Magnifier: Leveraging the Fine-Grained Hardware Information and Their Temporal Patterns for Concurrent LoRa Decoding

Weiwei Chen, *Member, IEEE*, Shuai Wang, *Senior Member, IEEE*, and Tian He, *Fellow, IEEE*

***Abstract*—LoRa Wide Area Network (LoRaWAN) is famous for its low power consumption and wide coverage area. A critical issue that constrains the scalability of LoRaWAN is how to support concurrent packet reception efficiently. To this end, our work first carefully investigates all the hardware imperfections between a LoRa transceiver pair and studies their impacts on the received signal's temporal patterns with fine granularity. Such imperfections and the associated temporal patterns are intrinsic and unique for any transceiver pair in a wireless system. Therefore, they provide a brand-new and highly influential perspective for identifying different packets. Motivated by this, we propose Magnifier, which integrates the two factors mentioned above for LoRa concurrent decoding. Though some prior work leverage partial hardware imperfections, they fail to systematically study and exploit the temporal patterns caused by the imperfections for concurrent packet reception. We also evaluate the performance of Magnifier with 40 transmitters (commodity LoRa chips) and one receiver (USRP B210). Both critical complexity analysis and extensive experiments show that with the same hardware settings and the same scale of complexity, Magnifier can support 3× to 8× throughput of other existing schemes.**

***Index Terms*—Lora, concurrent reception, system offsets, temporal patterns.**

I. Introduction

NOWADAYS, we have witnessed the tremendous growth of the Internet of Things (IoT) networks. According to Cisco's Virtual Network Index (VNI) [1], the M2M connections from IoT are expected to reach 14.7 billion by 2023. As a low-cost and wide coverage solution, LPWAN is indispensable in IoT for supporting low-rate and infrequent traffic [2], such as smart meters, smart farming, etc.

Manuscript received 3 November 2022; revised 13 April 2023; accepted 20 June 2023. Date of publication 4 July 2023; date of current version 4 April 2024. This work was supported in part by the National Nature Science Fundation of China under Grant 62172092, in part by Science and Technology Innovation 2030 - Major Project under Grant 2021ZD0114202, and in part by the National Natural Science Foundation of China under Grant 62272098. Recommended for acceptance by V. P Ranga Rao. *(Corresponding author: Shuai Wang.)*

Weiwei Chen is with the Department of Computer Engineering and Science, Shanghai University, Shanghai 200444, China (e-mail: chen.ava.0012@gmail.com).

Shuai Wang and Tian He are with the Department of Computer Science and Engineering, Southeast University, Nanjing 211189, China (e-mail: shuaiwang@seu.edu.cn; tianhe@cs.umn.edu).

Digital Object Identifier 10.1109/TMC.2023.3292279

LoRa, due to its low deployment cost and high performance, has quickly become one of the representative technologies in LPWAN [3]. LoRa adopts Chirp Spread Spectrum (CSS) as its modulation scheme. In CSS, signals with different Spreading Factors (SF) or bandwidths are nearly orthogonal. Each transmitter chooses its own SF and bandwidth independently to avoid collision with other nodes [4]. Besides, no sophisticated MAC protocol, but ALOHA is used to compete for a channel. As a LoRa receiver (usually a LoRa gateway) connects to many transmitters, packet collisions (transmissions with the same central frequency, SF, and bandwidth) are inevitable. If not handled properly, this issue will cause high packet reception failure and constrain the network scalability significantly.

Prior works regarding concurrent reception in LoRaWAN include solutions from MAC layer [5], [6], [7] and PHY layer [8], [9], [10], [11], [12], [13], [14], [15], [16], [17]. MAC layer approaches strive to avoid collisions by designing various scheduling schemes on top of LoRaWAN's own MAC protocol stack. In contrast, the PHY layer approaches separate concurrently received packets from IQ signals directly without performing additional scheduling strategies.

In this paper, we focus on PHY layer approaches. PHY layer concurrent LoRa reception has drawn extensive research attention. Most of them, however, only use coarse-grained information [8], [9], [10], [11], [12], [13], such as the time-misalignment between different packets and their average received power for concurrent decoding. As detailed in the following section, such information is inaccurate, leading to poor decoding performance when the number of concurrently transmitted packet increases.

Fine-grained hardware imperfections include the Central Frequency Offset (CFO); the Sampling Time Offset (STO, the time offset between signals arrive at the receiver and the receiver starts to sample them); and the Sampling Frequency Offset (SFO) between a transceiver pair. In wireless systems, especially low-cost LoRaWAN, imperfections are intrinsic and unique. Moreover, the phase and amplitude distribution of symbols in the same packet due to such imperfections demonstrate strong temporal correlations. In particular, if the time-domain LoRa signal is transformed into the frequency domain spectrum, in the short term, the location and the phase drifts of the frequency peaks that carry several consecutive symbols in a packet stay close to each other. However, in the long term, these features evolve with time and can be very different if the duration of

the two symbols is considerable (e.g., the sth symbol and the $s + 20$th symbol in a packet). As long as the fine-grained features and their temporal patterns are appropriately tracked, they can be used to identify symbols from different packets.

Compared with the coarse-grained features, the fine-grained offset features and their temporal patterns provide higher dimensional information with greater accuracy, resulting in remarkably lower decoding errors and higher concurrency. Though Choir [16], CIC [17], and PCube [15] also discuss part of the hardware offsets information, such as CFO, in their designs, none of them consider all of the offsets. Let alone exploiting the associated temporal patterns for discriminating concurrently received packets.

Motivated by this, we propose Magnifier that leverages three offsets due to hardware imperfections, e.g., CFO, STO, and SFO, and the associated temporal patterns for concurrent LoRa reception. A crucial challenge in implementing Magnifier is that the overall temporal patterns are unknown before decoding packets. To address this issue, Magnifier first extracts the initial frequency drifts, phase, and amplitude information from the preambles in a packet. Given the initial features, symbols of the same packet are identified one by one based on the short-term temporal patterns (fine-grained features of adjacent symbols in a packet are close to each other). The fine-grained features of the already identified symbols are then used to estimate and construct a packet's long-term temporal patterns. Magnifier further leverages the temporal patterns to refine the decoding result. To our best knowledge, this is the first work that leverages the fine-grained temporal patterns of a packet caused by various system offsets for concurrent LoRa reception. The main contributions are listed as follows.

1) We systematically investigate the fine-grained CFO, STO, and SFO and their impact on the received signal's short and long-term temporal patterns. An accurate signal model includes the three offsets is established. We also introduce low-complexity approaches to extract the temporal patterns from low Signal to Noise Ratio (SNR) packets.
2) We propose Magnifier, a low-complexity framework that leverages the fine-grained features and their temporal patterns to entangle concurrent reception. Our design only requires one receiver with one receiving antenna. In addition, the long-term temporal pattern helps to compensate for the overall frequency drifts and correct the corresponding decoding errors in a packet. Note that this issue has yet to be studied in existing works.
3) We theoretically analyze the complexity of Magnifier, and conduct extensive evaluations to examine its performance. According to the results, Magnifier's complexity is in the same order as other existing approaches. However, its throughput can be as high as $3\times$ to $8\times$ of existing schemes.

The rest of the paper is organized as follows. Motivations are provided in Section II. Section III discusses the overall design of Magnifier. Detailed chip level signal model is presented in Section IV. Section V presents the detailed design of each module in Magnifier. It also analyzes its complexity. Evaluations are shown in Section VI. Section VII summarizes prior works. We conclude the paper in Section VIII.

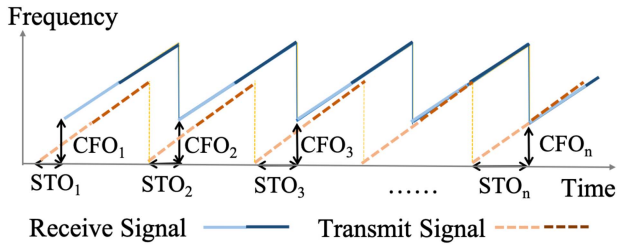


Fig. 1. An illustration of offsets between the transceiver pair. The slope difference between the transmitting and receiving signal represents the SFO of the transceiver pair.

II. Motivations

A. Fine-Grained Signal Offsets and Their Temporal Patterns

We first briefly discuss three fine-grained offsets. In CSS (Chirp Spread Spectrum), a symbol is modulated by a sequence of frequency components. Each frequency component is named a chip, and the set of chips that carry one symbol is called a chirp. In general, an offset consists of two parts: a *chirp level* and a *chip level* offset. The chirp level offset is measured with an integer number of chips or frequency bins, while the chip level offsets are measured with a fractional number of chips or frequency bins.

1) Three Types of Offsets:

1) *Central Frequency Offset (CFO):* CFO can be modeled as a coarse-grained chirp level component with an integer number of frequency bins and a fine-grained chip level component with a fractional number of frequency bins.
2) *Initial Sampling Time Offsets (STO):* The initial STO is the time offset between the first LoRa chip arriving at the receiver and the receiver starting sampling it. Similar to CFO, the initial STO is also composed of a chirp level and a chip level component. With CSS, apart from CFO, the frequency peak of the received signal experiences an additional frequency drift due to STO.
3) *Sampling Frequency Offsets (SFO):* SFO is device dependent and varies from one transceiver pair to another. Similar to CFO, the SFO is also caused by the discrepancy between the transmitter and receiver oscillator.

Put everything together, Fig. 1 illustrates CFO, STO, and SFO in a LoRa system. In the figure, there are a set of transmitting and receiving preambles. For illustration purpose, a chirp only contains two chips. E.g., the light blue line denotes the first chip of a chirp, whereas the dark blue line denotes the second chip of the same chirp. The gap between the transmitting and receiving frequency stands for the CFO. Likewise, the slope of the two signals represent the sampling rate at the transmitter or the receiver. For instance, the higher the slope, the larger the number of samples received in a given time, and the higher the sampling rate. Due to SFO, STO changes with time. E.g., refer to Fig. 1, as the sampling rate at the receiver is lower than that of the transmitter, the sampling time at the receiver lags longer and longer, leading to a higher STO as time goes by. In the first chirp, STO_1 is smaller than one chip, however, when receiving the nth chirp, the STO_n reaches one chip.

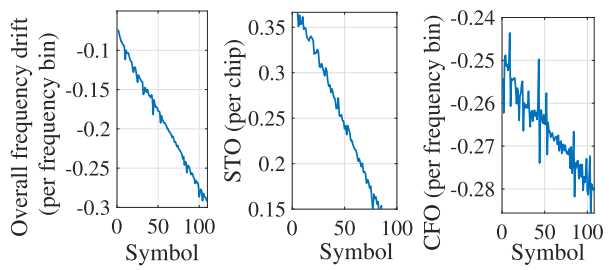


Fig. 2. Illustration of the temporal patterns of the frequency drift, the chip level STO, and the chip level CFO of a commercial SX1278 device.

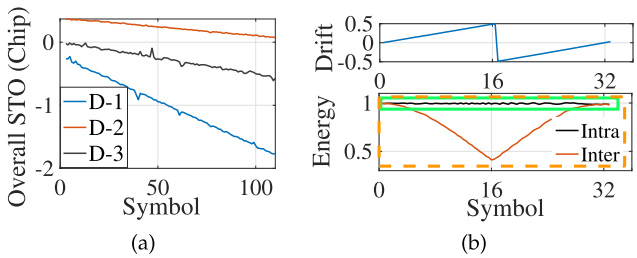


Fig. 3. (a). Overall STO for three commodity LoRa devices with SFO=6.7e-6, 2.4e-6 and 1.1e-6. Here the three devices are labeled as D-1, D-2 and D-3. (b). The energy of the frequency peak of each symbol in a packet. "FBB" refers to the overall "Fractional Frequency Bin".

2) The Temporal Patterns of the Offsets: Fig. 2 plots the temporal patterns of the frequency drift, STO, and CFO of a commercial SX1278 device. It shows that the fine-grained features do not stay constant in a packet but drift slowly with time. Moreover, CFO also experiences slight changes in a packet. According to the user manual [18], for commodity LoRa devices (such as SX1276/77/78), the overall increments or decrements of CFO during the whole packet is smaller than one-third of a frequency bin.

Moreover, the existence of SFO changes the sampling time offsets between the transceiver, leading to a time-varying STO of the received signals. SFO is a minimal value should a LoRa node function properly. For instance, Fig. 3(a) plots fine-grained STO of three packets from three SX1278 LoRa devices. According to the figure, when SF=10, the SFO of the three devices are 1.3e-5, 4.8e-6, and 2.2e-6 chips per sample, leading to 0.013, 0.0049, and 0.0023 chips STO shift per symbol. Herefore, from the short-term perspective, it is safe to assume STO stays constant within one symbol, and the STO of adjacent symbols are close. From a long-term standpoint, however, after receiving several symbols (e.g., several tens of symbols), the cumulative time shift due to SFO is significant, resulting in a very different STO. E.g., the STO of the three devices mentioned above shifted 0.39, 0.15, and 0.07 chips after receiving 30 symbols.

To summarize, the fine-grained chip-level features and the associated temporal patterns carry critical information and thus can be exploited for identifying an ongoing packet.

B. The Significance of Magnifier

1) Fine-Grained Chip-Level Features and Their Temporal Patterns are Intrinsic and Unique for Each Packet: Due to the hardware discrepancy, different transceiver pairs have various SFO, CFO, and initial STO. The chip level offsets are marginal, but their temporal patterns usually have high identification resolution. For instance, refer to Fig. 3(a), though the SFO of the three devices is similar and close to 0, the changing rate (the temporal pattern) of the chip level STO for the first device is 1.7 and 5 times higher than that of the second and the third device.

2) Fine-Grained Chip-Level Features and Their Temporal Patterns are Accurate and Multi-Dimensional, Leading to High Packet Identification Resolution: Fig. 3(b) plots the energy information collected from an SX1278 LoRa device (with Bandwith = 250KHz, SF = 10). The black line shows the energy of the actual frequency peak obtained with chip-level information. The red line is the energy with chirp-level information (the integer frequency bin). Due to STO and SFO, the energy of the integer frequency bin oscillates significantly from peaks to valleys as the chip-level frequency drift slides from 0 to ± 0.5 frequency bin. In the worst case, the energy of the integer frequency bin is only 40% of the actual peak. It implies that signal features obtained with chip-level information are far more accurate compared with chirp-level approaches, leading to high decoding efficiency.

Moreover, Magnifier exploits multi-dimensional features and their temporal patterns to classify packets. E.g., we can carefully trace the time-varying chip level features such as STO, the phase, energy, and the location of the target frequency bins based on their temporal patterns. Two packets are successfully separated when one of the four features is different. In general, the probability that two packets' features are similar decays exponentially with the number of features to be used. Magnifier has a significantly lower likelihood of misidentifying two packets than the approaches utilizing one or two features. Magnifier thus provides higher identification resolution than existing approaches.

3) Magnifier Paves the Way to Recover Decoding Errors That Existing Works Cannot Correctly Handle: The temporal patterns of chip-level information help to reconstruct the original LoRa signal with great precision. The accurate signal model is then adopted to recover detection errors from two aspects. First, as will be detailed in Section V-F, due to SFO, the frequency of drifts keep evolving with time during the packet. By carefully tracing the temporal patterns of the STO and CFO, Magnifier compensates for the overall frequency drifts and correctly recovers every symbol in a packet. Existing schemes, however, cannot handle such drifts properly as they fail to consider their temporal patterns. Therefore, consecutive symbol errors occur when the overall frequency drift exceeds 0.5 or -0.5 frequency bins. E.g., when the overall frequency drift exceeds 0.5 frequency bin after the ith symbol, the remaining symbols denoted as $\{s_j | j > i\}$ will be demodulated as $\{s_j + 1 | j > i\}$.

Second, symbol collision occurs when symbols from two or more packets occupy the same or adjacent frequency bins and can only be recovered with a precise signal model based on chip-level information. The chirp-level approaches, thus, are not able to address this issue as they cannot describe the exact features of a frequency peak. Recall that as Choir and PCube ignore the chip-level temporal patterns the offsets, they cannot recover errors due to symbol collisions and cumulative frequency drifts.

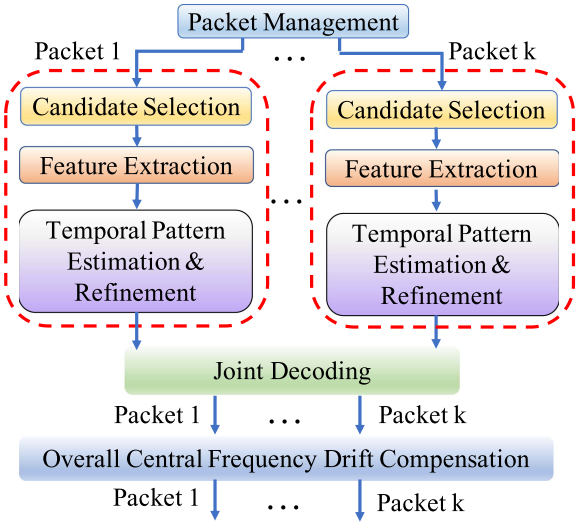


Fig. 4. The design architecture of magnifier.

III. Design Overview

This section introduces the overall design of Magnifier. Without loss of generality, in this paper, *packet collision* refers to the case when the transmission duration of two or more packets overlap with each other. Moreover, *symbol collision* represents the situation that two or more symbols are concurrently transmitted on the same or adjacent frequency bins). Note that symbol collision is more difficult to handle than packet collision.

The chip-level features used in Magnifier are the STO, the central frequency drift, the phase, and the amplitude of the frequency peak. These four features represent the combined effects of CFO, STO, SFO, and the received power of a LoRa symbol. Magnifier first extracts the chip-level features of a set of frequency peaks and uses the temporal patterns of a packet to identify whether a frequency peak belongs to the packet or not. The receiver's sampling rate equals the transmission bandwidth. Fig. 4 shows the overall design architecture. In particular, a packet management module is set on top to detect the arrival of new packets. It also performs chirp-level synchronization for those packets.

The Candidate Selection module selects a set of frequency bins for each symbol of an ongoing packet. The following modules will investigate the chip-level information of the frequency set. By choosing a candidate frequency set, the computation complexity of Magnifier is maintained at a level comparable to that of existing approaches.

The feature extraction module analyzes the chip level information of each frequency candidate and assigns at most one frequency bin as the decoded symbol to the packet. The assignment is determined with the short-term temporal patterns of the fine-grained features. We call the frequency bin assigned to a symbol the *frequency-symbol pair*. Notice that, at this stage, some frequency bins may be wrongly paired with some symbols. The following module is responsible for detecting and correcting these pairing errors.

The frequency-symbol pairs and their chip level features are fed to the Temporal Pattern Estimation & Feature Refinement module. This module uses the decoded frequency-symbol pairs to estimate the long-term temporal patterns of all the symbols in a packet. The long-term temporal patterns are then adopted for refining the decoding result. In particular, for symbols that have yet to be decoded (are yet to assign a frequency bin) or have been decoded incorrectly (with the wrong frequency-symbol pair), this module strives to assign the correct frequency bins to them.

The previous three modules can decode most of the symbols in a packet. However, when symbol collision happens, the chip-level features of such frequency bins will be the combination of all the collided symbols. The joint Decoding module detects the collisions and assigns the collided frequency bins to the target symbols.

With the estimated temporal patterns, the last module compensates for the time-varying frequency drifts and corrects the decoding errors due to such drifts.

TABLE I
Notations

Δ_m	The overall STO of the mth symbol.
N	The number of chips in a symbol.
ζ	The SFO of a transceiver pair.
δ_m	The CFO of the mth symbol.
η_m	The overall central frequency drift.
a_m	The amplitude of the frequency peak of the mth symbol.
s_m	The value of the mth symbol transmitted by the packet.
ϕ_m	The phase of the frequency bin carries symbol m .
α_m	The phase difference between symbol m and $(m-1)$.

IV. Chip Level Fine-Grained Offsets Extractions

A. Inspecting the Impact of Offsets on the Received Signal

This section introduces the signal models and chip-level parameter calculations. The temporal patterns of the chip-level features are also discussed. Recall that prior work [19] establishes signal models regarding CFO and SFO. However, it fails to consider STO and the temporal patterns of offsets. To our best knowledge, this is the first work that discusses three key offsets and their temporal patterns with great accuracy. Notations are listed in Table I.

1) A Primer on CSS Modulation/Demodulation: A LoRa symbol is divided into a set of chips. The number of chips in a symbol is determined by its Spreading Factor (SF), and the duration of a chip is determined by the transmission bandwidth. Via finely tuning SF and the bandwidth, different degrees of reliability and date rate trade-off can be achieved. Let N be the number of chips, s_m be the mth symbol, and the nth chip in the symbol (denoted as $x_{m,n}$) is

$$x_{m,n} = \begin{cases} e^{j2\pi\left(\frac{n}{2}+s_m-\frac{N}{2}\right)\frac{n}{N}} & 0 \le n < N - s_m, \\ e^{j2\pi\left(\frac{n}{2}+s_m-\frac{3N}{2}\right)\frac{n}{N}} & N - s_m \le n < N. \end{cases} \quad (1)$$

The received signal will first be de-chirped and transformed to the frequency domain. The frequency bin with the highest value is demodulated as the transmitted symbol.

2) Modeling the Received Signal With Offsets: Denote the overall STO at the beginning of the mth symbol as Δ_m. Since the variation of the SFO and the STO within a symbol is negligible,

we assume they stay constant within a symbol but vary from symbol to symbol. Let ζ_m be the SFO of the mth symbol, and denote $k_m = \zeta_m + 1$. If the transmitter sends $[(k_m)n]$ chips in symbol m, the receiver will get n samples. Here the operation $[x]$ means to take the integer part of x. The STO of all the chips in symbol m is approximated as

$$\Delta_m = \Delta_0 + \sum_{i=0}^{m} \zeta_i N. \tag{2}$$

Note that, as shown in Section II, the SFO in a packet is relatively stable. In this case, we can approximate SFO as a constant ζ. Let $y_{m,n}$ be the nth chip of the mth symbol at the receiver, p_m be the average transmission power, h_m be the channel state of the mth symbol, $a_m = h_m\sqrt{p_m}$. $|a_m|$ is the average amplitude of the mth symbol. $y_{m,n}$ becomes

$$y_{m,n} = \begin{cases} a_m e^{j2\pi q_1(n)} & 0 \le n < N - s_m, \\ a_m e^{j2\pi q_2(n)} & N - s_m \le n < N. \end{cases}$$

Here Δ_m and δ_m is the STO and CFO of the symbol, and

$$q_1(n) = \left(\frac{k_m n + \Delta_m}{2} + s_m - \frac{N}{2}\right)\frac{k_m n + \Delta_m}{N} + \frac{k_m \delta_m n}{N},$$

$$q_2(n) = \left(\frac{k_m n + \Delta_m}{2} + s_m - \frac{3N}{2}\right)\frac{k_m n + \Delta_m}{N} + \frac{k_m \delta_m n}{N}.$$

When chip-level synchronization is achieved ($|\Delta_m| < 1$), the de-chirped signal $r_{m,n}$ is

$$\begin{aligned} r_{m,n} &= y_{m,n} e^{j2\pi\left(-\frac{n^2}{2N} + \frac{n}{2}\right)} \\ &\approx \begin{cases} a_m e^{j2\pi\phi_m} e^{j2\pi\frac{f_m \cdot n}{2N}} & 0 \le n < N - s_m \\ a_m e^{j2\pi\phi'_m} e^{j2\pi\frac{(f_m+N)n}{2N}} & N - s_m \le n < N \end{cases}, \end{aligned} \tag{3}$$

where

$$\begin{aligned} \phi_m &= \angle h_m + \frac{\Delta_m^2}{2N} + \frac{\Delta_m s_m}{N} - \frac{\Delta_m}{2} + \sum_{z=0}^{m-1} k_m \delta_z \\ &\approx \angle h_m + \frac{\Delta_m s_m}{2N} - \frac{\Delta_m}{2} + \sum_{z=0}^{m-1} \delta_z, \end{aligned} \tag{4a}$$

$$\phi'_m = \phi_m - \Delta_m, \tag{4b}$$

$$\begin{aligned} f_m &= \Delta_m k_m + k_m s_m + \frac{(k_m^2 - k_m)n}{2} + k_m \delta_m \\ &\approx \Delta_m + s_m + \delta_m. \end{aligned} \tag{4c}$$

Here the operation $\angle x$ means to take the angle of the complex value x. f_m is the frequency peak of symbol m. ϕ_m is defined as the phase of the target frequency peak at symbol m. (4a) and (4c) holds as $|\Delta_m| << N$, $\frac{\Delta_m^2}{2N} \to 0$, and the SFO between a transceiver pair k_m is close to 1.

With (4a) and (4c), we can express the chip level central frequency drift η_m and the phase differences between symbol m and $m-1$ (denoted as α_m) as

$$\eta_m = f_m - s_m = \Delta_m + \delta_m, \tag{5a}$$

$$\begin{aligned} \alpha_m &= \phi_m - \phi_{m-1} \\ &\approx \frac{\Delta_m s_m}{N} - \frac{\Delta_{m-1} s_{m-1}}{N} - \frac{\Delta_m}{2} + \frac{\Delta_{m-1}}{2} + \delta_{m-1}. \end{aligned} \tag{5b}$$

Note that, as a LoRa symbol lasts no more than several tens of milliseconds and two transmitted symbols are within the coherent time, (5b) assumes that the channel state between adjacent symbols remains constant. Moreover, since we only consider α_m, the difference between ϕ_m and ϕ_{m-1}, the channel effect, which may experience considerable variation during the whole packet will also be safely excluded.

B. Chip-Level Parameter Calculation

The chip-level parameters to be used in Magnifier are listed in the tuple $\{\eta_m, \Delta_m, \alpha_m, a_m\}$, here a_m is the amplitude of the frequency peak (the square root of the energy of the frequency peak). Due to LoRa signals' low SNR, these parameters cannot be calculated directly from the time domain. We hence use frequency domain signals to derive the parameters so that the processing gain in CSS can be fully exploited, resulting in highly accurate parameter estimation. Given a frequency bin f_m to be inspected, let $f_m = f_{m,1} + \eta_m$ and $f_{m,1}$ is the integer part of f_m. For any chip n in symbol m, we first compensate the integer part of the frequency peak f_m and obtain $z_{m,n}$ as

$$z_{m,n} = \begin{cases} x_{m,n} e^{-\frac{j2\pi f_{m,1} n}{N}} & 0 \le n < N - s_m, \\ x_{m,n} e^{-\frac{j2\pi (f_{m,1}+N) n}{N}} & N - s_m \le n < N. \end{cases} \tag{6}$$

Define $\vec{d}_{m,1} = \{z_{m,n} | 0 \le n < N - s_m\}$, and $\vec{d}_{m,2} = \{z_{m,n} | N - s_m \le n \le N\}$. Magnifier first correlates $\vec{d}_{m,1}$ and $\vec{d}_{m,2}$ with all the fractional frequency sets $\vec{g}_{i,1} = \{e^{-j2\pi \frac{i}{I}\frac{n}{N}} | 0 \le n \le N - s_m\}$ and $\vec{g}_{i,2} = \{e^{-j2\pi \frac{i}{I}\frac{n}{N}} | N - s_m \le n \le N\}$ respectively. Here i is an integer within $(-\frac{I}{2}, \frac{I}{2})$, and $\frac{i}{I}$ is the frequency component of vector $\vec{g}_{i,1}$ and $\vec{g}_{i,2}$. We set $I = 100$ to achieve better complexity and performance trade-off. Let the index of the fractional frequency set with the largest correlation be i^* (the index results in the largest amplitude of the frequency peak), and the chip-level features can be obtained with the following equation.

$$i^* = \arg\max_i |\vec{g}_{i,1} \cdot \vec{d}_{m,1}| + |\vec{g}_{i,2} \cdot \vec{d}_{m,2}|, \tag{7a}$$

$$\eta_m \approx \frac{i^*}{I}, \tag{7b}$$

$$a_m = |\vec{g}_{i^*,1} \cdot \vec{d}_{m,1}| + |\vec{g}_{i^*,2} \cdot \vec{d}_{m,2}|, \tag{7c}$$

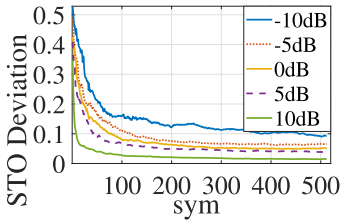
$$\phi_m = \angle(\vec{g}_{i^*,1} \cdot \vec{d}_{m,1}), \tag{7d}$$

$$\phi'_m = \angle(\vec{g}_{i^*,2} \cdot \vec{d}_{m,2}), \tag{7e}$$

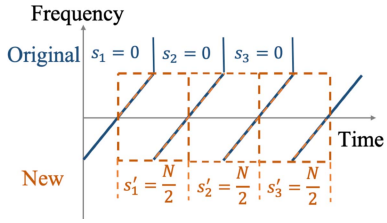
$$\Delta_m = \phi_m - \phi'_m, \tag{7f}$$

$$\alpha_m = \phi_m - \phi_{m-1}. \tag{7g}$$

Here $\vec{a} \cdot \vec{b}$ means to take the inner product of vector $\vec{a}$ and $\vec{b}$. (7f) is obtained via (3). Since we can combine N samples in a symbol to compute the fractional frequency peak η_m and its amplitude a_m, N times processing gain is still preserved to guarantee the accuracy of η_m and a_m. As to Δ_m, to make accurate



(a) STO Convergence



(b) STO Calculation

Fig. 5. (a). STO estimation deviation versus s_m under various SNR with SF10. In this case, there are 1024 frequency tone in a symbol. (b). An illustration as how to calculate the initial STO of a packet.

parameter calculation, we compute it only when $0.1N \leq s_m \leq 0.9\ N$. Similarly, α_m is calculated when $0.1N \leq s_m \leq 0.9\ N$ and $0.1N \leq s_{m-1} \leq 0.9\ N$. Therefore, in Feature Extraction module, for symbol s_m with $s_m < 0.1\ N$ or $s_m > 0.9\ N$, only η_m and a_m are used to identify different packets. In the Temporal Pattern Estimation & Feature Refinement module, Δ_m and α_m can be carefully recovered and used for entangling different packets even when $s_m < 0.1\ N$ or $s_m > 0.9\ N$.

The reason why we only estimate STO when $0.1N \leq s_m \leq 0.9\ N$ is the following. As has been illustrated in (3-4) (Section IV-A2), a chirp is composed of two parts, one starts from chip $0 to N - s_m - 1$ (used for estimating ϕ_m), and the other starts from chip $N - s_m$ to $N - 1$ (used for estimating ϕ'_m). The processing gain of the first and the second part is $N - s_m$ and s_m respectively. When s_m approaches to 0 or N, the processing gain of the first or the second part degrades significantly, leading to inaccurate STO estimation. Magnifier, therefore, does not use the STO feature to identify different symbols when s_m is close to 0 or N.

We further plot the average deviation of STO estimation when evaluated with different s_m in Fig. 5(a). The figure is obtained with the following steps. First, we receive one packet from a commodity LoRa chip. The SNR of the received packet is 50 dB. The STO estimated with the received packet is used as the ground truth of as its SNR is very high. We then add Gaussian noise to the received packet to manipulate its SNR. The SNR of the evaluated symbols ranges from -10 dB to 10 dB. Since the performance for $s_m = n$ and $s_m = N - n$ are similar, we only study the case when $s_m < N/2$. According to the figure, we can observe that for various SNR, STO estimation experiences large deviation when s_m is lower than 100 ($\approx 0.1\ N$), and converges to a relatively steady value when $s_m \geq 0.1\ N$. We therefore compute STO value when $0.1N \leq s_m \leq 0.9\ N$.

Note that, as SNR keeps decreasing (SNR lower than -10 dB), STO deviation converges with a larger value, e.g., $s_m \geq 0.1\ N$. However, as this work only focuses on extending the network concurrency but not improving reception quality for extremely low SNR packet, computing the STO value when $0.1N \leq s_m \leq 0.9\ N$ is efficient enough to retrieve accurate STO values. We leave the concurrent reception for extremely low SNR packet as our future work.

C. Initial Feature Extraction

In Magnifier, it is essential to retrieve the initial states from the preambles. Luckily, as will be detailed in Section V-A, a LoRa packet leverages multiple preambles to synchronize a transciever pair. The initial features extracted from multiple preambles also achieve multiple folds of processing gain compared with the payload symbols. Therefore, the initial features can be estimated accurately with high probability.

Even though, preambles are transmitted with the symbol $s_m = 0$, and STO cannot be extracted directly from (7f) as $\vec{d}_{m,2}$ is empty when $s_m = 0$. Luckily, due to the cyclic feature of the preambles, and the characteristics of CSS, by resetting the starting and ending time of the decoding window in the middle of two consecutive preambles, we can decode the preamble as symbol $N/2$ (where $N = 2^{sf}$ is the number of symbols that can be used in a chirp, and sf is the Spreading Factor used in the packet). There are two reasons to calculate STO of preambles in this way.

First, since STO stays almost constant in a chirp, we can approximate the STO estimated in the middle of the chirp to the value estimated at the beginning of a chirp; and

Second, as has been discussed in the previous comments, the STO estimation converges when $0.1N \leq s_m \leq 0.9\ N$, we thus set $s_m = N/2$ to calculate the STO of the preambles.

As the symbol carried by the newly generated preamble is $\frac{N}{2}$, we can easily calculate its STO with (7f). Denote the STO of the mth preamble and the newly generated symbol as $\Delta_{p,m}$ and $\Delta'_{p,m}$ respectively. $\Delta'_{p,m}$ can be expressed as $\Delta'_{p,m} \approx \frac{\Delta_{p,m}+\Delta_{p,m+1}}{2}$, where $\Delta_{p,m}$ and $\Delta_{p,m-1}$ are the STO of the mth and $(m+1)$th preamble. Since the difference between $\Delta_{p,m}$ and $\Delta_{p,m-1}$ is marginal, the initial STO of the first symbol in a packet can be approximated as $\Delta_0 \approx \Delta'_{p,n_{p-1}}$, where n_p is the index of the last preamble.

V. Design Architecture of Magnifier

This section describes each module in Magnifier first. Magnifier's computation complexity analysis is provided later.

A. Packet Management

The module is responsible for managing the arrival of new packets, and extracting their initial states. As shown in [3], a LoRa packet contains a set of preambles and 2.25 down-chirps (also named Start Frame Delimiter: SFD) as its header.

Correlation detection is performed to search the preambles and the SFD of a packet so as to achieve chirp level synchronization between the transceiver pair. To this end, we use a sliding window with n_{sf} (according to [20], n_{sf} is lower bounded by 6) preambles and 2.25 SFD to locate the last chip of the SFD. Once

been located the de-chirped preambles and the SFD should be demodulated with the same frequency bin. When the last chirp has been determined, we can extend the starting time of the sliding window so that more preamble chirps are included in the sliding window and used for estimating STO.

If the initial state fails, the consecutive symbols cannot be decoded with high probability. However, we argue that, the offsets information extracted from consecutive preambles achieves n_{sf} times of SNR compared with the symbols. Moreover, the 2.25 preambles can also be used for decoding a packet. The overall SNR gain of the preambles is $n_{sf} + 2.25 \geq 8.25$ higher than a normal symbol. If the initial state with more than 8.25 times SNR gain (corresponds to 9.1 dB) cannot be decoded, the probability to decode packet is very low. We leave how to decode concurrent packets with extremely low SNR as our future work.

Moreover, to suppress the frequency leakage from other packets (especially those with high received power), a Hamming window is applied to search the preambles. Once a newly arrived packet synchronizes (chirp level synchronization) to the receiver, its initial states will be extracted, and signals will be fed to the next module.

We next give a brief analysis regarding Magnifier's effectiveness of detecting consecutive non-interfering preambles for initial state estimation. In Magnifier, the corrupted preambles that are interfered with other transmissions can be easily identified as their amplitude and CFO values will deviate from the non-interfered preambles. Once been identified, Magnifier discards them before calculating the initial STO. Given k concurrent packets with SF=q, and assume a symbol distributes its energy to two frequency bins due to frequency offset (the two closest frequency bins to the frequency peak that carries the symbol), the probability that a preamble collide with other transmissions is $p = 1 - (1 - 2/2^q)^{k-1}$. E.g., for 14, 24 and 30 nodes with SF8, SF10 and SF12 (the maximum capacity for Magnifier as shown in the evaluations), p=9.7%, 4.4% and 1.4% respectively. With 6 or more preambles, the probability to retrieve at least one correct STO value is higher than 99%

B. Candidate Selection

As shown before, padding $(I-1)N$ zeros and performing a NI size FFT for extracting chip-level features will either incur extremely high computation complexity (with large I) or affect the accuracy of chip-level features (with small I). We further observe that it is useless to calculate the chip-level features of a frequency bin that does not carry any symbols. Therefore, only a set of candidate frequency bins are to be inspected with (7) discussed in Section IV-B. We first perform a synthesis correlation to determine the candidate set. In particular, let

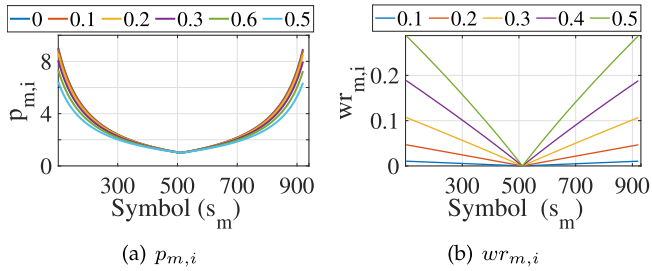
$$\vec{e}_{m,i,1} = \left\{ e^{j\frac{2\pi i n}{N}} | 0 \leq n < N - i \right\}, \tag{8a}$$

$$\vec{e}_{m,i,2} = \left\{ e^{j\frac{2\pi i n}{N}} | N - i \leq n < N \right\}, \tag{8b}$$

$$\vec{r}_{m,i,1} = \{ r_{m,n} | 0 \leq n < N - i \}, \tag{8c}$$

$$\vec{r}_{m,i,2} = \{ r_{m,n} | N - i \leq n < N \}, \tag{8d}$$

$$w_{m,i,1} = |\vec{e}_{m,i,1} \cdot \vec{r}_{m,i,1}|, \tag{8e}$$



(a) $p_{m,i}$ (b) $wr_{m,i}$

Fig. 6. (a). The integer frequency bin's amplitude ratio ($p_{m,i}$). (b). Amplitude ratio deviation of the integer frequency bin to the frequency bin with the fractional part ($wr_{m,i}$) for different symbols. Different lines indicate the ratio with various fractional frequency bins (η_m).

$$w_{m,i,2} = |\vec{e}_{m,i,2} \cdot \vec{r}_{m,i,2}|, \tag{8f}$$

$$w'_{m,i} = w_{m,i,1} + w_{m,i,2}, \tag{8g}$$

$$p_{m,i} = \max\left\{ \frac{w_{m,i,2}}{w_{m,i,1}}, \frac{w_{m,i,1}}{w_{m,i,2}} \right\}, \tag{8h}$$

$$wr_{m,i} = |p_{m,i} - \max\left\{ \frac{s_m}{N - s_m}, \frac{N - s_m}{s_m} \right\}|. \tag{8i}$$

Here $w'_{m,i}$ in (8g) is used to represent the overall amplitude of a symbol. Notice that, unlike traditional work, the amplitude of a symbol carried by a frequency peak (say symbol i) is not obtained with a standard FFT approach. Instead, we compute the correlation of the first $N - i$ and the last i chips separately and combine the absolute value of the two terms to derive the overall amplitude of a frequency peak. The reason why we calculate the amplitude with (8g) is the following. According to (3) in Section IV, due to STO, there is a sudden phase shift between the former $N - i$ and the later i chip-sets. For instance, according to (3) and (4b), the such phase shift is Δ_m. As Δ_m is unknown at this stage, we use the absolute value of the two chip-set to approximate the amplitude of a frequency bin. Note that prior work uses the standard FFT approach to compute the amplitude of a frequency bin and fails to handle such drifts, leading to higher amplitude estimation inaccuracy. $p_{m,i}$ is the ratio of $w_{m,i,1}$ and $w_{m,i,2}$, and $wr_{m,i}$ is the ratio difference.

Similar to the approach discussed in [9], $wr_{m,i}$ indicates the likelihood that the current transmitted symbol is s_m. Specifically, the smaller the $wr_{m,i}$, the higher the chance that frequency bin f_i is mapped to symbol s_m. $wr_{m,i}$ can be treated as a chirp-level feature. Ideally, when $\eta_m = 0$, there is no fractional frequency drift; $wr_{m,i}$ approaches 0 if frequency bin f_i carries the mth symbol. Unfortunately, the non-zero fractional frequency drifts η_m invisible in chirp-level information leads to inaccurate $wr_{m,i}$ estimation. Fig. 6 shows $p_{m,i}$ and $wr_{m,i}$ with various η_m. According to the figure, the maximum ratio difference is upper bound by 0.3 when $0.1N \leq m_s \leq 0.9\ N$. Moreover, refer to Fig. 3(b), the amplitude of the integer frequency bin is larger than 60% of the actual one. Therefore, f_i is added into the candidate set once $wr_{m,i} \leq 0.3$ and $0.6\bar{a} \leq w'_{m,i} \leq \frac{\bar{a}}{0.6}$. Here $\bar{a}$ is the average amplitude of the packet. This value can be derived from the preambles. As to symbols smaller than $0.1 * N$ or larger

than $0.9 * N$, only the amplitude $w'_{m,i}$ is used to determine if a frequency bin will be added to the set.

The next module only investigates the chip-level features of all frequency bins in the candidate set, and the Candidate Selection module reduces the complexity significantly.

C. Feature Extraction

The fine-grained chip-level features of each frequency bin in the Candidate set are evaluated for every symbol in a packet. The initial CFO, STO, amplitude, and phase of the frequency peaks of a newly arrived packet are first extracted from the preambles with (7) discussed in Section IV-A. Regarding the transmitted symbols, one of the frequency bins in the candidate set will be assigned to the packet if possible. Recall that users' chip-level features keep evolving with time. However, the exact temporal patterns of such information are yet to be identified and cannot be used to assign frequency bins to the packet at this stage. We address this problem by leveraging the short-term temporal pattern of the fine-grained features. In other words, chip-level features change slowly with time, and adjacent symbols in one packet have similar fine-grained chip-level features.

In detail, we first define a tuple representing the chip level features of the mth symbol as $\{\eta_m, \Delta_m, \alpha_m, a_m\}$. A threshold tuple $\{\tau_d, \tau_D, \tau_\phi, \tau_a\}$ is also defined. For the mth symbol, among all the frequency bin candidates, the frequency bin i (with frequency drift, STO, phase difference, and average amplitude denoted as $\tilde{\eta}_i$, $\tilde{\Delta}_i$, $\tilde{\alpha}_i$, and $\tilde{a}_i$ respectively) satisfies the following conditions will be selected.

$$|\tilde{\eta}_i - \eta_{m-1} \pm v| \leq \tau_d \tag{9a}$$

$$|\tilde{\Delta}_i - \Delta_{m-1} \pm v| \leq \tau_D \tag{9b}$$

$$|(\tilde{\phi}_i - \phi_{m-1}) - \alpha_{m-1} \pm 2v\pi| \leq \tau_\phi \tag{9c}$$

$$\frac{|\tilde{a}_i - a_{m-1}|}{a_{m-1}} \leq \tau_a \tag{9d}$$

Here $\{\eta_{m-1}, \Delta_{m-1}, \alpha_{m-1}, a_{m-1}\}$ is the extracted features of the $(m-1)$th symbol, and $v \in \{-1, 0, 1\}$. v is introduced in (9) as η_{m-1} and Δ_{m-1} can be larger than one frequency bin or one chip. Recall that (7) simply wraps η_{m-1} to a fractional value. By introducing v, once $|\eta_{m-1} - \eta_m + 1| \leq \tau_d$, we update $\eta_m = \eta_m - 1$, and $\eta_m = \eta_m + 1$ when $|\eta_{m-1} - \eta_m - 2| \leq \tau_d$. Similar operations are performed on α_m, and Δ_m.

As to the thresholds, a larger threshold is more robust to environmental noise and frequency leakage from other concurrently transmitted packets. However, it will also increase the chance that more frequency bins satisfy (9), leading to decoding ambiguity. In the evaluations, we set a lower threshold for high SNR packets and a higher threshold for low SNR packets to strike a balance. Besides, if more than one frequency bins meet the requirement (9), or none of the frequency bin meets the requirement, no frequency bin is assigned to that symbol. In this case, $\{\eta_{m-1}, \Delta_{m-1}, \alpha_{m-1}, a_{m-1}\}$ will be updated with $\{\eta_{m-2}, \Delta_{m-2}, \alpha_{m-2}, a_{m-2}\}$. The procedure will halt if more than ten consecutive symbols are not assigned with any frequency bin. The results will be passed to the next module to purify the chip-level features.

Note that, when nodes are moving, the channel state between the transceiver will change. However, as will be explained in the next subsection, the channel variations have a marginal impact on the *changes* in the offsets in the system. Moreover, since a symbol lasts no more than several tens of milliseconds (most of the time, it only lasts several hundred microseconds), it is reasonable to assume that the channel state between several consecutive symbols stays constant, and the phase differences between consecutive symbols are close. Similarly, the amplitude of the channel also remains stable for consecutive symbols. Therefore, Magnifier's performance will not be affected significantly due to the channel dynamics. We will leave how to deal with a time-varying channel in our future work.

In this module, the STO and the phase difference (α_m) are not always available to determine a frequency-symbol pair. E.g., when $s_m > 0.9\ N$ or $s_m < 0.1\ N$, its STO information is not calculated as discussed in Section IV-B. When the $m-1$th symbol is not determined, we cannot extract ϕ_{m-1}, let along derive α_m. Such missing information will be constructed in the next module.

D. Temporal Pattern Estimation and Feature Refinement

Once a set of frequency bins and associated tuples $\{\eta_m, \Delta_m, \alpha_m, a_m\}$ are derived, the temporal patterns of all the symbols are estimated. First, we apply linear regression to approximate the η_m and Δ_m for all symbols (with or without an assigned frequency bin) in a packet. The reason that we use linear regression are the following. Offsets variations in a packet are mainly due to two factors: hardware imperfections, and channel state changes.

First, regarding hardware imperfections, as has been implied in Fig. 2, both CFO and STO experience a continuous, and quasi-linear drift. This is reasonable as CFO and SFO are caused by different oscillators used at the transmitter and the receiver side. Moreover, a properly designed oscillator can maintain a stable working frequency over time, especially for a short period of transmitting one single packet. Similarly, the temperature of a chip tends to stay constant when transmitting a single packet. It thus merely contributes to the offset changes within a packet. Therefore, we can ignore the non-linear changes of the offsets due to hardware imperfections;

Second, as to channel state changes, for a narrow band system such as LoRa, the STO and SFO *changes* due to channel variations (usually caused by moving objects in the environment or a moving transmitter or receiver) are negligible. Similarly, there are merely CFO *variations* when transmitting a single packet.

Put it together, as the randomness of hardware imperfections and channel state variations have a marginal impact on the changes of STO and CFO, it is reasonable to assume that CFO and STO are solely affected by SFO. Therefore, we can approximate them to be linear functions of time. It is worth to be noticed that Magnifier can easily extend the linear regression to non-linear regressions if necessary.

Once the temporal patterns of η_m and Δ_m are established, we use (4c) and (5b) to construct the temporal pattern of α_m. As to the amplitude feature, we use interpolation to estimate the amplitude of symbols that have not been assigned with a frequency bin.

The estimated tuples are denoted as $\{\eta'_m, \Delta'_m, \alpha'_m, a'_m\}$. We replace the already extracted features in the previous module ($\{\eta_m, \Delta_m, \alpha_m, a_m\}$) with the estimated ones to detect the wrongly paired frequency-symbol pairs. Moreover, as the estimated features are established, we use the complete information (which may not be available in the previous module) to decode symbols that have not been assigned with any frequency bin. If they cannot be decoded, e.g., some symbols experience symbol collision with other packets, we pass them to the next module.

E. Joint Decoding

The previous modules help decode most of the symbols in each packet. However, when two or more packets transmit data with the same frequency bin or adjacent frequency bins, neither chip-level nor chirp-level features can be preserved. To this end, a joint decoding scheme is developed.

For each symbol that has not been assigned a frequency bin, we first erase the frequency bins in that symbol duration generated by other concurrently transmitted packets. The procedure can be done by reconstructing their signals with the signal model introduced in (3). Let $\vec{q}$ be the set of packets whose symbol has not been determined; $\vec{v}$ be the set of frequency bins whose value is greater than a given threshold δ_a, z_v be a complex value of the zth frequency bin. $F_v(\vec{x})$ is the complex value of the vth frequency bin after FFT (note that $\vec{x}$ is the time domain signal).

$$\arg \min_{\{b_{u,i}|u\in\vec{q},i\in\vec{v}\}} \sum_{i\in\vec{v}} |F_v\left(\sum_{u\in\vec{p}} b_{u,i}\cdot\vec{r}_m\right) - z_v|^2 \tag{10a}$$

$$\text{s.t.} \sum_{i\in\vec{v}} b_{u,v} = 1, \tag{10b}$$

$$b_{u,v} \in \{0,1\}. \tag{10c}$$

Here $\vec{r}_m = \{r_{m,n}, 0 \le n < N\}$, and $r_{m,n}$ is defined in (3). Since collisions do not frequently happen as long as the number of current packets is small (when the number of concurrently transmitted packets is 30 with SF=10, the symbol collision probability is about 10%), the size of the symbol and frequency set to be determined is limited. Moreover, as the probability that three or more symbols collide on the same frequency bin is marginal, we only deal with the case when two symbols collide. The optimization problem can be resolved efficiently.

Note that, though Successive Interference Cancellation (SIC) as has been discussed in [14] can work well for two or three concurrent nodes. Its performance degrades severely with higher number of concurrent packets. For one thing, the interference caused by inaccurate channel estimation can corrupt the received signal significantly. For the other, SIC cannot be applied to separate packets with similar received energy. We therefore apply the joint decoding module instead of SIC to separate different nodes.

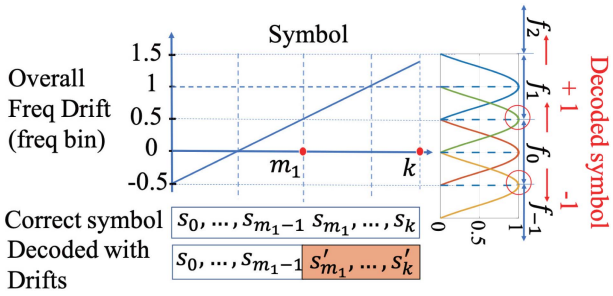


Fig. 7. Decoding error due to the lack of the overall STO information. When the overall frequency drift exceeds ± 0.5 chip, the decoded symbol will deviate ± 1 from the correct one.

F. Overall Central Frequency Drift Compensation

After a packet has been decoded, we use this module to compensate for each symbol's overall central frequency drift. According to (5a), the overall frequency drift is affected by both STO and CFO. Though the chirp level CFO can be easily estimated from preambles and stays relatively stable in a packet, the chip level CFO and STO grow with time. Simply compensating for the central frequency drift with CFO (as done by existing works) will lead to consecutive symbol errors when η_m exceeds 0.5 frequency bin.

Fig. 7 illustrates this issue. When the overall frequency drift is within $[-0.5, 0.5]$ frequency bin, the symbols are decoded correctly (denoted as $\{s_{m_1} \cdots s_{m_2-1}\}$). When the frequency drift exceeds 0.5 frequency bin but is no more than 1.5 frequency bins, the set of symbols decoded with chirp level CFO information, designated as $\{s'_{m_1} \cdots s'_k\}$, shifts one frequency bin. That is $s'_i = s_i + 1$.

Though the traditional approaches fail to handle the such problem, the cumulative frequency drifts can be easily compensated for with the temporal patterns of the fine-grained features. Recall to (4c), s_m is compensated as

$$s_m \approx [f_m - \Delta_m - \delta_{m,f}]. \tag{11}$$

Here $[x]$ means to take the integer part of x.

G. Complexity Analysis

We analyze the complexity of Magnifier by evaluating its complexity in each module.

1) The Complexity of Packet Management Module: Packet Management will be used to perform coarse-grained chirp-level synchronization. Denote the number of preambles to be P. This procedure needs to de-chirp $P + 2.25$ symbols and perform a standard $(N)^2$ size FFT for each of them (recall that N is the number of samples in a symbol), the associated complexity is $O((P + 2.25)N^2) \to O(N^2)$.

2) The Complexity of Candidate Selection Module: For the mth symbol in a packet, a standard FFT is required to calculate $p_{m,i}$, $wr_{m,i}$, and the chirp level amplitude information. Let L be the average packet length, and the overall complexity is $O(LN^2)$.

3) The Complexity of Feature Extraction Module: As has been mentioned in Section V-B, for all symbols in the packet, the complexity of calculating the chip level features for all frequency bins in the candidate set is $O(LK(NI))$. Note that K is usually significantly lower than N as most frequency bins are excluded from the Candidate Selection module. E.g., with $SF = 10$ and 40 concurrently transmitted packets, according to [13], since symbols with time-misalignment smaller than 10% cannot be distinguished, the average number of packets in the candidate set is about 5 ($N = 1024$, and $K << N$). Furthermore, with $I = 100$, KI and N are of the same scale. We hence approximate $O(LKNI)$ with $O(LN^2)$.

4) The Complexity of Temporal Pattern Estimation and Feature Refinement Module: The complexity of performing linear regressions to construct the temporal patterns of the features is $O(L^2) < O(LN^2)$.

5) The Complexity of the Joint Decoding Module: This module is performed via exhaustively enumerating all frequency-symbol pairs for the collided packets. Suppose the average number of collided symbols is C, and the pairing complexity is $O(K^C)$. Likewise, as the joint decoding module only deals with the case when at most, two symbols collide with each other, the complexity can be approximated with $O(K^2) < O(N^2)$. The overall complexity of this module is upper bounded by $O(LN^2)$.

6) The Complexity of Overall Frequency Drift Compensation Module: Refer to (11), the complexity of this module is $O(L)$. To summarize, the overall complexity of decoding one packet in Magnifier is $O(LN^2)$, comparable to the existing coarse-grained chirp-level-based approaches.

VI. Evaluations

This section first shows the experimental settings. The results will be presented later.

A. Experimental Settings

We adopt commodity LoRa chips SX1278 as the transmitters and a USRP B210 as the receiver. USRP is connected to a PC running GNU Radio to collect the raw signals from transmitters. MATLAB is then used to process the collected signal. The default setting is listed as follows: transmission bandwidth = 250 KHz; the transmission power = 10 dBm; and the sampling frequency at USRP = 250 KHz. Likewise, 40 nodes are deployed in the system. The experimental deployment is shown in Fig. 8(b). In the experiment, each of node randomly selects a time to start transmitting packets. The transmission interval is randomly chosen within a duration. Note that transmission duration affects the network's overall throughput, which will be studied in the results. The USRP listens to the channel all the time. In each round of evaluation, all the nodes use the same channel with the same SF to transmit data. The network performance with various SF (SF8, SF10 and SF12) are investigated in different rounds of evaluations. Moreover, the gateway listens to the channel, and receives data all the time when it is turned on. A round of experiment lasts 22 seconds. In this case, each node transmits about 100 packets. For a given SF, three individual rounds of evaluations are collected.

(a) (b)

Fig. 8. (a) Devices used for the evaluations. (b). Experimental deployment of a 40-node network. The size of the map is 1400 m×800 m. The red triangles represent the LoRa transmitters, and the blue circle is the receiver.

TABLE II
Standard and Maximum Deviations of the Chip Level Features Under Various SNR

	SNR	10dB	-0.3dB	-7.2dB	-10.2dB
SF=8	$\delta_m(\%)$	0.76/2.1	2.67/5.4	6.1/10.6	7.4/12.7
	Δ_m (%)	4.4/4.2	5.0/5.2	6.4/8.7	10.8/12.6
	$\alpha_m(\%)$	1.3/3.5	2.3/4.3	5.0/9.5	6.7/11.9
	a_m (%)	0.52/5.6	2.3/9.8	5.0/11.2	6.0/12.1
SF=10	δ_m (%)	0.49/2.9	2.0/5.6	3.3/7.9	5.8/10.2
	$\Delta_m(\%)$	1.8/3.2	2.9/4.5	5.6/6.7	7.0/8.3
	$\alpha_m(\%)$	0.48/2.6	2.1/4.8	2.9/5.9	3.5/7.8
	a_m (%)	1.1/2.5	2.9/4.3	2.6/5.4	3.6/7.04
SF=12	δ_m (%)	0.40/1.2	1.3/3.8	3.0/8.1	4.3/8.5
	Δ_m (%)	0.32/1.1	0.93/2.2	2.3/4.8	3.1/5.1
	α_m (%)	0.37/3.2	1.1/4.1	2.7/6.6	3.4/8.3
	a_m (%)	0.33/1.1	0.95/1.2	1.7/3.4	2.9/5.2

To ease the experiment, each node repeatedly transmits the same payload (the payload length is 12 Bytes) to the gateway. The coding rate is $1/2$. The preamble lasts for 10 symbols. 4.25 additional symbols follows the preambles (2 symbols with the device ID and 2.25 down-chirps). According to [18]. The packet duration for SF8, SF10, and SF12 is 47.4 ms, 156.7 ms, and 626.7 ms, respectively. Each node is assigned with a unique ID number so as to identify the packet from it. Moreover, since the average packet duration is known, it is easy to determine how many packets, and when they are received.

B. Evaluation Results

1) The Accuracy of Extracting the Chip Level Features Under Various SNR: We first show the accuracy of extracting the chip-level features under different SNR. We collect 100 packets with SNR = 30 dB and use them as the baseline. White Gaussian noise is added manually into each received packet to keep the chip level offset the same as the baseline packet. Moreover, coarse-grained chirp level synchronization is performed before computing the chip level offsets. Table II shows the Standard Derivation (STD) of different parameters with various SNR when SF = 8, 10, and 12. Since maximum estimation error of δ_m (central frequency drift), Δ_m (STO) and α_m (phase difference of the frequency peaks of consecutive symbols) are ± 1, and the deviations are upper bounded by 1. As to the amplitude, we evaluate the STD of the normalized amplitude. Here the normalized amplitude is defined as $\tilde{a}_m = \frac{|a_m - a'_m|}{a_m}$ where a'_m is the amplitude of the mth

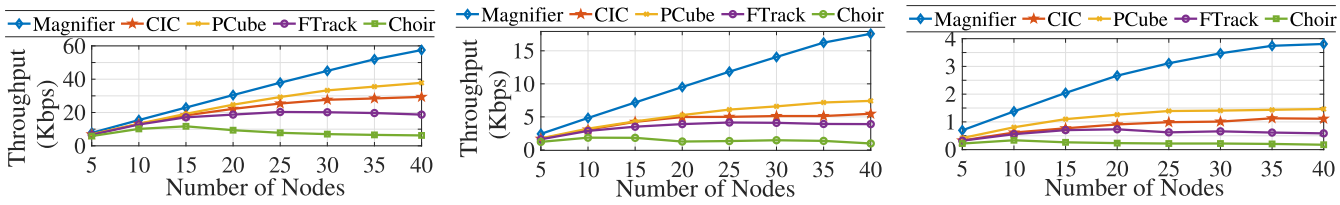


Fig. 9. Throughput Performance with different SF.

symbol with a different noise, and a_m is the value obtained with SNR $= 30$ dB.

The table shows that the higher the SF, the larger the processing gain and the lower the deviation. Moreover, for various SF, as the processing gain is properly leveraged in Magnifier, accurate chip-level parameter calculation can be achieved for signals with SNR far below 0 dB. E.g., when SNR is around -10 dB, the standard deviation is about 3% to 10.8%, and the maximum deviation of all parameters is less than $\pm 13\%$. Based on the results in the table, we set the thresholds used in Section V-C as follows. $\{\tau_d, \tau_D, \tau_\phi, \tau_a\} = \{k, k, k, k\}$ and $k = 0.15$ when the average SNR obtained from preambles is lower than -5 dB; $k = 0.1$ when -5 dB$<$SNR<5 dB; and $k = 0.08$ when SNR>5 dB. We leave how to select the optimal thresholds as our future work.

2) Comparing Magnifier's Performance With Existing Works: In this section, we first compare Magnifier's throughput performance with CIC [17], PCube [15], FTrack [13], and Choir [16]. All schemes' sampling rate is set to the transmitter's bandwidth (250 KHz), and PCube is realized with two receiving antennas to make a fair comparison.

Fig. 9 plots the throughput for various schemes with SF=8, 10, and 12. Here throughput is defined as the total number of bits (including preambles) collected by the receiver in one second. The number of transmitting nodes in the system increases from 5 to 40. The average packet duration for SF=8,10, and 12 is 47.4 ms, 156.7 ms, and 626.7 ms, respectively. The average packet interval is 160 ms. Since the packet duration for SF=8 is much smaller than that of the other SF, all the schemes with SF=8 show better throughput performance than other SF. Similarly, when SF=12, all schemes show significantly lower throughput than other SF.

The figure also shows that PCube provides higher throughput performance than CIC, FTrack, and Choir, as it can extract accurate phase information with two receiving antennas. Still, equipped with multi-dimensional and precise information, Magnifier shows 52% to 137% higher throughput performance than PCube. Notice that Magnifier only requires one receiving antenna instead of adopting two antennas. If MIMO is used in Magnifier, tremendous throughput improvement can be expected since the phase information discussed in PCube can be treated as one feature in Magnifier. As to CIC and FTrack, Magnifier provides 2.0× to 3.2× throughput improvement. According to our result, CIC only performs slightly better than FTrack. The phenomenon is partly because, in our evaluation, the sampling rate of CIC is the bandwidth. Recall that the sampling rate used in CIC in [17] is as high as eight times the bandwidth (9 dB additional processing gain). Choir shows the lowest throughput performance due to its inaccurate frequency drifts estimation.

We then define the average concurrency of a packet as follows. Let $c_{i,m}$ be the number of concurrently transmitted symbols from other packets when the mth symbol of packet i is transmitted. Two symbols are assumed to be transmitted concurrently only when one symbol's transmission duration overlaps with more than 15% of the other. The effective concurrency is defined as $\bar{c}_i = (\sum_m c_{i,m})/p_l$, where p_l is the number of symbols in a packet. For a packet with an effective concurrency of $k - 1 < \bar{c}_i \leq k$, where k is an integer, we then quantize its concurrency as k.

Fig. 10 shows the Symbol Error Rate (SER) versus the number of concurrently transmitted packets in the network with various SF. According to the figure, lower SF supports a lower degree of concurrency. The reason is evident: the lower the SF, the higher the spectrum leakage of a concurrently transmitted packet, and the higher the interference it brings to other nodes.

Another interesting observation is that with higher SF, all the schemes but Magnifier suffers from a higher symbol error rate even with a small number of concurrency. E.g., when SF=12, CIC, Pcube, and FTrack show more than 25% symbol error rate even when the number of concurrently transmitted nodes is 2. This can be easily understood as with the increase of the SF, the packet duration and the cumulative frequency drift grows larger. Once the overall frequency drift exceeds ± 0.5, the consecutive symbol error occurs. Magnifier, however, can compensate for such drift by tracking the temporal patterns of the fine-grained offsets while decoding a symbol. Hence, different SF and packet duration will not affect its performance.

Regarding the specific performance of different SF, we observe that Magnifier shows significantly better SER performance than other schemes. Specifically, with SF8, SF10, and SF12, Magnifier can support up to 13, 24, and 28 concurrently transmitted packets, respectively, when SER$\leq$35%. The achievable concurrent transmissions for CIC and PCube is 6 to 9, and for Choir and FTrack is 3 to 5.

To summarize, Magnifier's excellent performance comes from two aspects. First, contrary to FTrack and CIC (with a limited number of inaccurate features) or PCube (with phase difference information), or Choir (with inaccurate frequency drift information), Magnifier uses four accurate features and their temporal patterns. It hence provides better decoding ability when the degree of concurrency increases. Second, other schemes but Magnifier fail to trace and take care of the temporal pattern

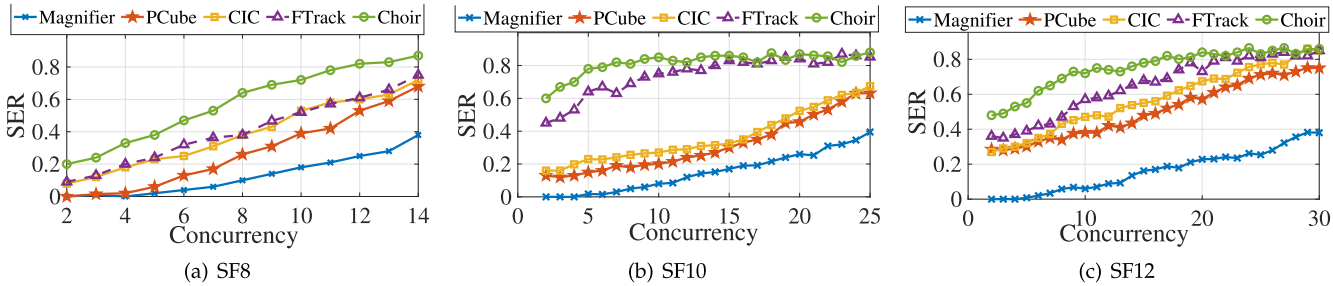


Fig. 10. SER versus the network concurrency with different SF.

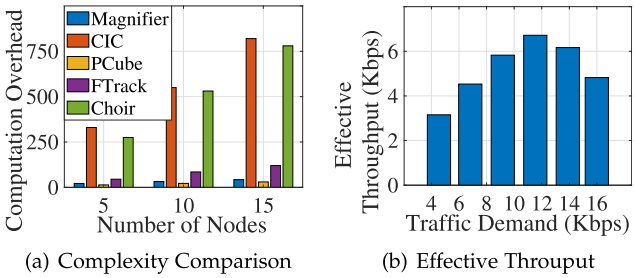


Fig. 11. Performance comparison with SF10.

of the fine-grained frequency drifts. It is also observed that as the payload size and packet transmission time increases, the throughput loss due to the cumulative frequency drift becomes even more remarkable.

C. Complexity Analysis

In this part, we evaluate the complexity of Magnifier with experiments. The performance trend for SF8, SF10, and SF12 are similar. Due to the page limitation, in the rest of the paper, we only discuss the case with SF10. We run all the schemes on a PC with 5, 10, and 15 concurrently transmitted nodes. The decoding time is normalized to the time consumed by a standard LoRa decoder. Fig. 11(a) shows the normalized decoding complexity. According to the figure, the complexity of Magnifier is comparable to that of PCube, and is much lower than that of Choir and CIC. It can be explained as follows. Due to the feature extraction approach and the candidate set selection algorithm discussed in the design, the complexity of extracting frequency offsets in Magnifier only grows linearly with the number of concurrently transmitted nodes in the network. Choir and CIC, however, extract the chip level fractional frequency offset by padding $K \times N$ (to increase the identification resolution, K is set to a large value, e.g., $K \geq 10$ in Choir) zeros at the end of the original signal and performing a large size FFT. The complexity thus is K^2 times that of a standard FFT.

D. Effective Network Throughput Versus Traffic Demands

Effective network throughput is defined as the number of successfully received payloads from the whole network. Traffic Demand represents the total amount of bits (including payloads, preambles, and SFD) sent out by all the nodes in the network. Fig. 11(b) shows the effective throughput of the 40 nodes network with different traffic demands. At first, the effective throughput increases with the traffic demand. However, as traffic demand keeps growing, the number of concurrently transmitted packets increases significantly, leading to high SER. The increase in SER further affects the effective throughput of the network.

E. Magnifier's Performance With Different SNR

We further investigate Magnifier's SER performance with different SNRs. Here we only discuss the case with SF=10. The range for high, medium, and low SNR is set as > 5 dB, $[-5\text{ dB}, 5\text{ dB}]$, and < -5 dB, respectively. Based on this setting, the portion of nodes with high, medium and low SNR is 18.2%, 61.6%, and 20.2%, respectively. Fig. 12 plots the CDF of SER for packets with various SNRs. In the figure, the average number of concurrently transmitted packets equals 10, 15, 20, and 25, respectively. We can observe that packets with high SNR are more robust to interference from other concurrent transmissions. In contrast, the SER for packets with low SNR proliferates with the increase of the concurrently transmitted packets. A higher degree of concurrency is expected to bring in higher interference, and low SNR packets are more error-prone to such interference. As to medium SNR packets, their performance is close to high SNR packets when the number of concurrently transmitted packets is below 20. However, it suffers explicit performance degradation when network concurrency grows.

VII. Related Works

LoRa concurrent reception has drawn extensive research attentions [21]. Several studies aim to improve concurrent reception in LoRaWAN via MAC or PHY layer solutions. We briefly summarize these approaches in this section.

A. MAC Layer Approaches for Collision Avoidance

MAC layer aims at leveraging scheduling schemes for collision avoidance. Prior work [6], [22], [23], [24] set different SF, bandwidth, and time slots for different users to eliminate transmission collisions. However, as no coordination is performed between different transmitters and the receiver, collisions still happen, and there is no practical way to handle such collisions. By leveraging the Channel Activity Detection (CAD) feature of LoRa radios, LMAC [5] proposes to detect preamble chirp to achieve reliable carrier sense. Prior work [25] analyzes the

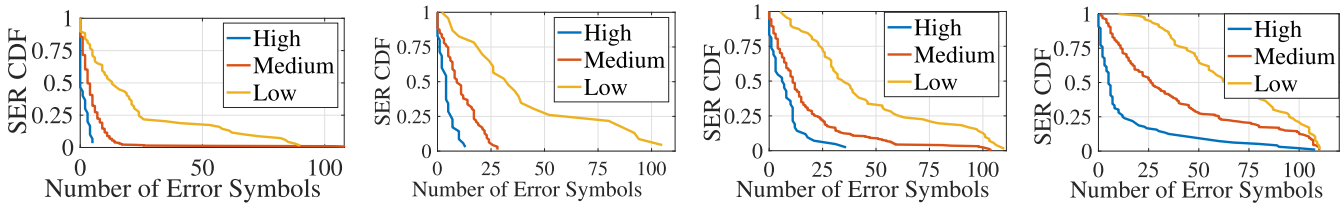


Fig. 12. CDF of erroneously decoded symbols per packet with different SNR when the number of concurrently transmitted packets equals.

limitations of LoRa Channel Activity Detection. It proposes a channel access method with a lightweight collision avoidance mechanism that can operate without a reliable Clear Channel Assessment procedure. Still, implementing CAD requires additional firmware changes at the transmitter. Besides, CAD cannot wholly resolve collisions when the transmitters are far from each other and thus cannot sense the transmissions from others.

B. PHY Layer Solutions for Concurrent LoRa Reception

1) Chirp Level Approaches: FTrack [13], CoLoRa [9], Pyramid [11], SCLora [8], NScale [10], AlignTrack [12] and OCT [26] exploit the fact that different packets arrive at the receiver at different time, and hence their symbol boundaries are different. FTrack [13] leverages the frequency continuity to identify if a frequency bin belongs to the target packet. Besides the different arrival times, Pyramid [11], AlignTrack [12], and SCLora [8] further adopt the average energy of each packet for parallel decoding. Linear and non-linear scaling are applied to scale different chips in CoLoRa [9], and NScale [10]. The energy distributions of frequency bins in two successive symbols are examined to map frequency bins to packets. Since all these schemes exploit the coarse-grained chirp-level features, they have poor packet identification resolution.

CurvingLoRa [27] also proposes a non-linear scaling approach to modulate and demodulate concurrent LoRa transmission. Though effective, the non-linear approach requires a brand new modulation and de-modulation scheme, leading to tremendous hardware change. Hence it cannot be applied to LoRaWAN composed of COTS LoRa chips.

Successive Interference Cancellation (SIC) is discussed in mLora [14] for jointly decoding multiple packets. When SIC is applied, the decoding efficiency depends on the accuracy of the signal model. Unfortunately, only chirp-level information is considered, and the chip-level features are entirely ignored, resulting in substantial decoding errors. According to [14], mLora can support two to three concurrent transmissions, which shows significantly lower efficiency than Magnifier. Netscatter [28] enables large-scale concurrent transmissions from backscatter devices. It only works for short-range communication as the backscattered signal is usually much lower than regular transmissions. It also requires strict synchronization of all transmitters, which is unsuitable for distributed implementation.

2) Schemes Partially Involve Chip-Level Information Into Consideration: Choir [16] uses a simplified model that ignores the existence of STO and SFO (assume STO=0 and SFO=0). Moreover, the temporal patterns of such offsets are neglected. E.g., CFO is considered to be fixed in the whole packet. With these assumptions, the location of the frequency peaks of the packet is assumed to remain constant. However, as discussed previously, the location of the frequency peak changes due to SFO. Simply adopting a fixed location of the frequency peak makes the receiver quickly lose track of the actual location of the peak, resulting in packet reception failure. Regarding CIC [17], it employs both the chirp level frequency and energy information and the chip-level frequency drift features (the same as Choir). In particular, it examines and filters out all frequency bins that do not exist in the whole symbol duration with chip-level offsets. The frequency bin sent by the packet under investigation emerges. Since the chirp-level time misalignment information and the chip-level CFO information proposed in Choir are inaccurate, CIC's performance degrades significantly when concurrently transmitted packets increase. As to PCube [15], instead of exploiting the excellent properties of the chip level features, it adopts a MIMO system to extract the phase difference between two antennas and remove the chip level effects. PCube, thus cannot be deployed in the system with a single receiving antenna. Moreover, though removing the chip-level offsets can help to obtain accurate and stable phase information of each packet, the phase information alone (*single dimension*) will quickly lose its reception power as the number of concurrently received packets increases. E.g., two packets with a similar Direction of Arrival (DoA) can merely be separated by the phase difference. It thus involves more receivers with independent channel states to resolve the ambiguity, leading to a cumbersome implementation.

C. Applying Offsets Information to Other Applications

Offsets exist for almost all wireless networks. For instance, in Radar systems, such as Impulse Radio Ultra-Wideband (IR UWB) [29] and mmWave Radar [30], via observing the changes in the reflected signal, we can estimate the location and the speed of an object with centimeter-level accuracy. The fine-grained STO is also used for sensing and tracking an object in Wi-Fi system as has been discussed in [31].

VIII. Conclusion

Magnifier, a concurrent LoRa reception framework, is proposed in this work. By elegantly extracting the fine-grained chip-level offsets such as CFO, STO, and SFO and their temporal

patterns, Magnifier enables accurate and multi-dimensional features for decoding concurrently transmitted packets. Extensive results imply that Magnifier provides 3× to 5× throughput of existing schemes with the same hardware settings.

Effective as it be, Magnifier still faces some limitations. For instance, Magnifier cannot decode packets with extremely low SNR efficiently. Likewise, the degree of network concurrency Magnifier can support is still limited as it is incapable of resolving large scale of frequency tone collisions. In future work, we will strive to extend such limitations by introducing multiple receiving antennas to our system, and explore the degree of freedom to increase its ability for concurrent reception.

REFERENCES

[1] V. N. I. Cisco, "Forecast and methodology, 2018–2023," *White Paper*, pp. 1–41, 2018.

[2] J. P. S. Sundaram, W. Du, and Z. Zhao, "A survey on LoRa networking: Research problems, current solutions, and open issues," *IEEE Commun. Surv. Tut.*, vol. 22, no. 1, pp. 371–388, First Quarter 2020.

[3] LoRa Alliance, "Lorawan specification," 2015.

[4] J. C. Liando, A. Gamage, A. W. Tengourtius, and M. Li, "Known and unknown facts of LoRa: Experiences from a large-scale measurement study," *ACM Trans. Sensor Netw.*, vol. 15, no. 2, pp. 1–35, 2019.

[5] A. Gamage, J. C. Liando, C. Gu, R. Tan, and M. Li, "LMAC: Efficient carrier-sense multiple access for LoRa," in *Proc. 26th Annu. Int. Conf. Mobile Comput. Netw.*, 2020, pp. 1–13.

[6] Y. Li, J. Yang, and J. Wang, "DyLoRa: Towards energy efficient dynamic LoRa transmission control," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 2312–2320.

[7] Z. Xu, J. Luo, Z. Yin, T. He, and F. Dong, "S-MAC: Achieving high scalability via adaptive scheduling in LPWAN," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 506–515.

[8] B. Hu, Z. Yin, S. Wang, Z. Xu, and T. He, "SCLoRA: Leveraging multi-dimensionality in decoding collided LoRa transmissions," in *Proc. IEEE 28th Int. Conf. Netw. Protoc.*, 2020, pp. 1–11.

[9] S. Tong, Z. Xu, and J. Wang, "CoLoRa: Enabling multi-packet reception in LoRa," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 2303–2311.

[10] S. Tong, J. Wang, and Y. Liu, "Combating packet collisions using non-stationary signal scaling in LPWANs," in *Proc. 18th Int. Conf. Mobile Syst. Appl. Serv.*, 2020, pp. 234–246.

[11] P. X. Zhenqiang Xu and J. Wang, "Pyramid: Real-time LoRa collision decoding with peak tracking," in *Proc. IEEE Conf. Comput. Commun.*, 2021, pp. 1–9.

[12] Q. Chen and J. Wang, "Aligntrack: Push the limit of LoRa collision decoding," in *Proc. IEEE 29th Int. Conf. Netw. Protoc.*, 2021, pp. 1–11.

[13] X. Xia, Y. Zheng, and T. Gu, "FTrack: Parallel decoding for LoRa transmissions," *IEEE/ACM Trans. Netw.*, vol. 28, no. 6, pp. 2573–2586, Dec. 2020.

[14] X. Wang, L. Kong, L. He, and G. Chen, "MLoRa: A multi-packet reception protocol in LoRa networks," in *Proc. IEEE 27th Int. Conf. Netw. Protoc.*, 2019, pp. 1–11.

[15] X. Xia, N. Hou, Y. Zheng, and T. Gu, "PCube: Scaling LoRa concurrent transmissions with reception diversities," in *Proc. 27th Annu. Int. Conf. Mobile Comput. Netw.*, 2021, pp. 670–683.

[16] R. Eletreby, D. Zhang, S. Kumar, and O. Yağan, "Empowering low-power wide area networks in urban settings," in *Proc. Conf. ACM Special Int. Group Data Commun.*, 2017, pp. 309–321.

[17] M. O. Shahid, M. Philipose, K. Chintalapudi, S. Banerjee, and B. Krishnaswamy, "Concurrent interference cancellation: Decoding multi-packet collisions in LoRa," in *Proc. ACM SIGCOMM Conf.*, 2021, pp. 503–515.

[18] SX1276, LoRa, "77/78/79 datasheet, rev. 4" Semtech, Mar., 2015.

[19] R. Ghanaatian, O. Afisiadis, M. Cotting, and A. Burg, "LoRa digital receiver analysis and implementation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, 2019, pp. 1498–1502.

[20] LoRa Alliance, "Lorawan specification v1.0.3," Accessed: Sep. 22, 2021. [Online]. Available: https://lora-alliance.org/resource-hub/lorawan-specification-v103

[21] W. Xiao, M. Kaneko, N. El Rachkidy, and A. Guitton, "Integrating LoRa collision decoding and MAC protocols for enabling IoT massive connectivity," *IEEE Internet Things Mag.*, vol. 5, no. 3, pp. 166–173, Sep. 2022.

[22] J. Haxhibeqiri, I. Moerman, and J. Hoebeke, "Low overhead scheduling of LoRa transmissions for improved scalability," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3097–3109, Apr. 2019.

[23] B. Reynders, Q. Wang, P. Tuset-Peiro, X. Vilajosana, and S. Pollin, "Improving reliability and scalability of lorawans through LoRaWANs scheduling," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1830–1842, Jun. 2018.

[24] F. Benkhelifa, Z. Qin, and J. A. McCann, "User fairness in energy harvesting-based LoRa networks with imperfect SF orthogonality," *IEEE Trans. Commun.*, vol. 69, no. 7, pp. 4319–4334, Jul. 2021.

[25] C. Pham and M. Ehsan, "Dense deployment of LoRa networks: Expectations and limits of channel activity detection and capture effect for radio channel access," *Sensors*, vol. 21, no. 3, 2021, Art. no. 825.

[26] Z. Wang, L. Kong, K. Xu, L. He, K. Wu, and G. Chen, "Online concurrent transmissions at LoRa gateway," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 2331–2340.

[27] C. Li, X. Guo, L. Shangguan, Z. Cao, and K. Jamieson, "Curvinglora to boost LoRa network throughput via concurrent transmission," in *Proc. 19th USENIX Symp. Netw. Syst. Des. Implementation*, 2022, pp. 879–895.

[28] M. Hessar, A. Najafi, and S. Gollakota, "Netscatter: Enabling large-scale backscatter networks," in *Proc. 16th USENIX Symp. Netw. Syst. Des. Implementation*, 2019, pp. 271–284.

[29] L. Lampe and K. Witrisal, "Challenges and recent advances in IR-UWB system design," in *Proc. Proc. IEEE Int. Symp. Circuits Syst.*, 2010, pp. 3288–3291.

[30] P. Zhao et al., "miD: Tracking and identifying people with millimeter wave radar," in *Proc. IEEE 15th Int. Conf. Distrib. Comput. Sensor Syst.*, 2019, pp. 33–40.

[31] Y. Ma, G. Zhou, and S. Wang, "WiFi sensing with channel state information: A survey," *ACM Comput. Surv.*, vol. 52, no. 3, pp. 1–36, 2019.

Weiwei Chen (Member, IEEE) received the PhD degree from the Hong Kong University of Science and Technology, Hong Kong, in 2013. She is now with the department of Computer Engineering and Science, Shanghai University. Prior to that, she was at the department of Computer Science and Engineering, Southeast University, China. Her research interests include Wireless Sensing and Localization, IoT networking, Cross Technology Communications (CTC), and mobile edge computing.

Shuai Wang (Senior Member, IEEE) received the BS and MS degrees from the Huazhong University of Science and Technology, China and the PhD degree from the Department of Computer Science and Engineering, University of Minnesota, in 2017. He is currently a professor and Department Head with the Department of Computer Engineering, Southeast University. His research interests include the Wireless Networks and Sensors, Cyber-physical Systems, and Data Science.

Tian He (Fellow, IEEE) is currently a professor with the Department of Computer Science and Engineering, University of Minnesota Twin Cities. He is the author or coauthor of more than 280 papers in premier network journals and conferences with more than 22 000 citations (H-Index 65). His research interests include wireless networks, networked sensing systems, Cyber–physical Systems, real-time embedded systems, and distributed systems. He was the recipient of the NSF CAREER Award, in 2009, the McKnight Land-Grant Chaired Professorship, in 2011, the China NSF Outstanding Overseas Young Researcher I and II, in 2012 and 2016, the George W. Taylor Distinguished Research Award, in 2015, and seven best paper awards in international conferences, including MobiCom, SenSys, and ICDCS. He has served a few general/program chair positions in international conferences and on many program committees and also has been an editorial board member for six international journals. He is an ACM Fellow.