PCube: Scaling LoRa Concurrent Transmissions with Reception Diversities

Xianjin Xia, Ningning Hou, Yuanqing Zheng
The Hong Kong Polytechnic University
Hong Kong, China
{csxxia,csyqzheng,csnhou}@comp.polyu.edu.hk

Tao Gu
Macquarie University
Sydney, Australia
tao.gu@mq.edu.au

ABSTRACT

This paper presents the design and implementation of PCube, a phase-based parallel packet decoder for concurrent transmissions of LoRa nodes. The key enabling technology behind PCube is a novel air-channel phase measurement technique which is able to extract phase differences of air-channels between LoRa nodes and multiple antennas of a gateway. PCube leverages the reception diversities of multiple receiving antennas of a gateway and scales the concurrent transmissions of a large number of LoRa nodes, even exceeding the number of receiving antennas at a gateway. As a phase-based parallel decoder, PCube provides a new dimension to resolve collisions and supports more concurrent transmissions by complementing time and frequency based parallel decoders. PCube is implemented and evaluated with synchronized software defined radios and off-the-shelf LoRa nodes in both indoors and outdoors. Results demonstrate that PCube can substantially outperform state-of-the-art works in terms of aggregated throughput by 4.9× and the number of concurrent nodes by up to 5×. More importantly, PCube scales well with the number of receiving antennas of a gateway, which is promising to break the barrier of concurrent transmissions.

CCS CONCEPTS

• **Networks** → **Network Protocol Design**; • **Computer Systems Organization** → *Embedded Systems*.

KEYWORDS

Low-Power Wide-Area Networks, LoRa, Concurrent Transmission

1 INTRODUCTION

Low Power Wide Area Networks (LPWANs) [30, 45, 56] are emerging as a compelling paradigm for connecting Internet-of-Things (IoT). In LPWANs, a gateway can cover tens of km^2 and collect data
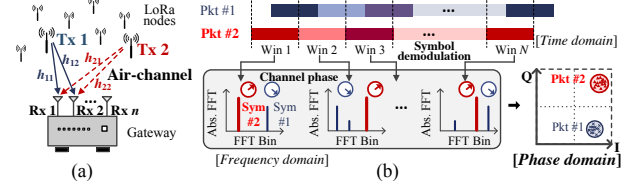
Figure 1: Illustration of parallel packet decoding in phase domain. (a) Multiple Rx antennas provide reception diversities, (b) Symbols of concurrent packets are clustered in I-Q plane.

from low-power sensors at low data rates (few kbps) for long-term operations (5∼10 years). With the rapid development of LoRa technology and IoT applications, large numbers of LoRa-enabled IoT devices will be densely deployed in large-scale IoT applications (*e.g.*, environment monitoring, smart metering, *etc.*). As the communication range of LoRa is long, the number of LoRa nodes covered by a gateway can be pretty large (*e.g.*, thousands of nodes per gateway). As all LoRa nodes operate in the same ISM band, the spectrum are likely to get crowded with an ever-increasing number of LoRa deployments. As a result, coexisting LoRaWANs would suffer serious intra- and inter-network interference and collisions, resulting in degraded network performance [3, 14]. To address this problem, we aim to support more concurrent LoRa transmissions in the shared spectrum.

MIMO (Multiple Input Multiple Output) technologies have been widely used in wireless systems (*e.g.*, WiFi, LTE) to enable concurrent transmissions and increase communication capacity [19, 20, 29, 36, 47, 54, 59, 60]. A recent work (Iris [12]) introduced Multi-User MIMO (MU-MIMO) to support concurrent transmissions of sensor nodes in LPWANs. However, the maximum concurrency supported by MIMO LPWANs is limited by the number of receiving (Rx) antennas of a gateway. It is hard to meet the capacity requirement of LoRaWANs, where the number of IoT nodes can be orders of magnitude larger than the number of Rx antennas. Ideally, we aim to support more concurrent transmissions than the number of Rx antennas of a gateway.

Latest advances of LoRaWAN (*e.g.*, Choir [8], FTrack [51], NScale [43]) explore parallel decoding of concurrent transmissions. Different from the MIMO-based method (*e.g.*, Iris), those parallel decoding methods leverage unique features of LoRa to separate concurrent packets in frequency and time domains. For example, Choir [8] leverages hardware imperfections and frequency diversities of LoRa nodes to classify colliding symbols into different packets according

to unique features in frequency domain. FTrack [51] exploits timing periodicity and edge misalignment of colliding symbols to separate collisions in time domain. These works demonstrate that it is possible to resolve collisions and support concurrent transmissions of multiple LoRa nodes with only one Rx antenna of a gateway.

Intuitively, is it possible to support more concurrent transmissions with multiple antennas at a gateway? This question motivates us to explore the possibility of combining the benefit of multi-antenna and LoRa-specific parallel decoding techniques to improve concurrent transmission performance beyond the limits of conventional MIMO and LoRa parallel decoders.

However, a simple combination of multiple antennas and existing parallel decoders (*e.g.*, Choir [8], FTrack [51]) does not lead to better performance in concurrent transmissions, because different antennas receive the same packets with almost the same time and frequency features. For example, if two colliding packets arrive at an antenna at the same time and cannot be separated in time domain, other collocated antennas cannot separate them in the time domain either. Ideally, we expect multiple antennas can provide reception diversities and add a new dimension so that we can resolve collisions and support more concurrent transmissions.

To this end, we leverage the fact that packets of different transmitters go through different wireless channels which imprint distinct channel features (*e.g.*, phase rotation) on their corresponding symbols, while the symbols from the same transmitter pass through the same wireless channel over the air (*i.e.*, *air-channel*). As illustrated in Figure 1(a), packets of two LoRa nodes pass through different air-channels, which lead to different extracted channel phases. We can extract channels from concurrent packets and separate colliding symbols based on the channel information. Figure 2 shows extracted channel phases of three colliding packets. The symbols are clustered into different groups in phase domain, which correspond to their air-channels. As such, the channel phases of received symbols essentially provide a new dimension to resolve collisions, which complements the existing time-domain and frequency-domain parallel decoders. More importantly, one additional Rx antenna creates N new air-channels to all N concurrent transmitters. In case that the air-channels of two packets produce similar phases at an Rx antenna, we can create reception diversities using multiple Rx antennas as shown in Figure 1(a), and separate collided packets at a new Rx antenna as illustrated in Figure 2(b,c). We expect to enable more concurrent transmissions with more Rx antennas of a gateway. As long as two transmitters are separated in physical space, we should be able to separate their concurrent transmissions in channel space by adding more reception diversities.

However, the design and implementation of a phase-based parallel decoder entails tremendous technical challenges in practice. First, it is challenging to measure channel phases from symbols of each packet, as signals of concurrent LoRa packets interleave together in demodulation windows. To address this problem, we leverage the fact that symbols of concurrent packets generally carry different payload data, which are encoded with different initial frequencies of LoRa chirps. As such, we can dechirp the signals in a demodulation window and separate concurrent symbols into distinctive FFT bins as illustrated in Figure 1(b). We then extract the phase of each frequency component from the FFT results to measure channel phases of concurrent symbols.

(a) Three concurrent packets with closely-aligned symbol edges

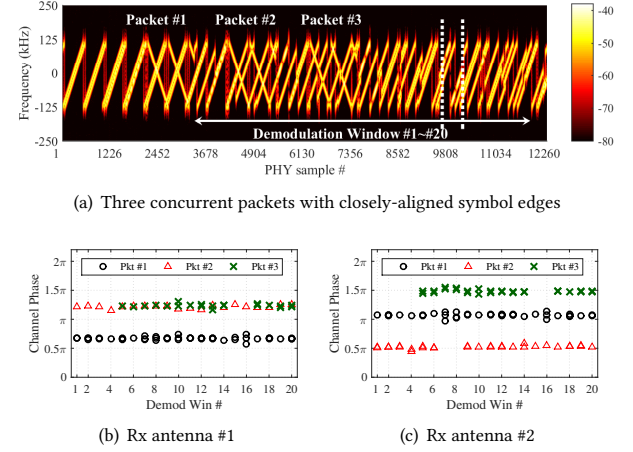(b) Rx antenna #1

(c) Rx antenna #2

Figure 2: Reception diversities help decode more concurrent packets: air-channels of Packet #2 and Packet #3 have the same phase at antenna #1, all three packets have distinctive channel phases at antenna #2.

Second, it is non-trivial to correctly extract air-channel information from raw phase measurements of concurrent symbols. The phases of symbols are affected not only by air-channels but also by various frequency and phase uncertainties of radio hardware, such as offset of carrier frequencies between transmitter and receiver (*i.e.*, CFO) and random clock offset of signal sampling (*i.e.*, STO) [2]. Besides, the hardware of LoRa modem also add random phase shifts to transmitted symbols, leading to unpredictable inter-symbol phase variance. To address these practical issues, we exploit the specific frame structure of LoRa preamble and SFD (*i.e.*, Start Frame Delimiter) to estimate and calibrate for frequency offsets. We further mitigate the impact of hardware-induced phase variance by examining the corresponding phase measurements of Rx-pairs. Finally, we obtain consistent phase measurements of air-channels as shown in Figure 2(b,c).

We design and implement PCube, a Phase-based Parallel Packet decoder for concurrent transmissions of LoRa nodes. PCube uses multiple Rx antennas at a gateway to resolve collisions and enable concurrent LoRa transmissions. PCube calibrates the frequency offsets of received signals to extract correct frame timing of each packet. PCube then measures phases of all concurrent symbols in a demodulation window. PCube mitigates inter-symbol phase variance and extracts air-channel phase of each symbol with Rx-pairs. Finally, PCube groups symbols to their corresponding packets according to distinct air-channel phases. PCube iterates to recover more packets from collisions.

We build a prototype system with 40 commodity off-the-shelf LoRa nodes and 8 synchronized USRPs as a gateway. We evaluate PCube via testbed experiments both indoors and outdoors. Our evaluations demonstrate that PCube can support up to 5× more concurrent transmissions than MIMO. The aggregated network throughput of PCube is 4.9× higher than the best throughput of existing LoRa parallel decoders and MIMO. PCube scales well with network size and the number of Rx antennas of a gateway.

The contributions of our paper are summarized as follows.

- We propose to separate LoRa packet collisions in the phase domain, which complements existing time-domain and frequency-domain approaches for LoRa collision recovery.
- We design techniques to estimate and calibrate for frequency and phase uncertainties of radio hardware, and present methods to reliably measure channel phases of concurrent packets in the presence of collisions.
- We design and implement a phase-based parallel packet decoder (PCube) for LoRa, which scales concurrent transmissions beyond the number of reception antennas of a gateway.

2 A PRIMER ON LORA

Chirp Spread Spectrum (CSS). LoRa adopts a Chirp Spread Spectrum (CSS) modulation technique, which modulates symbols with chirp signals in a pre-configured bandwidth (BW). The frequency of a chirp increases (*i.e.*, *up-chirp*) or decreases (*i.e.*, *down-chirp*) linearly over time at a rate $k = \frac{BW^2}{2^{SF}}$, where SF corresponds to the spreading factor of CSS. A *base chirp* sweeps from $-\frac{BW}{2}$ to $\frac{BW}{2}$ and can be represented as $C(t) = e^{j2\pi(\frac{k}{2}t - \frac{BW}{2})t}$. CSS changes initial frequencies of base chirps to modulate different symbols. The signal of a symbol can be represented as follows.

$$S(t, f_{sym}) = C(t) \cdot e^{j(2\pi f_{sym} t + \varphi_{sym})}, \tag{1}$$

where f_{sym} and φ_{sym} represent the initial frequency and initial phase of the chirp signal, respectively.

Demodulation. LoRa demodulates a symbol by measuring the initial frequency of chirp signal. The signal of a received symbol is represented as below.

$$y(t) = h \cdot S(t, f_{sym}) + n(t), \tag{2}$$

where h denotes the communication channel between transmitter and receiver and $n(t)$ denotes noises. To demodulate f_{sym} from $y(t)$, LoRa first de-chirps $y(t)$ by multiplying the conjugate of a base chirp ($C^{-1}(t)$) then performs Fast Fourier Transform (FFT), which produces $Z(f) = FFT(y(t) \cdot C^{-1}(t))$. The frequency response of FFT peak in $Z(f)$ corresponds to the demodulated frequency of a symbol as follows.

$$\tilde{f}_{sym} = \arg\max_{f_i} \|Z(f_i)\|, \tag{3}$$

where $f_i = i \cdot \frac{BW}{2^{SF}}$ denotes the frequency response of the ith FFT bin ($i = 0, 1, \cdots, 2^{SF} - 1$).

LoRa packet reception. A receiver continuously samples a channel to detect incoming packets. To facilitate packet detection, LoRa prepends a packet with a preamble which consists of a number of base chirps followed by a Start Frame Delimiter (SFD) composed of 2.25 down-chirps before the start of packet payload. A receiver extracts frame timing from a preamble, then locates and demodulates symbols in payload.

Concurrent transmission. When two LoRa nodes transmit simultaneously with the same SF and BW parameters, their signals collide at a gateway as follows.

$$y(t) = h_1 \cdot S_1(t, f_{sym1}) + h_2 \cdot S_2(t + \Delta\tau, f_{sym2}) + n(t), \tag{4}$$

where $S_1(t, f_{sym1})$ and $S_2(t, f_{sym2})$ correspond to the symbols of two nodes, $\Delta\tau$ denotes the time offset between two concurrent symbols, and h_i represents the communication channel from node i to receiver.

Air-channel. The communication channel from a transmitter to a receiver involves not only wireless channel over the air (*i.e.*, *air-channel*) but also RF chains of Tx and Rx radios. Due to hardware imperfection of radios, the end-to-end communication channel may vary over time. In contrast, air-channel remains invariant during a short time (*i.e.*, coherent time of wireless channel).

3 PCUBE IN A NUTSHELL

PCube develops a new paradigm of air-channel based concurrent transmissions. It is built on an observation that air-channel basically remains coherent and imprints consistent phase features on received symbols of a packet. The phase features can be regarded as hidden information encoded by the air-channel from a transmitter to a receiver. PCube recovers the hidden information and uses them to help decode concurrent transmissions.

Different from MIMO which estimates channels between Tx-Rx pairs by sending and receiving probe signals [19, 20, 38], PCube requires to extract air-channel from each symbol of a received data packet. PCube exploits the distinct air-channels traversed by concurrent packets to group symbols into their corresponding packets for parallel packet decoding. As air-channels change across Rx antennas, a gateway can create more spatial diversities by adding more Rx antennas to enhance its capability of parallel decoding. It turns out the spatial diversity can scale up combinatorially beyond the number of Rx antennas, which is promising to break the barrier of MIMO-based concurrent transmissions.

The air-channel based concurrent transmissions mainly target at communications of stationary IoT nodes or nodes with low-mobility. PCube can support a large number of LoRa nodes to transmit concurrently with non-orthogonal parameters (*e.g.*, the same carrier frequency, spreading factor and bandwidth). It is complementary to existing concurrent transmissions supported by LoRaWANs which use orthogonal parameters or multi-channels. PCube runs on a gateway with multiple Rx antennas. It aggregates the signals of all Rx antennas to calibrate frequency and phase for air-channel extraction. All computations involved in parallel packet decoding are put in the gateway side, and PCube requires no modification to end nodes.

4 AIR-CHANNEL BASED CONCURRENT TRANSMISSION

4.1 Challenges of Air-channel Extraction

PCube relies on the phases of air-channels to group concurrent symbols into their corresponding packets. It extracts air-channels from symbols of concurrent packets by overcoming a number of practical challenges.

(1) *Inter-packet interference.* Though the symbols of concurrent packets interleave together, they can be disentangled in frequency domain due to timing misalignment and frequency difference of chirps. For instance, we can use a standard demodulation method to demodulate the received signals of two concurrent symbols. It will produce two frequencies corresponding to the two symbols. As

long as the demodulated frequencies are separated in FFT Bins, we can measure the phase of each symbol from distinctive FFT Bins. However, the raw phase measurements correspond to the phases of end-to-end communication channels, which differ from the phases of air-channels.

(2) *Distortions of radio hardware.* The end-to-end communication channel h is composed of RF chains of transmitter and receiver radios (denoted by h^{rf}) and air-channel (denoted by h^{air}). The raw channel measurements (*i.e.*, $h = h^{air} \cdot h^{rf}$) from received symbols can be affected by various RF components of Tx and Rx radios. We summarize the primary sources of frequency and phase distortions introduced by radio hardware as below.

Central Frequency Offset (CFO). Due to hardware imperfection, the oscillator frequency of a LoRa node may be different from a gateway, resulting in central frequency offset. CFO can lead to frequency deviation of received symbols, as well as phase rotations across symbols of a packet.

Sampling Timing Offset (STO). Due to narrow bandwidth and low sampling rates of LoRa radio, the time offset between packet arrival and time of being sampled by a radio can be relatively long [2]. It can cause non-negligible distortions to the frequency and phase of received symbols.

Radio frequency leakage. LoRa radio is subject to frequency leakage when the frequency of transmitted signals changes from one chirp to another (*e.g.*, at the boundary of two symbols) [52]. It adds unpredictable phase shifts to transmitted symbols, leading to inter-symbol phase variance.

The preamble of a LoRa packet is conventionally designed for frequency and frame synchronization. Our work uses preambles for channel phase calibration. However, preambles may suffer from inter-packet interference. PCube develops a novel method to separate preambles of concurrent packets, and uses separated preambles (*i.e.*, collision-free) for frequency and phase calibration. In the following, we will investigate how various factors affect phase measurements and propose calibration methods for air-channel extraction.

4.2 Separating Concurrent Preambles

We detect preambles of concurrent LoRa packets by correlating received signals with a standard base chirp. Even when two preambles collide, their chirp frequencies are still separable in FFT due to misalignment of frame timing among concurrent transmissions [51]. As shown in Figure 3(b), the preamble chirp corresponds to $f_{sym} = 0$ (*i.e.*, Bin #1) after demodulation, while frequencies in other FFT bins correspond to interference. Ideally, we can clear the FFT bins of interference and use Bin #1 to restore a preamble chirp. However, it suffers information loss since CFO and STO can result in fractional frequencies represented by sidelobes.

Notice that an integer frequency (*i.e.*, when $\Delta f = 0$) can be precisely represented by a single FFT bin with the highest magnitude as shown in Figure 4(a,d). If we remove fractional part Δf from the raw frequency of preamble chirp, the resulting signal will be precisely represented by a single FFT bin, *i.e.*, Bin #1 in Figure 3(b). Then, we can safely clear interference frequencies in other FFT bins without affecting the preamble chirp. After that, we can add Δf
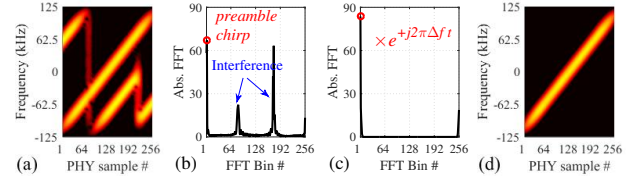

Figure 3: Separating preamble chirp from collisions: (a) collided chirps, (b) interference removal in FFT, (c) restore Δf, (d) separated preamble chirp.
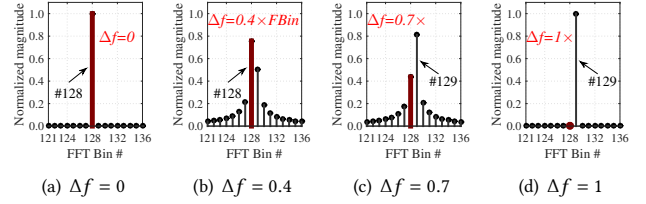

Figure 4: FFT representation of fractional frequencies (Δf). *FBin* denotes the frequency resolution of FFT.

back to Bin #1 to restore the original preamble chirp without loss of CFO/STO information, as illustrated in Figure 3(c,d).

In practice, we estimate the fractional frequency of a preamble chirp by searching Δf that can maximize the magnitude of FFT Bin #1 after compensating Δf to the received raw signal. As Δf is estimated based on the FFT magnitude of preamble chirp (*i.e.*, Bin #1), the method is resistant to noise and interference because the power of noise and interference do not accumulate in Bin #1. To accelerate the searching process, we first use grid search to find a coarse Δf within ±1 FFT bin and next use binary search to find $\Delta \tilde{f}$ in a confined range.

4.3 Compensating for CFO and STO

The residual frequency offsets of received signals (*i.e.*, CFO and STO) impact PCube in two aspects: (1) CFO and STO can lead to phase rotations across symbols of the same packet. The phase rotations, if not calibrated, would result in incorrect symbol grouping and decoding results. (2) CFO and STO may distort frame timing detection of a packet. Incorrect frame timing would lead to packet decoding errors. It may also introduce asynchronous frame timing issues among packet receptions of different antennas, impairing PCube's performance. We estimate CFO and STO of concurrent packets from their separated preambles.

Central Frequency Offset (CFO). The mismatched oscillator frequencies between transmitter and receiver radios result in CFO. We represent a received symbol with CFO as below.

$$y(t) = h^{air} \cdot e^{-j(2\pi\Delta f_{cfo}t + \varphi_{osc})} \cdot S(t, f_{sym}) + n(t), \quad (5)$$

where Δf_{cfo} and φ_{osc} are the oscillator frequency offset and phase offset between transmitter and receiver radios.

Intuitively, we can compare received preamble chirps with standard base-chirp and use Maximum Likelihood Estimation (MLE) to find Δf_{cfo}. This approach, however, does not work for LoRa
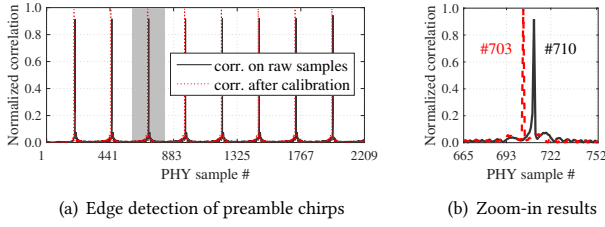
(a) Edge detection of preamble chirps

(b) Zoom-in results

Figure 5: Residual frequency offsets (*i.e.*, CFO and STO) lead to incorrect detection of frame timing.

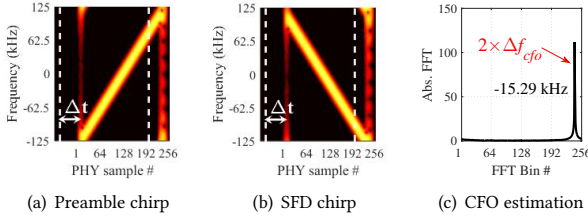(a) Preamble chirp

(b) SFD chirp

(c) CFO estimation

Figure 6: Estimating CFO with preamble up-chirp and SFD down-chirp. The white dashed lines indicate the real chirp edges.

because preamble chirps are not correctly located in the presence of CFO and STO. For instance, Figure 5 shows the edges of preamble chirps detected from received raw signals in comparison with the correct chirp edges. The preamble chirps are incorrectly located with several samples deviating from the real edges. We need a method that can reliably estimate CFO even if preamble chirps are located incorrectly.

Let $y_{pre}(t)$ denote the received signal of a preamble chirp. $y_{pre}(t+ \Delta t)$ denotes the detected preamble chirp, which deviates from real chirp edges with a time offset Δt due to impacts of CFO and STO. The goal is to estimate CFO with $y_{pre}(t + \Delta t)$.

We demonstrate that CFO can be reliably estimated with a preamble up-chirp and SFD down-chirp. The chirps in preamble and SFD share the same CFO and STO. We extract a preamble chirp and an SFD chirp based on the frame timing detected from received raw signals. Both chirps deviate Δt from their real edges and thus correspond to $y_{pre}(t+\Delta t)$ and $y_{sfd}(t+\Delta t)$, respectively, as illustrated in Figure 6(a,b). Since the same time offset (*i.e.*, Δt) transforms into opposite frequency offsets for preamble up-chirp and SFD down-chirp [43], we can remove the effect of timing offset Δt by multiplying $y_{pre}(t+\Delta t)$ with $y_{sfd}(t+\Delta t)$, which produces the following (noise $n(t)$ is omitted for clarity).

$$y_{pre}(t + \Delta t) \cdot y_{sfd}(t + \Delta t) = (h^{air})^2 \cdot e^{-j2\pi(2\Delta f_{cfo})t} \quad (6)$$

We perform FFT on the resulting signal of Eq.(6), as shown in Figure 6(c). The FFT peak indicates the integer frequency of $2\Delta f_{cfo}$.

Sampling Timing Offset (STO). Incoming signal $y(t)$ will be sampled by an Analog-to-Digital Converter (ADC) into discrete samples $y[n]$. The time offset between signal arrival and sampling

time of a receiver (*i.e.*, STO) introduces frequency and phase distortion to received chirps. As illustrated in Figure 7(a), ΔT_s denotes time offset of STO. The time offset would transform into a frequency offset Δf_{sto} and a phase offset φ_{sto} for a chirp signal. The received samples are essentially the signals as below.

$$y[n] = e^{j(2\pi\Delta f_{sto}t+\varphi_{sto})} \cdot y(t), \quad t = \frac{n}{F_s} \quad (7)$$

where $\Delta f_{sto} = \frac{BW^2}{2^{SF}}\Delta T_s$ and $\varphi_{sto} = 2\pi f_{sym}\Delta T_s$.

As STO is determined by both the arrival time of packet and sampling timing of receiver, STO changes across packets. It means that we cannot estimate STO in prior and calibrate for all packets. Instead, we should estimate and calibrate STO on a per-packet basis.

We estimate STO from the separated preambles of concurrent packets after CFO compensation. Since ΔT_s is basically less than a sample, Δf_{sto} influences the fractional part of FFT bins and can be determined and compensated with the searching algorithm as in Section 4.2. Figure 7(b) displays the CDF of Δf_{sto} measured from 500 LoRa packets. As expected, Δf_{sto} generally follows a uniform distribution in $[0, 1) \times \frac{BW}{2^{SF}}$.

4.4 Extracting Air-channel Phase

After compensating received signals for CFO and STO, we expect to obtain consistent phase measurements from symbols of the same packet. Figures 9(a) and (b) compare phase measurements from symbols of a packet before and after CFO and STO compensation. We see from the figure that phase measurements in preamble become invariant. However, there are inter-symbol phase variations in payload.

We find that the signals transmitted by commodity LoRa radio suffer from frequency leakages, which can lead to inter-symbol variation of phase measurements. Figure 8(a) presents a chirp signal transmitted by Semtech SX1276 radio. We can observe weak power leaking from main frequencies (*i.e.*, *frequency leakage*) when the chirp signal transits from the maximum frequency to the minimum. Specifically, we compare the samples transmitted by SX1276 against an ideal chirp signal of the same symbol. A phase shift is observed around positions of frequency leakage as shown in Figure 8(b). Figures 8(c,d) further compare phase measurements of chirp signals from windows A and C (*i.e.*, before and after the phase shift). The phase measurements differ by $165°$ because of the phase shift of transmitted samples. As frequency leakages also appear at the edges of payload symbols, it would add phase shifts to adjacent symbols, leading to inter-symbol variation of phase measurements as shown in the payload part of Figure 9(b).

We take CFO, STO and inter-symbol phase variation into account and characterize the received signal of a LoRa symbol as below.

$$y(t) = h^{air} \cdot \underbrace{e^{-j(2\pi\Delta f_{cfo}t+\varphi_{osc})}e^{j(2\pi\Delta f_{sto}t+\varphi_{sto})}e^{j\varphi_{var}}}_{h^{rf}} \cdot S(t, f_{sym}),$$

where φ_{var} represents the phase variance introduced by frequency leakages of LoRa radio.

Note that the initial phase of symbol $S(t, f_{sym})$ is φ_{sym}, and the phase of air-channel h^{air} is denoted by φ^{air}. After we remove CFO and STO, the phase measurement from $y(t)$ becomes $\phi = \varphi^{air} - \varphi_{osc} + \varphi_{sto} + \varphi_{var} + \varphi_{sym}$. As the goal is to extract phase of
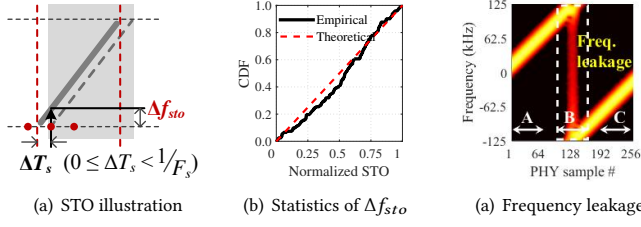
(a) STO illustration

(b) Statistics of Δf_{sto}

(a) Frequency leakage

(b) Phase shift

(c) FFT of Win A

(d) FFT of Win C

Figure 7: Illustration of STO and empirical measurement results.
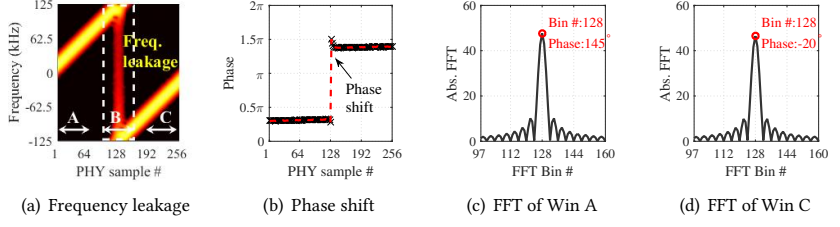
Figure 8: Illustration of frequency leakage and impacts on phase measurements.
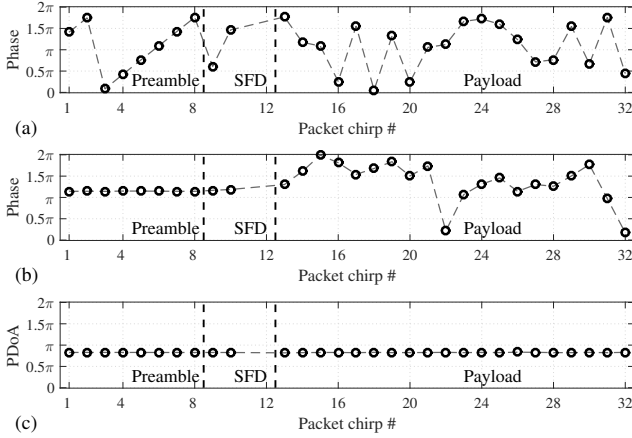

(a)

(b)

(c)

Figure 9: Phase measurements: (a) from received raw signals; (b) after compensating for CFO and STO; (c) after calibrating for both frequency and phase.

air-channel (*i.e.*, φ^{air}), we need to remove phase uncertainties of radio hardware (*i.e.*, $\varphi_{osc}, \varphi_{sto}, \varphi_{var}, \varphi_{sym}$) from ϕ to derive φ^{air}.

We use two synchronized Rx antennas (named an *Rx-pair*) to calibrate hardware phase uncertainties. For clarity, we denote the two antennas of an Rx-pair by Rx1 and Rx2 respectively. As the signals received by Rx1 and Rx2 correspond to the same packet, they share the same inter-symbol phase variation (φ_{var}) and symbol initial phase (φ_{sym}), because φ_{var} and φ_{sym} are determined by Tx radio and thus are invariant at Rx1 and Rx2. Besides, as Rx1 and Rx2 share the same clock source, the phase of oscillator frequency and STO remain the same. We can remove phase uncertainties of both Tx and Rx radios by subtracting the phase measurements of Rx1 and Rx2, which gives $\phi_1 - \phi_2 = \varphi_1^{air} - \varphi_2^{air}$.

Phase Difference of Air-channels (PDoA). $\varphi_1^{air} - \varphi_2^{air}$ represents the phase difference between air-channels from transmitter to Rx1 and Rx2, termed *Phase Difference of Air-channels (PDoA)*. In practice, we extract PDoA with a pair of Rx antennas. We separately process received signals of each antenna (*e.g.*, CFO and STO compensation) and measure phases of demodulated symbol (*i.e.*, ϕ_1 and ϕ_2). We extract PDoA of a symbol by subtracting the raw phase measurements of two antennas of the Rx-pair (*i.e.*, $\phi_1 - \phi_2$). Figure 9(c) presents the PDoA measurements from symbols of the packet

in Figure 9(a,b). We can observe that the PDoA stays constant across symbols in both preamble and payload of a packet.

4.5 PDoA based Parallel Decoding

PCube relies on an Rx-pair to receive and decode concurrent packets. Upon detecting concurrent LoRa packets, PCube separates the preamble of each packet from collisions, then uses separated preambles to calibrate for CFO and STO and extract symbol edges. After that, PCube can correctly locate the payload symbols of different packets and demodulate symbols for each packet.

Although we can separate preamble of a packet from collisions, the same method cannot be used to extract payload symbols of the packet without the prior knowledge on the initial frequency of payload chirps. In the presence of concurrent transmissions, more than one symbols will be detected in a demodulation window as shown in Figure 10(a,b).

Intuitively, as symbols of concurrent packets are received by the same Rx-pair with different air-channels, the multiple symbols detected in a demodulation window are expected to have different PDoAs. Figure 10(c) presents the PDoA measurements from symbols of the two packets shown in Figure 10(a). We see that the PDoAs of symbols form two horizontal lines corresponding to air-channels of the two packets. It motivates us to distinguish symbols of a target packet from interference by leveraging PDoA measurements.

However, as different packets usually have different CFOs and STOs, it is non-trivial to extract PDoAs for concurrent symbols that coexist in the same window. Fortunately, the CFOs and STOs of concurrent packets can be mitigated with a pair of synchronized Rx antennas. To be specific, PCube first dechirps raw signals from the same demodulation window of each Rx antenna and performs FFTs. Let $Z_1(f)$ and $Z_2(f)$ denote the FFT of dechirped signals of Rx1 and Rx2, respectively. Note that demodulated symbols appear at the same FFT bin in $Z_1(f)$ and $Z_2(f)$. Although symbols of different packets may have different CFOs and STOs, the corresponding symbols of the same packet which appear in the same FFT bins in $Z_1(f)$ and $Z_2(f)$ would share the same CFO and STO. As such, we can multiply $Z_1(f)$ with the conjugate of $Z_2(f)$ (denoted as $Z_2^*(f)$) for each FFT bin to remove the CFOs and STOs for all concurrent symbols in a demodulation window, which is represented as below.

$$\mathbb{Z}(f) = Z_1(f) \cdot Z_2^*(f). \tag{8}$$

Figure 10(d) plots the results of $\mathbb{Z}(f)$, where each frequency peak corresponds to a symbol and the phase readings of the peaks are PDoAs of the corresponding symbols.
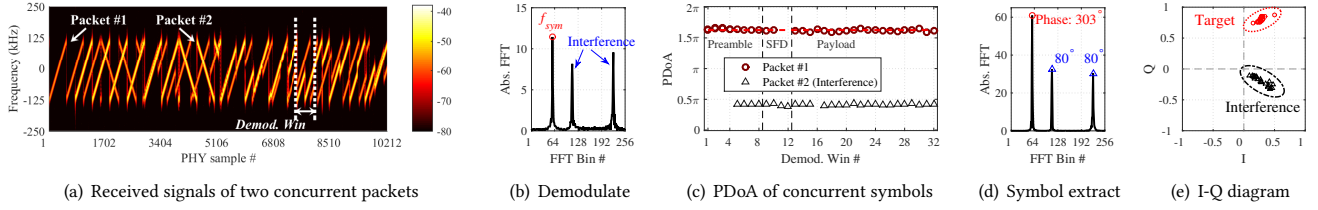
Figure 10: Illustration of extracting PDoA for concurrent symbols and PDoA based packet decoding.

To recover symbols of a target packet from collisions, PCube first demodulates the raw chirp signals in each demodulation window and removes CFOs/STOs for concurrent symbols according to Eq.(8). It extracts all demodulated symbols (target and interference) from each window and plots all detected symbols in an I-Q plane as shown in Figure 10(e) where each point represents a symbol. We see that demodulated symbols form two clusters in the I-Q plane, which correspond to different PDoAs of the two packets. PCube extracts the PDoA of a target packet from the separated preamble and selects the cluster of symbols that can best match the PDoA. Symbols in the selected cluster can then be decoded with a standard LoRa decoder, which finishes the extraction of a target packet from concurrent transmissions. After that, PCube aligns demodulation window with another packet and iterates the above operations to decode more concurrent packets.

4.6 Scaling with Rx Diversities

In case that two concurrent packets have similar air-channel phases, it would cause ambiguities to PDoA-based symbol classification and result in decoding errors. As the number of concurrent transmissions increases, we can anticipate serious *PDoA ambiguities* for packet decoding. PCube solves this problem by leveraging channel diversities of multiple Rx antennas of a gateway.

We leverage the fact that the same packet can be received by different antennas with different air-channels. If two packets suffer PDoA ambiguities at one Rx-pair because of the same angle-of-arrival (AoA) or similar channel conditions, they can still be separated at another Rx-pair. As such, an intuitive method is to iterate through all Rx-pairs to combat PDoA ambiguity for packet decoding. However, the method does not scale well because it becomes more likely to suffer from PDoA ambiguities at all Rx-pairs as the number of concurrent packets increases. As illustrated in Figure 11, a target packet (*e.g.*, Pkt #1) cannot be separated from interference at anyone of the Rx-pairs.

We observe that due to air-channel diversities among concurrent packets, PDoA measurements of different packets change differently across different Rx-pairs. We exploit the observation to combine PDoA measurements of multiple Rx-pairs to selectively strengthen a target packet. As illustrated in Figure 11 (Step 2), we firstly eliminate PDoA difference of a target packet across Rx-pairs by rotating the raw PDoA measurements of the target packet to phase 0 for each Rx-pair, and then combine all demodulated symbols of M Rx-pairs in the I-Q plane. As the rotated symbols of the target packet are aligned in phase, they will be coherently combined and strengthened. Meanwhile, the symbols of interference packets are
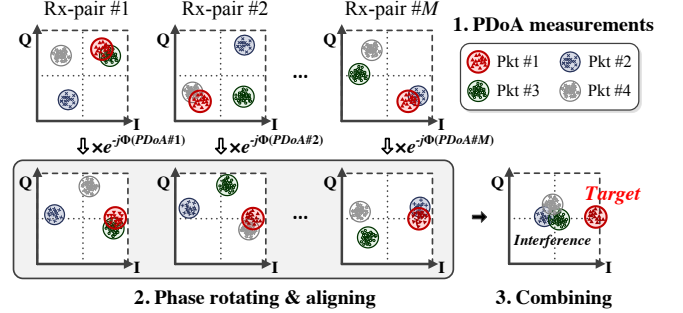

Figure 11: Combining PDoA measurements of multiple Rx-pairs.

incoherently combined and weakened due to phase heterogeneity from the target packet. As a result, symbols of the target packet and interference symbols are separated apart in the I-Q diagram as illustrated in Figure 11 (Step 3). Next, we can apply the PDoA based parallel decoding method (Section 4.5) to extract symbols of the target packet from collisions. The key step of Rx-pair rotation and combining is as follows.

$$\mathbb{Z}_{combine}(f) = \sum_{i=1}^{M} w_i \cdot \mathbb{Z}^{(i)}(f) \cdot e^{-jPDoA^{(i)}(f_{target})}. \quad (9)$$

By multiplying $\mathbb{Z}^{(i)}(f)$ with $e^{-jPDoA^{(i)}(f_{target})}$, we rotate the PDoA of target packet to phase 0, where $\mathbb{Z}^{(i)}(f)$ denotes the demodulated frequency results (in FFT) of the ith Rx-pair, and $PDoA^{(i)}(f_{target})$ represents the PDoA of a target packet at the ith Rx-pair measured from the separated preamble of the target packet. To mitigate the effect of heterogeneous SNRs among different Rx-pairs, we assign $w_i = \|\mathbb{Z}^{(i)}(f)\|/\sum_{k=1}^{M} \|\mathbb{Z}^{(i)}(f)\|$ based on the signal strength of each Rx-pair.

5 EVALUATION

5.1 Methodology

Multi-antenna gateway. We build a LoRa gateway using USRP SDRs (N210 with WBX daughterboards) based on the `gr-lora` open-source project [16]. We connect SDRs with an external clock source (CDA-2990) to form a synchronized multi-antenna system, as shown in Figure 12 (Left). The raw PHY samples are sent to a laptop through a 100 Gigabit Ethernet Switch (HUAWEI CloudEngine S1730S-L24T-A). The laptop runs PCube decoder implemented in MATLAB to process the raw PHY samples. A multi-antenna LoRa
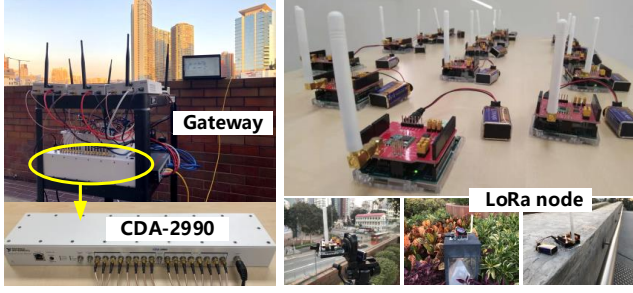
Figure 12: PCube implementation.

(a) Indoor layout   (b) Outdoor layout

Figure 13: Testbed settings of PCube.

gateway can be developed at low-cost similar to a widely available multi-antenna WiFi access point.

LoRa nodes. We use commodity LoRa nodes as transmitters. The LoRa nodes are composed of Dragino LoRa shields [7] embedded with Semtech SX1276 radios. We connect LoRa shields to Arduino Uno boards to configure the SX1276 chips. We set the default central frequency, bandwidth (BW), spreading factor (SF), coding rate (CR), and transmission power of LoRa communication as 925MHz, 250kHz, 10, 4/8, and 23 dBm, respectively.

Experiment setup. We evaluate PCube over four months of experiments in a university and neighborhoods spanning 1.08 km × 1.2 km. We conduct experiments in both indoors and outdoors (see Figure 13). The testbed consists of 40 LoRa nodes and a multi-antenna gateway. In outdoor experiments, the gateway is deployed at rooftop of a 54m high building as shown in Figure 12. LoRa nodes are configured to transmit in Low Data Rate (LDR) mode to better combat packet failures during experiments.

Metrics. We evaluate the performance of parallel decoding with three metrics: (1) *Packet Reception Ratio (PRR)* of concurrent packets, (2) *network throughput* defined as the aggregated data rate of correctly decoded packets, and (3) *maximum number of concurrent packets* that can be decoded.

Comparison. We compare PCube against four benchmarks: (1) *LoRaWAN* — a standard LoRa packet decoder without parallel decoding capability; (2) *Choir* [8] — a parallel decoder with collision resolving in frequency domain; (3) *FTrack* [51] — a parallel decoder with collision resolving in time domain; and (4) *MIMO* — an oracle Multi-User MIMO scheme using multiple Rx antennas where the channels between transmitters and Rx antennas are measured in prior. Note that the original decoders of Choir, FTrack and standard LoRaWAN do not use multi-antenna. For fair comparisons, we decode signals of each Rx antenna and select the best decoding results as their final results.

5.2 Air-channel Measurement

Intra-packet coherence. We first test the coherence of air-channels during a packet reception. We measure the difference between PDoAs of symbols and the average PDoA of a packet (*i.e.*, termed *phase deviation*) to characterize the intra-packet coherence of air-channel. As shown in Figure 14(a), PDoA measurements may vary across symbols when a packet is received with poor SNRs. The phase deviation generally follows a Gaussian distribution as shown
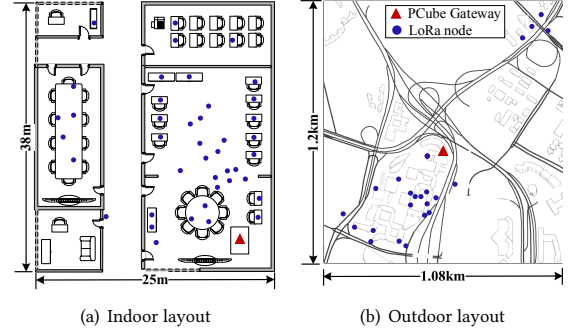
in Figure 14(b). Figure 14(c) displays deviations of PDoA measurements under different SNRs ranging from -10dB to 20dB. We see that phase deviations are smaller than 0.05π when SNR<0dB, and approach to 0 as SNRs increase. It validates that air-channels can be affected by channel conditions but generally remain stable within a packet duration.

Concurrent channel measurements. We next evaluate PDoA measurements in the presence of concurrent transmissions. We first measure PDoA from collision-free packets to obtain groundtruth for performance evaluation. Then, we add up PHY samples of two packets and measure PDoA of each packet using the interleaved signals. Figure 15(a) displays the PDoA measurement errors. We see 70% of the measurements have $< 0.1\pi$ phase errors, indicating that the impact of inter-packet interference is small.

We further examine the impacts of frequency gaps of concurrent symbols on PDoA measurements. We change the initial frequency and power of interfering symbols, and present the results in Figure 15(b-d). We have three observations: (1) Interference may distort the PDoA measurement of a target symbol if the frequency gap between concurrent symbols is smaller than 7 FFT Bins (*i.e.*, $7 \times \frac{BW}{2^{SF}}$). (2) Interference may cause large phase distortions (*e.g.*, $> 0.2\pi$) while the interference power is comparable or higher than the power of target symbol as shown in Figure 15(c,d). (3) Interference symbols do not distort PDoA measurements when the frequency gap is larger than 7 FFT Bins, even when the interference power is strong. It means that we can reliably measure air-channels of concurrent symbols and group symbols accordingly to the correct packets when the chirp frequency of concurrent symbols differ by 7 or more FFT Bins. This enables PCube to handle parallel packet decoding in most cases because both the frequency difference and timing misalignment of concurrent symbols contribute to gaps of chirp frequency. In case that the frequency gaps fall within 7 FFT Bins, the PDoA measurements of concurrent symbols are likely to suffer distortions due to inter-symbol interference. This problem can be mitigated by combining multiple Rx-pairs for reliable PDoA measurements, which is evaluated in the following experiment.

Multi-antenna gains. This experiment investigates the effects of multiple Rx antennas on PDoA measurements. Figure 16(a) evaluates the deviations of PDoA measurements from low-SNR packets using different numbers of Rx antennas. As expected, the deviations become smaller when the data of more Rx-pairs are combined. It
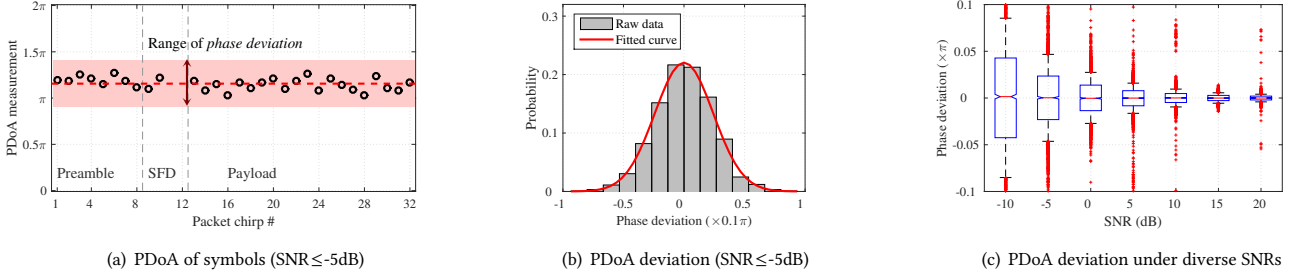
(a) PDoA of symbols (SNR≤-5dB)   (b) PDoA deviation (SNR≤-5dB)   (c) PDoA deviation under diverse SNRs

Figure 14: PDoA measurements of a single packet: (a) PDoA measurements when SNR≤-5dB, (b) PDoA deviation follows Gaussian distribution, (c) PDoA deviations under different SNRs.

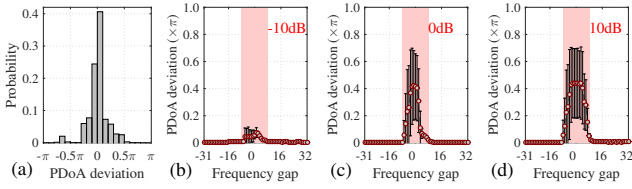(a) PDoA deviation   (b) Frequency gap   (c) Frequency gap   (d) Frequency gap

Figure 15: PDoA measurements of concurrent packets: (a) PDoA measurement errors of concurrent symbols, (b-d) Impacts of interference power on PDoA measurements.

demonstrates that more Rx-pairs are beneficial to combating low SNRs.

Figures 16(b) compares the normalized magnitudes of target symbols and interference when we combine data of 4 Rx-pairs. We see that the magnitude of 90% target symbols are higher than 0.95. In contrast, 90% interference symbols have magnitude lower than 0.8. Figures 16(c) and (d) examine the impacts of the number of Rx antennas on normalized magnitudes of target and interference symbols, respectively. We observe that the magnitude of target symbols are close to 1 across various SNRs, as Rx-pair combining can align target symbols and enhance signal strength. In contrast, the normalized magnitude of interference symbols become weaker as more Rx-pairs are combined, since interference symbols are generally mis-aligned in phase. As a result, target symbols and interference can be separated apart farther in I-Q plane and thus be distinguished more reliably when more Rx-pairs are employed.

5.3 Parallel Decoding Performance

Scalability performance. We conduct this experiment with indoor settings as shown in Figure 13(a). We setup a gateway with up to 8 Rx antennas using 8 synchronized USRP SDRs. The number of LoRa nodes increases from 1 to 40. To ensure that all nodes are synchronized and transmit simultaneously, we use a control node to broadcast beacons every five seconds. Other nodes listen to beacons and respond a 20-Byte data packet. The gateway receives the signals of concurrent packets and decodes with different decoders to evaluate the scalability performance.

Figure 17(a) presents the measured throughput of different decoders. The throughput of standard LoRaWAN is the lowest since it cannot decode concurrent packets. Parallel decoders PCube and

FTrack yield increasing throughput as more nodes transmit simultaneously. The throughput of PCube increases faster than FTrack due to benefits of multiple antennas. The throughput of PCube approaches the ideal upper-bounds of throughput produced by oracle MIMO when the number of concurrent packets is 1∼8. However, the throughput of MIMO stops increasing when the number of concurrent transmissions exceeds 8 (*i.e.*, the maximum number of Rx antennas). In contrast, PCube continuously increases and can support more concurrent transmissions beyond the number of Rx antennas. PCube yields a maximum throughput of 42kbps when decoding 40 concurrent packets, which is 4.9× higher than the best throughput produced by existing parallel decoders (*i.e.*, FTrack and MIMO).

We evaluate the benefits of multiple Rx-pairs by comparing performance of PCube when using 2 Rx antennas and 8 antennas. PCube (8 Rx) uses all data of 8 Rx antennas for Rx-pair combining. PCube (2 Rx) corresponds to the PDoA-based parallel decoder without Rx-pair combining as presented in Section 4.5. For fairness, we run PCube (2 Rx) with data of different Rx-pairs of 8 antennas and select the best decoding results. As shown in Figure 17(a), PCube (8 Rx) is capable of decoding more concurrent packets than PCube (2 Rx). The throughput increment of PCube (8 Rx) over PCube (2 Rx) is more than 50%.

Figure 17(b) examines Packet Reception Ratio (PRR) performance. The PRR of PCube stays close to 1 and decreases slightly to 0.88 when the number of concurrent packets increases to 40. As expected, the PRR of MIMO decreases dramatically when the number of concurrent packets exceeds 8. Similarly, PRRs of Choir and FTrack drop as more nodes transmit simultaneously, indicating limited decoding capabilities (*e.g.*, FTrack≤4 and Choir≤2 packets).

We define decoding capability as the maximum number of concurrent packets that can be decoded under certain PRR criteria. Figure 17(c) evaluates the decoding capability of PCube under 2∼8 Rx antennas. Different from MIMO whose decoding capability increases linearly with the number of Rx antennas, the decoding capability of PCube first increases 'super-linearly' then reaches the maximum when more than 6 Rx antennas are employed. PCube can break the limit of MIMO by decoding more packets than the number of antennas. For example, PCube can concurrently decode 37 packets with PRR>90% when 8 antennas are used, which is 4.6× of the decoding capability of conventional MIMO.

(a) PDoA deviation vs. # of antennas   (b) Target vs. interference (4 Rx-pairs)   (c) Magnitude of target symbol   (d) Magnitude of interference
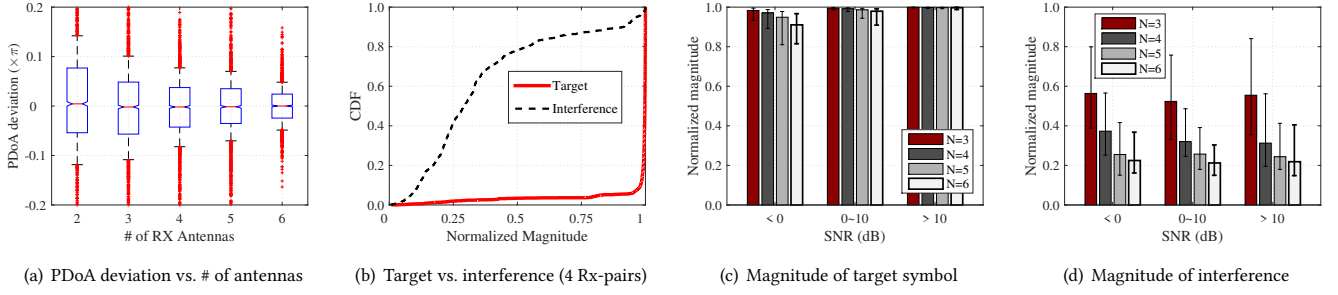
Figure 16: PDoA measurements with multiple antennas: (a) Impacts of multi-antenna on PDoA measurements (SNR<0dB), (b) Magnitudes of target symbols and interference when combining 4 Rx-pairs, (c,d) Impacts of antenna number (N) on magnitudes of target symbols and interference, respectively.

(a) Throughput   (b) Packet Reception Ratio   (c) Decoding capability   (d) Computation overhead
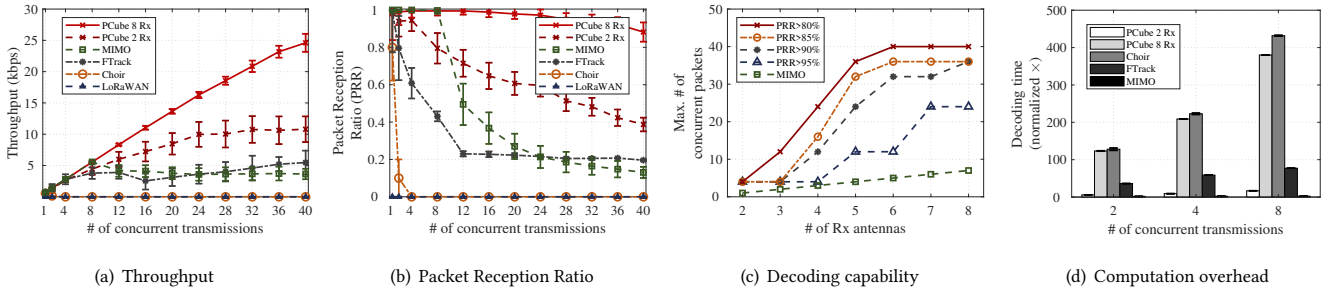
Figure 17: Scalability evaluation: (a,b) Throughput and Packet Reception Ratio as the number of concurrent transmissions increases; (c) Decoding capability of PCube and MIMO; and (d) Comparison of computation overhead.

Computation overhead. Figure 18(a) compares time overheads of various parallel decoders. We run these decoders on a desktop to decode 2, 4 and 8 concurrent packets and measure the decoding time respectively. We use the decoding time of standard LoRa decoder as a benchmark to normalize time measurements of parallel decoders. We see that MIMO runs the fastest, and Choir spends the longest time due to high overhead of frequency feature extraction (*e.g.*, 10,240 samples for each FFT). PCube has shorter decoding time than FTrack when using 2 Rx antennas. When more antennas are used, PCube's decoding time increases yet is still shorter than the time of Choir. As expected, it generally takes more time to decode more concurrent packets. It means that PCube can handle concurrent transmissions on demand in a flexible manner. If no collision occurs or a few packets collide together, we use less Rx antennas (*e.g.*, 2 Rx) for lower computation overhead. To support large-scale LoRaWANs with more concurrent transmissions, we can increase the number of antennas of gateways for higher decoding capability. In this case, PCube indeed incurs higher computational overhead to gateways. It will not be a problem because gateways are usually powerful enough in terms of both computation capability and power supply. In practice, we can reduce the decoding time using high-end gateways and hardware acceleration (*e.g.*, FPGA).

SNR & packet configurations. This experiment examines the impacts of SNR and LoRa packet configuration on PCube performance. We set up 20 nodes to transmit simultaneously and run PCube to decode packets using 2, 4 and 8 Rx antennas, respectively. We carry out experiments outdoors as shown in Figure 13(b), and change the transmission power and locations of LoRa nodes to evaluate a wide range of SNRs. The results are presented in Figure 18(a). We observe that PCube performs better under high SNRs; and more Rx antennas can help decode more packets. For example, the PRR of PCube when SNR>0dB increases from 55.8% to 88.1% as the number of Rx antennas increases from 4 to 8. The PRR improvement is more substantial when SNR<0dB.

Figures 18(b) and (c) evaluate the PRR of PCube with different settings of LoRa Spreading Factor (SF) and Bandwidth (BW) when SNR>0dB. Generally, PCube performs better with larger SF and BW. It means that large SF and BW settings can better support concurrent transmissions. But larger SF and BW also correspond to higher costs of packet decoding and power consumption. An interesting result is that by adding more antennas (*e.g.*, 8 Rx), PCube can decode packets of small SF and BW with PRR>90%. It allows LoRa nodes to transmit concurrently with power efficient parameters and achieve comparable PRR performance.

Near-far effects. In this experiment, we set up two LoRa nodes (A and B) to transmit concurrently. We change the distances from two nodes to gateway to study the impact of near-far effects on PCube performance. We carry out experiments in an outdoor open space. Both nodes have line-of-sight path to the gateway. Node A is placed at a fixed location with 80m away from the gateway.
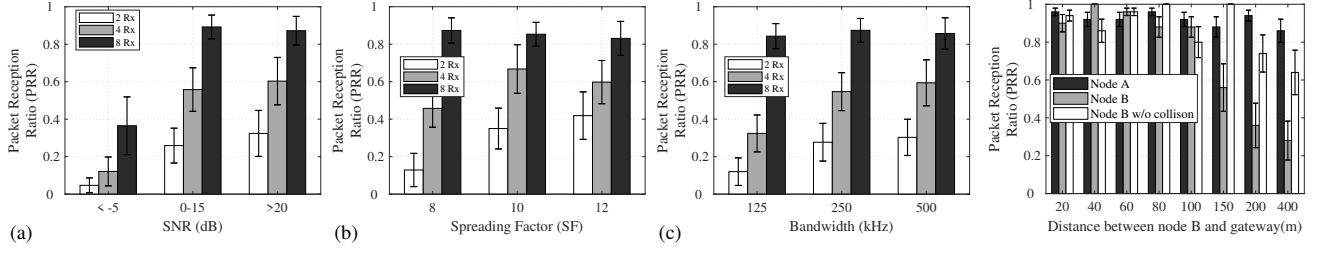
(a)
SNR (dB)

(b)
Spreading Factor (SF)

(c)
Bandwidth (kHz)

Distance between node B and gateway(m)

Figure 18: Impacts of (a) SNR (b) spreading factor and (c) bandwidth on PCube when decoding 20 concurrent packets.

Figure 19: Impact of near-far effects.

We move node B to 8 locations 20m~400m from the gateway. The gateway receives 100 collided packets of the two nodes at each location.

Figure 19 compares PRRs of two nodes at different locations. The PRRs of node B without collision are plotted as baseline comparison for the performance of collision recovery of node B's packets at different locations. We see that the PRR of node A stays higher than 85% across all settings, regardless of the location of node B. This is because the signal strength of node A is high enough for packet decoding. In contrast, the PRR of node B drops dramatically when node B moves to 150m away or farther from the gateway, where the signal strength of node B becomes much weaker than that of node A. We observe that weak transmissions (*e.g.*, node B) are more likely to suffer packet failures in presence of strong concurrent transmissions (*e.g.*, node A).

Proximity of AoA & node locations. If LoRa transmitters are adjacent in physical locations, their air-channels are likely to resemble each other. This experiment tests the performance of PCube on decoding collided packets for closely-located transmitters with the same AoA. We perform the experiment indoors with two LoRa nodes as illustrated in Figure 20(a). The two nodes are aligned along the median line of an Rx-pair of a gateway. We move node 2 along the line to change distances between node 1 and node 2, during which the two nodes are aligned with the same angle in relative to the Rx-pair.

We expect that PDoA measurements would be around phase 0 for both nodes because the two nodes have equal distance to the two Rx antennas. Surprisingly, we observe that even when node 1 and node 2 collocate with a distance shorter than 1cm, their collided packets can still be separated in phase domain as shown in Figure 20(b). Note that although the two nodes are aligned, their signals can be received by an Rx antenna not only from the direct path, but also from many indirect paths due to multi-path effects. As a result, even though the length of direct paths are equal for the two transmitters, their signals may propagate through different paths to reach two antennas of the Rx-pair. Besides, even if two nodes are aligned to an Rx-pair, they can be mis-aligned to other Rx-pairs.

Figure 20(c) presents the packet decoding performance of PCube with different node locations. We see that the packets of both nodes are received with ratios higher than 90%, although the two nodes are close to each other. In particular, the packet reception ratio further increases to >95% as the number of Rx antennas increases from 2 to 6. As a matter of fact, even if some collided packets are
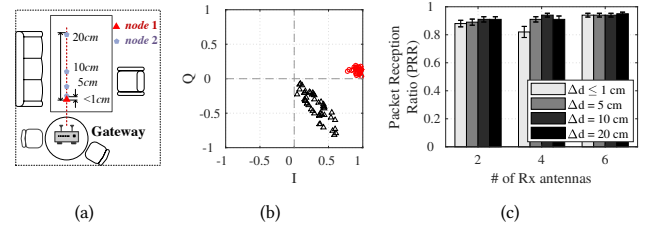

(a)

(b)

(c)

Figure 20: Impact of proximity of node locations: (a) Experiment settings; (b) PDoA measurements from collided packets of the two nodes when $\Delta d \leq 1$cm; and (c) Packet reception performance.

aligned in phase domain in some rare cases, we can still separate the packets by creating reception diversities with more Rx antennas because collided packets are not likely to have phases aligned at all antennas. As long as the air-channel phases of collided packets are different to one of the many receiver pairs, PCube can support their concurrent transmissions.

6 RELATED WORK

Last few years have seen advances in LoRa technology such as performance optimization [1, 5, 6, 9, 13, 22, 27, 34, 39, 57], LoRa backscatter [18, 21, 28, 37, 40], LoRa sensing [4, 10, 32, 35, 53, 58], and LoRa security [17, 23, 24, 49], *etc.*

Our work is most related to concurrent transmissions for LoRa [26, 50, 61]. Choir [8] exploits the frequency offsets introduced by LoRa hardware to separate LoRa collisions. In practice, however, the extracted frequency offset is not reliable to classify colliding symbols due to many influencing factors (*e.g.*, STO, radio frequency leakage). FTrack [51] leverages the misalignment of LoRa chirps in time domain and detects the continuity of chirps within demodulation windows to resolve LoRa collisions. mLoRa [48] detects the time offset between concurrent packets based on preamble correlation results and separates collision samples in physical layer. CoLoRa [44] classifies LoRa symbols to their corresponding LoRa packets according to the power level of the same frequency in different demodulation windows. NScale [43] amplifies the time offsets between colliding packets with non-stationary signal scaling. SCLoRa [25] resolves collisions by leveraging multi-dimensionality and jointly considering frequency and time features. ALIGNER

[33] proposes to avoid the potential ACK collision in concurrent transmissions for low power wireless networks. Those previous works mainly resolve collisions by leveraging the time domain and the frequency domain information. Our work complements and enhances the previous works by exploring a new dimension – phase domain. Our work can help scale concurrent transmissions of LoRa nodes by leveraging reception diversities of multiple Rx antennas of a gateway.

In MAC layer, LMAC [11] enables carrier sensing of LoRa nodes to avoid collisions. S-MAC [55] models the channel access problem as a channel scheduling problem and reduces collisions. The MAC layer protocols can mitigate the collision problem but cannot fundamentally avoid collisions such as hidden terminals.

We draw strength from previous efforts that aim to enable concurrent transmission in Multi-User MIMO systems. IAC [15] presents interference alignment and cancellation for decoding concurrent sender-receiver pairs in MIMO networks. SAM [41] uses chain-decoding to reliably decode concurrent frame transmissions and implements with high performance software defined radios [42]. n$^+$ [31] allows nodes to contend in random access not just for transmission time but also for the degree of freedom provided by multiple antennas of WiFi stations. JMB [38] measures and synchronizes the phase of multiple transmitters in a distributed manner to beamform their signals and communicate with multiple clients in wireless LAN.

Our work is also related to virtual MIMO in information theory [46] which aims to support multiple transmitters to transmit concurrently and make receivers collaborate to jointly decode the concurrent transmissions. Virtual MIMO, however, requires the symbol-level synchronization among distributed transmitters and the transmission of raw physical layer samples among distributed receivers. In practice, the tight synchronization is hard to achieve especially for low-cost IoT devices and the raw sample transmission incur high communication overhead. Unlike virtual MIMO, our work aims to enable concurrent transmissions of a large number of single-antenna LoRa nodes without synchronization. Our work scales the concurrency by exploiting the reception diversities of multiple Rx antennas of a gateway.

7 DISCUSSION

Dynamic change of CFO/STO. We note that as STO is determined by both the arrival time of a packet and sampling time of a receiver, it generally varies across different packets. CFO may also change over time due to hardware imperfection (*e.g.*, clock drift). Despite that, CFO and STO are relatively stable within a packet duration and thus can be estimated and calibrated on a per-packet basis. Even if clock drift takes places within a packet, as CFO and STO dynamics remain the same across synchronized Rx antennas, the time-varying features of CFO and STO can be well mitigated during PDoA extraction with synchronized Rx-pairs.

Communication fairness. PCube can fairly decode all concurrent transmissions when the SNRs of packets are good. However, if a weak packet collides with a strong packet, the weak transmission is more likely to fail due to low SNR of the packet. Essentially, it is

because CFO and STO could be estimated and compensated incorrectly under low SNRs, which in turn leads to measurement errors of channel phase and incorrect symbol classification.

Power consumption. PCube improves the decoding capability of LoRa gateways to decode more concurrent packets that otherwise could not be decoded due to collisions. If a gateway is capable of decoding collided packets of concurrent transmissions, LoRa nodes do not need to re-transmit when collision occurs, which also brings energy benefits to LoRaWANs. More importantly, PCube does not require modifications to the battery-powered LoRa nodes; it puts all computation overheads at the gateway side. Although PCube incurs higher overhead and more power consumption to gateways when the number of collided packets increases, gateways with sufficient computational resource and power supply can handle the overheads in practice. We believe such a design trade-off is worthy for power-constrained LoRa transmitters.

Node mobility. If a node moves, the channel between node and gateway may change across symbols of a packet. If we use the time-varying channel phases to classify symbols of concurrent packets, it may lead to errors in symbol classification and packet decoding. In this case, we can jointly use features in time, frequency and phase domains for collision resolving. For example, as the symbol timing of packets are generally stable in presence of node mobility, we can firstly separate packets from collisions based on symbol timing (*e.g.*, FTrack [51]). For the packets that cannot be separated in time domain, we can check their phase-domain features to recover more packets from collisions. PCube complements existing LoRa parallel decoders by resolving collisions from a new dimension. We plan to study node mobility and its impact on PDoA measurements and extend our current classification model that assumes static PDoAs in the future.

8 CONCLUSION

This paper presents PCube — a phase-based parallel decoder that can scale the concurrent transmissions of LoRa nodes with reception diversities of multiple receiving antennas of a gateway. We overcome several practical challenges in accurately extracting air-channel dependent phase features while canceling various influencing factors such as CFO, STO, and radio frequency leakage. Experiment results demonstrate that PCube can scale communication concurrency beyond the number of receiving antennas. We believe the design of PCube has wider implication. For example, the method of measuring Phase Difference of Air-channels (*i.e.*, PDoA) is promising in developing various systems such as phase-based localization and tracking, wireless physical layer security, and downlink MIMO for LoRaWANs.

9 ACKNOWLEDGEMENTS

REFERENCES

[1] Artur Balanuta, Nuno Pereira, Swarun Kumar, and Anthony Rowe. 2020. A cloud-optimized link layer for low-power wide-area networks. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services (MobiSys'20)*. 247–259.

[2] C. Bernier, F. Dehmas, and N. Deparis. 2020. Low Complexity LoRa Frame Synchronization for Ultra-Low Power Software-Defined Radios. *IEEE Transactions on Communications* 68, 5 (2020), 3140–3152. https://doi.org/10.1109/TCOMM.2020.2974464

[3] Martin C. Bor, Utz Roedig, Thiemo Voigt, and Juan M. Alonso. 2016. Do LoRa Low-Power Wide-Area Networks Scale?. In *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '16)*. Association for Computing Machinery, 59–67.

[4] Lili Chen, Jie Xiong, Xiaojiang Chen, Sunghoon Ivan Lee, Kai Chen, Dianhe Han, Dingyi Fang, Zhanyong Tang, and Zheng Wang. 2019. WideSee: Towards wide-area contactless wireless sensing. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems (SenSys'19)*. 258–270.

[5] Silvia Demetri, Marco Zúñiga, Gian Pietro Picco, Fernando Kuipers, Lorenzo Bruzzone, and Thomas Telkamp. 2019. Automated estimation of link quality for LoRa: A remote sensing approach. In *2019 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN'19)*. IEEE, 145–156.

[6] Adwait Dongare, Revathy Narayanan, Akshay Gadre, Anh Luong, Artur Balanuta, Swarun Kumar, Bob Iannucci, and Anthony Rowe. 2018. 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'18)*. IEEE, 60–71.

[7] Dragino. 2021. *LoRa Shield for Arduino*. Retrieved Mar 25, 2021 from http://www.dragino.com/products/module/item/102-lora-shield.html

[8] R. Eletreby, D. Zhang, S. Kumar, and O. Yagan. 2017. Empowering Low-Power Wide Area Networks in Urban Settings. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication (SIGCOMM'17)*. 309–321.

[9] Akshay Gadre, Revathy Narayanan, Anh Luong, Anthony Rowe, Bob Iannucci, and Swarun Kumar. 2020. Frequency configuration for low-power wide-area networks in a heartbeat. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI'20)*. 339–352.

[10] Akshay Gadre, Fan Yi, Anthony Rowe, Bob Iannucci, and Swarun Kumar. 2020. Quick (and Dirty) Aggregate Queries on Low-Power WANs. In *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN'20)*. IEEE, 277–288.

[11] Amalinda Gamage, Jansen C. Liando, Chaojie Gu, Rui Tan, and Mo Li. 2020. LMAC: Efficient Carrier-Sense Multiple Access for LoRa. In *The 26th Annual International Conference on Mobile Computing and Networking (MobiCom'20)*. Article 43, 13 pages.

[12] Chuhan Gao, Mehrdad Hessar, Krishna Chintalapudi, and Bodhi Priyantha. 2019. Blind Distributed MU-MIMO for IoT Networking over VHF Narrowband Spectrum. In *The 25th Annual International Conference on Mobile Computing and Networking* (Los Cabos, Mexico) *(MobiCom'19)*. Association for Computing Machinery, New York, NY, USA, Article 26, 17 pages.

[13] Weifeng Gao, Zhiwei Zhao, and Geyong Min. 2020. AdapLoRa: Resource Adaptation for Maximizing Network Lifetime in LoRa networks. In *2020 IEEE 28th International Conference on Network Protocols (ICNP'20)*. IEEE, 1–11.

[14] Branden Ghena, Joshua Adkins, Longfei Shangguan, Kyle Jamieson, Philip Levis, and Prabal Dutta. 2019. Challenge: Unlicensed LPWANs Are Not Yet the Path to Ubiquitous Connectivity. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom'19)*. Article 43, 12 pages.

[15] Shyamnath Gollakota, Samuel David Perli, and Dina Katabi. 2009. Interference Alignment and Cancellation. In *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication* (Barcelona, Spain) *(SIGCOMM '09)*. Association for Computing Machinery, New York, NY, USA, 159–170. https://doi.org/10.1145/1592568.1592588

[16] Gr-LoRa GitHub community. 2021. *gr-lora projects*. Retrieved Mar 15, 2021 from https://github.com/rpp0/gr-lora

[17] Chaojie Gu, Linshan Jiang, Rui Tan, Mo Li, and Jun Huang. 2020. Attack-aware data timestamping in low-power synchronization-free LoRaWAN. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS'20)*. 100–110.

[18] Xiuzhen Guo, Longfei Shangguan, Yuan He, Jia Zhang, Haotian Jiang, Awais Ahmad Siddiqi, and Yunhao Liu. 2020. Aloba: rethinking ON-OFF keying modulation for ambient LoRa backscatter. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems (SenSys'20)*. 192–204.

[19] Ezzeldin Hamed, Hariharan Rahul, Mohammed A. Abdelghany, and Dina Katabi. 2016. Real-Time Distributed MIMO Systems. In *Proceedings of the 2016 ACM SIGCOMM Conference* (Florianopolis, Brazil) *(SIGCOMM '16)*. Association for Computing Machinery, New York, NY, USA, 412–425.

[20] Ezzeldin Hamed, Hariharan Rahul, and Bahar Partov. 2018. Chorus: Truly Distributed Distributed-MIMO. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication* (Budapest, Hungary) *(SIGCOMM '18)*. Association for Computing Machinery, New York, NY, USA, 461–475.

[21] Mehrdad Hessar, Ali Najafi, and Shyamnath Gollakota. 2019. Netscatter: Enabling large-scale backscatter networks. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI'19)*. 271–284.

[22] Mehrdad Hessar, Ali Najafi, Vikram Iyer, and Shyamnath Gollakota. 2020. TinySDR: Low-Power SDR Platform for Over-the-Air Programmable IoT Testbeds. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI'20)*. 1031–1046.

[23] Ningning Hou, Xianjin Xia, and Yuanqing Zheng. 2021. Jamming of LoRa PHY and Countermeasure. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE.

[24] Ningning Hou and Yuanqing Zheng. 2020. Cloaklora: A covert channel over LoRa phy. In *2020 IEEE 28th International Conference on Network Protocols (ICNP'20)*. IEEE, 1–11.

[25] Bin Hu, Zhimeng Yin, Shuai Wang, Zhuqing Xu, and Tian He. 2020. SCLoRa: Leveraging Multi-Dimensionality in Decoding Collided LoRa Transmissions. In *2020 IEEE 28th International Conference on Network Protocols (ICNP'20)*. IEEE, 1–11.

[26] Qianyi Huang, Zhiqing Luo, Jin Zhang, Wei Wang, and Qian Zhang. 2020. Lo-Radar: Enabling Concurrent Radar Sensing and LoRa Communication. *IEEE Transactions on Mobile Computing* 01 (nov 2020). https://doi.org/10.1109/TMC.2020.3035797

[27] Linshan Jiang, Rui Tan, and Arvind Easwaran. 2020. Resilience Bounds of Network Clock Synchronization with Fault Correction. *ACM Trans. Sen. Netw.* 16, 4, Article 38 (Sept. 2020), 30 pages. https://doi.org/10.1145/3409804

[28] Mohamad Katanbaf, Anthony Weinand, and Vamsi Talla. 2021. Simplifying Backscatter Deployment: Full-Duplex LoRa Backscatter. In *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI'21)*.

[29] Minsung Kim, Salvatore Mandrà, Davide Venturelli, and Kyle Jamieson. 2021. Physics-inspired heuristics for soft MIMO detection in 5G new radio and beyond. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking (MobiCom'21)*. 42–55.

[30] Jansen C Liando, Amalinda Gamage, Agustinus W Tengourtius, and Mo Li. 2019. Known and unknown facts of LoRa: Experiences from a large-scale measurement study. *ACM Transactions on Sensor Networks (TOSN)* 15, 2 (2019), 1–35.

[31] Kate Ching-Ju Lin, Shyamnath Gollakota, and Dina Katabi. 2011. Random Access Heterogeneous MIMO Networks. In *Proceedings of the ACM SIGCOMM 2011 Conference* (Toronto, Ontario, Canada) *(SIGCOMM '11)*. Association for Computing Machinery, New York, NY, USA, 146–157. https://doi.org/10.1145/2018436.2018454

[32] Yuxiang Lin, Wei Dong, Yi Gao, and Tao Gu. 2021. SateLoc: A Virtual Fingerprinting Approach to Outdoor LoRa Localization Using Satellite Images. *ACM Trans. Sen. Netw.* 17, 4, Article 43 (July 2021), 28 pages.

[33] Daibo Liu, Zhichao Cao, Mengshu Hou, Huigui Rong, and Hongbo Jiang. 2020. Pushing the Limits of Transmission Concurrency for Low Power Wireless Networks. *ACM Trans. Sen. Netw.* 16, 4, Article 40 (Sept. 2020), 29 pages.

[34] Jun Liu, Weitao Xu, Sanjay Jha, and Wen Hu. 2020. Nephalai: towards LPWAN C-RAN with physical layer compression. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking (MobiCom'20)*. 1–12.

[35] Rajalakshmi Nandakumar, Vikram Iyer, and Shyamnath Gollakota. 2018. 3D localization for sub-centimeter sized devices. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems (SenSys'18)*. 108–119.

[36] R. Narayanan, S. Kumar, and S. R. Murthy. 2020. Cross Technology Distributed MIMO for Low Power IoT. *IEEE Transactions on Mobile Computing* (2020), 1–1. https://doi.org/10.1109/TMC.2020.3029218

[37] Yao Peng, Longfei Shangguan, Yue Hu, Yujie Qian, Xianshang Lin, Xiaojiang Chen, Dingyi Fang, and Kyle Jamieson. 2018. PLoRa: A passive long-range data network from ambient LoRa transmissions. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication (SIGCOMM'18)*. 147–160.

[38] Hariharan Shankar Rahul, Swarun Kumar, and Dina Katabi. 2012. JMB: Scaling Wireless Capacity with User Demands. In *Proceedings of the ACM SIGCOMM 2012 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication* (Helsinki, Finland) *(SIGCOMM '12)*. Association for Computing Machinery, New York, NY, USA, 235–246.

[39] Yaman Sangar and Bhuvana Krishnaswamy. 2020. WiChronos: energy-efficient modulation for long-range, large-scale wireless networks. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking (MobiCom'20)*. 1–14.

[40] Vamsi Talla, Mehrdad Hessar, Bryce Kellogg, Ali Najafi, Joshua R Smith, and Shyamnath Gollakota. 2017. Lora backscatter: Enabling the vision of ubiquitous connectivity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (UbiComp'17)* 1, 3 (2017), 1–24.

[41] Kun Tan, He Liu, Ji Fang, Wei Wang, Jiansong Zhang, Mi Chen, and Geoffrey M. Voelker. 2009. SAM: Enabling Practical Spatial Multiple Access in Wireless LAN. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking* (Beijing, China) *(MobiCom '09)*. Association for Computing Machinery, New York, NY, USA, 49–60. https://doi.org/10.1145/1614320.1614327

[42] Kun Tan, He Liu, Jiansong Zhang, Yongguang Zhang, Ji Fang, and Geoffrey M. Voelker. 2011. Sora: High-Performance Software Radio Using General-Purpose Multi-Core Processors. *Commun. ACM* 54, 1 (Jan. 2011), 99–107. https://doi.org/10.1145/1866739.1866760

[43] Shuai Tong, Jiliang Wang, and Yunhao Liu. 2020. Combating Packet Collisions Using Non-Stationary Signal Scaling in LPWANs. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services (MobiSys '20)*. 234–246.

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

[45] Benny Vejlgaard, Mads Lauridsen, Huan Nguyen, Istvan Z. Kovacs, Preben Mogensen, and Mads Sorensen. 2017. Coverage and Capacity Analysis of Sigfox, LoRa, GPRS, and NB-IoT. In *Proceedings of the 2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*. 1–5.

[46] Pramod Viswanath and David N. C. Tse. 2003. Sum capacity of the vector Gaussian broadcast channel and uplink–downlink duality. *IEEE Transactions on Information Theory* 49, 8 (2003), 1912–1921.

[47] Shuai Wang, Woojae Jeong, Jinhwan Jung, and Song Min Kim. 2020. X-MIMO: cross-technology multi-user MIMO. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems (SenSys'20)*. 218–231.

[48] Xiong Wang, Linghe Kong, Liang He, and Guihai Chen. 2019. mLoRa: A Multi-Packet Reception Protocol in LoRa networks. In *2019 IEEE 27th International Conference on Network Protocols (ICNP'19)*. IEEE, 1–11.

[49] Xiong Wang, Linghe Kong, Zucheng Wu, Long Cheng, Chenren Xu, and Guihai Chen. 2020. SLoRa: towards secure LoRa communications with fine-grained physical layer features. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems (SenSys'20)*. 258–270.

[50] Zhe Wang, Linghe Kong, Kangjie Xu, Liang He, Kaishun Wu, and Guihai Chen. 2020. Online concurrent transmissions at LoRa gateway. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2331–2340.

[51] Xianjin Xia, Yuanqing Zheng, and Tao Gu. 2019. FTrack: Parallel decoding for LoRa transmissions. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems (SenSys'19)*. 192–204.

[52] Xianjin Xia, Yuanqing Zheng, and Tao Gu. 2020. LiteNap: Downclocking LoRa Reception. In *IEEE INFOCOM'20*. 2321–2330.

[53] Binbin Xie and Jie Xiong. 2020. Combating interference for long range LoRa sensing. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems (SenSys'20)*. 69–81.

[54] Yaxiong Xie, Yanbo Zhang, Jansen Christian Liando, and Mo Li. 2018. SWAN: Stitched Wi-Fi ANtennas. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking* (New Delhi, India) *(MobiCom '18)*. Association for Computing Machinery, New York, NY, USA, 51–66.

[55] Zhuqing Xu, Junzhou Luo, Zhimeng Yin, Tian He, and Fang Dong. 2020. S-MAC: Achieving High Scalability via Adaptive Scheduling in LPWAN. In *IEEE INFOCOM'20*. 506–515.

[56] Deliang Yang, Xianghui Zhang, Xuan Huang, Liqian Shen, Jun Huang, Xiangmao Chang, and Guoliang Xing. 2020. Understanding power consumption of NB-IoT in the wild: tool and large-scale measurement. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking (MobiCom'20)*. 1–13.

[57] Mingran Yang, Junbo Zhang, Akshay Gadre, Zaoxing Liu, Swarun Kumar, and Vyas Sekar. 2020. Joltik: enabling energy-efficient "future-proof" analytics on low-power wide-area networks. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking (MobiCom'20)*. 1–14.

[58] Fusang Zhang, Zhaoxin Chang, Kai Niu, Jie Xiong, Beihong Jin, Qin Lv, and Daqing Zhang. 2020. Exploring lora for long-range through-wall sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (UbiComp'20)* 4, 2 (2020), 1–27.

[59] Renjie Zhao, Timothy Woodford, Teng Wei, Kun Qian, and Xinyu Zhang. 2020. M-Cube: A millimeter-wave massive MIMO software radio. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–14.

[60] Renjie Zhao, Fengyuan Zhu, Yuda Feng, Siyuan Peng, Xiaohua Tian, Hui Yu, and Xinbing Wang. 2019. OFDMA-enabled Wi-Fi backscatter. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom'19)*. 1–15.

[61] Marco Zimmerling, Luca Mottola, and Silvia Santini. 2020. Synchronous transmissions in low-power wireless: A survey of communication protocols and network services. *ACM Computing Surveys (CSUR)* 53, 6 (2020), 1–39.