The following publication S. Yu, X. Xia, N. Hou, Y. Zheng and T. Gu, "XGate: Scaling LoRa Communications to Massive Logical Channels," in IEEE Transactions on Networking, vol. 33, no. 5, pp. 2352-2366, Oct. 2025 is available at https://doi.org/10.1109/TON.2025.3560408.

# XGate: Scaling LoRa Communications to Massive Logical Channels

Shiming Yu, *Student Member, IEEE, ACM*, Xianjin Xia, *Member, IEEE, ACM*, Ningning Hou, *Member, IEEE, ACM*, Yuanqing Zheng, *Senior Member, IEEE, Member, ACM*, and Tao Gu, *Fellow, IEEE, Member, ACM*

***Abstract*—LoRa is a promising technology that provides widespread low-power IoT connectivity. With its capabilities for multi-channel communication, orthogonal transmission, and spectrum sharing, LoRaWAN is poised to connect millions of IoT devices across thousands of logical channels. However, current LoRa gateways rely on hardwired Rx chains that cover less than 1% of these channels, restricting the potential for large-scale LoRa communications. This paper introduces XGate, a groundbreaking gateway design that uses a single Rx chain to simultaneously receive packets from all logical channels, enabling scalable LoRa transmission and flexible network access. Unlike the hardwired Rx chains in existing gateway designs, XGate dynamically allocates resources, including software-controlled Rx chains and demodulators, based on the extracted meta-information of incoming packets. XGate overcomes several challenges to efficiently detect incoming packets without prior knowledge of their parameter configurations. Evaluations demonstrate that XGate enhances LoRa concurrent transmissions by 8.4× compared to state-of-the-art solutions.**

***Index Terms*—Internet of Things, LPWAN, LoRa, Logical Channel**

## I. Introduction

Low-Power Wide-Area Networks (LPWANs) have become a promising solution for connecting numerous physical devices to the Internet. Among these technologies, LoRa stands out with its extensive coverage and ultra-low power communication capabilities, making it particularly suitable for low data-rate IoT applications [1]–[9] like smart agriculture [10]–[12] and smart metering [13]–[15]. With its potential to provide widespread, cost-effective IoT connectivity across entire cities, LoRa is well-positioned to become the leading IoT technology.

Manuscript received 24 July 2024; revised 12 December 2024, 11 February 2025 and 24 March 2025; accepted 8 April 2025; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor xxx. Date of publication xxx; date of current version xxx; This work was supported in part by the Hong Kong General Research Fund (GRF) under Grant 15218022, 15231424, 15211924, 15206123, in part by the National Nature Science Foundation of China (NSFC) under Grant 62102336, and in part by the Innovation Capability Support Program of Shaanxi (No. 2023-CX-TD-08) Shaanxi Qinchuangyuan "scientists+engineers" team (No. 2023KXJ040). Xianjin Xia and Yuanqing Zheng are the corresponding authors.

Shiming Yu, Xianjin Xia, and Yuanqing Zheng are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China (email: shiming.yu@connect.polyu.hk; xianjin.xia@polyu.edu.hk; yuanqing.zheng@polyu.edu.hk)

Ningning Hou and Tao Gu are with the School of Computing, Macquarie University, Sydney, NSW 2109, Australia (email: ningning.hou@mq.edu.au; tao.gu@mq.edu.au)

Digital Object Identifier xxxx

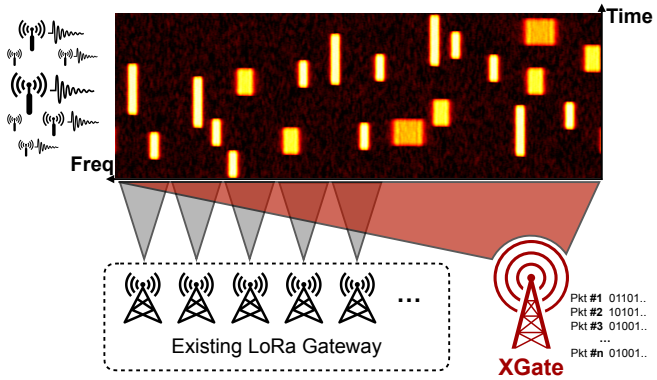



Fig. 1. Massive LoRa nodes can freely select from thousands of logical channels for transmission in a "come-and-be-served" manner. XGate covers all available logical channels with auto-configuration.

LoRa employs a series of designs at both the PHY and MAC layers to facilitate extensive IoT connectivity. *Multi-channel communication*: LoRa packets use a narrow bandwidth (*e.g.*, 125 kHz by default), allowing the entire LoRa spectrum to be divided into numerous sub-channels. This enables multiple LoRa nodes to communicate simultaneously on different physical channels without interfering with each other. *Orthogonal communication*: The LoRa PHY layer modulates packets using various parameters such as bandwidth (BW) and Spreading Factors (SF). Signals with different SF or BW can be orthogonal, allowing them to be demodulated even when transmitted on the same physical channel. This means that orthogonal *logical channels* with distinct packet parameters can be established over a single physical channel. For instance, the US902-928 MHz ISM band can be split into 208 physical channels. By utilizing these orthogonal configurations, thousands of logical channels can be created to support concurrent transmissions.

Traditional gateways use multiple hardwired Rx chains to receive LoRa packets from only a limited number of logical channels (*e.g.*, nine channels [16]). Due to this constraint, all connected nodes are forced to operate on a few channels, resulting in significant packet collisions and reduced network performance. Additionally, as a gateway's operating channels are typically pre-configured, they may not be well-suited to adapt to dynamic network conditions. Current gateways do not offer enough channels for LoRa nodes to effectively adapt

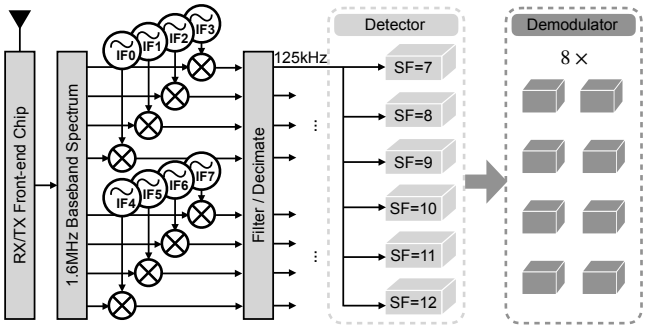



Fig. 2. Block diagram of a COTS gateway [16].

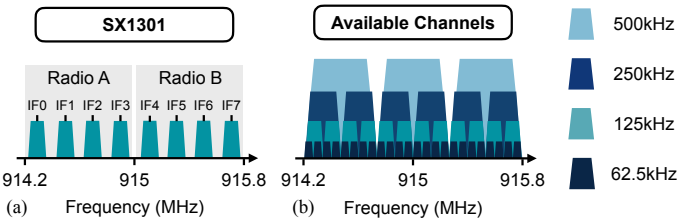



Fig. 3. (a) A feasible channel configuration of a COTS gateway; (b) Diverse channel settings for LoRa.

to the changing conditions. While it is possible to increase the number of gateways or Rx chains per gateway to expand the available channels, this approach would require a large number of gateways and Rx chains to adequately cover all logical channels across the entire LoRa spectrum.

Rather than assembling multiple Rx chains and fixing each one to a specific channel like existing gateways, our approach aims to support all available LoRa channels using a single Rx chain through auto-configuration. We accomplish this by efficiently detecting incoming packets from any channel on-the-fly and dynamically allocating software-controlled radio resources (*e.g.*, Rx chains, packet demodulators) on demand for packet reception and demodulation. As illustrated in Fig. 1, our design allows LoRa nodes to seamlessly connect to deployed networks in a "come-and-be-served" manner, eliminating the need for complex channel configurations for both nodes and gateways. This innovative gateway design also enables LoRa nodes to freely choose logical channels and parameter settings for their subsequent transmissions, allowing for rapid adaptation to changing network conditions. In addition to breaking the barrier of massive IoT connections and enabling scalable concurrent transmissions across all LoRa logical channels, we believe this design has significant potential to support new communication and networking paradigms for LoRaWAN.

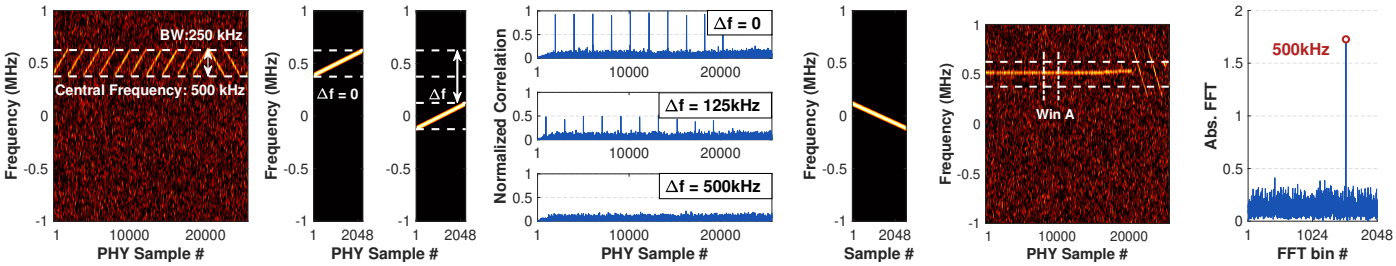
To address these challenges, this paper introduces *XGate*, an innovative gateway design that facilitates flexible packet reception across numerous LoRa channels. The core of XGate's functionality lies in separating the LoRa packet reception process into two distinct phases: *packet detection* and *packet decoding*. During the packet detection phase, a packet detector scans the entire Rx spectrum to identify incoming packets and extract their specific configurations. XGate utilizes a pool of software-based Rx chains and packet decoders, which can be dynamically configured based on the detection results, allowing for flexible reception on particular LoRa channels.

Implementing this idea in practice presents substantial challenges. Unlike traditional gateways that monitor fixed channels, XGate aims to receive packets from all logical channels within the Rx spectrum. Since packets can arrive on any logical channel without prior knowledge, identifying a packet across a broad spectrum is complex, not to mention recovering the packet's parameters. The challenge intensifies when multiple packets arrive simultaneously from different channels. Existing gateways can use preset channel information to filter out interference from unwanted channels, but with XGate, signals from different sub-channels are combined in a shared spectrum, leading to potential *cross-channel interference* that complicates the detection of concurrent packets. Additionally, detecting packets using raw signals from a wide Rx spectrum, rather than the narrow-band signals of a single LoRa channel, may result in SNR losses due to increased noise and interference. Therefore, it is crucial, yet challenging, for XGate to detect and receive packets without sacrificing sensitivity.

XGate introduces an innovative design that enables the detection and reception of LoRa packets across a wide spectrum while maintaining high resilience to noise and interference. We discovered that a chirp signal from one channel can be used to dechirp signals from other channels, provided they share the same chirp slope (*i.e.*, the rate of frequency change over time). The dechirped signals display periodic patterns in both the frequency and time domains, due to the repetitive structure of a LoRa preamble, which is utilized to detect incoming packets. We leverage the fact that the frequency and periodicity of dechirped signals inherently convey information about a packet's channel and chirp profiles (*e.g.*, bandwidth, spreading factor). Concurrent packets with different channels or chirp configurations typically manifest as distinct frequencies and unique periodicity patterns, which can be separated in the frequency and time domains. Based on these insights, XGate employs a novel strategy that tracks the frequency changes of dechirped signals across consecutive detection windows to accurately identify communication channels and parameters for concurrent packets. XGate scans the signals in multiple rounds using various chirp slopes and window sizes, where short and long windows are used together to balance sensitivity and detection delays. Finally, XGate configures software Rx chains with the appropriate parameters to receive the detected packets.

We implemented and evaluated XGate using Software Defined Radio (SDR) and commercial off-the-shelf (COTS) LoRa nodes. The results show that XGate significantly outperforms existing solutions. Specifically, XGate supports up to 8.4× more concurrent transmissions in the same spectrum compared to the state-of-the-art [17]. Additionally, XGate can be seamlessly integrated with existing LoRa parallel decoding techniques [17], [18] to support 78% more concurrent transmissions.

In summary, our work offers the following contributions: (1) XGate is the first gateway design of its kind that enables the scaling of LoRa concurrent transmissions across logical




(a) A received preamble (b) Local up-chirps (c) Correlation detection (d) Down-chirp (e) Dechirp results (f) FFT in Win A

Fig. 4. Packet detection in a large signal spectrum (2 MHz): (a) Raw signals of a LoRa preamble; (b,c) Correlation detection results with local up-chirps in different channel settings; (d-f) Results of dechirping preamble signals with a local down-chirp.

channels, revolutionizing the potential for massive IoT communications; (2) We introduce novel techniques to efficiently detect channel activities across all LoRa channels within a given spectrum and accurately extract the configurations of incoming packets; (3) We implement and evaluate XGate to validate its effectiveness, with our evaluations demonstrating significant improvements achieved by our design. Furthermore, we believe the design of XGate will have broader impacts on spectrum management, network planning, and channel adaptation in LoRaWAN operations and deployments.

## II. Packet Reception of COTS Gateways

The background of LoRa physical layer and channel settings is introduced in [19]. COTS LoRa gateways are typically equipped with Semtech chipsets [20]. Figure 2 shows a simplified packet reception workflow and functional blocks.

*RF reception*. The RF front-end of a gateway receives signals over the air and converts the incoming RF signals to the baseband. As illustrated in Fig. 2, a COTS gateway (*e.g.*, SX1301) uses two front-end radios with each covering 0.8 MHz RF spectrum [16].

*IF channel reception*. The radio feeds received baseband signals to a few parallel Rx chains (*e.g.*, IF0∼7), which can be programmed to receive on different physical channels within the spectrum. An Rx chain only passes targeted frequencies while filtering undesired frequencies, and then decimates signals according to channel bandwidth. The channel frequencies of the Rx chains can be configured individually.

*Packet detection*. A gateway uses LoRa packet detector to monitor the eight channels for new packet arrivals. A packet detector searches for LoRa preambles of various SFs (*e.g.*, SF7∼12) on each of the IF channels. Packets with different SFs can be detected simultaneously in the same channel.

*Packet demodulation*. Upon detecting a packet, a COTS gateway uses the corresponding SF of a detected LoRa preamble to demodulate the packet. However, a COTS gateway can demodulate only a few packets concurrently due to resource constraints. If many more packets arrive at the same time, even if detected, some packets cannot be demodulated.

Fig. 3(a) illustrates a feasible channel configuration of a COTS gateway. The central frequencies of two front-end radios are set to 914.6 MHz and 915.4 MHz, respectively. The whole spectrum is partitioned into eight non-overlapping physical channels with each 125 kHz wide. The gateway operates on the eight channels simultaneously to detect and receive incoming packets. Note that the operating channels of a gateway are usually configured beforehand and stay invariant after installation. It typically requires in-situ spectrum measurement and analysis to select and configure operating channels for deployed gateways, which can incur high measurement and maintenance overhead especially when the network size is large and scales over time.

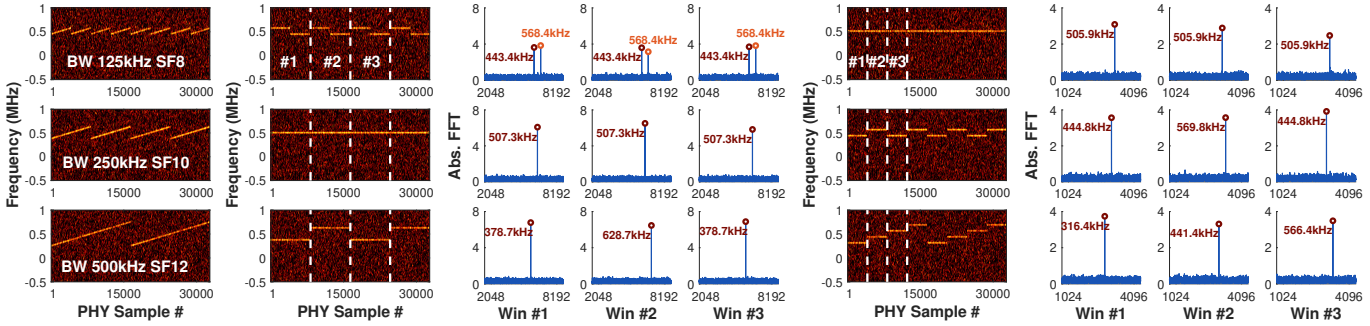
We observe that there is still a large gap in using COTS gateways to support massive LoRa communications. As all connected nodes can only use the few operating channels of a COTS gateway, it can lead to poor spectrum utilization and sub-optimal network performance due to collisions. In contrast, LoRaWAN supports a variety of channel frequencies and bandwidths. Thus, the actual channel settings available for LoRa nodes can be rather diverse. As shown in Fig. 3(b), much more logical channels can be created in the same spectrum when three new bandwidths other than 125 kHz are used. Ideally, if a gateway can cover all logical channels in the spectrum, it can not only support more nodes with reduced collisions but also allow nodes to freely choose among diverse channel settings to better adapt to changing network conditions. However, due to the large amounts of LoRa logical channels, it is challenging and even impractical to use COTS gateways to cover every channel with a dedicated Rx chain. It calls for a better gateway design that can detect and receive packets from all possible LoRa channels in a large spectrum.

## III. XGate Design

### A. Overview

XGate maintains a pool of ”soft-wired” Rx chains that can be dynamically configured on-the-fly to receive signals at any frequency with various bandwidth settings. A key component of XGate is a robust packet detector capable of scanning a wide spectrum to identify the detailed parameters of incoming packets (*e.g.*, central frequency, bandwidth, and spreading factor) and flexibly configuring the software Rx chains to receive the detected packets.

However, implementing XGate presents significant challenges. First, given the potentially large number of LoRa logical channels (e.g., over hundreds), detecting packets across so many possible channels without prior knowledge of their meta-information is complex. Second, while XGate monitors




(a) Preamble signals (b) Dechirp (setting #1) (c) FFTs of three windows (setting #1) (d) Dechirp (setting #2) (e) FFTs of three windows (setting #2)

Fig. 5. Detect BW-SF configurations: (a) Signals of three packets having the same chirp slope but different BW-SF settings; (b,c) Dechirped results with a down-chirp in (250 kHz, SF10); (d,e) Dechirped results with a down-chirp in (125 kHz, SF8).

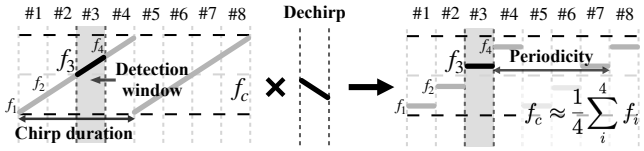



Fig. 6. Dechirp with short windows.

the entire Rx spectrum to detect LoRa packets, signals originally separated into different sub-channels can cause *cross-channel interference*. Correctly detecting concurrent packets from different sub-channels and determining their detailed parameters is a major challenge for XGate. Additionally, since an Rx spectrum covers wider frequency bands than a single LoRa channel, it includes more noise and interference, which can degrade signal quality. Receiving packets without compromising sensitivity is challenging. We present novel techniques to address these challenges in the following subsections.

### B. Packet Detection over Logical Channels

*1) Detecting Packet and Meta-information:* As a packet may arrive from any logical channel of an Rx spectrum that is unknown to a gateway, XGate needs to detect not only the arrival of an incoming packet but also the channel and the parameter configuration of the packet.

**Packet detection**. XGate processes the raw signals from the entire Rx spectrum to detect incoming packets. A standard LoRa packet detection method involves correlation detection, where incoming signals are correlated with a locally-generated LoRa preamble to detect packets on a specific LoRa channel. This method is commonly used in existing systems [16], [21]. Can we extend this approach to detect packets across the entire Rx spectrum? Figure 4(a) shows a LoRa packet received within a 2 MHz-wide spectrum. We correlate these signals with a local preamble chirp that has the same spreading factor and bandwidth but different channel settings, as depicted in Fig. 4(b). As illustrated in Figure 4(c), we only observe the highest correlation peaks when the central frequencies of the local chirp and the received signals match exactly (*i.e.*, $\Delta f = 0$). This indicates that the channel information (*e.g.*, central frequency) of a packet is necessary to initiate correlation-based detection for a specific channel. Clearly, this method is not suitable for XGate, as the channel information of an incoming packet is not known in advance.

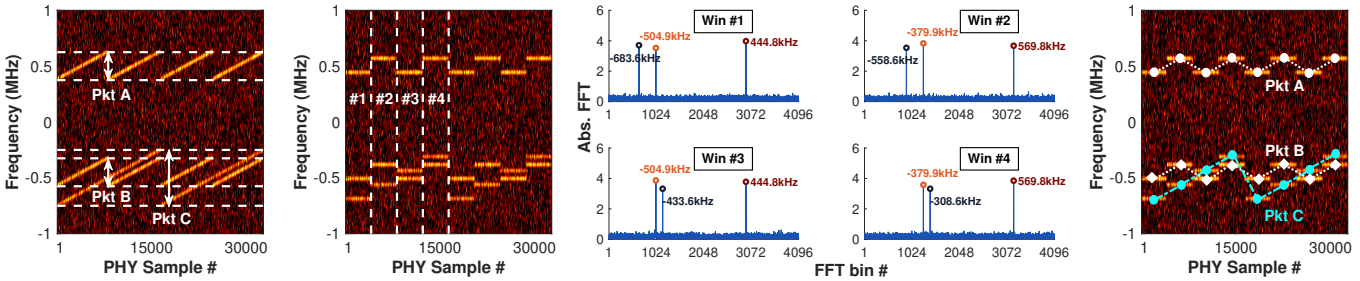
How can we detect a packet without knowing its channel? Fortunately, since the Rx spectrum of a gateway encompasses many narrow LoRa channels, the signals from all LoRa logical channels are physically received. This allows us to freely access signals from any LoRa logical channel within the Rx spectrum. For example, we can use a down-chirp signal centered at frequency 0, as illustrated in Fig. 4(d), to dechirp the signals shown in Fig. 4(a), resulting in the output depicted in Fig. 4(e). If a LoRa preamble is present, we will detect the same tone frequency in successive windows. This enables us to detect a LoRa packet from the Rx spectrum even when the packet's channel is unknown.

XGate extends the dechirp operation from a single LoRa channel to an entire Rx spectrum to detect a packet, which can be represented as follows:

$$h_i e^{j2\pi f_i t} C_i(t) \cdot C_i^{-1}(t + \Delta t_i) = h_i e^{j2\pi f_i t} \cdot e^{j2\pi \Delta f_i t}, \quad (1)$$

where $h_i e^{j2\pi f_i t} C_i(t)$ denotes a received preamble chirp in a LoRa channel centered at frequency $f_i$, and $C_i^{-1}(t)$ is a local down-chirp centered at frequency 0 with the corresponding $BW_i$ and $SF_i$. Here, $\Delta t_i$ models the misaligned timing between symbol edges of incoming signals and the local down-chirp, which leads to a frequency component $\Delta f_i = (BW_i^2/2^{SF_i})\Delta t_i$ in the dechirped results. Theoretically, the tone-frequency of dechirped signals (Fig. 4(f)) is essentially $(f_i + \Delta f_i)$. Since edge misalignment $\Delta t_i$ is within $\pm 0.5\times$ of a chirp duration, the resulting $\Delta f_i$ shall be within $\pm 0.5\times$ of the chirp's bandwidth (*i.e.*, $BW_i$). The tone-frequency $(f_i + \Delta f_i)$ deviates at most $\frac{BW_i}{2}$ from $f_i$ (*i.e.*, a channel's central frequency). Hereby, the tone-frequency of dechirped signals not only indicates the presence of an incoming packet, but also roughly estimates its central frequency.

**Extracting packet parameters**. After detecting an incoming packet, XGate must extract the packet's BW and SF configurations for demodulation. Since LoRa chirps with different BWs or SFs are typically orthogonal, if the BW/SF of the down-chirp in Eq.(1) differs from that of the received packet, the packet signals will not be dechirped to a tone frequency and will not be detected. By leveraging chirp orthogonality,




(a) Preambles of three packets (b) Dechirped results (c) FFTs in consecutive windows (d) Freq. tracks of three packets

Fig. 7. Detect concurrent packets: (a) Compound signals of packets A, B in (250 kHz, SF10) and packet C in (500 kHz, SF12); (b,c) Dechirped frequencies of signals from the same packet change at a fixed rate ($\Delta f$=125 kHz) over consecutive windows; (d) Track frequency changes to separate the three packets.

we can iterate through all BW-SF combinations to identify the correct configurations of a packet.

However, this method may encounter ambiguous detection because a packet can be dechirped by various chirps with different BWs or SFs. For example, the BW and SF of the three signals shown in Fig. 5(a) are (125 kHz, SF8), (250 kHz, SF10) and (500 kHz, SF12), respectively. When using a down-chirp with BW 250 kHz and SF10 to dechirp the three signals, strong tone frequencies can be detected from not only the (250 kHz, SF10) signal but also the other two signals. This occurs because they share the same *chirp slope*, defined as $k = BW^2/2^{SF}$, making them not entirely orthogonal and allowing them to be dechirped by the same down-chirp with a slope of $-k$. Specifically, Fig. 5(b,c) illustrates the dechirped frequencies of these signals across three successive windows. Identical single-tone frequencies are detected in successive windows for (250 kHz, SF10). However, for (125 kHz, SF8), two frequencies are detected repeatedly in each window, creating ambiguity as to whether there is only one packet at (125 kHz, SF8) or two packets at (250 kHz, SF10).

XGate resolves such ambiguities by utilizing the fact that chirp signals with different bandwidths (BW) or spreading factors (SF) have varying chirp lengths. Specifically, by using a shorter down-chirp (*e.g.*, the shortest chirp length among the three signals) to dechirp the signals shown in Figure 5(a), we obtain a unique single-tone frequency in each short window. The frequency changes across windows, displaying distinctive periodic patterns for the three packets, as illustrated in Figure 5(d,e). By detecting the periodicity of these frequencies with short windows, XGate can differentiate the three packets and accurately recover the BW and SF for each one.

**Putting it together**. XGate employs down-chirps with different slopes for packet detection, rather than iterating through all bandwidth-spreading factor (BW-SF) combinations. Since multiple BW-SF settings can share the same chirp slope, we can efficiently examine fewer slopes instead of all possible BW-SF settings to detect packets. To prevent ambiguities, we use a short window to dechirp and identify signal patterns, typically selecting a down-chirp with the shortest length for a given slope.

We rewrite Eq.(1) to represent the dechirp operation of a short window as below:

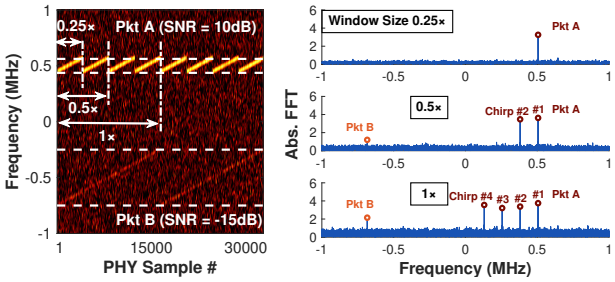
$$h_i e^{j2\pi f_i t} \cdot e^{j2\pi f_k t} C_w(k_i, t) \cdot C_w^{-1}(k_i, t) = h_i e^{j2\pi f_i t} \cdot e^{j2\pi f_k t}, \quad (2)$$

where $e^{j2\pi f_k t} C_w(k_i, t)$ is part of a long up-chirp located in the target window, $f_k$ denotes the starting frequency of the chirp segment (Fig. 6), $k_i$ is the chirp slope, and $C_w^{-1}(k_i, t)$ is a corresponding down-chirp in slope $k_i$. According to Eq.(2), the tone frequency shown in any window of Fig. 5(e) is essentially $(f_i + f_k)$, which signifies the starting frequency of the chirp segment in a short window. The starting frequency $(f_i + f_k)$ would change across windows and the periodicity of frequency changes equals the length of a complete chirp signal as Fig. 6 illustrates.

XGate identifies the periodicity of dechirped frequencies to determine the chirp length of received signals, allowing for the unambiguous restoration of the signals' bandwidth (BW) and spreading factor (SF) configurations. Additionally, XGate calculates the average of the frequencies (*i.e.*, $(f_i + f_k)$'s) detected from different windows, which approximates $f_i$, to estimate the channel frequency of a received packet. A more precise method for estimating the channel frequency is detailed in §III-D.

*2) Detection of Multiple Packets:* When multiple nodes transmit through different logical channels, XGate is designed to detect and receive all packets concurrently. Specifically, since concurrent packets have orthogonal parameters (*i.e.*, different chirp slopes), XGate can separate their signals by using down-chirps with different slopes to detect orthogonal packets in a divide-and-conquer manner. However, if the signals from packets on different logical channels share a common chirp slope, they will interfere with each other, resulting in *cross-channel interference*. For instance, as shown in Fig. 7(a), three packets with two chirp configurations, (250 kHz, SF10) and (500 kHz, SF12), have the same chirp slope. When the compound signals of the entire spectrum are dechirped, three frequencies from different packets are detected in each window, as illustrated in Fig. 7(c), which introduces interference in the packet detection and parameter extraction process. In such cases, XGate must distinguish the frequencies of the three packets and use the frequencies of each packet separately to ensure accurate packet detection and parameter extraction.

If two packets are apart in the frequency spectrum (*e.g.*, Pkt




(a) Preamble signals (b) Dechirped frequencies

Fig. 8. Use long windows to detect weak packets: (a) Received signals of two packets, *i.e.*, Pkt A in (125 kHz, SF8) and Pkt B in (500 kHz, SF12); (b) Detected results with 0.25×, 0.5×, and 1× of the chirp length of Pkt B.

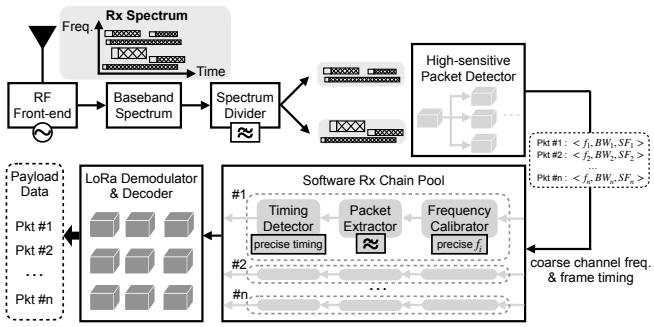



Fig. 9. System architecture of XGate.

A in Fig. 7(b)), we can easily separate concurrent packets by checking the ranges of detected frequencies. However, such a method may not function well if the frequencies of concurrent packets are close or overlapping each other (*e.g.*, Pkt B and C in Fig. 7(b)). To achieve the highest communication concurrency, we expect XGate can correctly receive packets even if the packets have overlapping frequencies.

XGate leverages the signal structures of LoRa preambles to separate concurrent packets in the same chirp slope. The intuition is that the dechirped frequencies of preamble chirps from the same packet (*i.e.*, starting frequencies of preamble chirps) would change across windows at a fixed rate $\Delta f = k_i T_{win}$ that is determined by the chirp slope ($k_i$) and the length of a detection window ($T_{win}$). By contrast, the dechirped frequencies of preamble chirps from different packets will not change in $\Delta f$ across windows (see Fig. 7(c)). Based on this observation, XGate checks frequency changes across consecutive windows against $\Delta f$ to track the preamble of a particular packet. XGate iteratively uses the method to disentangle preamble signals of concurrent packets. As shown in Fig. 7(d), even though the frequencies of packets B and C are overlapping, we can separately track preamble signals for the two packets and recover packet configurations from their separated frequency patterns.

To summarize, XGate can detect all concurrent packets within an Rx spectrum through the following two steps: (1) Iterate through all chirp configurations to scan the Rx spectrum for LoRa preambles and packets' meta-information. Concurrent packets with different chirp slopes are detected separately in different iterations, without interfering with each other due to the orthogonality of their chirp signals. (2) For concurrent packets sharing the same chirp slope, their signal frequencies are detected simultaneously. XGate iterates through all detected frequencies and tracks frequency changes across consecutive detection windows to disentangle colliding preambles, enabling concurrent packet detection and meta-information extraction.

### C. Enhancing Packet Sensitivity

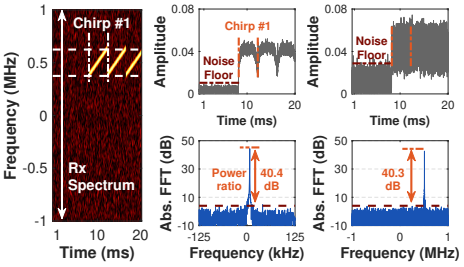
While a short detection window helps avoid ambiguities in packet parameters, it can lead to *SNR losses* when detecting packets with large SFs. This is because short windows spread the signal power of a long chirp across multiple windows. Fig. 8 illustrates two packets received from links with different SNRs: Packet A (SF8) comes from a strong link, while Packet B, using SF12, travels through a weaker link. When XGate uses a short window (*e.g.*, 0.25× the chirp length of Pkt B), it successfully detects Packet A but fails to detect Packet B due to SNR losses caused by the short windows, as shown in Fig. 8(b). We address this issue by enhancing XGate to reliably detect both long-chirp and short-chirp packets.

A practical approach is to use longer windows for packet detection, which can aggregate power from more samples of a chirp signal, thereby enhancing detection sensitivity. As illustrated in Fig. 8(b), the weak frequency of Packet B is detected as the window size increases from 0.25× to 0.5× the chirp length of Packet B, and the detected frequency becomes stronger with a window size of 1×. However, longer windows do not provide additional power gains for Packet A. Instead, when the window size exceeds the chirp length of Packet A, multiple chirps of Packet A are included in the same window and detected as distinct frequencies (see Fig. 8(b)), which can introduce more interference and create ambiguity issues in detecting packet parameters.

How can we achieve high sensitivity while avoiding the drawbacks of long detection windows (*e.g.*, interference, ambiguity)? We observe that although different window sizes yield different frequency results, there are intrinsic relationships among these frequencies. Specifically, for a long-chirp packet (*e.g.*, Pkt B), the detected frequencies remain consistent across different window sizes. In contrast, for Pkt A, multiple short chirps are captured within a long window, and the frequencies from these chirps are equally spaced by a fixed interval equal to the chirp bandwidth, as illustrated in Fig. 8(b).

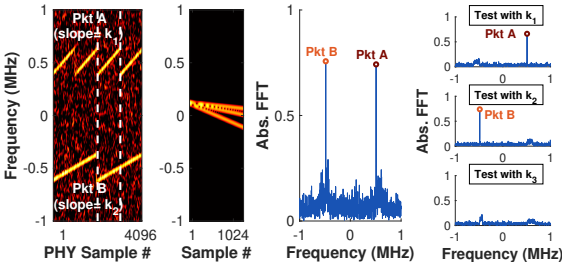
Leveraging these observations, XGate uses the diverse detection results from multiple window sizes for mutual verification, which helps resolve the interference and ambiguity issues associated with long-window detection. For example, after detecting Pkt A with a window size of 0.25×, we can apply a window size of 1× to confirm that those four frequencies indeed belong to Pkt A (see Fig. 8). We then use this additional information to manually remove the frequencies of Pkt A from the raw detection results of the long windows (*i.e.*, 1×) and focus on detecting the frequencies from Pkt B.

In practice, XGate scans the Rx spectrum in multiple rounds to reliably detect packets with varying SNRs and parameter configurations. Initially, XGate detects high-SNR packets us-




(a) Rx signals (b) 250 kHz (c) 2 MHz

Fig. 10. (a) Raw signals in a 2 MHz spectrum; (b,c) Time-domain and frequency-domain views of Chirp #1 under 250 kHz and 2 MHz bands.




(a) (b)

Fig. 11. Speed up packet detection: (a) Detect multiple slopes simultaneously, A and B are detected in one round; (b) Dechirp tests for each slope, A detected with slope $k_1$ and B in $k_2$.

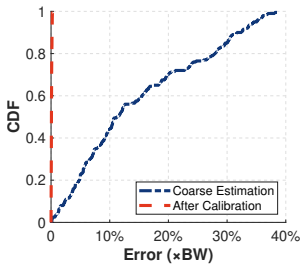



Fig. 12. Estimation errors of packets' central frequencies before and after calibration.

ing short windows and then gradually increases the window sizes to search for weaker packets (*i.e.*, usually in large SFs). To mitigate the impact of high-SNR packets on the reception of weaker packets, XGate utilizes the information from high-SNR packets detected in short-window rounds to facilitate interference cancellation in the detection results of longer windows. Additionally, if some packets cannot be reliably detected due to interference and noise in a single round (*e.g.*, Pkt B with window size 0.5× in Fig. 8), XGate can aggregate the detection results from multiple rounds to verify the results, effectively reducing false positive and false negative errors.

### *D. Massive Packet Reception*

Figure 9 illustrates the general workflow of packet reception in XGate. The RF radio front end receives signals across a wide spectrum and down-converts them to baseband as digitized samples. XGate scans the entire Rx spectrum to detect LoRa packets and determine their parameters, including channel frequencies, bandwidths, and spreading factors. Using the detected parameters, XGate configures software Rx chains to receive packets in parallel. These Rx chains are created and activated on-demand when new packets are detected. A parameterized Rx chain extracts the signals of a packet from the Rx spectrum, which are then passed to a standard LoRa decoder for demodulation and decoding.

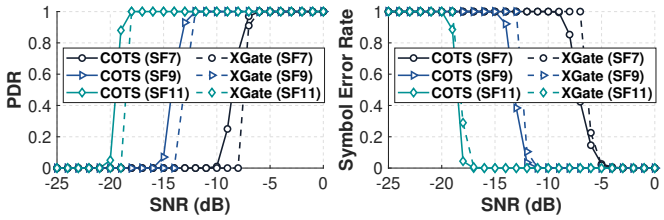
Note that a packet will not be received if it cannot be detected. We address several practical issues that may affect the sensitivity and accuracy of packet detection, which are crucial to the success of XGate.

**Detecting large spectrum vs. small spectrum**. When XGate scans the entire Rx spectrum to detect packets, rather than focusing on a specific LoRa channel, a primary concern is whether the sensitivity of packet detection might be compromised due to the increased noise and interference present in a broader spectrum compared to a single LoRa channel. Fig. 10 illustrates a comparison of signals from the same chirps in a packet received over a 2 MHz spectrum before and after filtering by the target channel (250 kHz). As expected, the time-domain signal with a 2 MHz spectrum exhibits a higher noise floor than the 250 kHz signal. Interestingly, we observe minimal changes in the noise floors in the frequency domain after dechirping both signals. This is because noise is uniformly distributed across the entire spectrum, and while a larger spectrum increases the frequency range, it does not increase the noise strength per frequency bin. Since XGate detects packets in the frequency domain, the detection sensitivity is not significantly affected by the spectrum's bandwidth. However, a large spectrum may experience intensified cross-channel interference, potentially increasing packet detection errors for XGate. To mitigate interference, XGate divides a larger Rx spectrum into multiple smaller chunks (as shown in the *Spectrum Divider* in Fig. 9) and applies packet detectors to each smaller spectrum separately to detect packets.

**Accelerating packet detection**. XGate generally requires iterating through all chirp slopes in multiple rounds to detect LoRa packets and configurations, which can incur high overheads and long delays. In practice, however, due to the random and infrequent natures of LoRaWAN communications (*e.g.*, $\leq$1 % duty cycles with frequency hopping [22]), LoRa packets often appear sparsely in a particular spectrum. It would be inefficient to iteratively detect packets for every chirp slope as many chirp profiles can be absent in the received signals.

To speed up packet detection for sparse traffics, XGate extends a packet detector to simultaneous monitoring for multiple chirp slopes. This can be achieved by extending the dechirp operation in Eq.(2) with a synthetic signal of $m$ down-chirps (*i.e.*, $\sum_{i=1}^{m} C_w^{-1}(k_i, t)$ as shown in Fig. 11(a)), instead of a single down-chirp (*i.e.*, $C_w^{-1}(k_i, t)$ in Eq.(2)). Then, XGate can perform one round of spectrum scanning to detect packets for at most $m$ chirp slopes. We expect to detect positives when a packet in any of the $m$ chirp slopes is present. XGate next recovers the correct chirp slopes of incoming packets using $m$ tests with each testing for one chirp slope. For these test operations, rather than repeating packet detection to detect the presence of a particular chirp slope, we apply a lightweight dechirp operation to the received signals and detect frequency magnitude. We can detect strong frequency peaks if received signals contain a corresponding chirp profile (see Fig. 11(b)). Empirical studies show that we can complete testing for one chirp slope with 2∼3 dechirp operations in general. The tests for $m$ chirp slopes have negligible costs compared to packet detection. In practice, LoRa nodes usually configure with SF7∼SF12 in three BWs (125 kHz, 250 kHz and 500 kHz), which involves 10 chirp slopes. XGate can use 2∼3 rounds of spectrum scanning to complete packet detection for all chirp slopes with the extended method.

**Refining packet configuration**. Recall that XGate detects




(a) Packet detection (b) Packet decoding

Fig. 13. Performance of single packet reception

the frequency patterns of dechirped signals to determine a packet's parameter configurations. The SF and BW of a packet are inferred from the periodicity of frequency changes across windows, which is highly resilient to errors. However, the central frequency of a packet is coarsely estimated as the average of dechirped frequencies, which can be error-prone due to random time offsets between detection windows and chirp signals (*i.e.*, $\Delta t$ in Eq.(1)) that can alter the estimated frequencies. Our measurements indicate that this coarse estimation of the central frequency often deviates from the actual frequency of a packet by 5% to 35% of the signal bandwidth (see Fig. 12). If not corrected, these deviations can negatively impact subsequent packet reception and decoding operations.

Our insight is that although the estimations of a packet's channel frequency may not be entirely accurate, the detected meta-information allows us to roughly identify the frame structure of a LoRa packet (*e.g.*, the preamble, SFD, *etc*). By leveraging the signal structure of a LoRa packet, we can apply existing time-offset correction methods [23], [24] to mitigate the effects of misaligned detection windows. The rationale is that different chirps within the same packet experience the same time offset deviation from the detection windows. However, this time offset has opposite frequency effects on the preamble's up-chirps and the SFD's down-chirps [23], [24]. We can use two coarsely located chirps from the preamble and SFD parts of a packet, denoted by $C_{pre}(t)$ and $C^{-1}_{sfd}(t)$, to cancel out the effects of time offset as below:

$$h_i e^{j2\pi f_i t} C_{pre}(t+\Delta t_i) \cdot h_i e^{j2\pi f_i t} C^{-1}_{sfd}(t+\Delta t_i) = h_i^2 e^{j2\pi(2f_i)t}, \quad (3)$$

where $f_i$ denotes the central frequency of the $i^{th}$ packet, $h_i$ models the impact of wireless channel, and $\Delta t_i$ stands for the time offset between detection windows and chirps. The frequency component from the resulting signals of Eq.(3) gives the accurate central frequency of the packet (*i.e.*, $f_i$).

XGate uses Eq.(3) to refine the detection of channel frequency. The frequency mismatch between up- and down-chirps can be eliminated. As plotted in Fig. 12, the refined results achieve almost $100\,\%$ accuracy. We note that Eq.(3) not only removes the effects of window time offset but also calibrates potential CFOs between sender and receiver radios. After frequency calibrations, a software Rx chain of XGate can correctly filter the interference of unwanted channels and detect precise frame timing from CFO-free signals for reliable packet demodulation and decoding (see Fig. 9).

## IV. Evaluation

### A. Methodology

**Implementation.** We implement XGate on a software-defined radio platform (USRP N210) using the `gr-lora` project [25]. The USRP is used to receive signals from LoRa nodes and forward the received samples to a workstation running XGate for processing. We utilize COTS LoRa nodes equipped with Semtech SX1276 [26] as transmitters, with an Arduino Uno used to configure transmission parameters. Our testbed consists of 40 LoRa nodes and six gateways (*i.e.*, four COTS gateways, two USRP gateways). We conducted experiments on our campus, covering an area of $1.08\,\mathrm{km}\times 1.2\,\mathrm{km}$, which represents a typical urban environment, as shown in Fig. 16(a). The gateways are installed on the roof of a 20-story building. All nodes operate in the $915\,\mathrm{MHz}$ ISM band.

**Experiment Setup.** We conduct extensive experiments to evaluate XGate, aiming to answer the following questions: (1) How many concurrent LoRa transmissions can be supported by XGate? (§IV-B) and (2) How does XGate perform with practical network settings? (§IV-C)

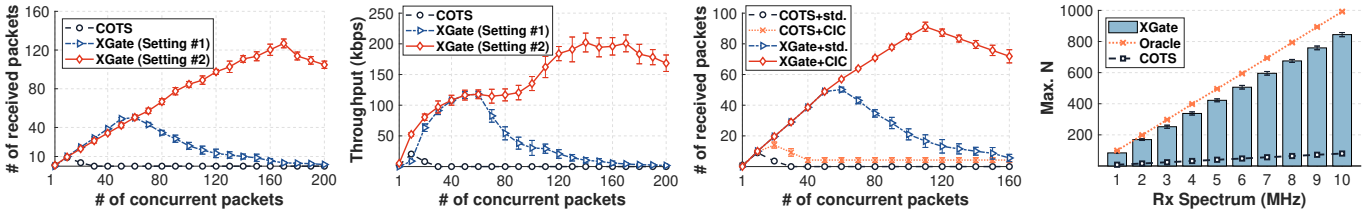
We collect over 10,000 packet traces from more than 200 links within our testbed, encompassing a range of channel conditions (*e.g.*, indoor and outdoor, low and high SNRs). All evaluations involving $<$40 nodes are conducted through real-world experiments in the testbed. To assess the performance of massive concurrent transmissions (*e.g.*, thousands of LoRa nodes), we synthesize traffic based on the collected traces. Specifically, we combine multiple packet traces with random time offsets to emulate real-world network traffic. These synthesized traces are then replayed to evaluate the performance of XGate when more than 40 nodes are involved.

**Baseline.** We compare XGate with two baselines: (1) *COTS gateways* [16] receive LoRa packets with multiple Rx chains concurrently. The Rx chains of COTS gateways are pre-configured; all nodes communicate in fixed channels supported by the Rx chains of COTS gateway. (2) *CIC* [17] adopts collision resolving technique to receive concurrent LoRa transmissions in the same LoRa channel.

### B. Basic Reception Performance

**Packet detection & decoding**. We first evaluate XGate in detecting and receiving packets from LoRa logical channels. We set up one gateway to receive packets from one LoRa node under different SNR conditions. The Rx spectrum of gateway is $1.6\,\mathrm{MHz}$. The LoRa node randomly chooses a channel (*i.e.*, central frequency) for every packet with a default bandwidth of $125\,\mathrm{kHz}$. We analyzed $>$100 packets for each SNR condition. XGate detects and receives a packet from the Rx spectrum without knowing the channel or parameter configuration. For comparison, we use a COTS LoRa gateway, which knows the channels and parameter configurations in advance.

Figure 13 compares the packet detection ratios and decoding errors of XGate with a COTS gateway. We observe that even without the prior knowledge of channel or parameter configuration, XGate achieves comparable performance in packet detection and decoding to that of a COTS gateway which knows all the meta-information in advance. XGate




(a) Concurrent packet reception (b) Throughput (c) Comparison with SOTA (d) Concurrency vs. Rx Bandwidth

Fig. 14. Performance of concurrent reception: (a,b) different numbers of concurrent transmissions in a 1.6 MHz spectrum; (c) comparison with state-of-the-art; (d) concurrency under different spectrum bandwidths.

can reliably detect a packet from unknown logical channels in ultra-low SNRs (*e.g.*, -12.4 dB for SF9 and -18.2 dB for SF11), and use detected parameters to decode incoming packets correctly. Compared to a COTS gateway, the absence of channel information in XGate only leads to 0.3 dB $\sim$ 0.6 dB SNR losses on packet detection and decoding while achieving $>80\,\%$ packet detection ratio and $<20\,\%$ symbol error rate.

**Gains on concurrency**. We next evaluate the performance of XGate in supporting LoRa concurrent transmissions. We compare XGate against a COTS LoRa gateway. We configure both XGate and the COTS gateway to operate within the same Rx spectrum of 1.6 MHz. We control a varying number of LoRa nodes to transmit concurrently on different LoRa channels. The COTS gateway (SX1301) has nine Rx chains that receive at nine frequencies distributed equally in the Rx spectrum with a fixed 125 kHz bandwidth. We adopt two settings of channel partition for XGate. For Setting #1, XGate uses the same channel partition scheme as in the COTS gateway. By varying SF from 7 to 12, 54 LoRa logical channels are supported. Different from COTS gateway detecting and receiving packets on channels known in advance, XGate detects the channel and parameter configurations of packets on-the-fly. For Setting #2, XGate supports all LoRa logical channels in a 1.6 MHz spectrum. LoRa nodes randomly choose packet parameters from those supported by a COTS radio [26]. In particular, we test four bandwidths (*i.e.*, 62.5 kHz, 125 kHz, 250 kHz and 500 kHz) and seven SFs (*i.e.*, 6$\sim$12). The total number of LoRa logical channels is 168.

Figure 14(a) plots the number of received packets when different numbers of nodes transmit concurrently. The number of received packets first increases as more nodes transmit concurrently, then reaches the maximum and starts to drop when the number of concurrent nodes exceeds the number of available channels because of multiple packets colliding in the same channel. As expected, a COTS gateway can receive nine concurrent packets at most. By contrast, XGate (Setting #1) can receive up to 51 concurrent packets, *i.e.*, one packet per logical channel. This number increases to 126 for XGate (Setting #2), *i.e.*, 14$\times$ higher than the COTS gateway. The performance gain in concurrency is mainly because XGate can support more LoRa logical channels.

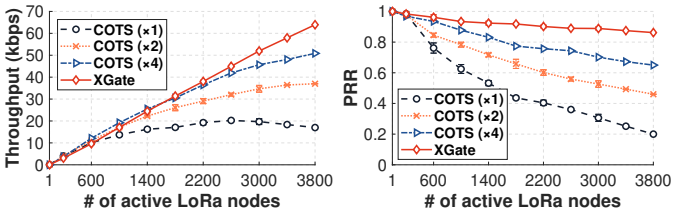
Figure 14(b) compares the aggregate network throughput of XGate with a COTS gateway. The highest throughput of a COTS gateway is 20.4 kbps on average when nine packets are transmitted concurrently. XGate can achieve a maximum throughput of 118 kbps in Setting #1 and 200 kbps in Setting #2, which are 5.8$\times$ and 9.8$\times$ higher than a COTS gateway.

**Comparison with the state-of-the-art.** This experiment compares XGate with existing LoRa concurrent transmission strategies. We consider two state-of-the-art (SOTA) strategies (*i.e.*, COTS gateway [16] and CIC [17]) and evaluate four schemes that combine different gateway paradigms and packet decoders, *i.e.*, COTS gateway with a standard LoRa decoder (*COTS+std.*), COTS gateway with a CIC decoder (*COTS+CIC*), XGate (Setting #1) with a standard LoRa decoder (*XGate+std.*), and XGate (Setting #1) with a CIC decoder (*XGate+CIC*). We control various numbers of LoRa nodes to transmit concurrently on different channels using the same method as in the former experiment. When a COTS gateway is used, only nine concurrent channels (*i.e.*, one channel per Rx chain) are available in a 1.6 MHz spectrum, while XGate supports 54 logical channels in the same spectrum.

Figure 14(c) presents the number of received packets of the four schemes under different numbers of concurrent transmissions. As expected, a COTS gateway with the standard LoRa decoder receives nine concurrent packets at most, *i.e.*, one packet per Rx chain. When more than nine nodes transmit concurrently, collisions will occur on some channels, resulting in packet loss. CIC recovers two packets from collisions on average. It increases the maximum concurrency of a COTS gateway to 15 packets. Unlike CIC , XGate increases concurrency in the dimension of LoRa logical channels. XGate with a standard decoder receives 51 concurrent packets in the same spectrum without collisions, which is 3.4$\times$ higher than the concurrency of the compound SOTA scheme (*i.e.*, CIC's parallel decoding atop multiple Rx chains of a COTS gateway). Moreover, XGate can jointly work with CIC to support up to 91 concurrent transmissions (*i.e.*, 78 % more than XGate with a standard LoRa decoder) as shown in Fig. 14(c).

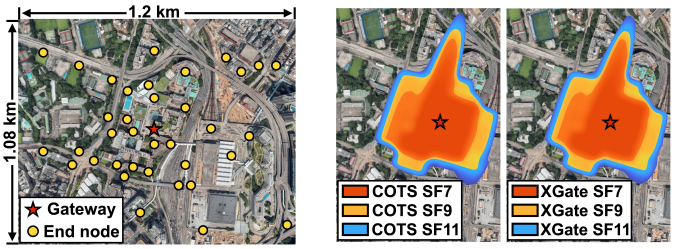
**Scaling with Rx bandwidth**. This experiment evaluates the maximum number of concurrent transmissions supported by XGate under different Rx spectrum settings. We increase the Rx spectrum of a gateway from 1 MHz to 10 MHz and set up various LoRa nodes to transmit concurrently using the same method as XGate (Setting #2). We compare XGate with two benchmarks: (1) COTS gateway that uses dedicated Rx chains to receive pre-configured channels (*i.e.*, eight 125 kHz LoRa channels per 1 MHz); and (2) Oracle method that can divide a given spectrum into the maximum number of logical channels and know the parameters in advance.

We measure the maximum number of concurrent transmissions and present the results in Figure 14(d). The maximum




(a) Throughput (b) Packet Reception Ratio

Fig. 15. Performance of massive packet reception.




(a) Campus-scale testbed (b) Coverage comparison

Fig. 16. (a) Deployment map of testbed (each dot represents multiple surrounding sites); (b) Coverage comparison between XGate and COTS.

concurrency increases with the bandwidth of Rx spectrum for all three strategies. The COTS method increases the slowest, and the Oracle increases the fastest. XGate matches closely with Oracle, *i.e.*, within 85 %. Specifically, when spectrum bandwidth increases to 10 MHz, XGate receives 844 concurrent packets which is 10.5× higher than the COTS gateway method. The results indicate that XGate can support higher concurrency with an increased Rx spectrum.

**Massive reception performance**. This experiment evaluates XGate in supporting massive IoT communications in practical network settings. In particular, we deploy LoRa gateways in our testbed area (1.08 km×1.2 km) to cover up to 3,800 IoT sensors using a 4 MHz ISM spectrum. Each IoT sensor transmits a 20-Byte message every 30 min with a duty cycle ≤1%. We use a trace-driven method to investigate communications for thousands of sensors. Specifically, we measure LoRa packet profiles from more than 200 sites with SNR ranging from −20 dB to 10 dB and use the collected data to synthesize the signals of received packets with randomly selected link profiles. We consider using a single XGate gateway with a 4 MHz Rx spectrum or various COTS gateways with each covering 1 MHz only. Sensors can randomly choose channel frequency (in the 4 MHz spectrum), bandwidth (62.5 kHz, 125 kHz, 250 kHz and 500 kHz) and spreading factor (SF6∼12) for every transmission when XGate is used. We use COTS gateways for benchmarking. LoRa nodes select only from the pre-configured channels to transmit packets with an ALOHA-based MAC.

Figure 15 compares the aggregated throughput and packet reception ratio (PRR) of XGate and COTS gateways under different numbers of active nodes. We see that the throughput of a COTS gateway first increases and then becomes saturated when the number of nodes exceeds 200. As more nodes transmit actively, it increases the chance of collisions and decreases PRRs (Fig. 15(b)). A COTS gateway can only support 400 nodes with PRR>80 %. When four COTS gateways are used, up to 1600 nodes can transmit actively with PRR>80 %. By contrast, XGate can use fewer gateways to serve more active nodes with even higher PRRs because many LoRa logical channels are used to reduce potential collisions. As shown in Fig. 15, a single XGate gateway can support 3,800 active nodes with PRR>85 % in the same 4 MHz spectrum, achieving a throughput close to 64 kbps.

**Gateway coverage**. This experiment compares the coverage of XGate with that of a COTS gateway in an urban environment. To obtain the coverage maps, we divide our testbed area as shown in Figure 16(a) into 40×36 grids. We put LoRa nodes in each grid to send packets at 915 MHz with a default bandwidth of 125 kHz and fixed SFs (*i.e.*, 7, 9, and 11). We measure the SNRs of LoRa packets transmitted from each grid and plot a grid in colors if packets from the grid can be received with PRR>90 %. Figure 16(b) visualizes the coverage area of a COTS gateway and XGate with different spreading factors. We see that XGate achieves almost the same coverage as the COTS gateway. Due to the severe signal blockage in our deployment area, the maximum communication ranges are 286 m, 340 m and 406 m for SF7, SF9 and SF11, respectively. The differences in communication ranges are smaller than 20 m between XGate and a COTS gateway.

### C. Microbenchmarks

**Impact of cross-channel interference**. We conducted experiments to assess the impact of cross-channel interference on XGate's performance. Initially, we examined the effect of frequency gaps between concurrent packets. Two LoRa nodes with strong SNRs transmitted simultaneously using identical parameters (BW125 kHz, SF9), but with varying central frequencies. Our findings in Figure 17(a) indicate that XGate reliably detects all packets across different frequency settings. However, when concurrent transmissions overlap in frequency (frequency gap < BW), some packets encounter decoding errors due to interference. These errors can be partially mitigated using a parallel LoRa decoder, such as CIC. We note that XGate+CIC performs best when the frequency gap is 0, which is exactly the collision problem that CIC targets. The performance drops significantly when the frequency gap increases 25% where the packets partially overlap, causing cross-channel interference that CIC cannot totally handle. As the frequency gap increases, we observe the PRR of XGate+CIC gradually increases due to the decreased interference power. Furthermore, we evaluated XGate under varying intensities of cross-channel interference by increasing the number of concurrent transmissions within a fixed 1 MHz spectrum. XGate consistently detects all packets accurately, but as the number of concurrent transmissions (>8), decoding errors arise due to intensified interference. As shown in Figure 17(b), the packet detection method of XGate exhibits high resilience to interference, primarily affecting packet decoding rather than detection. This resilience is attributed to the fixed signal patterns in the preamble, which are less susceptible to interference compared to the random patterns in payload symbols.

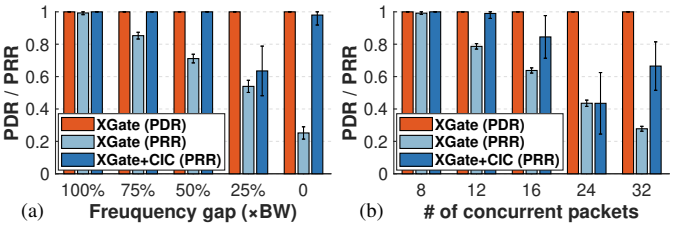
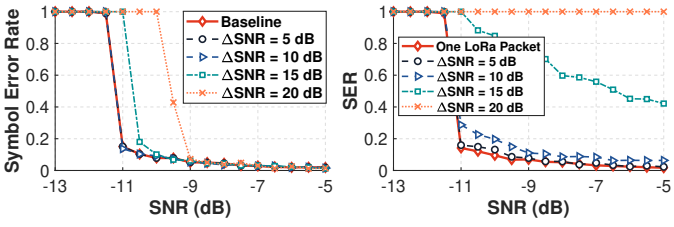



Fig. 17. Impacts of cross-channel interference under (a) two packets with different frequency gaps, and (b) different numbers of concurrent packets.




(a) Near-far in different channels (b) Near-far in same channel

Fig. 19. Near-far SER performance.

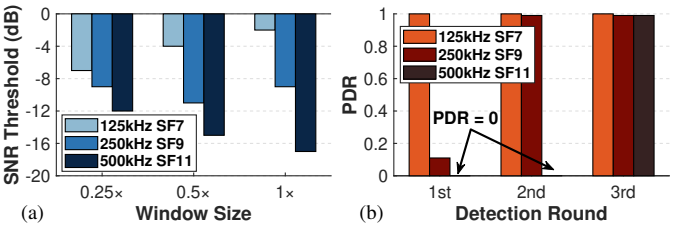



Fig. 18. Impact of window size on (a) single packet and (b) concurrent packet detection.

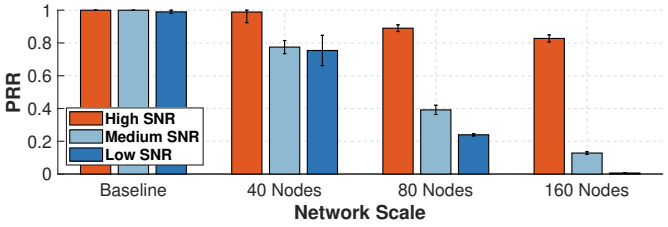



Fig. 20. Near-far effect at different network scales.

**Impact of detection window**. This experiment evaluates how the setting of detection windows in XGate affects the performance of packet detection. We set up three LoRa nodes and a gateway in the testbed. The SNRs of links from the three nodes to the gateway are -2 dB, -10 dB, and -16 dB, respectively. The BW and SF configurations of the three nodes are (125 kHz, SF7), (250 kHz, SF9), and (500 kHz, SF11). We employ three types of detection windows for XGate, *i.e.*, 0.25×, 0.5×, and 1× of the chirp length of (500 kHz, SF11).

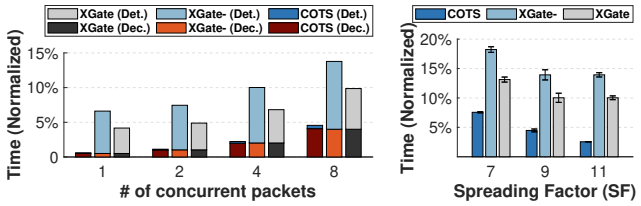
We first examine the packet detection capability of XGate with different detection windows. We control the three nodes to transmit individually with corresponding BW and SF, and change signal SNRs by varying the transmit power of LoRa nodes. We measure the minimum SNRs (*i.e.*, SNR threshold) of packets that XGate can detect reliably. Fig. 18(a) plots the results. We see that XGate generally gives the highest sensitivity when the length of detection windows equals the chirp length of a packet. For instance, the chirp length of a packet in (125 kHz, SF7) is 0.25× of the chirp length of (500 kHz, SF11). XGate, with a detection window of 0.25×, detects a packet of (125 kHz, SF7) at the highest sensitivity (*i.e.*, −7 dB). For the packets of (500 kHz, SF11), the SNR sensitivity increases (*e.g.*, from −12 dB to −17 dB) as detection window enlarges from 0.25× to 1× of the chirp length. Whereas for packets in (250 kHz, SF9), the SNR threshold first decreases and then increases. The optimal sensitivity (*i.e.*, −11 dB) is produced by window size 0.5×.

Next, we evaluate the full packet detection scheme of XGate, which detects packets in multiple rounds using different windows. We control the three nodes to transmit concurrently on three logical channels. XGate is configured to use three types of windows (*i.e.*, 0.25×, 0.5×, and 1×) to detect packets from all possible logical channels in three rounds. Fig. 18(b) plots the packet detection ratios (PDRs) of the three nodes after each round of detection. As expected, the packets of (125 kHz, SF7) are detected with 100 % PDR in the first round, whereas packets from the other two nodes cannot be detected due to poor link qualities. The packets of (250 kHz, SF9), and (500 kHz, SF11) are gradually detected in the second and third rounds, as longer detection windows (*e.g.*, 0.5× and 1×) are adopted.

**Near-far effect**. In practice, concurrent packets may differ in SNRs due to the different distances and heterogeneous link qualities of LoRa nodes (*i.e.*, near-far effects [27]). This experiment investigates the impact of SNR differences between concurrent packets on XGate performance. We set up two LoRa nodes (*i.e.*, one node in higher Tx power and the other in lower power) to transmit concurrently. Both nodes use a default BW 125 kHz. We configure SF7 for the strong node and SF9 for the weak node. In particular, we maintain a fixed SNR difference (*i.e.*, ΔSNR) between the two nodes. We change SNRs of the weak node from −13 dB to −5 dB and measure Symbol Error Rates (SERs) of both nodes under different SNR conditions (SERs of strong packets are not displayed). We also measure SERs from the same packets of the weak node with the strong node absent and use the results for baseline comparison.

First, we transmit packets at different channels. Fig. 19(a) plots the SERs of weak packet reception in various ΔSNR settings. We see that the SER remains the same as the baseline when ΔSNR≤10 dB. It means that XGate can correctly detect and receive weak packets as if no strong packets are present when the SNR difference between concurrent packets is smaller than 10 dB. As the SNR difference increases to 15 dB and 20 dB, though the concurrent packets use orthogonal SFs, a strong packet still increases the background noise power of a weak packet which in turn leads to higher SERs for weak packet reception (see Figure 19(a)).

Next, we transmit packets at the same channel. As shown in Figure 19(b), the SF9 link slightly suffers from interference when ΔSNR is 10 dB, and worsens as ΔSNR increases. we see that the near-far problem within the same channel has a greater influence on decoding performance because spectrum




(a) Detect & decode time (b) Overall time vs. SF

Fig. 21. Time overhead of COTS gateway and XGate.

| Enhancement for LoRaWAN | System Name | Method | Target Scenario |
|---|---|---|---|
| Throughput | HyLink [41] | Parallel CSS | High Throughput Apps |
| | MaLoRaGW [42] | Multi-antenna Gateway | |
| Coverage | XCopy [39] | Combine Retransmissions | Weak Packet Reception |
| | NELoRa [38] | Neural-enhanced Decoding | |
| Concurrency | FTrack [18] | Frequency Tracking | Single-channel Collision |
| | CoLoRa [43] | Peak Ratio Tracking | |
| | CIC [17] | Sub-symbol Cancellation | |
| | Mc-LoRa [44] | Signal Wiper | Cross-channel Collision |
| | **XGate** | Auto-configuration | Scale to Massive Channels |

TABLE I
COMPARISONS BETWEEN XGATE AND RELATED WORKS.

sharing leads to increased interference between two links.

We further conduct evaluations to assess the near-far effect when multiple nodes transmit simultaneously. The tests are being performed on network scales of 40, 80, and 120 nodes, evenly distributed in the testbed area, with nodes connecting to the gateway at SNRs ranging from -5 dB to 5 dB. We categorize the nodes into three SNR regimes: high SNR (2-5 dB), medium SNR (-2-2 dB), and low SNR (-5–2 dB). By controlling all nodes to transmit concurrently, we measure the packet reception performance of nodes in different SNR regimes, as well as their individual performance without interference. Our preliminary results show that when nodes communicate individually, the PRRs for each SNR regime are close to 100%. However, as the number of concurrent transmissions increases, the PRRs of medium- and low-SNR packets experience a significant drop, while high-SNR packets only see a slight decrease. This drop in performance for medium- and low-SNR packets can be attributed to increased background noise caused by concurrent high-SNR packets sharing the same spectrum.

**Real-time performance**. This experiment evaluates the time overheads of XGate. We utilize XGate on a workstation to handle varying numbers of concurrent packets (125 kHz, SF9) and measure the time required for packet detection and decoding. XGate- indicates XGate without detection acceleration that sequentially iterates each chirp slope. To ensure a fair comparison, we implement a software-based COTS gateway on the same platform as XGate. Both the COTS and XGate use the same number of demodulators and decoders for each experimental setup. We normalize the processing time as percentage of the packet's air time. Additionally, we measure the time costs of operations in COTS for benchmarking purposes. As shown in Figure 21(a), the packet decoding costs are similar for both XGate and the COTS. The primary overhead for XGate arises from packet detection. Armed with the detection acceleration technique, the detection time of XGate is reduced by 40%. Notably, the total processing time for eight packets in XGate is less than 10% of a packet's duration. This result demonstrates that XGate can perform both packet detection and decoding operations in real-time.

We further examine the overheads of XGate when receiving packets with different spreading factors. Specifically, we use XGate to receive 8 concurrent LoRa packets, each from a different central frequency. For each test, we configure the 8 concurrent packets with spreading factors SF7, SF9, and SF11, and present the results in Figure 21(b). The results show that the normalized time overhead generally decreases as the spreading factor increases. The longest processing time for XGate is less than 12% of a packet's duration, indicating that both detection and decoding operations can be completed in real-time within the duration of a LoRa packet.

## V. RELATED WORK

**Spectrum Sensing.** Spectrum sensing and channel occupancy detection [28] are crucial for identifying idle channels to facilitate media access and avoid interference. LoRa employs Channel Activity Detection (CAD) to detect packets [21], [23], [29], [30]. The LMAC protocol [29] uses CAD-based carrier-sense multiple access (CSMA) on pre-defined logical channels. LoRadar [21] utilizes CAD results from a narrow band channel to determine channel occupancy across a wider band. Falcon [30] enhances spectrum sensing by transmitting data over weak LoRa links through the detection and interference of ongoing LoRa transmissions. Unlike existing methods that operate on pre-configured channels, XGate can sense all logical channels within the Rx spectrum. Its co-designed packet detector and decoder allow XGate to receive incoming packets without prior knowledge of their specific configurations.

**Scalability Enhancement.** Considerable efforts have been made to enhance the scalability of wireless networks. At PHY layer, parallel decoding allows a gateway to resolve collisions. PCube [24] utilizes phase differences among synchronized antennas to separate collided symbols, while CIC [17] uses sub-symbol features to cancel interference from multi-packet collisions. At the MAC layer, LMAC [29] employs channel sensing to prevent collisions, and CurvingLoRa [31] uses nonlinear chirps for parallel transmissions. LoRa backscatter techniques, such as PLoRa [32] and Saiyan [33], generate additional LoRa links by shifting the frequencies of carrier LoRa packets. In contrast to these approaches, XGate utilizes all logical channels to support extensive IoT communications.

**Coverage extension.** Extending coverage and receiving ultra-weak packets are fundamental challenges for LoRa, and have been the focus of extensive research [30], [34]–[40]. Charm [36] enhances LoRaWAN coverage by leveraging the spatial diversity of multiple gateways. NELoRa [38] employs deep learning techniques to decode LoRa packets in ultra-low SNR conditions. MALoRa [37] improves LoRa packet reception by constructively combining weak signals from a multi-antenna gateway. XGate can be integrated with these methods to enhance network scalability and reliability.

**High throughput LoRaWAN.** Efforts have been made to improve LoRaWAN regarding throughput [41], [45]–[54]. MaLoRaGW [42] implements MU-MIMO for LoRa to improve uplink and downlink throughput. HyLink [41] proposes

concurrent chirp (de)modulation over the same packet to improve throughput. FDLoRa [49] leverages full-duplex gateways to enhance scalable downlink communications. Compared with these works, XGate mainly focuses on scalable uplink communication and can work together with these works to push the throughput limit of LoRaWAN.

## VI. Discussion

**XGate vs. SoTA.** Table I provides a comparison between XGate and related works regarding application scenarios and applied techniques. The quantitative comparison of concurrency between XGate and prior arts is presented in [19]. XGate targets supporting ultra-high concurrency for LoRa. The prior arts on scalability mainly focus on resolving single-channel collision [17], [18] and cross-channel collision [44] within a few logical channels supported by COTS gateway. By contrast, XGate scales LoRa communication to massive available logical channels with auto-configured software Rx chains. XGate is complementary to these existing methods of throughput improvement [41], coverage extension [39], and collision recovery [17], as they can be seamlessly integrated into XGate's architecture.

**Overheads of XGate.** The primary advantage of XGate is its ability to comprehensively cover all LoRa logical channels within the Rx spectrum. Its packet reception can be dynamically configured via software, offering enhanced flexibility and supporting greater concurrency. However, XGate incurs higher computational and energy overheads due to its wider bandwidth reception (*e.g.*, at the MHz level), as well as the detection of concurrent packets and meta-information. The overhead can be reduced if partial information is available to XGate to accelerate the slope iteration operations. It experiences slight SNR losses ($\sim$0.6 dB) due to the absence of packet meta-information. XGate can operate with existing end nodes without requiring modifications, but it needs access to the PHY raw signal to implement new techniques, which is not supported by COTS gateways. We believe that these performance trade-offs, such as increased overheads and minor SNR loss, can be managed by high-end gateways and improved through hardware and software optimization. We plan to enhance the Rx sensitivity of XGate in future work.

**Logical channel selection.** XGate opens up a wide range of channel configurations for LoRa nodes, enabling them to freely select channels within the Rx spectrum without needing to negotiate with a gateway (*i.e.*, come-and-be-served). XGate automatically detects the parameters of incoming packets and dynamically configures software Rx chains to receive them. In our experiments, we evenly distributed LoRa nodes across all logical channels to maximize concurrency. In practical scenarios, nodes can choose channels based on the link states and workloads of the logical channels. We plan to explore logical channel selection strategies and network scheduling methods to further enhance XGate's capabilities in the future.

**Cross-channel Interference.** Recent studies have addressed the issue of cross-channel interference and proposed solutions like signal viper [44] and self-dechirp [55] to mitigate it. While XGate is robust against cross-channel interference during the packet detection stage, it struggles with precise decoding (as shown in Fig. 17) because its basic version uses a standard LoRa decoder. To overcome this limitation, XGate can integrate with previous solutions by replacing the standard LoRa decoders with state-of-the-art decoders, such as Mc-LoRa [44].

**Near-far effect across logical channels.** The near-far effect is a common challenge in wireless communication, especially in LoRa networks that span wide areas with varying SNR conditions. Although LoRa demodulation is designed to resist noise, strong packets can still overpower weaker ones. Experiments shown in Fig. 19 and Fig. 20 demonstrate that the near-far effect significantly affects packet reception across concurrent logical channels. Future work will focus on improving reception robustness across these channels.

## VII. Conclusion

This paper introduces XGate, an innovative gateway paradigm for LoRaWAN that revolutionizes packet detection and reception. Unlike traditional methods, XGate employs a single Rx chain to efficiently scan the entire Rx spectrum, detecting incoming packets across thousands of logical channels. It dynamically configures resources on-the-fly, such as software-controlled Rx chains and packet demodulators, to receive all detected packets. This breakthrough allows for scalable LoRa concurrent transmissions across all available logical channels, effectively overcoming the limitations of massive LoRa communications. XGate paves the way for new possibilities in LoRaWAN communication and networking paradigms, enabling dynamic channel configuration, data-rate adaptation, spectrum sharing, and more.

## References


[1] K. Cui, Q. Yang, L. Shen, Y. Zheng, F. Xiao, and J. Han, "Towards isac-empowered mmwave radars by capturing modulated vibrations," *IEEE Transactions on Mobile Computing*, 2024.

[2] S. Ji, X. Zhang, Y. Zheng, and M. Li, "Construct 3d hand skeleton with commercial wifi," in *ACM SenSys*, 2023, pp. 322–334.

[3] K. Wang, Z. Zhou, and Z. Li, "Latte: Layer algorithm-aware training time estimation for heterogeneous federated learning," in *ACM MobiCom*, 2024, pp. 1470–1484.

[4] Q. Yang and Y. Zheng, "Aquahelper: Underwater sos transmission and detection in swimming pools," in *ACM SenSys*, 2023, pp. 294–307.

[5] J. Cao, J. Chen, C. Lin, Y. Liu, K. Wang, and Z. Li, "Practical gaze tracking on any surface with your phone," *IEEE Transactions on Mobile Computing*, 2024.

[6] K. Wang, J. Cao, Z. Zhou, and Z. Li, "Swapnet: Efficient swapping for dnn inference on edge ai devices beyond the memory budget," *IEEE Transactions on Mobile Computing*, 2024.

[7] K. Cui, L. Shen, Y. Zheng, F. Xiao, and J. Han, "Talk2radar: Talking to mmwave radars via smartphone speaker," in *IEEE INFOCOM*. IEEE, 2024, pp. 2358–2367.

[8] K. Cui, Q. Yang, Y. Zheng, and J. Han, "mmripple: Communicating with mmwave radars through smartphone vibration," in *IEEE/ACM IPSN*, 2023, pp. 149–162.

[9] Q. Yang, K. Cui, and Y. Zheng, "Voshield: Voice liveness detection with sound field dynamics," in *IEEE INFOCOM*. IEEE, 2023, pp. 1–10.

[10] Y. Ren, W. Sun, J. Du, H. Zeng, Y. Dong, M. Zhang, S. Chen, Y. Liu, T. Li, and Z. Cao, "Demeter: Reliable cross-soil lpwan with low-cost signal polarization alignment," in *ACM MobiCom*, 2024, pp. 230–245.

[11] Y. Chen, K. Yang, Z. An, B. Holder, L. Paloutzian, K. M. Bali, and W. Du, "Marlp: Time-series forecasting control for agricultural managed aquifer recharge," in *ACM KDD*, 2024, pp. 4862–4872.

[12] Y. Ren, A. Gamage, L. Liu, M. Li, S. Chen, Y. Dong, and Z. Cao, "Sateriot: High-performance ground-space networking for rural iot," in *ACM MobiCom*, 2024.

[13] P. Kumari, R. Mishra, H. P. Gupta, T. Dutta, and S. K. Das, "An energy efficient smart metering system using edge computing in lora network," *IEEE Transactions on Sustainable Computing*, vol. 7, no. 4, pp. 786–798, 2021.

[14] Y. Cheng, H. Saputra, L. M. Goh, and Y. Wu, "Secure smart metering based on lora technology," in *2018 IEEE 4th International Conference on Identity, Security, and Behavior Analysis (ISBA)*, 2018, pp. 1–8.

[15] L. Shen, Q. Yang, K. Cui, Y. Zheng, X.-Y. Wei, J. Liu, and J. Han, "Fedconv: A learning-on-model paradigm for heterogeneous federated clients," in *ACM MobiSys*, 2024, pp. 398–411.

[16] Semtech. (2022, oct) Sx1301. [Online]. Available: https://www.semtech.com/products/wireless-rf/lora-core/sx1301

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

[18] X. Xia, Y. Zheng, and T. Gu, "Ftrack: Parallel decoding for lora transmissions," in *ACM SenSys*, 2019, pp. 192–204.

[19] S. Yu, X. Xia, N. Hou, Y. Zheng, and T. Gu, "Revolutionizing lora gateway with xgate: Scalable concurrent transmission across massive logical channels," in *ACM MobiCom*, 2024, pp. 482–496.

[20] Semtech. (2023, Jan.) Lora gateway chips and reference designs. [Online]. Available: https://www.semtech.com/products/wireless-rf/lora-core

[21] F. Yu, X. Zheng, L. Liu, and H. Ma, "Loradar: An efficient lora channel occupancy acquirer based on cross-channel scanning," in *IEEE INFOCOM*, 2022, pp. 540–549.

[22] L. Alliance. (2022, Oct.) Lorawan specification. "https://lora-alliance.org/about-lorawan".

[23] N. Hou, X. Xia, and Y. Zheng, "Jamming of lora phy and countermeasure," in *IEEE INFOCOM*. IEEE, 2021, pp. 1–10.

[24] X. Xia, N. Hou, Y. Zheng, and T. Gu, "Pcube: scaling lora concurrent transmissions with reception diversities," in *ACM MobiCom*, 2021, pp. 670–683.

[25] Gr-LoRa GitHub community. (2021, Jul) gr-lora projects. "https://github.com/rpp0/gr-lora".

[26] Semtech. (2022, Dec.) Sx1276. [Online]. Available: https://www.semtech.com/products/wireless-rf/lora-connect/sx1276

[27] M. Hessar, A. Najafi *et al.*, "Netscatter: Enabling large-scale backscatter networks," in *USENIX NSDI*, 2019.

[28] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE communications surveys & tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[29] A. Gamage, J. C. Liando, C. Gu, R. Tan, and M. Li, "Lmac: Efficient carrier-sense multiple access for lora," in *ACM MobiCom*, 2020, pp. 1–13.

[30] S. Tong, Z. Shen, Y. Liu, and J. Wang, "Combating link dynamics for reliable lora connection in urban settings," in *ACM MobiCom*, 2021, pp. 642–655.

[31] C. Li, X. Guo, L. Shangguan, Z. Cao, and K. Jamieson, "CurvingLoRa to boost LoRa network throughput via concurrent transmission," in *USENIX NSDI*, 2022, pp. 879–895.

[32] V. Talla, M. Hessar, B. Kellogg, A. Najafi, J. R. Smith, and S. Gollakota, "Lora backscatter: Enabling the vision of ubiquitous connectivity," *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, vol. 1, no. 3, pp. 1–24, 2017.

[33] X. Guo, L. Shangguan, Y. He, N. Jing, J. Zhang, H. Jiang, and Y. Liu, "Saiyan: Design and implementation of a low-power demodulator for lora backscatter systems," in *Proc. USENIX NSDI*, 2022, pp. 437–451.

[34] H. Yang, Z. Sun, H. Liu, X. Xia, Y. Zhang, T. Gu, G. Hancke, and W. Xu, "Chirpkey: a chirp-level information-based key generation scheme for lora networks via perturbed compressed sensing," in *IEEE INFOCOM*. IEEE, 2023, pp. 1–10.

[35] J. Liu, W. Xu, S. Jha, and W. Hu, "Nephalai: towards lpwan c-ran with physical layer compression," in *ACM MobiCom*, 2020, pp. 1–12.

[36] 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 *ACM/IEEE IPSN*, 2018, pp. 60–71.

[37] N. Hou, X. Xia, and Y. Zheng, "Don't miss weak packets: Boosting lora reception with antenna diversities," *ACM Trans. Sen. Netw.*, vol. 19, no. 2, feb 2023.

[38] 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 *ACM SenSys*, 2021, pp. 56–68.

[39] X. Xia, Q. Chen, N. Hou, Y. Zheng, and M. Li, "Xcopy: Boosting weak links for reliable lora communication," in *ACM MobiCom*, 2023, pp. 1–15.

[40] X. Xia, Y. Zheng, and T. Gu, "Litenap: Downclocking lora reception," *IEEE/ACM Transactions on Networking*, vol. 29, no. 6, pp. 2632–2645, 2021.

[41] X. Xia, Q. Chen, N. Hou, and Y. Zheng, "Hylink: Towards high throughput lpwans with lora compatible communication," in *ACM SenSys*, 2022, pp. 578–591.

[42] H. Pirayesh, S. Zhang, P. K. Sangdeh, and H. Zeng, "Maloragw: Multi-user mimo transmission for lora," in *ACM SenSys*, 2022, pp. 179–192.

[43] S. Tong, Z. Xu, and J. Wang, "CoLoRa: Enabling Multi-Packet Reception in LoRa," in *IEEEINFOCOM*, 2020, pp. 2303–2311.

[44] F. Yu, X. Zheng, L. Liu, and H. Ma, "Enabling concurrency for non-orthogonal lora channels," in *ACM MobiCom*, 2023, pp. 1–15.

[45] N. Hou, X. Xia, and Y. Zheng, "Cloaklora: A covert channel over lora phy," *IEEE/ACM Transactions on Networking*, vol. 31, no. 3, pp. 1159–1172, 2022.

[46] C. Li, Y. Ren, S. Tong, S. I. Siam, M. Zhang, J. Wang, Y. Liu, and Z. Cao, "Chirptransformer: Versatile lora encoding for low-power wide-area iot," in *ACM MobiSys*, 2024, pp. 479–491.

[47] N. Hou, X. Xia, Y. Wang, and Y. Zheng, "One shot for all: Quick and accurate data aggregation for lpwans," *IEEE/ACM Transactions on Networking*, 2024.

[48] A. Gadre, R. Narayanan, A. Luong, A. Rowe, B. Iannucci, and S. Kumar, "Frequency configuration for low-power wide-area networks in a heartbeat." in *USENIX NSDI*, 2020, pp. 339–352.

[49] S. Yu, X. Xia, Z. Zhang, N. Hou, and Y. Zheng, "Fdlora: Tackling downlink-uplink asymmetry with full-duplex lora gateways," in *ACM SenSys*, 2024, pp. 281–294.

[50] J. Jiang, Z. Xu, F. Dang, and J. Wang, "Long-range ambient lora backscatter with parallel decoding," in *ACM MobiCom*, 2021, pp. 684–696.

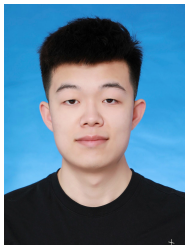
[51] R. Trüb, R. Da Forno, A. Biri, J. Beutel, and L. Thiele, "Lsr: Energy-efficient multi-modulation communication for inhomogeneous wireless iot networks," *ACM Transactions on Internet of Things*, vol. 4, no. 2, pp. 1–36, 2023.

[52] C. Shao and O. Muta, "Tonari: Reactive detection of close physical contact using unlicensed lpwan signals," *ACM Transactions on Internet of Things*, vol. 5, no. 2, pp. 1–30, 2024.

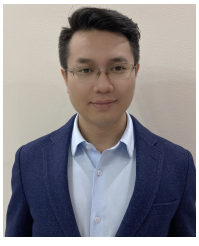
[53] J. Luo, Z. Xu, J. Lin, C. Chen, and R. Xiong, "Ch-mac: Achieving low-latency reliable communication via coding and hopping in lpwan," *ACM Transactions on Internet of Things*, vol. 4, no. 4, pp. 1–25, 2023.

[54] A.-U.-H. Ahmar, E. Aras, T. D. Nguyen, S. Michiels, W. Joosen, and D. Hughes, "Design of a robust mac protocol for lora," *ACM Transactions on Internet of Things*, vol. 4, no. 1, pp. 1–25, 2023.

[55] F. Yu, X. Zheng, Y. Ma, L. Liu, and H. Ma, "Resolve cross-channel interference for lora," in *IEEE ICDCS*, 2024, pp. 1027–1038.

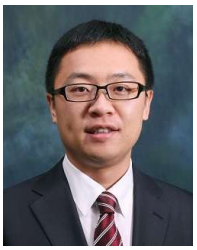



**Shiming Yu** (Student Member, IEEE) received the B.E. degree from University of Electronic Science and Technology of China in 2022. He is currently a PhD candidate in the Department of Computing, The Hong Kong Polytechnic University. His research interests include low-power wide-area networks, wireless networking, mobile computing. He is a Student Member of ACM.



**Xianjin Xia** (Member, IEEE) received the B.S, M.Sc and PhD degrees in Computer Science from Northwestern Polytechnical University, Xi'an, China, in 2010, 2013 and 2018, respectively. He is currently a Research Assistant Professor (RAP) in the Department of Computing, The Hong Kong Polytechnic University. His research interests include low-power wide-area networks, localization, mobile computing. He is a Member of ACM.

**Ningning Hou** (Member, IEEE) received the B.S. degree from Beijing University of Posts and Telecommunication in 2017 and received the PhD degree from Hong Kong Polytechnic University in 2021. She is currently a Lecturer in the School of Computing at Macquarie University. Her research interests include low-power wide-ares networks, physical layer security, IoT, and smart sensing. She is a Member of ACM.



**Yuanqing Zheng** (Senior Member, IEEE) received the B.S. degree in Electrical Engineering and the M.E. degree in Communication and Information System from Beijing Normal University, Beijing, China, in 2007 and 2010 respectively. He received the PhD degree in School of Computer Engineering from Nanyang Technological University in 2014. He is currently an Associate Professor with the Department of Computing in Hong Kong Polytechnic University. His research interests include Wireless Networking and Mobile Computing, Acoustic and RF Sensing, and Internet of Things. He is a member of ACM. He received the Best Paper Award in IEEE INFOCOM 2020 and the Best Paper Candidate in IEEE INFOCOM 2021. He has been serving on the editorial board of IEEE Transactions on Wireless Communications since 2021.

**Tao Gu** (Fellow, IEEE) received the bachelor's degree from the Huazhong University of Science and Technology, the M.Sc. degree from Nanyang Technological University, Singapore, and the Ph.D. degree in computer science from the National University of Singapore. He is currently a Professor with the Department of Computing, Macquarie University, Sydney. His current research interests include Internet of Things, ubiquitous computing, embedded AI, wireless sensor networks, and Big Data analytics. He is a Distinguished Member of ACM.