CONST: Exploiting Spatial-Temporal Correlation for Multi-Gateway based Reliable LoRa Reception

Zeyu Zhang* Weiwei Chen* Junwen Wang Shuai Wang† Tian He

Southeast University
{zeyu_zhang, junwen_wang, shuaiwang, tianhe}@seu.edu.cn, chen.ava.0012@gmail.com

2022 IEEE 30th International Conference on Network Protocols (ICNP) | 978-1-6654-8234-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICNP55882.2022.9940424

***Abstract*—As a representative technology of low power wide area network, LoRa has been widely adopted to many applications. A fundamental question in LoRa is how to improve its reception quality in ultra-low SNR scenarios. Different from existing studies that exploit either spatial or temporal correlation for LoRa reception recovery, this paper jointly leverages the fine-grained spatial-temporal correlation among multiple gateways. We exploit the spatial and temporal correlation in LoRa packets to jointly process received signals so that the fine-grained offsets including Central Frequency Offset (CFO), Sampling Time Offset (STO) and Sampling Frequency Offset (SFO) are well compensated, and signals from multiple gateways are combined coherently. Moreover, a deep learning based soft decoding scheme is developed to integrate the energy distribution of each symbol into the decoder to further enhance the coding gain in a LoRa packet. We evaluate our work with commodity LoRa devices (i.e., Semtech SX1278) and gateways (i.e., USRP-B210) in both indoor and outdoor environments. Extensive experiment results show that our work achieves 4.6dB higher signal-to-noise ratio (SNR) and 1.5× lower bit error rate (BER) compared with existing approaches.**

***Index Terms*—LPWAN, LoRa, Reliable Reception, Coherent decoding**

I. Introduction

Recently, LoRa has become one of the representative technologies in Low Power Wide Area Network (LPWAN). According to Semtech's LoRa Technology Overview [1], by the end of 2021, 191 million LoRa nodes have been deployed, with more than 100 operators in over 100 countries. As a long-range solution, it is inevitable that the LoRa signal experiences severe channel fading, resulting in an extremely low signal-to-noise ratio (SNR) and high decoding errors at the gateway. How to enhance the reception capability for the low SNR packet is essential to save energy, prolong the life cycle of low-powered LoRa nodes, and extend the coverage of the network.

To enhance the reception capability for the low SNR packet, existing approaches either exploit the temporal or spatial correlations in the system. When temporal correlation is considered, NELoRa [2] and prior work [3] strive to enhance the SNR of a received packet at the signal level. Researchers leverage soft decoding to jointly optimize the modulation and decoding procedure [4]–[6].

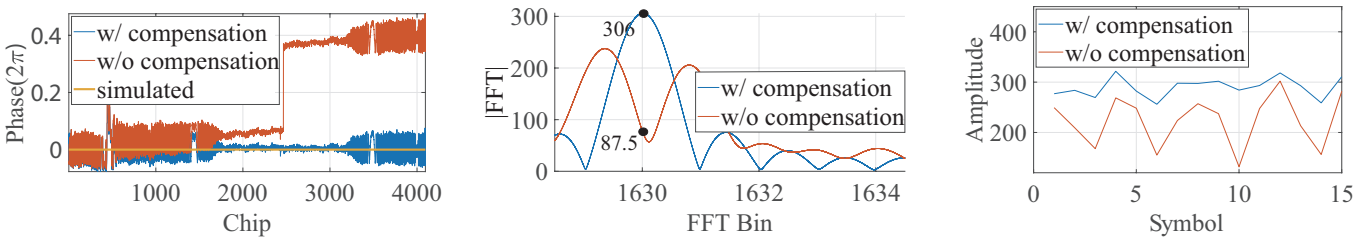
* Both authors contributed equally to this work. † Shuai Wang is the corresponding author.

Spatial Correlation \ Temporal Correlation	W/O	W/
W/O	[12]	[2]–[6]
W/	[7]–[10]	This Work

TABLE I
COMPARISON OF CONST WITH EXISTING SOLUTIONS

When spatial correlation is investigated, joint data processing from multiple gateways that focuses on recovering error bits has been discussed in OPR [7]. Though easy to be implemented, the lack of signal level information significantly constrains its error correction capability. A series of studies propose joint signal processing approaches to improve the decoding efficiency [8]–[11]. These approaches, however, are challenged by another critical issue. Signals received from multiple gateways suffer from different hardware offsets. Moreover, as a narrow-band technology, the duration of a LoRa packet is long (e.g., a LoRa packet lasts for tens to hundreds of milliseconds). With a long packet duration, hardware offsets cause significant signal variations. Such variations should be carefully compensated and signals from different gateways should be aligned perfectly before being combined. Existing studies fail to consider these issues. Therefore, the macro-diversity gain provided by multiple gateways is yet to be thoroughly leveraged for recovering ultra-low SNR packets. In addition, the aforementioned work does not explore the SNR improvement provided by LoRa PHY, such as the interleaving and FEC modules.

To address these issues, a fundamental question to be asked is how to jointly maximize the spatial and temporal correlations in both signal and coding layers from multiple gateways to improve the reception quality of ultra-low SNR LoRa packets. To answer this question, we propose CONST, a framework that enables reliable receptions for ultra-low SNR LoRa packets. Specifically, we observe that the hardware offsets from different packets exhibit stable temporal patterns. Such temporal patterns can be extracted reliably even for packets with extremely low SNR. This motivates us to leverage the temporal correlations in each packet for compensating the offsets, and thus facilitating coherent packets combining from multiple gateways. The major differences between CONST and existing studies are listed in Table I. Existing studies either consider temporal correlation or spatial correlation, while this



(a) The phases of different chips (b) The frequency distribution (c) The amplitude of peaks

Fig. 1. (a) The phases of different chips in a LoRa symbol with and without offsets compensation. (b) The frequency distribution of a LoRa symbol with and without offsets compensation. (c) The amplitude of the actual peaks carrying different symbols in a LoRa packet with and without offsets compensation.

work jointly exploits these two types of correlations.

Although the idea is straightforward, it is a nontrivial task in practice. Firstly, it is challenging to jointly estimate and compensate temporal patterns from multiple different gateways. Secondly, we need a wise design to leverage the temporal correlations for joint demodulation and decoding of a LoRa packet. CONST performs the following key steps to tackle the aforementioned problems. For each symbol to be decoded, we select a set of candidate frequency peaks and pass them to the cloud. The cloud jointly estimates and compensates the signals so that signals from different gateways can be combined coherently. It then adopts a Neural Network based Soft Decoder to recover the payload of a LoRa packet.

The main contributions are summarized as follows.

- To the best of our knowledge, we are the first to jointly investigate the spatial and temporal correlations of LoRa packets among multiple gateways. We propose a multi-gateway LoRa reception framework, called CONST, that exploits both spatial and temporal correlations in the received signal for ultra-low SNR LoRa reception.
- To exploit the spatial and temporal correlations, we jointly estimate and compensate the offsets among multiple gateways. The compensated signals from different gateways are then combined coherently. Coding level temporal correlations are also studied to further improve the packet reception rate. Neural network based approaches are proposed to extract such correlations.
- We conduct extensive experiments to evaluate the performance of CONST. In detail, commodity LoRa devices (Semtech SX1278), multiple USRP-B210 and a PC are utilized as LoRa senders, gateways and the cloud respectively. Experiment results show that our work achieves 4.7dB SNR improvement and a 150% lower bit error rate compared with existing studies.

II. Motivations

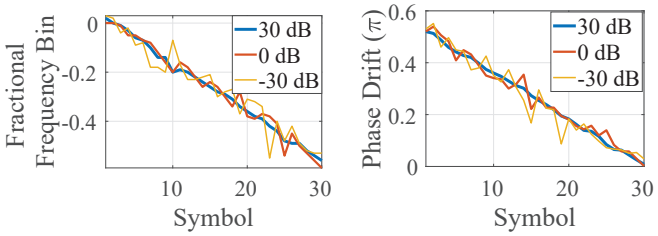
This section introduces spatial and temporal correlations in LoRa packets which have great potential to enhance the reception quality of ultra-low SNR LoRa links.

A. Offsets in LoRa systems

In general, three types of offsets, the Central Frequency Offset (CFO), the Sampling Time Offset (STO), and the Sampling Frequency Offset (SFO), exist in a LoRa system. CFO represents the center frequency offset between transceivers and is caused by an inherent mismatch between low-cost crystal oscillators and their nominal frequency values. STO is the time offset between the time that the first LoRa chip arrives at a receiver and the time that a receiver starts sampling it. SFO represents the difference in the sampling frequency of transceivers caused by the mismatch between the oscillators [3]. Previous work has shown that these offsets affect bit error rates. In this paper, we accurately model the effect of these offsets on received signals and their temporal patterns.

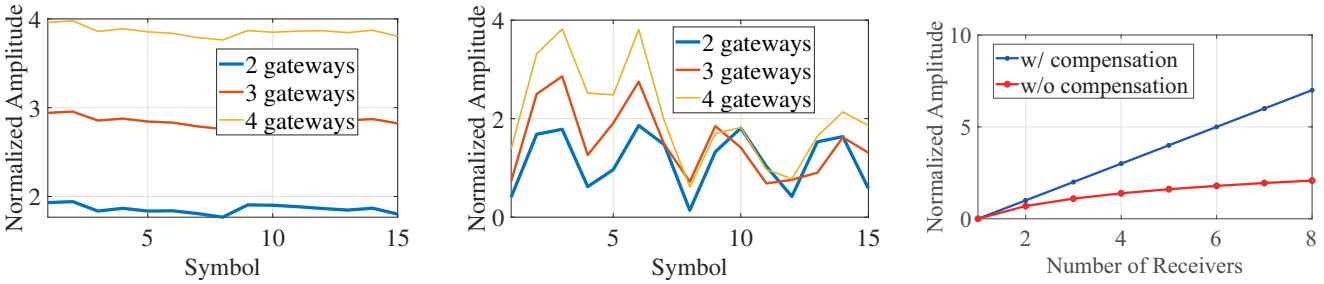
B. Temporal correlation of chips in a symbol due to offsets

The temporal correlation of chips in a symbol is related to the offsets in the system. Fig. 1(a) shows the phases of different chips in a symbol. In this experiment, the phase differences with offsets before and after fine-grained offsets compensation are collected from a commodity LoRa node (i.e., SX1278). The spreading factor is set to 12, the Bandwidth is 250kbps, and the transmitted symbol is 1630. The orange line refers to the case when no offsets present in the received signal after de-chirping, and it is generated by simulation. In Fig. 1(a), for all three lines, the integer part of the frequency component is removed. The phases with offsets increase with time because there is a fractional frequency drift in the received signal caused by offsets (chips are not aligned perfectly with the integer frequency bin). Moreover, we observe a sudden phase shift between two chips in a symbol. The phase shifts are the joint effects of both STO and CSS (Chirp Spread Spectrum). When no offsets exist in the system, the phases of chips without offsets are aligned perfectly, and hence stay close to 0 rad. After carefully compensating the offsets, the compensated



(a) Fractional frequency offsets (b) Phase Shift

Fig. 2. (a) The fractional frequency offsets of different symbols in a LoRa packet. (b) The sudden phase shift within different symbols in a LoRa packet.



(a) Effect of coherent combining (b) Effect of non-coherent combining (c) Macro-diversity gain of multiple receivers

Fig. 3. (a) The normalized amplitude of the actual frequency peak that carries the LoRa symbol with coherent combining; (b) The normalized amplitude of the actual frequency peak that carries the LoRa symbol with non-coherent combining; (c) The achievable macro-diversity gain for signals with or without coherent combining.

phase difference converges to the simulated one, leading to the highest amplitude of the received LoRa symbol.

Fig. 1(b) plots the frequency domain signal before and after offsets compensation. Referring to the red line, due to offsets, a fractional frequency drift is introduced, and the frequency peak deviates from the integer frequency bin. Moreover, the phase shifts caused by STO further reduce the amplitude of the strong frequency peak, and eventually split it into two weak peaks. According to the Nyquist sampling theorem, when the bandwidth of a signal is equal to the sampling rate, the sampling rate is less than the Nyquist frequency. Therefore, the resulting discrete time series is distorted. The additional peak introduces strong interference to the original peak. Fig. 1(c) shows the amplitude of the frequency peaks of a LoRa symbol. According to the figure, the received signal's SNR is improved significantly if the offset compensation is correctly performed.

C. Temporal correlation of symbols in a packet

Fig. 2 shows the fractional part of frequency peak, as well as the phase shift of different symbols in a packet. The blue, red, and orange line represents the STO and frequency drifts of the same packet with SNR=30dB, 0dB, and -30dB respectively. According to the figure, three lines stay close together under various SNRs, which implies that stable fine-grained information is available even if the SNR of a packet is extremely low. Because the offsets are caused by the transceiver pair, they are invariant to channel changes (such as changes in surrounding features or Lora node movements).

Moreover, we observe that the location of the fractional frequency peak and the phase shift in different symbols change with time. Simply using fixed offsets (e.g., the initial offsets) to process the whole packet will lead to inefficient offsets compensation and introduce unnecessary noise into the received signal. Capturing the temporal pattern of the received signal helps to trace and compensate the offsets.

In addition, LoRa PHY utilizes Hamming code as its Forward Error Correction (FEC) module. An interleaving module is also added after FEC to redistribute coded bits to different symbols. Interleaving and FEC together provide another layer of temporal correlation. Since soft decoding adopts the energy information of IQ signals for mapping a LoRa symbol to the zero-one-bit PHY payloads, it opens up the opportunity to fully explore both chip level and symbol level temporal correlation in a LoRa packet.

D. Spatial correlation from different gateways

Different copies of the same packet experience different offsets at different gateways. If macro-diversity is utilized effectively, symbols from different copies need to be combined coherently. For instance, the information of the actual frequency peak (the exact location, amplitude, and phase of the peak) observed from different gateways is very different. Combining signals from multiple gateways without compensating the aforementioned offsets introduces additional noise to the combined signal, leading to an inefficient signal combining.

In Fig. 3(a), after compensating the offsets, signals are aligned perfectly for demodulation. As shown in Fig. 3(b), however, since offsets change with time, if a fixed and not sufficiently accurate compensation is performed, the macro-diversity brought by multiple gateways degrades quickly as time evolves. Fig. 3(c) shows that the macro-diversity gain grows almost linearly with the number of gateways when signals from different gateways are combined coherently. When signals are not aligned perfectly, the diversity gain quickly vanishes with the increase of the gateway nodes.

III. System Overview and Design Challenges

A. System Overview

In this paper, CONST, a multi-gateway based reliable reception framework, is proposed for receiving LoRa signals with extremely low SNR. On the one hand, the bottleneck of LoRaWAN lies in the power constraint of remote LoRa nodes, and priority should be given to reducing power consumption and extending transmission range. On the other hand, both energy and computation resources at the gateway and the cloud are sufficient. Therefore, we intend to trade the computation power at the cloud for better performance of the remote LoRa nodes suffering from extremely high channel loss. The design overview is shown in Fig. 4. In detail, it contains two main parts: Signal Pre-Processing and Joint Decoding.

1) Signal Pre-Processing: This part extracts and compresses the critical information in each packet. The compressed information is passed to the cloud for further processing. It

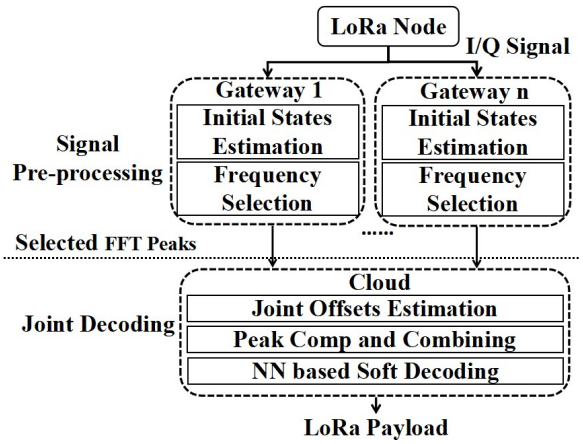


Fig. 4. Design Overview

is composed of two modules: Initial States Estimation and Frequency Selection.

To estimate offsets in preambles accurately, we concatenate preambles into a super symbol. The initial Central Frequency Offset (CFO) and Sampling Time Offset (STO) of a packet are extracted from the super symbol [13]. For example, given eight preambles, the super symbol is eight times of an ordinary symbol, leading to 9dB extra processing gain when used to examine the initial states. The integer part of the frequency offset is also retrieved from this module.

After estimating the temporal pattern of preambles, we introduce a frequency selection module to select a set of frequency peaks for each symbol of a packet. Frequency sets and the initial state are transferred to the cloud and will be further processed to determine the actual peaks that carry LoRa symbols. Note that instead of transmitting I/Q signal to the cloud, only a set of frequency peaks will be passed to the cloud, leading to significantly low bandwidth consumption.

2) Joint Decoding: This part explores the spatial and temporal correlation of different copies of the same packet from multiple gateways to jointly decode the whole packet. Each gateway synchronizes and receives the signals individually. A gateway can send several bits to the cloud to indicate the start of a new packet, and then send a set of frequency peak candidates for each symbol to the cloud. Beside this, no further approaches are required to synchronize the signals at the cloud. It is composed of three modules: the Joint Offset Estimation module, the Peak Compensation and Combining module, and the Neural Network based soft decoding module.

In the Joint Offset Estimation module, frequency peaks from multiple gateways are jointly examined so that one frequency peak will be selected for each symbol. The selected frequency peaks are utilized to estimate the offsets for each packet.

The Peak Compensation and Combining module utilizes estimated offsets to compensate for signals from different gateways, and then combines them coherently. The main difference between Charm [10] and our scheme is that signals cannot be aligned coherently if the offsets are not properly compensated (refer to Fig. 3(b)).

In the Neural Network based soft decoding module, a deep learning based approach is proposed to learn and deduce the payload of a LoRa packet. Notice that, here the payload refers to the 0-1 bits of the LoRa PHY layer after de-interleaving and FEC decoding, which will be passed to the upper layer.

B. Design Challenge

1) How to jointly maximize the estimation accuracy of the fine-grained offsets from different gateways: In an ultra-low SNR packet, some noisy peaks can be mistakenly treated as the actual frequency peaks carrying the LoRa symbols. Given multiple copies of the same packet, it is challenging to leverage macro-diversity so that actual peaks and the corresponding fine-grained offsets are correctly extracted from noise. This problem is addressed in Sec. V-B.

2) How to reduce the bandwidth consumption between gateways and the cloud: Transmitting raw IQ streams to the cloud consumes a large amount of bandwidth, resulting in network congestion and high costs. How to compress the IQ signals so as to save bandwidth is essential when implementing CONST. We will address this problem in Sec. V-B.

3) How to integrate continuous IQ signals into the discrete FEC decoding: LoRa PHY introduces a series of coding procedures to map payload bits into symbols transmitted to the air. We need to provide an efficient design to transform IQ signals into the probability distribution of every 0-1 bit and take the best advantage of the temporal correlation provided by coding layer. We will address this problem in Sec. V-C.

IV. Design of Signal Pre-processing

A. A Primer On LoRa Modulation/Demodulation

Before presenting the offset estimation module, we first introduce the basic modulation/demodulation mechanism of LoRa. In this paper, the sampling rate at the receiver is set to the transmission bandwidth of the transmitter, and the number of samples collected from one symbol at the receiver equals the number of chips in a symbol. Notations are listed in Table II.

Symbol	Notation
$x_s[n]$	The nth chip of the sth symbol in the time domain.
N	The total number of chips in a symbol.
h_s	The channel state of the sth symbol.
m_s	The sth symbol be to transmitted.
k	The ratio of the sampling frequency at the receiver to the transmitter.
δ_s	CFO between the transceiver pair at the sth symbol.
β_s	The fractional part of STO in the sth symbol.
$\Delta_{f,s}$	The central frequency drift of the sth symbol caused by CFO and STO
Φ_s	The starting phase of the first chip in the sth symbol.
$\Delta_{\phi,s}$	The phase shifts between chip $N-m_s-1$ and $N-m_s$.

TABLE II
Notations

LoRa PHY is a patented technology developed by Semtech. It is famous for its low power consumption and long transmission distance [14]. LoRa adopts Chirp Spread Spectrum (CSS) as its modulation scheme. It also introduces different Spreading Factor (SF) and Coding Rate (CR) which can be set dynamically to achieve different reliability and data rate trade-off. Specifically, the higher the SF, the larger the CR, the higher the transmission reliability and the lower the effective data rate [15]–[17].

Let N be the number of chips of a LoRa symbol. $w_s[n]$, the nth chip of the sth symbol (denoted as symbol m_s) at the transmitter, is expressed as

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

When the signal is demodulated, it will firstly be de-chirped. The receiver then performs an FFT on the de-chirped signal. The frequency bin with the highest amplitude is treated as the transmitted symbol [18].

B. Initial State Estimation

While CFO and STO affect the location of frequency peaks, SFO caused by hardware differences leads to small differences in sampling frequency between transceiver pairs. As shown in Fig. 2, STO and the central frequency drift are time-varying due to SFO. It is essential to estimate their initial states and temporal patterns.

1) Offsets calculation: We present the signal model with the consideration of various offsets. In the preamble part, symbols transmit 0; in the payload part, symbols transmit different value m_s. Here m_s can be any value within $[0, N)$. With the presence of offsets, the received LoRa signal, denoted as $x_s[n]$, is represented as

$$x_s[n] \approx \begin{cases} h_s e^{j2\pi\left(\frac{1}{2N}(kn+\beta_s)^2+\left(\frac{m_s}{N}-\frac{1}{2}\right)(kn+\beta_s)+\frac{k\delta_s n}{N}+\sum_{i=0}^{s-1}\delta_i\right)} & 0 \le n < N-m_s, \\ h_s e^{j2\pi\left(\frac{1}{2N}(kn+\beta_s)^2+\left(\frac{m_s}{N}-\frac{3}{2}\right)(kn+\beta_s)+\frac{k\delta_s n}{N}+\sum_{i=0}^{s-1}\delta_i\right)} & N-m_s \le n < N. \end{cases}$$

Here h_s, δ_s and β_s represent the channel state of the sth symbol, the CFO and the STO between the transceiver respectively. k is the ratio of the sampling frequency at the receiver to the transmitter. For example, when the transmitter transmits one chip (sample), the receiver will receive k samples. δ_s stays relatively constant for a whole packet. $(k-1)$ can be interpreted as the normalized SFO. The de-chirped signal, denoted as $d_s[n]$, is approximated as

$$d_s[n] \approx \begin{cases} h_s e^{j2\pi\left(\Phi_s+(km_s+\Delta_{f,s})\frac{n}{N}\right)} & 0 \le n < N-m_s \\ h_s e^{j2\pi\left(\Phi_s-\Delta_{\phi,s}+(km_s+\Delta_{f,s})\frac{n}{N}\right)} & N-m_s \le n < N, \end{cases} \quad (2)$$

Here,

$$\Delta_{f,s} = k\delta_s + k\beta_s + N(\frac{1}{2}-\frac{k}{2}) \approx \delta_s + \beta_s, \quad (3a)$$

$$\Phi_s = \frac{m_s\beta_s}{N} - \frac{\beta_s}{2} + \sum_{i=0}^{s-1}\delta_i, \quad (3b)$$

$$\Delta_{\phi,s} = \beta_s - \frac{\Delta_{f,s}}{N} \approx \beta_s, \quad (3c)$$

$$\beta_s \approx \beta_0 + (k-1)(s-1)N. \quad (3d)$$

β_0 is the initial STO, and k is close to 1. $\Delta_{f,s}$ is the central frequency drift of symbol m_s. Δ_{ϕ,m_s} is the sudden phase shifts between chip $N-m_s-1$ and $N-m_s$. Φ_s represents the starting phase of the first chip in the sth symbol. Moreover, Eq. (3d) assumes a fixed STO in the whole symbol as the change of STO in a symbol is marginal.

Remarks: From Eq. (3), we find that both cumulative frequency offsets and phase shifts within a symbol increase or decrease linearly with time.

2) Estimating the offsets of preambles: When the offsets in preambles are calculated, m_s is set to 0. Eq. (2) can be simplified to

$$d_s[n] = h_s e^{j2\pi\left(-\frac{\beta_s}{2}+\sum_{i=0}^{s-1}\delta_i+(\delta_s+\beta_s)\frac{n}{N}\right)}, \quad 0 \le n < N, \quad (4)$$

where,

$$\beta_s \approx \beta_0 + (k-1)(s-1)N \quad (5)$$

Define the signal after offsets compensation as $d'_s[n](\beta_s, \delta_s)$ (compensated with STO=β_s, and CFO=δ_s). When preambles are compensated, $d'_s[n](\beta_s, \delta_s)$ is expressed as

$$d'_s[n](\beta_s, \delta_s) = d_s[n]e^{-j2\pi\left(-\frac{\beta_s}{2}+s\delta_0+(\delta_0+\beta_s)\frac{n}{N}\right)}. \quad (6)$$

Here CFO is assumed to be stable for preambles, and we utilize δ_0 to denote it.

The offsets in preambles (the initial state of payload) are approximated by optimizing the following problem

$$a_p^* = \frac{1}{m_p}|\max_{\beta_s,\delta_0}\sum_{s=0}^{m_p}(\sum_{n=0}^{N-1} d'_s[n](\beta_s, \delta_s))| \quad (7a)$$

$$s.t. (2), (3d). \quad (7b)$$

Here the channel state is assumed to be stable for a small number of symbol duration, and m_p is the number of preambles of a packet. The objective function in Eq.(7) maximizes the amplitude of frequency peaks with offsets compensation, and a_p^* is the maximum amplitude of the initial preambles. The initial STO and CFO information can be resolved by searching the tuple $\{\beta_s, \delta_0\}$ that maximizes the amplitude of compensated preambles. The initial frequency offsets and the STO information will be transmitted to the cloud.

C. Frequency Selection

When SNR is high, m_s can be detected from the highest energy peak on the spectrum. However, with extremely low SNR, the frequency peak representing m_s may be overwhelmed by the environmental noise. We therefore introduced a neural network-based filter discussed in [2] to suppress the noise in the received signal. Even though, in an ultra-low SNR environment, after performing FFT on the pre-processed signal, there may exist a set of peaks with high energy, leading to decoding ambiguity. We therefore transmit a set of frequency bins to the cloud, which collects the information from multiple gateways to decode the symbol.

The set of frequency peaks transmitted to the cloud is determined as follows. Let m be the frequency bin to be transmitted in the frequency set, t_s and τ_f be its estimated STO and frequency drift respectively. We compute the amplitude of a frequency bin and split it into two parts (denoted as $fb_{m,1}$ and $fb_{m,2}$) with the following equation.

$$fb_{m,1} = \sum_{n\geq 0}^{n<N-m} d_s[n]e^{-j2\pi(\tau_f+m)\frac{n}{N}} \tag{8a}$$

$$fb_{m,2} = \sum_{n\geq N-m}^{n<N} d_s[n]e^{-j2\pi(\tau_f+m)\frac{n}{N}} \tag{8b}$$

$$a_m = |fb_{s,1}| + |fb_{s,2}| \tag{8c}$$

Here a_m is the maximum amplitude of the frequency bin after offset compensation. The set of frequency peaks whose a_m is higher than the pre-defined threshold is passed to the cloud for further processing. There exists a trade-off between decoding accuracy and the system bandwidth requirement. Lower threshold results in more frequency peaks being transferred to the cloud, which requires higher bandwidth but gives better decoding performance. We set the threshold to γa_p^*, where a_p^* is the maximum amplitude of preambles. We will study the impact of the different γ on the overall performance.

Once the frequency peak set is selected, for each frequency peak m in the set, $|fb_{m,1}|$, $|fb_{m,2}|$ and the exact location of the frequency peak $(\tau_f + m)$ will be transmitted to the cloud. Notice that, for each frequency peak to be transmitted, we transmit two real value (i.e., $|fb_{s,1}|$ and $|fb_{s,2}|$) and the phase difference between them (i.e., $\angle fb_{m,1} - \angle fb_{m,2}$) to the cloud where offsets estimation will be performed. Moreover, in our approach, the raw IQ signals are not transmitted to the cloud, which reduces bandwidth consumption significantly.

V. Design of Joint Decoding

A. Joint Offsets Estimation

With the ever-changing offsets, the temporal patterns of the actual frequency peaks also change with time. In this module, we estimate the temporal patterns of the frequency peaks to facilitate finely offsets compensation. For each symbol, the cloud receives multiple frequency sets from different gateways. Let $P_{g,s}$ be the frequency set of symbol s transmitted to the cloud from gateway g. For frequency peak $m \in P_{g,s}$, $a_{g,s,m}$ is the amplitude of frequency peak m (refer to Eq. (8)). If m does not belong to $P_{g,s}$, $a_{g,s,m} = 0$. We first generate a frequency sequence F with length l (l is the packet length), and $F = \{f_1, f_2, \cdots, f_l\}$. F will be used to estimate the offsets of the packets from different gateways, and for any $1 \leq s < l$, $f_s = \arg\max_m \sum_g a_{g,s,m}$. The intuition is that the noise will be suppressed by adding the amplitude of the same peak from different packets together. The probability that the true frequency sequence (a sequence of frequency peaks that carry LoRa symbols in a packet) is chosen to estimate the offsets can be improved. Fig. 5 further illustrates the process. It demonstrates how the spatial correlation is achieved.

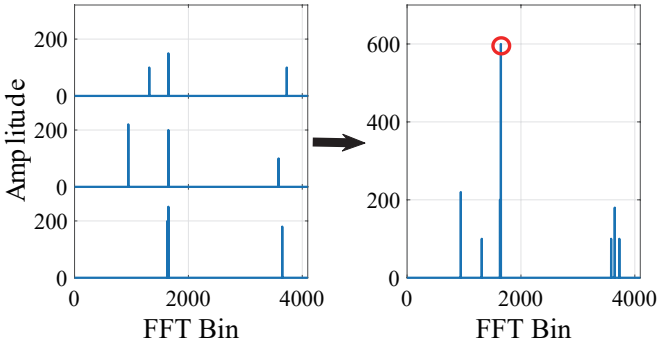


Fig. 5. The selection of frequency sequence. By combining signals from multiple gateways, the amplitude of the frequency peak that carries the correct LoRa symbol will be enhanced. However, the peaks caused by noise will be suppressed when signals from multiple gateways are jointly considered.

Though the joint frequency sequence selection improves the probability that correct symbols are selected by F, some frequency peaks in F may be incorrect. The incorrect frequency peaks in F are treated as outliers when estimating the temporal patterns of offsets. As long as there are enough correct frequency peaks, we have a high chance of making an accurate offset estimation. Moreover, if incorrect peaks are chosen in this step, they will be corrected by the following two modules.

When F is determined, we generate a frequency sequence for gateway g, denoted as F_g. A frequency offset sequence associated with F_g is denoted as DF_g. We first set $F_g = \emptyset$ and $DF_g = \emptyset$. For any symbol $1 \leq s < l$, if $a_{g,s,f_s} > 0$, we add f_s into F_g, and add the frequency offset of f_s from gateway g into DF_g. F_g and DF_g as well as initial preambles, are sent to the RANSAC filter [19] to estimate the frequency offset of the whole packet. We utilize a RANSAC filter to estimate the offsets due to its excellent performance in resisting the influence of outliers, which helps to obtain a stable estimation of frequency offsets. Moreover, according to Eq. (3a), given CFO estimated from the initial state and frequency offsets, the STO for each symbol is derived.

B. Peak Compensation and Combining

To fully exploit the spatial time correlation in a LoRa symbol, offsets should be carefully compensated. Since the phase shifts caused by STO further reduce the amplitude of

the correct frequency peak, we compensate such phase shift and obtain the compensated amplitude as follows.

$$dt_{g,s} = df_{g,s} - \delta_{g,0}, \tag{9a}$$

$$fb_{g,s,m} = |fb_{g,s,m,1} + e^{j2\pi dt_{g,s}} fb_{g,s,m,2}|, \tag{9b}$$

$$fb^*_{s,m} = \sum_g fb_{g,s,m} \tag{9c}$$

where $df_{g,s}$, $\delta_{g,0}$, $dt_{g,s}$ is the estimated frequency offset of the sth symbol from gateway g, the CFO of the pacekt from gateway g, and the estimated STO respectively. $fb_{g,s,m,1}$ and $fb_{g,s,m,2}$ are two frequency peaks transferred to the cloud from gateway g. $fb^*_{s,m}$ is the amplitude of the combined frequency peak after compensation. All of the frequency peaks and their compensated and combined amplitude ($fb^*_{s,m}$) are sent to the Neural Network based Soft Decoding module.

C. Neural Network based Soft Decoding

According to the demodulation process of LoRa, after dechirp and FFT, there is only one peak in the spectrum. The peak position represents the symbol transmitted by the packet. In the ultra-low SNR scenario, LoRa suffers from decoding errors after signal combining. We design a neural network based soft decoding module so that the SNR information of the frequency peaks and the coding gain (the temporal correlation) in the packet can be jointly exploited. As shown in Fig. 6, the decoding module contains four parts: Probability Evaluation module, Soft Gray Decoding module, Soft Deinterleaving module, and Soft Hamming Decoding module.

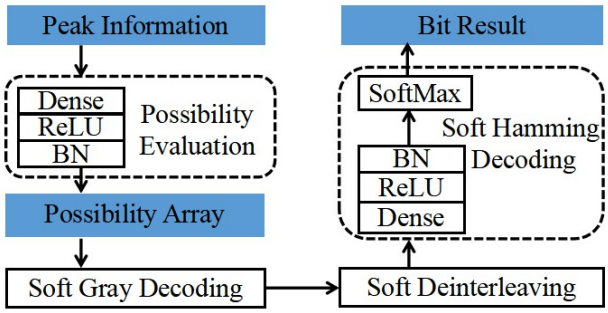


Fig. 6. Architecture of our Neural Network based Soft Decoding

a) Probability Evaluation: Intuitively, the higher the energy of a compensated frequency peak, the higher the chance that the frequency peak carries the transmitted symbol. Since multiple bits constitute a symbol, the probability distribution of symbols in a packet can be transformed into the probability distribution of each bit in a packet. The Probability Evaluation module converts I/Q signals to probabilities.

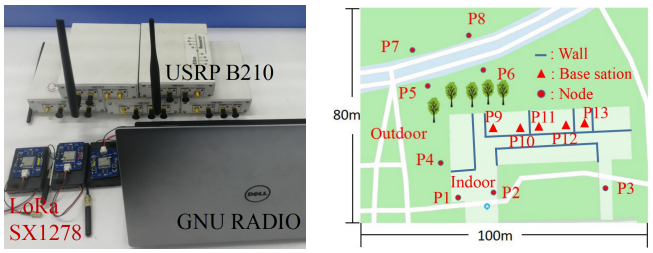
The probability of $bit = 1$ for each symbol is related to the spectral coefficient, the noise power, and the location of the bit. To generate the probability of the nth bit in symbol s, the combined peaks obtained from the previous module and the average noise power are input to this module.

To evaluate the probability of each bit, dense layer and ReLU are applied to fit complex nonlinear functions. Each dense layer is followed by a batch normalization layer that normalizes the mean and the variance of the layer input to preserve the physical meaning of the probability and speed up convergence. We use Bayesian optimization to tune the network hyper-parameters, which improves network performance.

b) Gray Decoding and Deinterleaving: After obtaining the probability distribution of each bit, LoRa PHY uses Gray code [20] and interleaving [21] [22] [23] to re-distribute the payload bits to enhance the reception reliability of a packet. We carefully retrieve linear mapping before and after gray code by reverse engineering. Similarly, deinterleaving can also be treated as a linear mapping. In detail, let M_G and M_I be the gray decoding and deinterleaving mapping. Let P be the probability set, for any bit i in a packet, p_i represents the probability that the ith bit equals 1. P_1 and P_2 denotes the probability set after gray decoding, and deinterleaving, respectively. The linear mapping can be represented as $P_1 = M_G P$, and $P_2 = M_I P_1$.

c) Soft Hamming decoding: Soft decoding [14] converts the traditional Hamming decoding into a maximum likelihood searching problem. In particular, it selects a set of bits sequence in a packet, which has the minimum Manhattan distance (the maximum likelihood) to the standard Hamming encoded sequence.

We use DENSE+ReLU layers to map the probability distribution of all the bits in the packet to high-dimensional feature space for clustering and classify them into 16 decoding results. So far, our neural network has completed the entire LoRa decoding process.



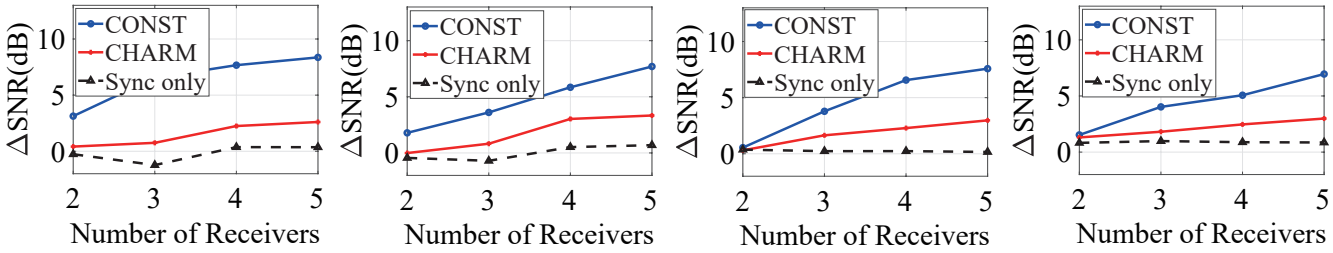
(a) USRP B210 and SX1278 nodes (b) Experiment scenarios

Fig. 7. Evaluation equipment and scenarios

VI. Evaluations

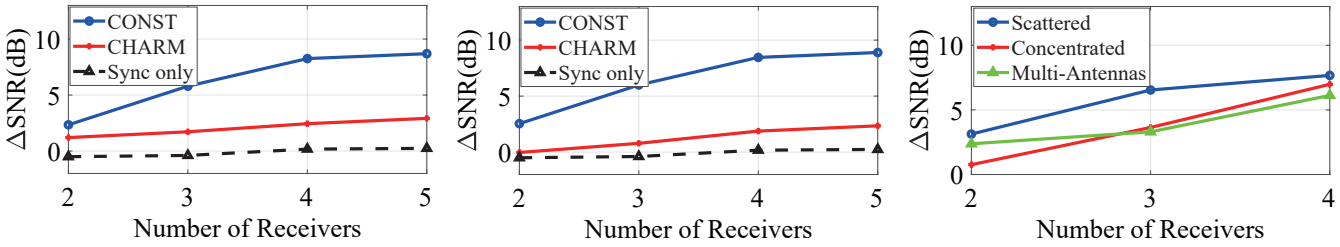
In this section, we set up a test platform to evaluate the performance of our design. We utilize five USRP B210 as five gateways to receive LoRa Packets. These USRP B210 are connected to computers configured with the hardware driver UHD of the USRP and GNU radio. A computer works as the cloud and then collects the data from five USRP. We deploy several commercial SX1278 LoRa nodes to transmit packets and set a long duty cycle for each LoRa node so that collision rarely happens. The equipment used for evaluations is shown in Fig. 7(a).

We deploy the gateways in different rooms and place SX1278 LoRa commercial chips at different locations on campus. The distribution of each node and the base station



(a) SNR improvement for Group2 nodes when SF=10 (b) SNR improvement for Group3 nodes when SF=10 (c) SNR improvement for Group2 nodes when SF=8 (d) SNR improvement for Group3 nodes when SF=8

Fig. 8. The performance of coherent combining versus different number of gateways



(a) SNR improvement when package length equals 20 symbols (b) SNR improvement when package length equals 80 symbols (c) SNR improvement when receivers' positions are different

Fig. 9. The performance of coherent combining with different packet lengths and receivers' locations

is shown in Fig. 7(b). The gateways are marked as triangles. Though the deployment area is small in the evaluations, since our gateways are placed in the room, the received signal is attenuated by multiple layers of walls and trees, and the received power at various gateways is low.

By adjusting the distance between LoRa nodes and the USRP, we control the SNR of the signal. We divide cases into three groups: (a) Group 1: LoRa commercial chips are deployed relatively close to the USRP equipment, such as $P3$ and $P1$ (with average SNR ranging from 6.5dB to 12dB); (b) Group 2: commercial chips are deployed far away from USRP devices such as $P5$ or $P6$ (with average SNR ranging from -18.5 to 21.5dB); and (c) Group 3: $P7$ and $P8$, from where the signals are much weaker and the SNR approaches -30dB. In each experiment, we transmit 100 LoRa packets and set $BW = 250kHz$.

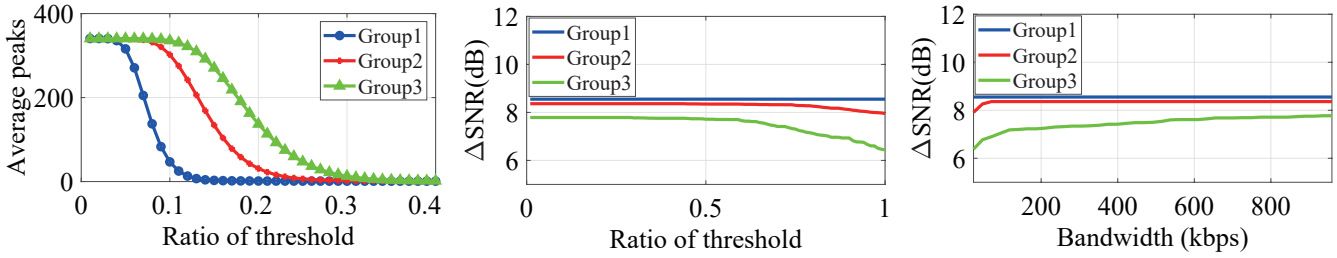
A. Performance improvement regarding to coherent combining

1) Comparing CONST's performance with existing work: In this subsection, we compare the results of CONST with Charm, and the baseline scheme in which signals are synchronized and combined directly. The baseline scheme is named "Sync only". Here we compare the performance of different schemes in a low SNR regime.

Fig. 8(a) and Fig. 8(b) show the SNR improvement of packets from Group 2 and Group 3 with $SF = 10$. Fig. 8(c) and Fig. 8(d) represents the performance of the two groups with $SF = 8$. From the figures, we observe that, compared with Charm, the SNR improvement of CONST grows faster with the number of gateways. For instance, with $SF = 10$, CONST provides 3.13dB to 8.36dB SNR improvement for various settings as the number of gateways increases from 2 to 5 for Group2 nodes. Charm, however, only achieves 0.61dB to 2.61dB improvement gain when the number of gateways grows from 2 to 5. Furthermore, with $SF = 10$, when the number of gateways is 5, the SNR gain of CONST for Group 2 and Group 3 nodes is 5.76dB and 3.36dB higher than that of Charm respectively. As to the case when $SF = 8$, according to Fig. 8(c) and 8(d), the SNR gain of CONST for Group 2 and Group 3 nodes is 4.6dB and 4dB higher than that of Charm given 5 gateways. The reason for this gap is that, as illustrated in Fig. 3, to combine the signals coherently, the offset must be finely compensated. However, Charm cannot effectively compensate for such offsets. Moreover, CONST extracts information from the frequency domain signal to take full advantage of the processing gain in CSS, whereas Charm obtains phase information from the time domain.

2) The impact of the packet length on the SNR improvement: In order to explore the effect of packet length on different coherent combining methods, we selected LoRa packets of different lengths received by Group2 nodes. The payload length is set to 20 Bytes and 80 Bytes, and the SF is 10.

As is shown in Fig. 9(a) and Fig. 9(b), the SNR improvement of Charm degrades as the packet lengthens, and the SNR gain of 5 gateways drops from 2.92 dB to 2.34 dB while the SNR gain of CONST remains around 8.75dB. The reason is that Charm ignores SFO and assumes a stable STO and CFO. However, the phase and frequency that the symbol needs to compensate for change with time, and gradually deviate from the estimated value as the LoRa package becomes longer, leading to inaccurate compensation.



(a) The average number of peaks that each symbol needs to transmit with different thresholds

(b) SNR improvement with different thresholds

(c) SNR improvement with different bandwidths

Fig. 10. The impact of the threshold on bandwidth and coherent combining

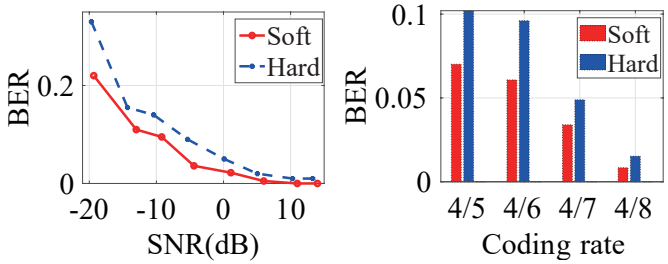
3) The impact of the location distribution of the gateways on the SNR improvement: To explore the effect of receivers' location on our coherent combining method, we investigate the SNR gain for Group2 nodes in the following three cases: (a) disperse four gateways at different places, i.e., $P9$, $P10$, $P11$, and $P12$ in Fig. 7(b); (b) concentrating the gateways at the same place $P13$ and $P10$; (c) using USRP N310 with multi-antenna synchronization at $P13$ and $P10$. The payload length is set to 100 Bytes, and the SF is 10.

The results shown in Fig. 9(c) illustrate that the SNR improvement of CONST degrades when the receiver's locations are concentrated. This is because the signals received by the scattered antennas are more independent than those received by the clustered antennas and multi-antenna systems, and the signals received by scattered antennas tend to obtain higher diversity gain.

4) The impact of the bandwidth consumption on the SNR improvement: In order to explore the effect of peak selection on our coherent combining methods, we evaluate the average SNR gain of LoRa packets with $SF = 10$, $BW = 250kHz$, and with different SNRs. When 96 bits are used to quantize the information of a peak, transmitting the information of a peak of each symbol to the cloud requires 23.4 kbps.

The relationship between the threshold and the number of peaks to be transmitted is shown in Fig. 10(a). With the same threshold, more peaks will be transmitted to the cloud as SNR decreases. For example, when the threshold equals $0.2a_p^*$, an average of 1.3 peaks need to be transmitted per symbol for packets from Group1 nodes, and an average of 8.09 peaks need to be transmitted per symbol for packets from Group2 nodes.

The results shown in Fig. 10(b) and Fig. 10(c) illustrate that the SNR improvement of CONST increases as the amount of peak information increases, and the lower the SNR, the greater the impact of the amount of peak information transmitted. Initially, the improvement goes quickly. However, as the number of peaks increases to five, the SNR improvement for different SNRs converges. It implies that when bandwidth is 120 kbps, CONST achieves a good trade-off between SNR and bandwidth overhead.



(a) The BER of different decoding schemes with different SNRs

(b) The BER of different decoding schemes at different coding rates

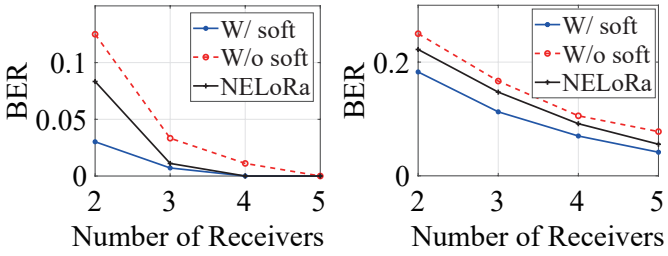
Fig. 11. The performance of soft decoding

B. The performance improvement brought by soft decoding

1) The impact of different SNRs on the performance of Neural Network based Soft Decoder: We first study the performance of soft decoding under different SNRs. In this experiment, we set $SF = 8$, $CR = 4/5$, and the offsets are not compensated. As demonstrated in Fig. 11(a), the performance of soft decoding is better than traditional decoding (hard decoding), and the Bit Error Rate (BER) is only 63% of the traditional hard decoding when $SNR = -20$dB. When BER is 0.22, the SNR of soft decoding is about -20dB, while the SNR of traditional decode is about -18dB.

2) The impact of coding rate on the performance of Neural Network based Soft Decoder: In order to test the performance of soft decoding with different coding rates (CR), we test the BER of soft decoding and the traditional decoding schemes for packets from Group 2 nodes. The payload length is set to 100 Bytes, and the SF is 10.

As shown in Fig. 11(b), the BER of soft decoding when $CR = 4/6$ and $CR = 4/8$ are 63.2% and 54.9% of the traditional hard decoding. When $CR = 4/7$, the BER of soft decoding compared to hard decoding is 69.5% and not further improved compared with $CR = 4/6$. The reason is that hard decoding has no error correction capability when $CR = 4/5$ and $CR = 4/6$, while hard decoding has the possibility to correct erroneous bits when $CR = 4/7$ and $CR = 4/8$ [20]. Therefore, the performance of hard decoding is significantly improved when $CR = 4/7$.



(a) BER of CONST and NELoRa when $SF = 10$, $SNR = -30$ dB (b) BER of CONST and NELoRa when $SF = 8$, $SNR = -30$ dB

Fig. 12. The performance of soft decoding versus the number of gateways

C. Overall performance

To demonstrate the combined performance of signal combining and soft decoding, we set SX1278 LoRa chips in Group3 to send packets with a payload length of 100 Bytes with $SF = 10$ and $CR = 4/5$.

As shown in Fig. 12, with low SNR, LoRa still has decoding errors after signal combining. With our design, the BER decreases as the number of receivers increases. Each additional receiver introduces an observable drop in BER for Group 3 nodes.

When $SF = 10$ and there are two receivers, the soft decoding (w/ soft) BER is 33% of the traditional decoding (w/o soft) and about 42% of the BER of NELoRa. Errors are almost eliminated when there are four receivers. When $SF = 8$, the processing gain brought by CSS modulation is much lower than that of SF10. Therefore, when $SNR = -30dB$, the BER is much higher than that of $SF = 10$. In this case, the errors that can be corrected by soft decoding decrease due to the increase in error density. When there are two receivers, the BER of soft decoding is still 73% of traditional decoding and about 82% of NELoRa.

Note that both our work and NELoRa use DNNs, but they are used for different purposes. While CONST uses it to estimate the probability of bit and classify probability matrix into standard Hamming encoded sequences, NELoRa utilizes it to classify the chirp symbols.

There are three major differences in detail. Firstly, NELoRa does not compensate for STO that increases linearly with time, and this drift will cause a drop in peak energy during demodulation. Secondly, NELoRa does not consider that the position of the frequency peak varies linearly with time, which makes the signal features inconspicuous and reduces the classification performance. Thirdly, CONST contains soft decoding and can further exploit the temporal correlation of the coding layer.

VII. Related Work

Prior work enhances the reliable reception capability of LoRa from two perspectives. One is utilizing multiple gateways to recover low-SNR or heavily interfered packets collaboratively, and the other is to improve the encoding and decoding system.

A. Improving the receiving capability in PHY layer

In work related to improving the reliable receiving capability of the physical layer, some work has used coherent combining to enhance the reliable receiving capability of LoRa under the condition of low SNR, such as Charm [10]. This work aligns LoRa data packets in time and uses LoRa data packets as the granularity to compensate the CFO. There are some work that reduces the bandwidth usage of this system [9] and improved synchronization method [8], but these work do not take into account the intra-chip level information, i.e., SFO between each transceiver. Although this is done to improve the SNR through diversity gain, the signals are not compensated accurately, thereby introducing additional noise to the combined signal. Some work [2], [24] deal with the received signal symbol by symbol but do not consider the time correlation introduced by LoRa PHY, e.g., the interleaving and FEC module.

B. Channel coding in MAC layer

In MAC layer, a lot of studies realize signal error correction by leveraging an additional level of coding scheme. They are set on top of existing LoRa PHY protocols to improve the reception rate of a low SNR packet. One representative work is DaRe [12], which is an application layer coding that uses redundant data to retrieve lost data, which uses convolutional codes, and fountain codes. These tasks reduce the coding efficiency, increase the communication overhead, and do not make full use of the physical layer I/Q signal to improve the processing gain.

There are also some work [4]–[6] that utilize the decoding gain of I/Q signals, but the physical layer encoding settings of LoRa commercial nodes, such as gray code, are not fully considered, so they cannot be applied to commercial chips.

VIII. Conclusions

In this paper, we propose a novel multi-gateway based collaborative recovery method based on the coherent combining of LoRa signals called CONST. CONST can be deployed on programmable gateways, without changing the hardware or firmware of commodity LoRa end devices. By adopting the spatial and temporal correlations to jointly process the received signals, CONST further improves the effect of coherent combining and decoding. The experiment results show that our work demonstrates 2 times higher SNR, and a 1.5 times lower bit error rate compared with existing studies.

IX. Acknowledgement

This work was supported in part by the Fundamental Research Funds for the Central Universities, China National Key R&D Program 2018YFB2100302, National Natural Science Foundation of China under Grant No. 61902066, National Nature Science Fundation of China under Grant No 62172092 and Natural Science Foundation of Jiangsu Province under Grant No. BK20190336.

REFERENCES

[1] O. Pospisil, R. Fujdiak, K. Mikhaylov, H. Ruotsalainen, and J. Misurec, "Testbed for lorawan security: Design and validation through man-in-the-middle attacks study," *Applied Sciences*, vol. 11, no. 16, p. 7642, 2021.

[2] C. Li, H. Guo, S. Tong, X. Zeng, Z. Cao, M. Zhang, Q. Yan, L. Xiao, J. Wang, and Y. Liu, "Nelora: Towards ultra-low snr lora communication with neural-enhanced demodulation," in *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, 2021, pp. 56–68.

[3] C. Bernier, F. Dehmas, and N. Deparis, "Low complexity lora frame synchronization for ultra-low power software-defined radios," *IEEE Transactions on Communications*, vol. 68, no. 5, pp. 3140–3152, 2020.

[4] G. Baruffa, L. Rugini, V. Mecarelli, L. Germani, and F. Frescura, "Coded lora performance in wireless channels," in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2019, pp. 1–6.

[5] T. Elshabrawy and J. Robert, "Enhancing lora capacity using non-binary single parity check codes," in *2018 14th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2018, pp. 1–7.

[6] G. Baruffa, L. Rugini, L. Germani, and F. Frescura, "Error probability performance of chirp modulation in uncoded and coded lora systems," *Digital Signal Processing*, vol. 106, p. 102828, 2020.

[7] A. Balanuta, N. Pereira, S. Kumar, and A. Rowe, "A cloud-optimized link layer for low-power wide-area networks," in *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, 2020, pp. 247–259.

[8] X. Xia and Y. Zheng, "Connecting lorawans deep inside a building," in *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, 2020, pp. 312–313.

[9] J. Liu, W. Xu, and W. Hu, "Energy efficient lpwan decoding via joint sparse approximation," in *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, 2018, pp. 325–326.

[10] A. Dongare, R. Narayanan, A. Gadre, A. Luong, A. Balanuta, S. Kumar, B. Iannucci, and A. Rowe, "Charm: exploiting geographical diversity through coherent combining in low-power wide-area networks," in *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2018, pp. 60–71.

[11] C. S. You, J. S. Yeom, and B. C. Jung, "Performance analysis of cooperative low-power wide-area network for energy-efficient b5g systems," *Electronics*, vol. 9, no. 4, p. 680, 2020.

[12] P. Marcelis, N. Kouvelas, V. S. Rao, and V. Prasad, "Dare: Data recovery through application layer coding for lorawan," *IEEE Transactions on Mobile Computing*, 2020.

[13] S. Wang, S. M. Kim, and T. He, "Symbol-level cross-technology communication via payload encoding," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2018, pp. 500–510.

[14] A. Marquet, N. Montavont, and G. Z. Papadopoulos, "Towards an sdr implementation of lora: Reverse-engineering, demodulation strategies and assessment over rayleigh channel," *Computer Communications*, vol. 153, pp. 595–605, 2020.

[15] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," *ieee communications surveys & tutorials*, vol. 19, no. 2, pp. 855–873, 2017.

[16] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of lora: Long range & low power networks for the internet of things," *Sensors*, vol. 16, no. 9, p. 1466, 2016.

[17] R. S. Sinha, Y. Wei, and S.-H. Hwang, "A survey on lpwa technology: Lora and nb-iot," *Ict Express*, vol. 3, no. 1, pp. 14–21, 2017.

[18] B. Hu, Z. Yin, S. Wang, Z. Xu, and T. He, "Sclora: Leveraging multi-dimensionality in decoding collided lora transmissions," in *2020 IEEE 28th International Conference on Network Protocols (ICNP)*. IEEE, 2020, pp. 1–11.

[19] K. G. Derpanis, "Overview of the ransac algorithm," *Image Rochester NY*, vol. 4, no. 1, pp. 2–3, 2010.

[20] J. Tapparel, "Complete reverse engineering of lora phy," 2019.

[21] P. Robyns, P. Quax, W. Lamotte, and W. Thenaers, "A multi-channel software decoder for the lora modulation scheme." in *IoTBDS*, 2018, pp. 41–51.

[22] M. Knight and B. Seeber, "Decoding lora: Realizing a modern lpwan with sdr," in *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016.

[23] S. Wang, W. Jeong, J. Jung, and S. M. Kim, "X-mimo: Cross-technology multi-user mimo," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 218–231.

[24] K. Sun, Z. Yin, W. Chen, S. Wang, Z. Zhang, and T. He, "Partial symbol recovery for interference resilience in low-power wide area networks," in *2021 IEEE 29th International Conference on Network Protocols (ICNP)*. IEEE, 2021, pp. 1–11.