BIC-LoRa: Bits in Chirp Shapes to Boost Throughput in LoRa

Geonhee Lee*, Eunjeong Park*, Mingyu Park‡, Jeongyeup Paek‡, Saewoong Bahk*

*Department of ECE and INMC, Seoul National University, Seoul, Republic of Korea, 08826
‡Department of Computer Science and Engineering, Chung-Ang University, Seoul, Republic of Korea, 06974
{ghlee,ejpark}@netlab.snu.ac.kr,{hello0922,jpaek}@cau.ac.kr,sbahk@snu.ac.kr

ABSTRACT

LoRa is a low-power long-range radio technology for wide-area IoT connectivity with exceptional receiver sensitivity thanks to its chirp spread spectrum (CSS) modulation. However, insufficient data rate has always been an Achilles' heel of LoRa. This paper proposes a novel PHY-layer design, *BIC-LoRa*, that leverages non-linear chirp shapes as pictograph to enhance LoRa's data rate. It employs multiple non-linear chirps for modulation whose shapes encode additional bits to boost data rate while maintaining resilience to low SINR scenarios. To the best of our knowledge, this is the first attempt to enhance the data rate of LoRa by embedding bits in the chirp shapes. Furthermore, *BIC-LoRa* fully leverages the characteristics of non-linear chirps, allowing it to deal with packet collisions and increase network throughput. We implement *BIC-LoRa* in GNURadio and MATLAB, and compare its performance against CurveALOHA and standard LoRaWAN through real experiments on USRP B210 software-defined radios. Evaluation results demonstrate that *BIC-LoRa* achieves up to 29.4% improvement in data rate for a single link and 32% improvement in overall network throughput while retaining the low SNR robustness of LoRa in real-world scenarios.

KEYWORDS

LoRa, Chirp Spread Spectrum (CSS), PHY-layer, Non-linear chirp

1 INTRODUCTION

LoRa[1] is an open-source low-power long-range radio technology operating at unlicensed ISM sub-GHz bands for wide-area Internet of Things (IoT) connectivity [21]. The primary attribute of LoRa is the exceptional receiver sensitivity which allows extensive communication range (2-5 km in urban areas and 15 km in suburban areas) [2] with low power. This enables large-scale IoT applications such as smart agriculture [31], smart city [5, 26], smart industrial control [14, 28], or advanced metering infrastructure [7, 38] which rely on long-distance connectivity for periodic sensor data collection. With ongoing technological advancements, the adoption of LoRa communication for IoT applications is on the rise.

However, LoRa achieves its ultra-low receiver sensitivity using the *chirp spread spectrum (CSS)* modulation which inherently sacrifices data rate [24] for long range. Furthermore, since LoRa functions without explicit coordination due to its stringent power budget, its bandwidth and frame size are typically restricted by regional regulations. For example, Table 1 lists the PHY bit rates and the maximum payload sizes according to the LoRa configurations

---

[1]LoRa Alliance®, https://lora-alliance.org/

Mode	Config.	PHY rate	Max. payload
0	SF12 / 125 kHz	250 bps	51 byte
1	SF11 / 125 kHz	440 bps	51 byte
2	SF10 / 125 kHz	980 bps	51 byte
3	SF9 / 125 kHz	1760 bps	115 byte
4	SF8 / 125 kHz	3125 bps	222 byte
5	SF7 / 125 kHz	5470 bps	222 byte
6	SF7 / 250 kHz	11000 bps	222 byte

Table 1: LoRa configurations in EU868 regional specification [3].

in EU868 regional specifications [3]. These bit rates are significantly lower than other popular wireless technologies such as Wi-Fi, Bluetooth, Zigbee, or even NB-IoT. For these reasons, insufficient data rate has always been an Achilles' heel of LoRa.

Numerous prior work have contributed to enhancing the usability of LoRa. However, most studies have mainly focused on improving the 'overall network throughput' via collision resolution [6, 23, 29, 34, 42], channel activity detection (CAD) [12, 19] or MAC layer design [8, 15, 27, 33], but not the 'individual link data rate'. In other words, although these proposals provide better channel utilization by resolving collisions, maximum payload size and the throughput of a single link is still limited. There have been attempts to increase individual link throughput through PHY-layer modifications [4, 11, 25, 41], but these approaches either have only marginal gains or require a sacrifice in communication range. In this work, we aim to boost the individual link data rate of LoRa while retaining the communication range as well as leveraging the collision resolution capability of prior work.

To this end, we propose a novel PHY-layer design, *BIC-LoRa*, that uses non-linear chirp shapes as pictographs to enhance LoRa's data rate. Instead of the standard linear chirps, *BIC-LoRa* employs multiple types of non-linear chirps for modulation whose shapes encode additional bits to boost data rate while maintaining resilience to low SINR scenarios. Specifically, N-kinds of chirps can represent $\log_2 N$ additional bits per chirp based on its shape. Since the standard CSS can embed as many data bits as the spreading factor (SF) per chirp, *BIC-LoRa* can improve throughput by $\frac{\log_2 N}{SF}$%. At a high-level, the modulation and demodulation process of *BIC-LoRa* is as follows;

(1) Transmitter divides and groups the data to be sent by 'SF + $\log_2 N$' bits per group.
(2) Each group is modulated to a chirp with its offset given by SF data bits (same as the standard LoRa) and a specific non-linear shape determined by the last $\log_2 N$ bits of the data.
(3) Receiver multiplies the received signal with an appropriate counterpart down-chirp per symbol via demodulation process.
(4) The matched chirp shape is translated to $\log_2 N$ data bits, and are added to the demodulated SF data bits to deliver '$SF+\log_2 N$' bits per chirp, $\log_2 N$ more bits per chirp than standard LoRa.
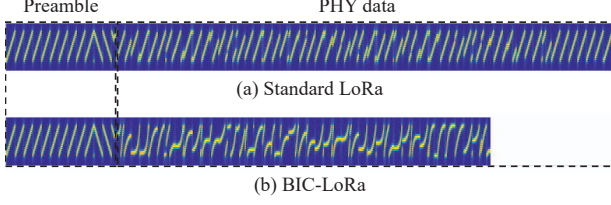
Figure 1: Modulated packets for standard LoRa and *BIC-LoRa*.

Figure 2: Basics of chirp spread spectrum (CSS).

Figure 3: Capture Effect

To the best of our knowledge, this is the first attempt to enhance the data rate of LoRa by embedding bits in the chirp shapes.

BIC-LoRa can express $\log_2 N$ more bits for each chirp, resulting in several advantages. **Energy consumption and channel occupancy.** *BIC-LoRa* requires *less number of symbols* with the same modulation process (en-chirp) to transmit an identical frame as shown in Fig. 1. **Robustness to interference.** Recent studies have shown that de-chirped signals spread out between unmatched chirps by quasi-orthogonality called *energy scattering effect* [22, 24, 43]. By using various kinds of chirp shapes to embed additional data, *BIC-LoRa* naturally leverages these characteristics and resolves collision (and interference) without extra MAC design. **Throughput and capacity.** Since *BIC-LoRa* can embed more data bits with an identical number of symbols, it not only improves the data rate but also increases the number of data bytes in a frame (for a given dwell time limit regulation).

We implement *BIC-LoRa* in GNURadio and MATLAB for modulation and demodulation respectively. The prototype is installed on USRP B210 software-defined radio (SDR) devices, and compared against CurveALOHA [22] and standard LoRaWAN through real experiments. Evaluation results demonstrate that *BIC-LoRa* increases the overall network throughput by up to 32% while boosting the throughput of individual node by 29.4%. The contributions of *BIC-LoRa* can be summarized as follows:

- We propose *BIC-LoRa*, an innovative approach leveraging the concept of non-linear chirps as pictographs to expand the data-carrying capacity of chirps with minimal overhead.
- We carefully design *BIC-LoRa* to enhance the link data rate of LoRa without sacrificing the robustness against noise and interference as well as collisions.
- We implement *BIC-LoRa* on a SDR platform, and conduct real experiments to demonstrate that *BIC-LoRa* improves both the single link and overall network throughput significantly.

The remainder of this paper is organized as follows. We provide background on CSS and non-linear chirps in Section 2, and Section 3 discusses previous studies inspiring *BIC-LoRa*. The design of *BIC-LoRa* is presented in Section 4, and Section 5 describes how we implement and deploy *BIC-LoRa* for proof-of-concept. Evaluation results are presented in Section 6, and we discuss challenges and limitations in Section 7 before we conclude in Section 8.

2 BACKGROUND

This section provides a primer of chirp spread spectrum (CSS), a modulation technique of LoRa that this work mainly targets. Then, we introduce the non-linear chirps that inspire *BIC-LoRa* to achieve data rate enhancement and collision resolution while retaining the robustness to low SNR scenarios.
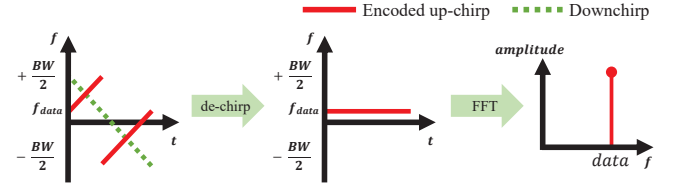
2.1 Chirp Spread Spectrum

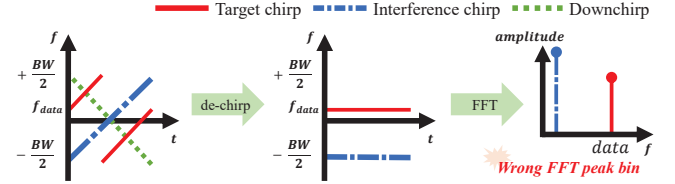**Overview.** Fig. 2 illustrates an example modulation and demodulation processes of standard LoRa. A LoRa transmitter encodes data into an *up-chirp* (solid red line) by shifting the initial frequency of a *base up-chirp*. A receiver applies a de-chirp that multiplies the received signal with a *base down-chirp* (dotted green line). Then, the receiver extracts the data symbol from the index of the frequency in the de-chirped signal using the Fast Fourier Transform (FFT).

Modulation. CSS modulates data using an *up-chirp* which increases frequency linearly. There are four main parameters for CSS modulation: spreading factor (SF), bandwidth (BW), center frequency (f_c), and a data value to be modulated (M). The modulation starts from the *base up-chirp* that changes frequency from $f_c - \frac{BW}{2}$ to $f_c + \frac{BW}{2}$ over time t, based on a linear function $f(t) = kt$, where k is a coefficient to fit into the range of symbol time and BW. Together with its counterpart demodulation scheme, this allows the impact of noise to be scattered across frequency and time domains, resulting in strong noise resilience. The base up-chirp $u(t)$ corresponding to $f(t)$ can be expressed as follows;

$$u(t) = e^{j2\pi(f_c + f(t))t}, t \in [0, \frac{2^{SF}}{BW}], f(t) \in [-\frac{BW}{2}, \frac{BW}{2}] \quad (1)$$

CSS embeds data by shifting this base up-chirp along the frequency axis, consistent with the value of the data. The amount of expressible data bits in a chirp is same as SF value, thus the value M ranges from 0 to 2^{SF}-1. Given a frequency resource of BW, expressing data M consequently necessitates a shift of $\frac{M}{2^{SF}} \times BW$. Therefore, an up-chirp including data M is derived as follows;

$$m(t) = e^{j2\pi\left(f_c + f(t) + \frac{M}{2^{SF}} \times BW\right)t}, M \in [0, 2^{SF}) \quad (2)$$

During this process, to stay within the given bandwidth, a 'fold' occurs if the frequency value exceeds $f_c + \frac{BW}{2}$, returning to ascend from $f_c - \frac{BW}{2}$ as shown in Fig. 2. With this modulation strategy, LoRa can place SF *bits* in a chirp. In addition, CSS applies Hamming code for error correction. Therefore the proportion of bits transmitted by given coding rate (CR) is given as $\frac{CR}{4+CR}$. As a result, physical bit
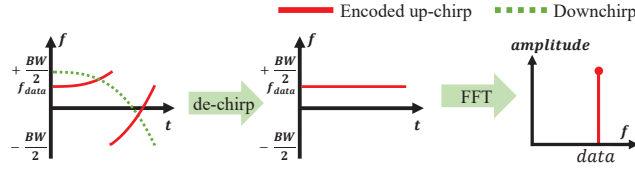
Figure 4: Modulation and demodulation with non-linear chirp.

Figure 5: Energy scattering effect in time difference scenario.

rate is calculated as follows;

$$\text{Data rate} = SF \times \frac{BW}{2^{SF}} \times \frac{CR}{4+CR} \text{ (bps)} \qquad (3)$$

Demodulation. Similar to the modulation process, CSS employs *base down-chirp* to demodulate received signal. The base down-chirp can be obtained by conjugating the base up-chirp as follows;

$$d(t) = conj(e^{j2\pi f(t)t}) = \overline{e^{j2\pi f(t)t}} = e^{-j2\pi f(t)t} \qquad (4)$$

The complex conjugate eliminates the increasing frequency component and retains only the shifted frequency component of the input symbol by multiplying it with the received input chirp as follows;

$$m(t) \times d(t) = e^{j2\pi\left(f_c+\frac{M}{2^{SF}}\times BW\right)t} \qquad (5)$$

Since the SF, BW, and f_c are deterministic, the LoRa receiver can readily extract data M from the extent of the frequency shift using FFT even with SNR below 0 dB. [44]

Limitations. Besides low data rate, one of the critical challenges in LoRa from a network perspective is the *capture effect* when multiple transmissions occur at the same time [30]. The capture effect refers to a phenomenon where a receiver decodes only the stronger signal from collided packets. It sometimes helps a receiver to successfully decode data from a collision, but sometimes the stronger may not be the intended target signal. Fig. 3 displays an example scenario where capture effect occurs. If an interfering chirp has higher power than the target chirp, the receiver may incorrectly receive wrong data, resulting in significant loss of performance.

2.2 Non-linear Chirp

To address the collision problem, Li *et al.* have proposed Curving-gLoRa which utilizes non-linear chirps [24]. With non-linear chirps, the linear function $f(t)$ is replaced with a curved function $f_i'(t)$ as follows;

$$f_i'(t) = k_i t^i, t \in [0, \frac{2^{SF}}{BW}], f_i'(t) \in [-\frac{BW}{2}, \frac{BW}{2}] \qquad (6)$$

where, i is a factor that decides the shape of the chirp, and k_i is a coefficient for the determined chirp. Fig. 4 is an example of non-linear chirp based modulation and demodulation with i=3. Even with the curved up- and down-chirps, the de-chirped results can be decoded in the same way as the conventional linear CSS due to the characteristics of complex conjugate. With this simple idea, a lot of advantages are brought while maintaining noise resilience of conventional CSS.

Energy scattering effect. It is a phenomenon where, during the demodulation FFT process for non-linear chirp, interference chirps that are not perfectly aligned with the receiving window do not concentrate their spectral energy into an FFT bin. For example, with
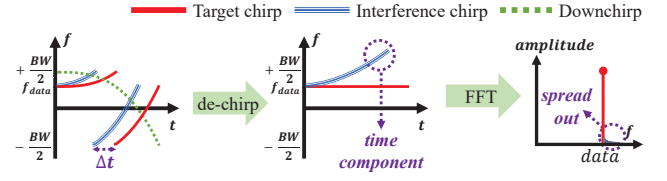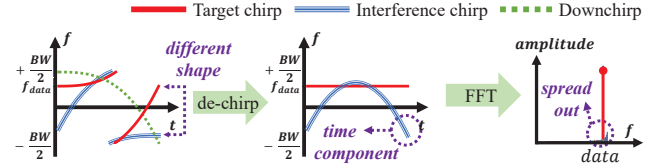

Figure 6: Quasi-orthogonality between different chirp shapes.

i=2 in Eq. (6), the de-chirped signal can be expressed as follows if there is a time offset Δt between the base up-chirp and down-chirp.

$$e^{j2\pi\left(f_c+f_2'(t+\Delta t)\right)t} \times e^{-j2\pi f_2'(t)t} = e^{j2\pi F_2(t)t} \qquad (7)$$

From the given Eq. (6), $F_2(t)$ is formulated as;

$$F_2(t) = f_c + f_2'(t + \Delta t) - f_2'(t) = f_c + 2k_2t\Delta t + k_2\Delta t^2 \qquad (8)$$

$F_2(t)$ retains a time-dependent component if $\Delta t \neq 0$ therefore the result of FFT will lead to energy scatter across multiple frequency bins rather than concentrating at a single point. Since the likelihood of Δt being 0 is negligible, the use of a non-linear chirp proves effective for handling interference. Fig. 5 displays an example scenario where the energy scattering effect occurs. The impact of collisions between chirps of the same shape diminishes unless they are perfectly aligned, thanks to the non-linear shape of chirps.

Quasi-orthogonality. Non-linear chirps can have an infinite number of shapes as long as they sweep through the given frequency range. If a receiver de-chirps a certain non-linear up-chirp of shape $f_i'(t)$ with a differently shaped down-chirp of $f_j'(t)$, this also results in the energy scattering effect across multiple FFT bins as follows;

$$e^{j2\pi\left(f_c+f_i'(t)\right)t} \times e^{-j2\pi f_j'(t)t} = e^{j2\pi G_{ij}(t)t} \qquad (9)$$

$$G_{ij} = f_c + f_i'(t) - f_j'(t) = f_c + k_i t^i - k_j t^j \qquad (10)$$

The time components remain as long as $i \neq j$, which means quasi-orthogonality exists between chirps of different shapes. Fig. 6 illustrates the quasi-orthogonality in a collision scenario. Even in the event of a perfectly aligned collision, the differences in shape can reduce the impact of the collision, providing effects similar to using distinct channels.

Previous studies [22, 24] have shown that non-linear chirps can be used to resolve collisions. However, we believe that there are still unexplored possibilities for the utilization of non-linear chirps beyond what has been explored so far. Based on this vision, we further exploit non-linear chirps for not only collision resolution but also to achieve throughput enhancement on both a single link and also the overall network, which are the main capabilities of *BIC-LoRa*.
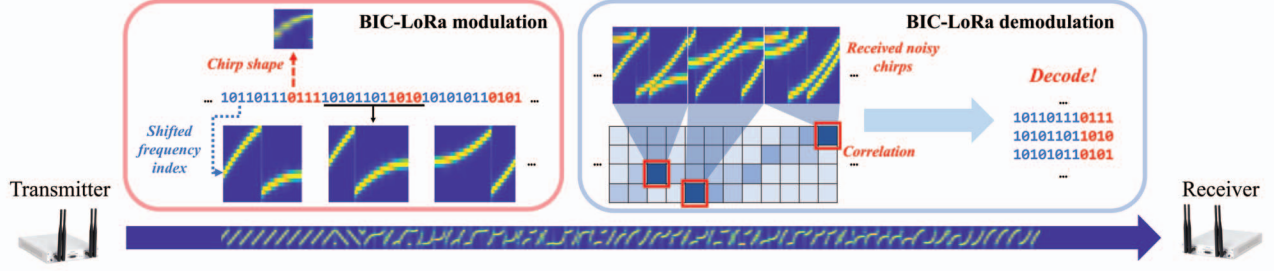
Figure 7: Overview illustration of *BIC-LoRa*.

3 RELATED WORK

BIC-LoRa is inspired by the latest advances in LoRa communication, the non-linear chirps. Traditional collision resolution techniques have addressed collision scenarios through signal processing in the time domain [17, 39, 40, 42], frequency domain [10, 36, 37] or both [34]. CurvingLoRa [24] was a pioneer for applying the concept of non-linear chirp in CSS [16, 18, 20] to LoRa. It implemented a PHY Layer that utilizes a specific non-linear chirp instead of a linear chirp, aiming to resolve collisions between concurrent transmissions simply by altering the shape of the chirp. While it significantly improves the concurrency of LoRa networks, it constructs the network with only one shape of non-linear chirp among the infinite possibilities. CurveALOHA [22] have designed a MAC Layer capitalizing on this remaining potential for improving on CurvingLoRa. To enhance network throughput, CurveALOHA allows packets with diverse chirp shapes to coexist within the network by leveraging quasi-orthogonality among these shapes. As a result, the network contains packets composed of various shapes, and the gateway conducts demodulation corresponding to the shape of each packet, enabling successful decoding even in the presence of collisions. CurveALOHA, through such operations, demonstrated higher network throughput performance compared to other MAC Layer designs such as [12]. Prism [32] enabled high-throughput LoRa backscatter by utilizing non-linear chirps in the process. However, these approaches still falls short of increasing the data rate of a single link, or binds themselves to a specific system which limits the usability of LoRa networks.

Various approaches have been proposed to enhance the throughput of a single link in LoRa communication. ICS-LoRa [11], SSK-LoRa [13], and PSK-LoRa [4] aims to increase single link throughput through their respective methods, but their extents of improvement are marginal. FBI-LoRa [25] increases throughput by encoding additional bits into the frequency bin index, but it comes at the cost of BER loss. Hylink [41] increases the throughput of a single link by transmitting multiple packets overlapped in strong link conditions (i.e., under good SNR). However, this method only gains benefits in strong links, which nullifies the advantage of LoRa's long communication range. Furthermore, because the approach primarily aims to improve the throughput of a single link, it does not well handle collisions. Unlike the approaches described above, *BIC-LoRa* utilizes concepts presented in CurvingLoRa and CurveALOHA to enhance the throughput of a single link while maintaining communication range and effectively handling collisions, ultimately leading to an increase in overall network throughput.
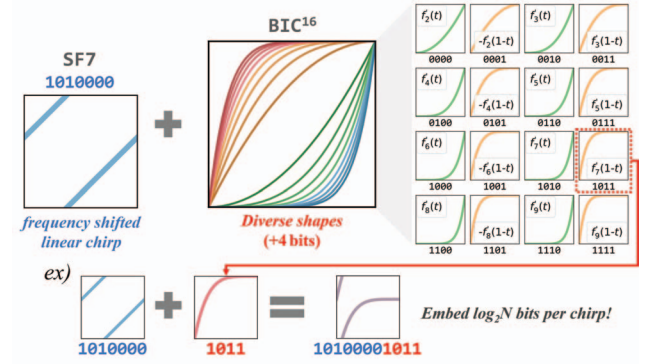

Figure 8: Modulation process with 16 different shapes of non-linear chirps to encode 4 additional bits per chirp.

4 *BIC-LORA* DESIGN

This section presents the design of *BIC-LoRa*, a novel enhancement to LoRa that aims to boost link data rate and network throughput.

4.1 System Overview

The primary goal of this research is to increase the data rate of LoRa. So, how can we boost the data rate, i.e., the amount of data transmitted per unit of time? We can either 1) reduce the time it takes to transmit the same amount of data, or 2) increase the amount of data sent in the same time frame. Although these two approaches exhibit slight variations, they yield the same outcome and share a common challenge.

Intuitively, these approaches undermine the fundamental design principles of LoRa modulation. LoRa's noise resilience is derived from the utilization of a broad frequency range and a long time duration within a single chirp, scattering the influence of noise across both temporal and spectral domains. However, the shortened chirp duration in the first approach, or the cramming of more information into the same bandwidth in the second approach, may lead to decreased time allotted to each frequency component, which could readily undermine these advantages. Moreover, achieving a significant increment in information content may necessitate sacrificing resilience to noise and interference.

Fig. 7 summarizes the overall workflow of *BIC-LoRa*. First, the transmitter exploits a modulation method utilizing a variety of non-linear chirps, resulting in the ability to encode a greater number of bits within a single symbol than conventional CSS. This allows for
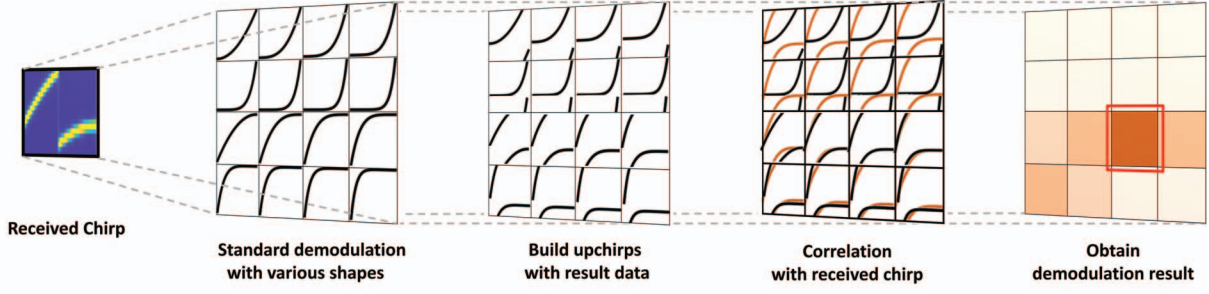
Figure 9: Demodulation process of *BIC-LoRa*-Base.

the quantity of data embedded in the initial frequency components of the chirp to be maintained at *SF* bits, while also enabling the encoding of extra bits in the chirp's shape. Then, the demodulation stage of *BIC-LoRa* leverages the properties of non-linear chirps to their fullest extent, preventing the reduction in robustness relative to conventional CSS demodulation. *BIC-LoRa* achieves robust reception in collision scenarios by calculating a correlation score based on cross-correlation operation in addition to the standard demodulation. This enables the receiver to decode additional data embedded in the chirp's shape. The following sections comprehensively describes this modulation and demodulation processes.

4.2 *BIC-LoRa* Modulation

In conventional LoRa modulation, a single chirp (or symbol) can hold data ranging from 0 to $2^{SF} - 1$, equivalent to *SF* bits. As a result, the modulation process involves partitioning the entire payload bit sequence into sets of *SF* bits, with each set forming a distinct chirp. We introduce an approach where a single chirp holds more bits than *SF*. This approach is made possible by embedding information within the non-linear shape of the chirp itself.

Let's assume we have a set of *N* different non-linear shapes. If one shape corresponds to one data item, then employing these shapes enables us to represent *N* information (e.g. 0~*N*-1), which is equivalent to $log_2 N$ bits. *BIC-LoRa* builds upon this intuition, allowing a single chirp to carry not just *SF* bits, but also an additional $log_2 N$ bits encoded in the shape of the chirp itself. Consequently, the modulation technique of *BIC-LoRa* divides the entire bitstream sequence into units of $SF + log_2 N$ bits. Then, the initial *SF* bits in the resulting unit serve as modulation data, while the following $log_2 N$ bits indicate the non-linear chirp shape to be used for modulation.

Fig. 8 illustrates a modulation example. With *SF*=7, a single chirp initially carries 7 bits. When employing a non-linear chirp set with *N*=16, an extra 4 bits of information can be encoded. The shape of the chirp is defined according to this extra 4 bits of data, creating a non-linear chirp that contains a total of 11 data bits per chirp. Then, a *BIC-LoRa* packet consists of a preamble and a Start-of-Frame Delimiter (SFD) utilizing linear chirps, along with a payload composed of non-linear chirps generated through the aforementioned modulation process.

The advantages of this approach can be explained in two ways. First, it results in a relatively shorter packet length for a given payload length. Since one symbol can carry more data, this gain can lead to a reduction in the duty cycle (power consumption) of nodes

and possibly improved packet delivery ratio for a given bit error rate. Secondly, it becomes possible to accommodate more payload for the identical frame length. This allows the LoRa network to boost maximum throughput for a given channel utilization or duty cycle limit.

Non-linear chirp set. *BIC-LoRa* transmits extra bits using the chirp shapes. Then, how many bits can we add? How much is the gain in data rate using *BIC-LoRa*? For example, if 2^{10} distinct non-linear chirps are used, we would be able to add 10 bits to a chirp. In this case, if SF7 is used, a single chirp could hold 17 bits instead of 7, resulting in a remarkable 143% increase in data rate. However, achieving this in practice poses significant challenges, primarily due to the issue of quasi-orthogonality between non-linear chirps. Even though there is an infinite number of possible shapes for non-linear chirps, there is a limit to the number of shapes that can be selected while preserving quasi-orthogonality between them.

In this work, we constrained the non-linear chirps as a simple polynomial as a proof-of-concept. The advantage of this approach lies in the ease and intuitiveness of expanding the chirp set. The form $f(t) = x^t$ sweeps the given frequency range with a convex shape, and we can obtain a concave form of $f(t)$ by applying a rotation transformation. Fig. 8 illustrates an example modulation process with 16 different shapes of $f(t)$ and its counterpart at different polynomial degrees. In this context, it is intuitively evident that the advantage of quasi-orthogonality among chirp shapes decreases as the polynomial's degree increases. Therefore, there will be a limit to the number of additional bits that can be accommodated while maintaining adequate performance, and we will address this in Section 6.

4.3 *BIC-LoRa* Demodulation

The demodulation must include the process to receive the additional bits embedded in the chirp's shape. Due to the nature of the demodulation process for non-linear chirps, getting the data in the chirp requires knowledge of the target chirp's shape. Unfortunately, the receiver lacks information about the target chirp's shape (since the shape also contains information). Thus, *BIC-LoRa*'s demodulation process involves extracting the carrying data d_{target} and the shape s_{target} of the target chirp C_{target}. We will first explain the demodulation method of *BIC-LoRa*-Base, which is used when there are no packet collisions, and then explain the whole demodulation process of *BIC-LoRa*.
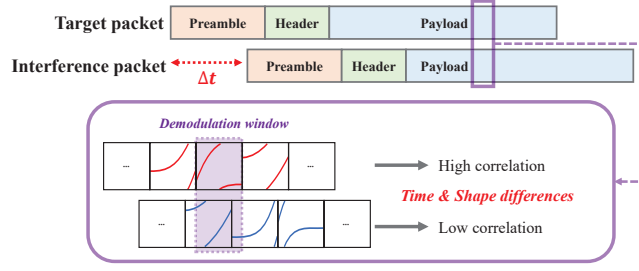
Figure 10: Demodulation process of *BIC-LoRa*.

BIC-LoRa-Base. Fig. 9 displays the overall process of the *BIC-LoRa*-Base. *BIC-LoRa*-Base's demodulation process aim at obtaining both the target chirp's shape and the data it carries. It begins by attempting standard demodulation (obtaining encoded chirp's data) on input chirp C_{in} with every possible N shapes $s_0, \ldots, s_{N-1}$. Then, we obtain corresponding N data values $d_0, \ldots, d_{N-1}$. The receiver's challenge is in distinguishing which value corresponds to the correct data encoded with the target shape.

This determination process starts with a straightforward intuition about non-linear chirps. Standard demodulation with an incorrect shape raises the probability of producing incorrect outcomes unrelated to the correct data. This is due to the dispersion of signal energy across multiple frequency bins, a consequence of the quasi-orthogonality of non-linear chirps. Furthermore, attempting to modulate the incorrectly obtained data with the corresponding non-linear shape results in a chirp that significantly deviates in shape from the original C_{in}.

This is a key point of *BIC-LoRa*-Base's demodulation process. The correct shape generates the correct data, forming a chirp similar to the input one, whereas an incorrect shape yields incorrect data, forming an incorrect chirp. Consequently, it's evident that the key difference between the correct and incorrect chirp lies in their correlation with the C_{in}. The cross-correlation value is a direct method to measure the degree of correlation between the input and the chirp formed through the aforementioned process. Therefore, *BIC-LoRa*-Base generates C_i $(d_i, s_i), (i \in [0, N-1])$ through the standard modulation process and calculates the cross-correlation value between this and C_{in}. Based on this intuition, the chirp C_{corr} with the highest cross-correlation value becomes the correct C_{target}, and we can obtain the data by processing the (d_{corr}, c_{corr}) that make up this chirp. In detail, *BIC-LoRa*-Base divides C_i (d_i, s_i) and C_{in} into n equal parts, multiplies all the cross-correlation values of each, and selects the chirp showing the largest value (n=8 by default).

Collision resolution with *BIC-LoRa*. However, *BIC-LoRa*-Base is insufficient for practical use in real network scenarios since it does not handle packet collisions; i.e. interference chirps can disrupt the demodulation of the target chirp. In cases where the interference chirp's signal strength exceeds that of the target chirp, the chirp with the highest cross-correlation value may not be the target chirp. This will lead to a failure to obtain correct data during demodulation. We tackle this issue by shifting the perspective from dealing with chirps to dealing with packets. Fig. 10 displays the operational concept of *BIC-LoRa*. *BIC-LoRa* addresses collisions by utilizing

the time difference between the interfering packet and the target packet. Details are explained below.

Thanks to LoRa's orthogonality among SFs, we only need to consider interfering chirps that use the same SF, which naturally results in identical symbol time duration. Moreover, interference does not exist in the form of a single chirp; instead, it exists in the form of a packet. Consequently, if a collision occurs between packets, the time difference between the start times of the chirps constituting these packets remain consistent across the entire packet.

With collision, the input chirp C_{in} includes not only the target chirp C_{target} but also the interfering chirps. The receiver can keep track of the arrival time of the preamble for each packet, enabling it to calculate the time offset Δt between the target chirp C_{target} and the interfering chirps. In reality, when a collision occurs between preambles that use linear chirps, packet recognition becomes challenging due to the capture effect. Therefore, Δt is calculated by performing a modulo operation with the length of the chirp on the time difference between the preambles. Using the obtained Δt, we can introduce another insight. Considering the moment of Δt, there exists a significant difference between the interfering chirps and the target chirp. While the target chirp has the same information and shape both before and after Δt, the interference chirps show no relationship between their information and shape.

In this situation, if we attempt demodulation with every shape, as was done in the *BIC-LoRa*-Base, we can generate a total of N chirps $C_0 \sim C_{N-1}$. Then we can divide the input and N chirps into front and back based on Δt, forming pairs such as $(C_{prev, in}, C_{next, in})$, $(C_{prev, 0}, C_{next, 0}), \ldots, (C_{prev, N-1}, C_{next, N-1})$. Furthermore, the demodulation of *BIC-LoRa* uses the cross-correlation score S_{corr} as a metric to find the shape of the target chirp, representing the shape of the target chirp with t, and the shapes of the interfering chirps with i (the earlier one) and j (the latter one). We denote the cross-correlation operation as $\otimes$.

$$S_{corr} = (C_{prev,k} \otimes C_{prev,in}) \times (C_{next,k} \otimes C_{next,in}) \tag{11}$$

For each C_k where k ranges from 0 to N-1, we can summarize how the S_{corr} values behave and how they can be utilized to determine the shape of the target chirp for different values of k:

- **'$k = t$':** In the case of $k = t$, both $C_{prev, k}$ and $C_{next, k}$ exhibit a strong connection to $C_{prev, in}$ and $C_{next, in}$. This is because the target chirp forms a complete chirp within the time window, suggesting that the value of S_{corr} will be the product of meaningful correlation values.

- **'$k = i$':** In this case, $C_{prev, k}$ is related to $C_{prev, in}$. By calculating S_{corr}, the front term will yield a specific correlation value due to the existence of a correlation between the two, but the latter term will approach zero, reflecting the lack of correlation. As a result, S_{corr} is likely to be negligible due to the latter term.

- **'$k = j$':** In this case, a similar phenomenon as in the previous case with $k = i$ occurs, and the value of S_{corr} becomes negligible.

- **'$k \neq i, j, t$':** In this case, $C_{prev, k}$ and $C_{next,k}$ are each unrelated to $(C_{prev, in}, C_{next,in})$. As a result, they have a low degree of correlation with each other. Therefore, the impact of C_k in the demodulation process can be considered negligible.

As S_{corr} exhibits the properties mentioned above depending on the value of k, we select the chirp with the highest S_{corr} value as the target chirp. Additionally, in cases where multiple packets collide, the aforementioned process is repeated, and the final S_{corr} is calculated by multiplying the scores for each packet. Subsequently, the demodulation process concludes by processing the data and shape associated with that chirp. *BIC-LoRa* operates based on correlation similar to *BIC-LoRa*-Base, but it effectively responds to collisions by utilizing timing information from colliding packets.

Frequency offset calibration. In LoRa modulation, carrier frequency offset (CFO) and symbol time offset (STO) may greatly influence transmission reliability. Therefore, to accurately decode signals affected by such offsets, calibration from the receiver's perspective is necessary. We calibrate offset using linear pilot chirps drawing insights from methods such as NScale [36] and CurvingLoRa [24], enabling reliable demodulation.

System overhead. For *BIC-LoRa* to be practically deployed, considerations must be given to overhead factors such as power consumption and computation. Firstly, although the modulation scheme of *BIC-LoRa* differs slightly from the conventional LoRa, it's power consumption remains the same [24], allowing it to be adequately employed in commodity LoRa devices. Second, it is evident that *BIC-LoRa* requires more computational overhead during Tx/Rx compared to conventional LoRa. To investigate this matter, we have measured the system overhead incurred during the Tx/Rx process. For Tx, we measured the time required to generate a single chirp. Across all spreading factors (SF) and sizes of chirp sets, Tx had an average computational load of 1.01x compared to classical LoRa. This indicates a manageable level of overhead increase, suggesting it does not pose a significant burden. For Rx, we obtained the time required to decode a single symbol for all SFs and sizes of the chirp set. In the case where SF is 12 and the size of the chirp set is 8, the maximum computational load reached up to 11.04x compared to classical LoRa. However in LoRa communication, gateways are typically higher-power platforms providing sufficient computational resources [23]. Therefore, *BIC-LoRa* can be adequately utilized in LoRa networks.

To summarize, *BIC-LoRa* can handle collisions when detected, and opt for *BIC-LoRa*-Base when no collision is perceived. (No Δt to use!). *BIC-LoRa* operates effectively even in the presence of low SNR and relatively strong interfering chirps, providing a means to acquire additional data embedded in the form of chirp shapes, which is unattainable through conventional methods.

5 IMPLEMENTATION

Hardware and software. We implement *BIC-LoRa* on a software-defined radio platform USRP B210 as a proof-of-concept to evaluate performance. The modulation process is implemented in GNURadio based on an open-source LoRa implementation [35], and the demodulation process is done in MATLAB.

For the indoor testbed, we installed a gateway and a transmitter device in an office building where the locations of the transmitter is illustrated in Fig. 11a. We have an area with dimensions of 34.8×89 m, and placed the gateway inside a room to avoid line-of-sight (LOS) paths to devices outside the room. Creating a controlled
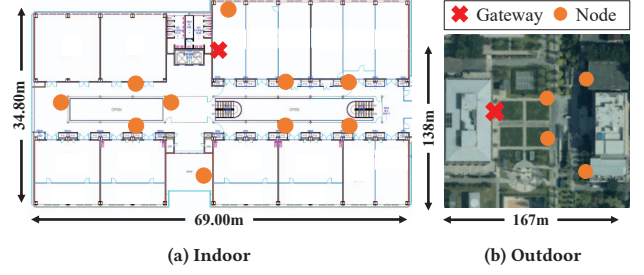

(a) Indoor   **(b) Outdoor**

Figure 11: Testbeds used for real-world evaluations.

System	CurveALOHA	ALOHA	BIC^N
Payload length	51	51	$51 + 5log_2 N$

Table 2: Payload length for evaluation.

amount of traffic load that results in sufficient number of collisions with a limited number of devices, without completely jamming the network, is challenging. Instead, we emulate collision scenarios by transmitting signals from various locations, saving the the raw signals, and combining them at the gateway. We collected data from a total of 10 locations, with each node transmitting a total of 100 packets with various packet intervals.

For the outdoor testbed, we installed a gateway and a transmitter on our university campus as shown in Fig. 11b. We collected data from a total of 4 locations, with each node transmitting a total of 50 packets with various packet intervals.

Evaluation Metrics. The main metrics to evaluate *BIC-LoRa* are symbol error rate (SER), packet delivery rate (PDR), and network throughput. SER indicates the robustness directly at the symbol level. PDR represents the ratio of packets that are successfully delivered, which shows the robustness at the packet level. Network throughput measures the overall throughput of unique data bytes that can be successfully received at the gateway from nodes in the network.

Comparison schemes. We compare *BIC-LoRa* against Curve-ALOHA [22], a state-of-the-art LoRa system, and also with basic ALOHA [1] with linear chirp used for LoRaWAN [3]. Within the original CurveALOHA scheme, when a node generates a packet, it selects either a random shape or the shape with the best PDR after obtaining an ACK from the gateway. However, to showcase the best performance of CurveALOHA in this comparison study, we ensured that all nodes use different shapes, eliminating collisions between nodes with the same shape.

Other settings. For LoRa configurations, we used 923 MHz ISM band, SF=10, Coding rate CR=3, BW=125 kHz, and 1 MHz sampling rate. For a fair comparison, it is essential to use an appropriate payload size according to the scheme and configurations. The maximum MAC payload for SF10 used in the experiments is set at 51 bytes, as indicated in Table 2. This is determined by the maximum time occupancy of the packet containing the payload. We experiment with the longest MAC payload, maximizing the number of symbols within a single packet aiming to deal with saturated traffic conditions. The length of a packet containing a specific payload
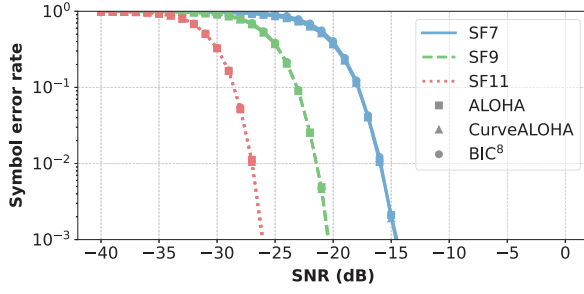
Figure 12: SNR vs. SER comparison for ALOHA, CurveALOHA, and *BIC-LoRa*.

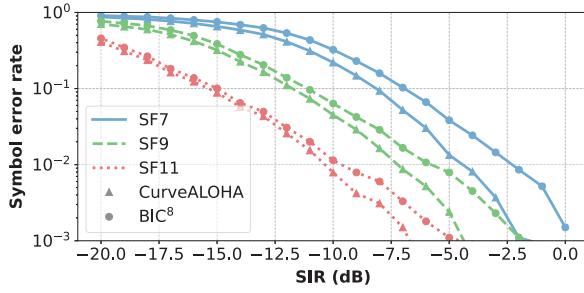**Figure 14: SIR threshold vs. SF for CurveALOHA and *BIC-LoRa*.**

Figure 13: SIR vs. SER comparison for CurveALOHA and *BIC-LoRa*. Note that ALOHA is subject to capture effect when SIR is below 0 dB, and thus has SER ≈ 1.
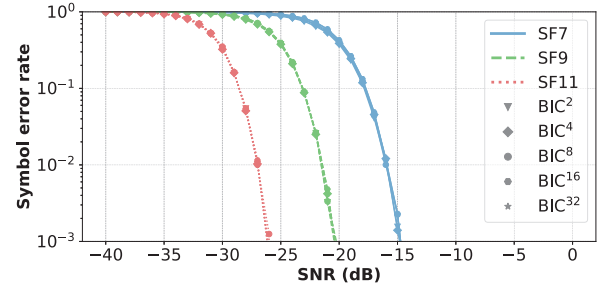
Figure 15: SNR vs. SER performance with various sizes of chirp set.

size (excluding the PHY header) is determined as follows;

$$l_{pkt} = 8 + \lceil \frac{2 * l_{payload} - (SF - 2) + l_{hdr} + l_{crc}}{SF} \rceil \times (4 + CR) \quad (12)$$

In the case of CurveALOHA and ALOHA, they contain *SF* bits per symbol as calculated in Eq. (12). However for *BIC-LoRa*, since one symbol carries more bits, the term '*SF*' increases to '*SF* + number of additional bit'. As a result, it becomes possible to accommodate a longer max payload while maintaining the same packet length, and we determine the payload length as Table 2 where we denote the *BIC-LoRa* with a chirp set size of N as BIC^N.

This implies an increase in the throughput of a single link. As a result of PHY encoding, the gain is $\frac{5log_2 N}{51} \times 100(\%)$ in our configuration. This can vary slightly depending on the system configuration, but it will be near $\frac{log_2 N}{SF} \times 100(\%)$.

6 PERFORMANCE EVALUATION

Our evaluation aims to address the following key questions:

- **Q1.** Despite an increase in throughput, can *BIC-LoRa* maintain noise resilience not to sacrifice communication range?
- **Q2.** Can *BIC-LoRa* handle packet collisions?
- **Q3.** The number of additional bits varies depending on the size of the non-linear chirp set used; how does this affect the performance of *BIC-LoRa*?
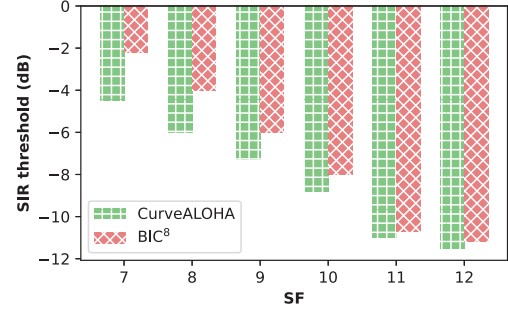- **Q4.** How much throughput gain does *BIC-LoRa* achieve in real experiments on real devices?

6.1 *BIC-LoRa* Performance analysis

In contrast to the ALOHA and CurveALOHA, *BIC-LoRa* involves extra processing steps in demodulation. Demodulation focuses on identifying the shape of the target chirp, which may have errors. To investigate the impact of errors caused by the incorrect identification of chirp shapes during the demodulation process, we conducted the following experiment. We compared *BIC-LoRa* with ALOHA and CurveALOHA by measuring the symbol error rate (SER) under various conditions using Monte Carlo simulations. The settings are as follows. Both CurveALOHA and *BIC-LoRa* utilize a non-linear chirp set with a size of 8, incorporating polynomial shapes ranging from the 2nd to the 5th degree for convex and concave (i.e., upper 8 chirps in Fig. 8. Then, we use MATLAB to create 100,000 symbols each using a linear chirp and a non-linear chirp with random shape in the non-linear chirp set.

Impact of Noise. We investigate whether *BIC-LoRa* could preserve the advantages of LoRa, specifically its noise resilience (or communication range). We introduce additive white Gaussian noise to the generated symbols, varying the SNR from -40 dB to 0 dB. Fig. 12 plots the SER with varying SF7, SF9, and SF11 respectively. The SER-SNR trend is observed to be similar across various SFs and systems. This demonstrates that *BIC-LoRa* can accurately obtain the shape and data carried in the chirp through correlation-based demodulation even in noisy conditions, confirming its noise resilience.

Impact of Collision. We next investigates how *BIC-LoRa*'s demodulation can respond to the presence of interference packets. Since collisions occur on a per-packet basis, we place interference chirps before and after the generated target chirp at random time points. The data and shape of the interference chirps are selected
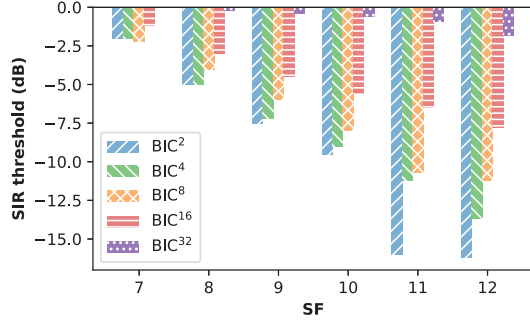
Figure 16: SIR threshold vs. SF with various sizes of chirp set.

Network traffic load (pkt/s)	2	4	6	8	10
Packet interval (ms)	4200	1700	867	450	200

Table 3: Packet interval for various traffic loads.

randomly, and the SIR varied from -20 dB to 0 dB by adjusting the interference chirp's strength. In the case of ALOHA based on linear chirps, it cannot handle collisions due to the capture effect. Therefore, we compare the performance of *BIC-LoRa* and CurveALOHA only. Fig. 13 plots the SER-SIR trend for SF7, 9, and 11. First observation is that the performance gap between CurveALOHA and *BIC-LoRa* is relatively small. In the case of CurveALOHA, decoding is performed with the target chirp's shape already determined, while in the case of *BIC-LoRa*, an additional demodulation process is required to ascertain the shape of the target chirp. When the interference chirp's strength is high, it affects the process of determining the chirp's shape, leading to a decrease in performance. However, *BIC-LoRa* minimizes the performance degradation by considering *delay*. In the case of SF11, the difference in the SIR threshold value where the SER reaches 1% is less than 1 dB. Furthermore, the SIR thresholds for all SFs are as shown in Fig. 14. It can be observed that as SF increases, the performance of *BIC-LoRa* improves, and this is related to the characteristics of LoRa. When SF increases, the time span of the chirp also lengthens, allowing for better distinction of the chirp's shape. In conclusion, *BIC-LoRa* has SIR threshold values lower than 0 dB, enabling it to cope with strong collision occurrences.

Impact of the size of the chirp set. As discussed in Section 4.2, the size of the chirp set can significantly affect the performance of *BIC-LoRa*. To investigate this effect, we compared the performances of *BIC-LoRa* with the size of the chirp set N set to 2, 4, 8, 16, 32 to add 1 to 5 bits data in shape. Chirp sets are created by expanding the chirp shape with the polynomial degree from 2 to 17, depending on N.

Firstly, Fig. 15 illustrates the SER-SNR trend according to N. This shows that noise resilience is well maintained regardless of the chirp set's size. However, Fig. 16 illustrates the SIR threshold for all SFs according to N. We can observe that as N increases, the SIR threshold rises significantly, indicating that it becomes increasingly challenging to cope with collisions. This is mainly due to the similarity in shape of high-order polynomials, and also due to precision of floating point numbers. As the distinction between the shapes decrease, it becomes harder to distinguish between them. It can also be observed that when extending the non-linear chirp set, the SIR threshold approaches zero and become susceptible to collisions. Thus, we conclude that the limit of embeddable bits to fully exploit the orthogonality of polynomial non-linear chirps is limited to 5 bits. Of course, the performance can vary significantly

depending on *how* the non-linear chirp set is constructed, This will be discussed in Section 7. Additionally, Section 6.2 demonstrates the impact of chirp set size in real-world experiments.

6.2 Performance in real-world experiments

So far we have evaluated how *BIC-LoRa* successfully handles collisions and noise under controlled settings. Here we conduct experiments to evaluate how *BIC-LoRa* performs in actual network scenarios. We begin with the indoor scenario in Fig. 11a where we deploy 10 nodes indoors, each situated at varying distances from the gateway. We control the overall network traffic load to vary the collision frequency between packets, enabling us to evaluate the collision-handling capabilities in network scenarios.

The traffic load is controlled by adjusting the packet interval for each node. Since the length (in time) of the packets used in the experiments is 796.67 ms ($\approx$800 ms), we configured the average packet interval as Table 3 to generate the desired traffic load. In addition, we configure the packet interval as an exponential random variable with the given average to create a more realistic scenario. As discussed in Section 5, the packets used in the experiments contain a payload of maximum length, generating the maximum throughput for each scheme.

Overall comparisons. Fig. 17 plots the results for 3 metrics: SER, PDR, and total network throughput. Thanks to the characteristics of non-linear chirps, CurveALOHA and *BIC-LoRa* exhibit significant advantages in SER and PDR relative to ALOHA, illustrated in Fig. 17a and Fig. 17b. This is because ALOHA is incapable of receiving packets that has collided, and the decoding capability of LoRa that uses linear chirp is inferior to non-linear chirps. In *BIC-LoRa*, every chirp is non-linear except for the preamble. Therefore in collision scenarios, demodulation is possible at SIR levels higher than the SIR thresholds shown earlier. Moreover, the SER and PDR performances of *BIC-LoRa* is similar to that of CurveALOHA, indicating that *BIC-LoRa* is proficient at distinguishing chirp shapes during demodulation discussed in Section 3. Fig. 17c provides a clear evidence of how *BIC-LoRa* significantly enhances overall network throughput with enhanced single-link throughput. In the case of ALOHA, as the network traffic load increases, there is a corresponding increase in collisions resulting in a decrease in the actual amount of throughput achieved. On the other hand, CurveALOHA and *BIC-LoRa* handle collisions well, ensuring that total throughput rises in line with demand until saturation. When comparing CurveALOHA with *BIC-LoRa*, if packets of the same length are transmitted, then *BIC-LoRa* contains more data since it embeds data within the chirps. Thanks to this reason, *BIC-LoRa* was able to increase the throughput of a single link by 29.4% by adding 3 bits in 1 symbol, boosting the overall network throughput by up to 32% when compared to CurveALOHA. (6 pkt/s, BIC[8])

Impact of the chirp set size. To examine how the overall network's performance is affected by the size of the chirp set N, experiments are carried out with $N \in \{2, 4, 8, 16\}$. The other settings are unchanged from the previous experiment. Fig. 18a shows that
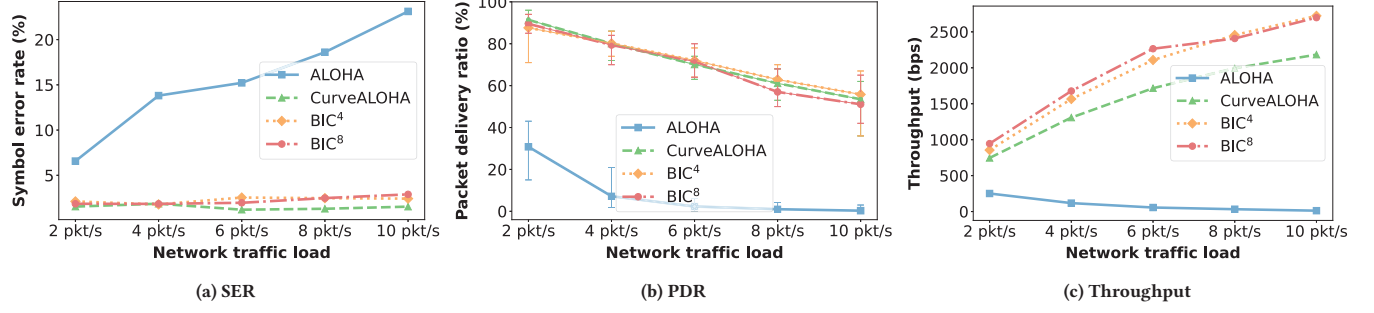
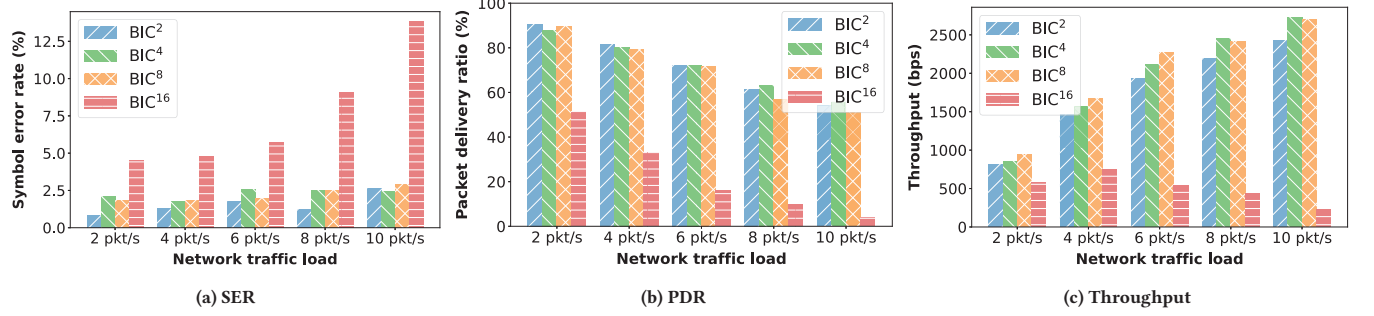**Figure 17: SER, PDR, and throughput results from indoor testbed evaluation.**

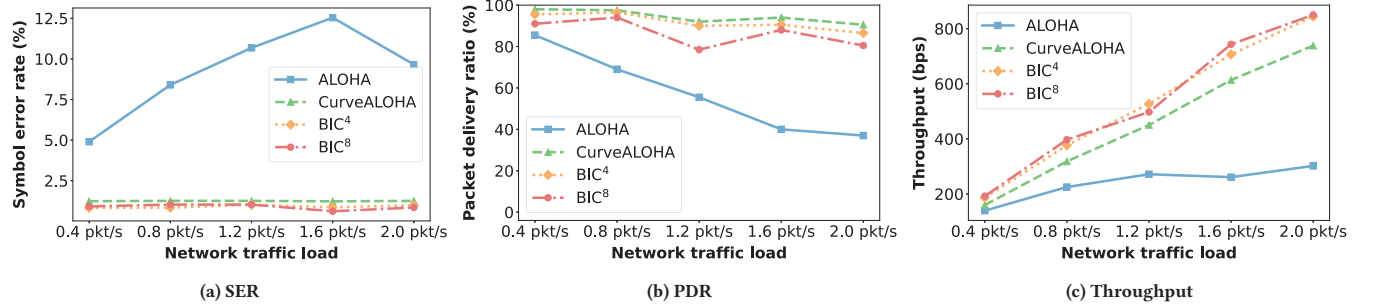**Figure 18: Impact of chirp set in indoor testbed network.**

Figure 19: SER, PDR, and Throughput results from outdoor testbed experiment.

in all traffic load scenarios, the SER performance degrades significantly when the highest N=16 is used. This is closely related to the SIR threshold for the corresponding N already shown in Fig. 16. When N=16, the ability to cope with collisions decreases sharply, leading to symbol errors. Fig. 18b and Fig. 18c display corresponding changes in PDR and throughput. When N is small, link error rates are lower; however, the throughput gain is small because only a small number of additional bits (1 for N=2) can be added. With N=16 or higher, the system exhibits performance similar to ALOHA, struggling to manage collisions and failing to accommodate the growing network traffic load. Hence, it is necessary to use an appropriate value of N when constructing the network, and this choice can be made based on the network's congestion level.

Outdoor experiment. Finally, we experiment on an outdoor testbed (Fig. 11b) with lower SNR links compared to the indoor testbed. The results in Fig. 19 indicate that, despite the fluctuations observed

in Fig. 19a and Fig. 19b due to the dynamic nature of the outdoor environment, *BIC-LoRa* consistently outperformed the compared schemes in terms of throughput (Fig. 19c) in all setups. *BIC-LoRa* increased the max throughput of a single link by 29.4%, boosting the overall network throughput by up to 24.7% (0.8 pkt/s, BIC^8).

7 DISCUSSION

Non-linear chirp set. The non-linear chirps used in *BIC-LoRa* are constructed in a simple pattern by expanding the polynomial degree. This approach allows for an intuitive expansion of the chirp set. However, increasing the polynomial degree resulted in an unexpected performance loss. This turned out to be due to the precision limitations in computation (i.e. float vs. double). Therefore, constructing a non-linear chirp set with more precise or different and distinct shapes can improve performance further. For example, when N=16, substituting $sin(x)$, $tan(x)$, $2^x - 1$, and $log2x + 1$ for
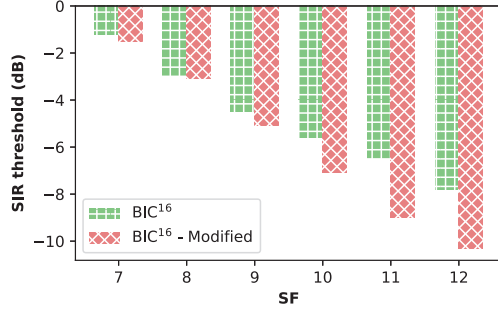
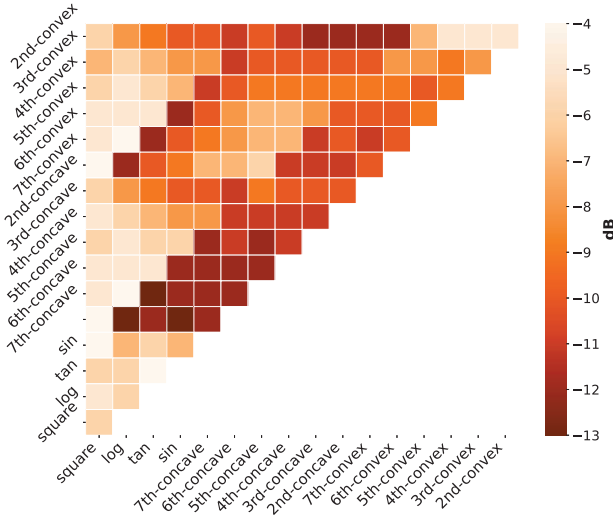**Figure 20: SIR threshold difference with modified chirp set.**

Figure 22: Coexistence with classical LoRa.

Interference SF	7	8	9	10	11
SIR Threshold (Non-linear, dB)	-24	-24	-25	-25	-25
SIR Threshold (Linear, dB)	-24	-24	-24	-24	-17

Table 4: SIR threshold for linear and non-linear chirps, SF12

Figure 21: SIR threshold heatmap for SF=9.

the four convex and concave chirp forms of x^8 and x^9 results in an enhancement to the SIR threshold in each SF as shown in Fig. 20.

Additionally, Fig. 21 plots the heatmap of the SIR threshold for collisions between non-linear chirp shapes when SF=9. Note that the bottom-right portion is symmetric to the top-left portion, thus omitted. Intuitively, as the difference in shapes of each chirp increases, lower SIR thresholds are obtained, indicating a larger quasi-orthogonality between the two. Additionally, as the difference in shape from linear chirp increases, the diagonal components representing energy scattering effects become more significant, leading to lower SIR thresholds. Therefore, when extending the selection of non-linear chirps, it is important to consider both quasi-orthogonality and energy scattering effect. The meticulous selection of such non-linear chirps may require a mathematical analysis, potentially paving the way for new research directions to delve deeper into the properties of the non-linear chirp itself.

Co-existence with classical LoRa. To investigate potential co-existence between *BIC-LoRa* and classical LoRa, we conducted experiments with one device using BIC[8] and another using classical LoRa at the same position while transmitting to a common gateway under contention at various TX rates. Fig. 22 plots the result. *BIC-LoRa* consistently outperforms classical LoRa, and it is intuitive
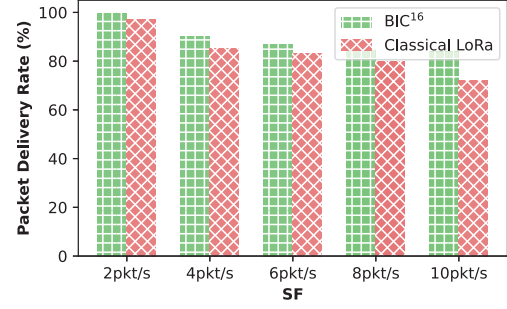
that PDR decreases for both schemes as the traffic load increases. However, there is no notable loss in PDR compared to running only one scheme at a time (Fig. 17b, Fig. 19b), indicating that there is no coexistence issue between *BIC-LoRa* and classical LoRa.

Imperfect orthogonality between different SFs. LoRa enables collision resolution through the orthogonality between SFs. That is, collisions between symbols using different SFs typically have minimal impact on each other's reception [9]. However, the introduction of non-linear chirps may alter this characteristic. To investigate this issue, we generate a target chirp with an arbitrary non-linear shape and SF12, and subjected it to collisions with interference chirps with arbitrary non-linear shapes and SFs ranging from SF7 to SF11, measuring the SIR threshold after 100,000 collisions. We then compared these thresholds to those obtained when using linear chirps. The results (Table 4) show that when the SF difference was 2 or more, the SIR threshold remained consistent, indicating preserved orthogonality. However, with a difference of 1 SF, there was a significant 8 dB difference in the SIR threshold. Nonetheless, the SIR threshold of -17 dB suggests substantial orthogonality, which could potentially be overcome through a more precise design of non-linear shapes.

Future work. We have investigated the impact of various chirp sets and sizes on *BIC-LoRa* for a single link. However, determining the optimal size of the chirp set from a network perspective is challenging. Therefore, methods such as observing the traffic load at the gateway, deriving the optimal chirp set, and dynamically re-configuring them on the nodes over the air can be considered to optimize performance in various scenarios. We leave this as our future work.

8 CONCLUSIONS

We presented *BIC-LoRa*, a new PHY-Layer design that embeds additional bits in the chirp shapes to enhance the data rate of LoRa with same number of symbols. *BIC-LoRa* fully leverages the characteristics of non-linear chirps, not only increasing the data rate of a single link but also retaining the resilience to packet collisions. We implemented *BIC-LoRa* on real software-defined radio platforms,

and demonstrated its superior performance compared to state-of-the-art schemes in terms of both single link by 29.4% and overall network throughput by 32% in real-world scenarios. To the best of our knowledge, this is the first work that enhances the link data rate of LoRa by embedding bits in the chirp shapes. We believe increasing the throughput can expand the usability of LoRa to cover more diverse and emerging IoT applications.

REFERENCES

[1] Norman Abramson. 1970. THE ALOHA SYSTEM—Another alternative for computer communications. In *Proceedings of the November 17-19, fall joint computer conference*. 281–285.

[2] Ferran Adelantado, Xavier Vilajosana, Pere Tuset-Peiro, Borja Martinez, Joan Melia-Segui, and Thomas Watteyne. 2017. Understanding the Limits of LoRaWAN. *IEEE Communications magazine* 55, 9 (2017), 34–40.

[3] LoRa Alliance®. 2022. *LoRaWAN® Regional Parameters RP002-1.0.4*. https://resources.lora-alliance.org/technical-specifications/rp002-1-0-4-regional-parameters

[4] Roberto Bomfin, Marwa Chafii, and Gerhard Fettweis. 2019. A novel modulation for IoT: PSK-LoRa. In *IEEE 89th Vehicular Technology Conference (VTC)*. 1–5.

[5] Marco Centenaro, Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. 2016. Long-Range Communications in Unlicensed Bands: the Rising Stars in the IoT and Smart City Scenarios. *IEEE Wireless Communications* 23, 5 (2016), 60–67.

[6] Qian Chen and Jiliang Wang. 2021. AlignTrack: Push the Limit of LoRa Collision Decoding. In *IEEE 29th International Conference on Network Protocols (ICNP)*. 1–11.

[7] Yao Cheng, Hendra Saputra, Leng Meng Goh, and Yongdong Wu. 2018. Secure Smart Metering Based on LoRa Technology. In *IEEE 4th International Conference on Identity, Security, and Behavior Analysis (ISBA)*. 1–8.

[8] Ulysse Coutaud, Martin Heusse, and Bernard Tourancheau. 2020. Fragmentation and Forward Error Correction for LoRaWAN small MTU networks. In *International Conference on Embedded Wireless Systems and Networks (EWSN)*. 289–294.

[9] Daniele Croce, Michele Gucciardo, Stefano Mangione, Giuseppe Santaromita, and Ilenia Tinnirello. 2018. Impact of LoRa Imperfect Orthogonality: Analysis of Link-Level Performance. *IEEE Communications Letters* 22, 4 (2018), 796–799. https://doi.org/10.1109/LCOMM.2018.2797057

[10] Rashad Eletreby, Diana Zhang, Swarun Kumar, and Osman Yağan. 2017. Empowering Low-Power Wide Area Networks in Urban Settings. In *Proceedings of the SIGCOMM Conference*. 309–321.

[11] Tallal Elshabrawy and Joerg Robert. 2019. Interleaved chirp spreading LoRa-based modulation. *IEEE Internet of Things Journal* 6, 2 (2019), 3855–3863.

[12] Amalinda Gamage, Jansen Liando, Chaojie Gu, Rui Tan, Mo Li, and Olivier Seller. 2023. LMAC: Efficient Carrier-Sense Multiple Access for LoRa. *ACM Transactions on Sensor Networks* 19, 2 (2023), 1–27.

[13] Muhammad Hanif and Ha H Nguyen. 2020. Slope-shift keying LoRa-based modulation. *IEEE internet of things journal* 8, 1 (2020), 211–221.

[14] Jetmir Haxhibeqiri, Abdulkadir Karaagac, Floris Van den Abeele, Wout Joseph, Ingrid Moerman, and Jeroen Hoebeke. 2017. LoRa Indoor Coverage and Performance in an Industrial Environment: Case Study. In *22nd IEEE international conference on emerging technologies and factory automation (ETFA)*. 1–8.

[15] Mehrdad Hessar, Ali Najafi, and Shyamnath Gollakota. 2019. NetScatter: Enabling Large-Scale Backscatter Networks. In *The 16th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*. 271–284.

[16] Nozhan Hosseini and David W Matolak. 2021. Nonlinear Quasi-Synchronous Multi User Chirp Spread Spectrum Signaling. *IEEE Transactions on Communications* 69, 5 (2021), 3079–3091.

[17] Raejoon Jung and Philip Levis. 2021. Receiving Colliding LoRa Packets with Hard Information Iterative Decoding. In *IEEE Global Communications Conference (GLOBECOM)*. 1–7.

[18] Muhammad Ajmal Khan, Raveendra K Rao, and Xianbin Wang. 2013. Performance of Quadratic and Exponential Multiuser Chirp Spread Spectrum Communication Systems. In *International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*. IEEE, 58–63.

[19] Nikolaos Kouvelas, Vijay S Rao, R Venkatesha Prasad, Gauri Tawde, and Koen Langendoen. 2020. p-CARMA: Politely Scaling LoRaWAN. In *International Conference on Embedded Wireless Systems and Networks (EWSN)*. 25–36.

[20] C Leśnik and A Kawalec. 2008. Modification of a Weighting Function for NLFM Radar Signal Designing. *Acta Physica Polonica A* 114, 6A (2008).

[21] Chenning Li and Zhichao Cao. 2022. LoRa Networking Techniques for Large-Scale and Long-Term IoT: A Down-to-Top Survey. *ACM Computing Survey* 55, 3, Article 52 (feb 2022), 36 pages. https://doi.org/10.1145/3494673

[22] Chenning Li, Zhichao Cao, and Li Xiao. 2022. CurveALOHA: Non-linear Chirps Enabled High Throughput Random Channel Access for LoRa. In *IEEE Conference on Computer Communications (INFOCOM)*. 520–529.

[23] Chenning Li, Hanqing Guo, Shuai Tong, Xiao Zeng, Zhichao Cao, Mi Zhang, Qiben Yan, Li Xiao, Jiliang Wang, and Yunhao Liu. 2021. NELoRa: Towards Ultra-low SNR LoRa Communication with Neural-enhanced Demodulation. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 56–68.

[24] Chenning Li, Xiuzhen Guo, Longfei Shangguan, Zhichao Cao, and Kyle Jamieson. 2022. CurvingLoRa to Boost LoRa Network Throughput via Concurrent Transmission. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*. 879–895.

[25] Huan Ma, Yi Fang, Guofa Cai, Guojun Han, and Yonghui Li. 2021. A new frequency-bin-index LoRa system for high-data-rate transmission: design and performance analysis. *IEEE Internet of Things Journal* 9, 14 (2021), 12515–12528.

[26] Davide Magrin, Marco Centenaro, and Lorenzo Vangelista. 2017. Performance Evaluation of LoRa Networks in a Smart City Scenario. In *IEEE International Conference on communications (ICC)*. 1–7.

[27] Paul J Marcelis, Vijay Rao, and R Venkatesha Prasad. 2017. DaRe: Data Recovery through Application Layer Coding for LoRaWAN. In *Proceedings of the 2nd International Conference on Internet-of-Things Design and Implementation (IoTDI)*. 97–108.

[28] Jorge Navarro-Ortiz, Sandra Sendra, Pablo Ameigeiras, and Juan M Lopez-Soler. 2018. Integration of LoRaWAN and 4G/5G for the Industrial Internet of Things. *IEEE Communications Magazine* 56, 2 (2018), 60–67.

[29] Mingyu Park and Jeongyeup Paek. 2023. Poster Abstract: Frequency-Hopped Chirp Spread Spectrum for Collision Resolution in LoRa Network. In *The 21st ACM Conference on Embedded Networked Sensor Systems (SenSys'23)* (Istanbul, Turkiye, 12-17). https://doi.org/10.1145/3625687.3628406

[30] Congduc Pham and Muhammad Ehsan. 2021. Dense Deployment of LoRa Networks: Expectations and Limits of Channel Activity Detection and Capture Effect for Radio Channel Access. *Sensors* 21, 3 (2021), 825.

[31] Qahhar Muhammad Qadir, Tarik A. Rashid, Nawzad K. Al-Salihi, Birzo Ismael, Alexander A. Kist, and Zhongwei Zhang. 2018. Low Power Wide Area Networks: A Survey of Enabling Technologies, Applications and Interoperability Needs. *IEEE Access* 6 (2018), 77454–77473. https://doi.org/10.1109/ACCESS.2018.2883151

[32] Yidong Ren, Puyu Cai, Jinyan Jiang, Jialuo Du, and Zhichao Cao. 2023. Prism: High-throughput LoRa backscatter with non-linear chirps. In *IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1–10.

[33] Yaman Sangar and Bhuvana Krishnaswamy. 2020. WiChronos : Energy-Efficient Modulation for Long-Range, Large-Scale Wireless Networks. In *Proceedings of the International Conference on Mobile Computing and Networking (MobiCom)*.

[34] Muhammad Osama Shahid, Millan Philipose, Krishna Chintalapudi, Suman Banerjee, and Bhuvana Krishnaswamy. 2021. Concurrent Interference Cancellation : Decoding Multi-Packet Collisions in LoRa. In *Proceedings of the ACM SIGCOMM Conference*. 503–515.

[35] Joachim Tapparel, Orion Afisiadis, Paul Mayoraz, Alexios Balatsoukas-Stimming, and Andreas Burg. 2020. An open-source LoRa physical layer prototype on GNU radio. In *IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 1–5.

[36] Shuai Tong, Jiliang Wang, and Yunhao Liu. 2020. Combating Packet Collisions Using Non-Stationary Signal Scaling in LPWANs. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services (MobiSys)*. 234–246.

[37] Shuai Tong, Zhenqiang Xu, and Jiliang Wang. 2020. CoLoRa: Enabling Multi-Packet Reception in LoRa. In *IEEE Conference on Computer Communications (INFOCOM)*. 2303–2311.

[38] Nadège Varsier and Jean Schwoerer. 2017. Capacity Limits of LoRaWAN Technology for Smart Metering Applications. In *IEEE international conference on communications (ICC)*. 1–6.

[39] Xiong Wang, Linghe Kong, Liang He, and Guihai Chen. 2019. mLoRa: A Multi-Packet Reception Protocol in LoRa networks. In *IEEE 27th International Conference on Network Protocols (ICNP)*. 1–11.

[40] Zhe Wang, Linghe Kong, Kangjie Xu, Liang He, Kaishun Wu, and Guihai Chen. 2020. Online Concurrent Transmissions at LoRa Gateway. In *IEEE Conference on Computer Communications (INFOCOM)*. 2331–2340.

[41] Xianjin Xia, Qianwu Chen, Ningning Hou, and Yuanqing Zheng. 2022. HyLink: Towards High Throughput LPWANs with LoRa Compatible Communication. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. 578–591.

[42] Xianjin Xia, Yuanqing Zheng, and Tao Gu. 2019. FTrack: Parallel decoding for LoRa transmissions. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems (SenSys)*. 192–204.

[43] Zhenqiang Xu, Shuai Tong, Pengjin Xie, and Jiliang Wang. 2020. FlipLoRa: Resolving collisions with up-down quasi-orthogonality. In *IEEE International Conference on Sensing, Communication, and Networking (SECON)*. 1–9.

[44] Yuguang Yao, Zijun Ma, and Zhichao Cao. 2019. LoSee: Long-Range Shared Bike Communication System Based on LoRaWAN Protocol. In *International Conference on Embedded Wireless Systems and Networks (EWSN)*. 407–412.