each overlapped chip independently and robustly. FORWARD is also orthometric to most prior works, flexible to co-work with other methods for different wireless scenarios.

VII. Conclusion

This paper proposes FORWARD, a novel physical layer design to forward the collision decomposition technology in ZigBee communications. On one hand, FORWARD mitigates the drawbacks of dependent decoding through the reference waveform design. On the other hand, FORWARD compensates the FOs for concurrent transmissions via a multi-dimension FO estimation design. Benefiting from these two core designs and their recursive operation, FORWARD can achieve a highly accurate decomposition for multi-packet collision. We implement FORWARD on USRP-based testbed. Performance results show that FORWARD reduces BER in collision decomposition by 4.96$\times$ compared to the state-of-the-art mZig. We also perform throughput simulations using trace-driven data, and results show that FORWARD outperforms mZig by 1.46$\sim$2.8$\times$ in general environments.

The future work of lies in two aspects. Firstly, FORWARD is an orthogonal approach with some existing collision-resolving solutions. Hence, finding a plausible method to combine FORWARD with others is still to-be-explored. Secondly, the core idea of FORWARD is theoretically feasible in other PSK- or FSK- based wireless networks and we believe it can be further extended for more widespread applications.

VIII. Acknowledgment

This work was supported in part by the National Key R&D Program of China 2018YFB1004703, in part by NSFC grant 61972253, 61672349, 61672353, and in part by CNS1814590 and CCF1909963 from National Science Foundation in US.

REFERENCES

[1] "ZigBee Standard," http://standards.ieee.org/getieee802/download/802.15.4-2011.pdf/.

[2] H. Y. Tung, K. F. Tsang, and et al, "The generic design of a high-traffic advanced metering infrastructure using zigbee," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 836–844, 2014.

[3] S. Wang, S. M. Kim, and et al, "Corlayer: A transparent link correlation layer for energy efficient broadcast," *IEEE/ACM Transactions on Networking*, vol. 23, no. 6, pp. 1970–1983, 2015.

[4] P. Yi, A. Iwayemi, and et al, "Developing ZigBee deployment guideline under wifi interference for smart grid applications," *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 110–120, 2011.

[5] S. Gollakota and D. Katabi, "ZigZag decoding: combating hidden terminals in wireless networks," *ACM SIGCOMM*, 2008.

[6] L. Kong and X. Liu, "mZig: Enabling multi-packet reception in zigbee," *ACM MOBICOM*, 2015.

[7] D. Tse and P. Viswanath, "Fundamentals of wireless communication," *Cambridge University Press*, 2005.

[8] "ZigBee Implementation on GNURadio," https://github.com/bastibl/gr-ieee802-15-4/.

[9] M. Garetto, T. Salonidis, and et al, "Modeling per-flow throughput and capturing starvation in csma multi-hop wireless networks," *IEEE/ACM Transactions on Networking*, vol. 16, no. 4, pp. 864–877, 2008.

[10] J. Kwak, C. H. Lee, and et al, "A high-order markov-chain-based scheduling algorithm for low delay in csma networks," *IEEE/ACM Transactions on Networking*, vol. 24, no. 4, pp. 2278–2290, 2016.

[11] F. Hermans, O. Rensfelt, and et al, "Collision-minimizing CSMA and its applications to wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, 2004.

[12] K. Xu, M. Gerla, and et al, "Effectiveness of rts/cts handshake in ieee 802.11 based ad hoc networks," *Ad Hoc Networks*, vol. 1, no. 1, pp. 107–123, 2003.

[13] S. Sen, N. Santhapuri, and et al, "Successive interference cancellation: Carving out MAC layer opportunities," *IEEE Transactions on Mobile Computing*, vol. 12, no. 2, pp. 346–357, 2013.

[14] D. Halperin, T. Anderson, and D. Wetherall, "Taking the sting out of carrier sense: interference cancellation for wireless lans," *ACM MOBICOM*, 2008.

[15] S. Katti, H. Rahul, and et al, "XORs in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 497–510, 2008.

[16] Q. Xiang, H. Zhang, and et al, "On optimal diversity in network-coding-based routing in wireless networks," *IEEE INFOCOM*, 2015.

[17] S. Gollakota, D. P. Samuel, and D. Katabi, "Interference alignment and cancellation," *ACM SIGCOMM Computer Communication Review*, 2009.

[18] M. Jain, J. I. Choi, and et al, "Practical, real-time, full duplex wireless," *ACM MOBICOM*, 2011.

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

[20] F. Ferrari, M. Zimmerling, and et al, "Efficient network flooding and time synchronization with glossy," *ACM/IEEE IPSN*, 2011.

[21] Y. Wang, Y. Liu, and et al, "Disco: Improving packet delivery via deliberate synchronized constructive interference," *IEEE Transactions on Parallel and Distributed Systems,*, vol. 26, no. 3, pp. 713–723, 2015.

[22] A. Hithnawi, S. Li, and et al., "Crosszig: combating cross-technology interference in low-power wireless networks," *ACM/IEEE IPSN*, 2016.

[23] F. Hermans, O. Rensfelt, and et al., "SoNIC: classifying interference in 802.15. 4 sensor networks," *ACM/IEEE IPSN*, 2013.

[24] D. Ismail, M. Rahman, and A. Saifullah, "Rnr: Reverse and replace decoding for collision recovery in wireless sensor networks," *IEEE SECON*, 2017.

[25] S. Saha and M. C. Chan, "Design and application of a many-to-one communication protocol," *IEEE INFOCOM*, 2017.