# Jamming of LoRa PHY and Countermeasure

NINGNING HOU, XIANJIN XIA, and YUANQING ZHENG*, The Hong Kong Polytechnic University, China

LoRaWAN forms a one-hop star topology where LoRa nodes send data via one-hop up-link transmission to a LoRa gateway. If the LoRa gateway can be jammed by attackers, it may not be able to receive any data from any nodes in the network. Our empirical study shows that although LoRa physical layer (PHY) is robust and resilient by design, it is still vulnerable to synchronized jamming chirps. Potential protection solutions (*e.g.*, collision recovery, parallel decoding) may fail to extract LoRa packets if an attacker transmits synchronized jamming chirps at higher power. To protect the LoRa PHY from such attacks, we propose a new protection method that can separate LoRa chirps from jamming chirps by leveraging their difference in power domain. We note that the new protection solution is orthogonal to existing solutions which leverage the chirp misalignment in time domain or the frequency disparity in frequency domain. We conduct experiments with COTS LoRa nodes and software defined radios (SDRs) with varied experiment settings such as different spreading factors, bandwidths, and code rates. Results show that synchronized jamming chirps at high power can jam all previous solutions, while our protection solution can effectively protect LoRa gateways from the jamming attacks.

Additional Key Words and Phrases: LoRa, LPWAN, Jamming attack, Collision recovery

**ACM Reference Format:**
Ningning Hou, Xianjin Xia, and Yuanqing Zheng. 2022. Jamming of LoRa PHY and Countermeasure. 1, 1 (August 2022), 22 pages. https://doi.org/XXXXXXX.XXXXXXX

## 1 INTRODUCTION

Low-power wide-area networks such as LoRaWAN are emerging technologies that enable long-range low-power wireless communication for battery-powered sensor nodes [1–3]. Compared with existing technologies such as Wi-Fi and ZigBee, LoRa has advantages of both long communication range and low power consumption. For example, a battery-powered LoRa node can communicate with a gateway 10 *km* away for years. These features make LoRa suitable for many innovative Internet of Things (IoT) applications [4, 5], such as smart electricity metering, smart homes, supply chain, and health care.

LoRa adopts chirp spread spectrum (CSS) modulation in physical layer (PHY), which is known to be resilient and robust to interference and noise. Benefiting from the long communication range, LoRaWAN forms a one-hop star topology, where a large number of LoRa nodes can send packets via one-hop up-link transmissions to a LoRa gateway, which greatly simplifies the network protocol design and facilitates data collection. In such a star topology, however, if a LoRa gateway is jammed by malicious attackers, the LoRa gateway may not be able to receive LoRa packets from any nodes in the network, leading to single point of failure. Neighbor gateways could help receive the packets in this case, but those gateways can also be under jamming attacks.

We note that wireless jamming has been extensively studied in literature [6] and LoRa jamming has also been attracting attention from both academia and industry recently. Some previous works [7–9] have demonstrated that it is indeed possible to jam LoRa nodes to some extent by emitting various jamming signals, while other measurement studies [1, 10, 11] show that LoRa nodes are inherently resilient and robust to interference and can even support parallel transmissions by resolving collisions. To better understand LoRa demodulation under jamming attacks, we conduct experiments with COTS LoRa nodes and SDRs. Our empirical study indicates that jamming attacks (*e.g.*, random interference and jamming chirps) may not necessarily affect packet receptions at LoRa gateways, meaning that LoRa by design is resilient to a certain type of jamming attacks and intentional interference.

*Yuanqing Zheng is the corresponding author.

Authors' address: Ningning Hou, ningning.hou@connect.polyu.hk; Xianjin Xia, xianjin.xia@polyu.edu.hk; Yuanqing Zheng, yqzheng@polyu.edu.hk, The Hong Kong Polytechnic University, Hong Kong, China.

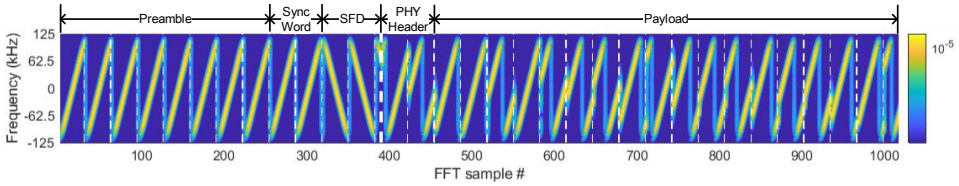



Fig. 1. LoRa packet structure.

By conducting deep analysis, however, we notice that if jamming chirps are well-aligned with LoRa chirps, LoRa gateways cannot extract LoRa chirps from jamming chirps any more. As such, a malicious attacker can send synchronized chirps at high power to jam LoRa chirps, which leads to dramatic performance degradation of LoRa communication. The key rationale behind jamming attack is that if the power of a jamming chirp is higher than that of a legitimate chirp, a gateway will demodulate the jamming chirp with the higher power instead of the legitimate chirp.

Recent collision recovery methods can resolve collisions and potentially protect legitimate chirps from jamming chirps. For example, time domain methods (*e.g.*, FTrack [10], mLoRa[12]) leverage misalignment edges of LoRa symbols. However, if LoRa chirps and jamming chirps are aligned, they cannot be separated in time domain. Frequency domain solutions (*e.g.*, Choir [11]) cannot help either since attackers can send jamming chirps at the same frequency of LoRa chirps. That means even if we assume LoRa gateways can be enhanced with existing collision recovery methods to extract legitimate packets, attackers can still attack the gateways with synchronized jamming chirps.

To further enhance LoRa PHY against synchronized jamming chirps, we propose a new protection method that separates LoRa chirps from jamming chirps by leveraging their difference in signal strength. We note that the new protection method is orthogonal to existing solutions which leverage timing information (*e.g.*, chirp boundary misalignment) or frequency information (*e.g.*, frequency disparity). As such, our protection method can be integrated with existing collision recovery solutions and complement each other.

We implement our jammer and protection method and conduct experiments with COTS LoRa nodes as well as SDRs. Experiment results show that well-synchronized jamming chirps at high transmission power can jam all previous solutions with very high success rates, while our protection method can effectively protect LoRa gateways from all known LoRa jamming attacks including synchronized jamming chirps.

Key contributions of this paper can be summarized as follows.

- We investigate the vulnerability of current LoRaWAN physical layer under jamming attacks. We expose the risk of LoRa gateways under the attack of synchronized jamming chirps, which could lead to single point of failure in LoRaWAN.
- We propose and implement a new collision recovery method as a countermeasure against the attack of synchronized jamming chirps by leveraging the difference in signal strength of jamming chirps and LoRa chirps.
- We conduct comprehensive experiments with COTS LoRa nodes as well as SDRs under various experiment settings. Experimental results demonstrate the effectiveness of our synchronized jamming methods and anti-jamming countermeasure.

This paper extends our previous work [13] mainly in the following four aspects. First, we study new types of jamming attacks (*e.g.*, consecutive SFDs) and evaluate their impacts on LoRa packet reception. Second, we propose an enhanced countermeasure combining RSS and channel phase information to protect gateways from synchronized jamming attacks. We also extend the primary countermeasure with rationales, hypotheses, algorithms, and implementation details. Third, we add a comprehensive literature review to discuss anti-jamming techniques. Finally, we provide more experiments to evaluate the performance of the jammer as well as two versions of countermeasures.

## 2 BACKGROUND AND SYSTEM MODEL

### 2.1 LoRa Packet Detection

LoRa adopts Chirp Spread Spectrum (CSS) modulation which modulates data in a chirp's initial frequency. Fig. 1 shows PHY samples of a LoRa packet collected with SDRs. There are several critical steps in receiving a LoRa packet. As

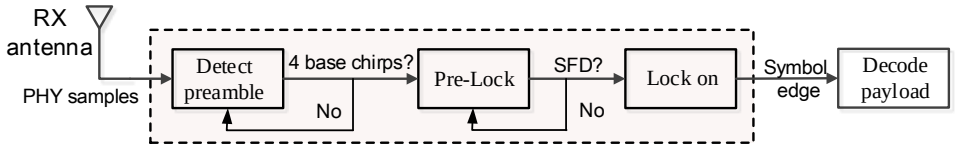



Fig. 2. Locking process at LoRa receiver.

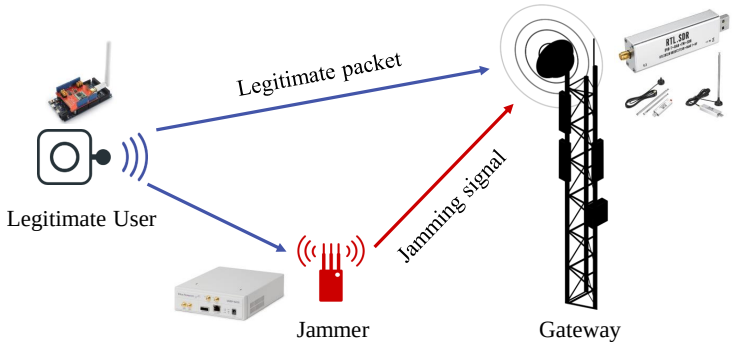



Fig. 3. Attack model. A legitimate user (LoRa node) sends legitimate LoRa packets to a gateway. A malicious jammer sniffs incoming LoRa packets and generates jamming radio signals accordingly. The gateway aims to receive legitimate packets.

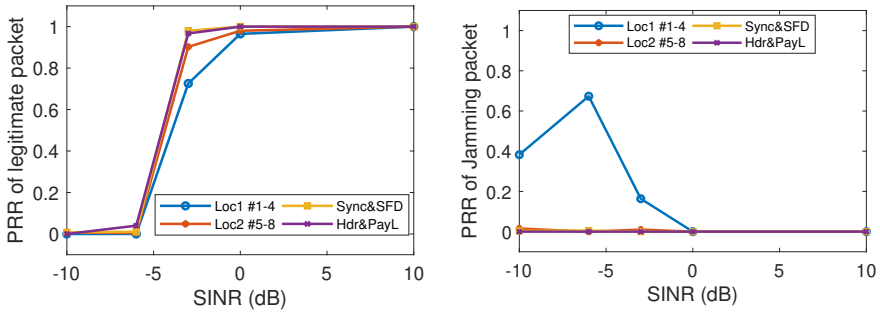
illustrated in Fig. 2, a LoRa receiver first detects arrivals of LoRa packets by detecting a preamble of more than four up-chirps. If more than four consecutive peaks appear in the correlation results, a receiver interprets that a LoRa packet is coming. After successful preamble detection, a LoRa receiver needs to accurately detect an SFD and determine the chirp boundaries of PHY header and payload. To this end, a LoRa receiver multiplies incoming PHY samples with an up-chirp and monitors continuous frequency for 2.25 chirp duration to determine the chirp boundary of the first chirp in PHY header and payload. A LoRa receiver can demodulate [10] the payload chirps and decode the incoming packet after locking on the chirp boundaries.

### 2.2 System Model and Assumptions

Fig. 3 illustrates the jamming model, which consists of an end user (which sends LoRa packets), a LoRa gateway (which receives LoRa packets), and a malicious jammer (which aims to jam LoRa communication).

We employ a commodity LoRa node [14] to transmit packets to a gateway as a legitimate user. For a legitimate user, we assume that it remains static and the transmission power of each packet remains stable. For a LoRa gateway, we assume that it is equipped with the latest parallel decoding algorithms (*e.g.*, FTrack, Choir) which can protect a gateway against some jamming attacks. We note that a LoRa gateway can use low-cost receive-only SDR (*e.g.*, RTL-SDR dongle) since it only needs to receive uplink LoRa packets rather than transmit LoRa packets to end users. Specifically, the communication going from a legitimate user to a gateway is called uplink; The communication going from a gateway to a legitimate user (*i.e.*, LoRa node) is called downlink. For downlink packet, the gateway can use COTS LoRa modules for transmission.

We assume that a jammer is equipped with SDR (*e.g.*, USRP N210) for sniffing incoming LoRa packets and generating jamming radio signals accordingly. The jamming radio can be random Gaussian noise or LoRa signals. In LoRaWAN, LoRa nodes typically adopt low duty cycle mode (*e.g.*, 1% duty cycle). Duty cycle is the fraction of time when a radio is in active states (*e.g.*, transmitting and receiving). As such, if a jammer constantly emits jamming signals at a high transmission power, the jammer can be easily detected and located. Therefore, we consider a jammer that adopts reactive jamming where the jammer stays quiet when the channel is idle, and starts emitting jamming signals when it detects on-going LoRa communication to selectively jam the LoRa communication. The objective of a jammer is to




(a) (b)

Fig. 4. Jamming packets collide with different parts of LoRa packets under different SINRs: (a) Packet Reception Rate of legitimate packets and (b) Packet Reception Rate of jamming packets.

jam the communication between LoRa nodes and a LoRa gateway. We assume that the jammer aims to jam a specific gateway rather than all gateways in a network.

On the other hand, we design and implement a countermeasure to protect LoRa communication by enhancing a LoRa gateway against jammer. Ideally, the countermeasure should not require any modification to LoRa nodes to support a large number of already deployed COTS LoRa nodes.

## 3 EMPIRICAL STUDY OF LORA JAMMING

LoRa jamming has been attracting wide attention due to the potential risk of single-point failure under jamming attacks. Previous works [7–9] have demonstrated that it is indeed possible to jam LoRa nodes to some extent by emitting various jamming signals, while other measurement studies [1, 10, 11] show that LoRa nodes are inherently resilient and robust to interference and can even support parallel transmissions by resolving collisions. In this section, we conduct empirical studies to evaluate the impact of a variety of prior jamming attacks on LoRa communication.

### 3.1 Prior Jamming Attacks and Empirical Study

*3.1.1 Jamming LoRa with Gaussian Noise.* Gaussian noise has been commonly used to jam wireless communication systems. However, LoRa is inherently robust to Gaussian noise. The rationale behind is that the de-chirp operation can concentrate the energy of a chirp to a single FFT bin while the power of Caussian noise is evenly distributed in all FFT bins. In principle, a jammer can also improve jamming performance by increasing transmission power. However, jamming brutally with a higher power is impractical. A jammer's transmission power may exceed the maximum transmission power restricted by regulation, resulting in easy exposure of the jammer.

*3.1.2 Jamming LoRa with Chirps.* Recent work [7] proposes to jam LoRa nodes with LoRa packets and generate collisions for legitimate LoRa communication. In this work, a jammer adopts the maximum transmission, while a legitimate LoRa node adopts a lower one to reduce power consumption. We carry out the following experiment to understand the performance of chirp-chirp jamming. Specifically, we evaluate the impact of jamming chirps on LoRa chirps when they collide. A jamming LoRa packet is in the same packet structure as a legitimate LoRa packet as illustrated in Fig. 1. In this experiment, a legitimate transmitter and a jammer are configured to use the same SF and bandwidth as a LoRa gateway supports parallel transmissions of LoRa packets with different parameter settings [1]. After setting the same parameters (*e.g.*, spreading factor, bandwidth, and central frequency), we vary the transmission power of a jammer and evaluate the impact of jamming chirps under different SINR.

Based on the reception process (Section 2.1), we investigate four scenarios where jamming chirps collide with different parts of LoRa packets: 1) collision with the first four base chirps; 2) collision with the last four base chirps; 3) collision with sync word and SFD; and 4) collision with PHY header and payload. We report the PRRs of legitimate packet and jamming packet in Fig. 4 (a) and (b), respectively. Several key observations are as follows.

First, the power of jamming packets needs to be orders of magnitude higher than that of a legitimate LoRa packet (*e.g.*, SINR $\leq -3\ dB$) to jam it. A gateway can receive legitimate packets with a high PRR (*e.g.*, $\geq 96.5\%$) if the jamming signal has comparable power (*e.g.*, SINR $\geq 0\ dB$) with the legitimate signal.

Second, a LoRa receiver is not likely to receive a late-comming jamming packet. This phenomenon is because a LoRa receiver tends to detect and lock on the preamble and SFD of an early-arrival packet. In our scenario, a legitimate packet arrives earlier than a jamming packet. Yet, we do observe the capture effect where a jamming packet colliding at the first four base chirps of a legitimate packet with stronger signal strength is selected and demodulated (*e.g.*, SINR $\leq -3\ dB$).

Third, the impact of collisions at PHY header and payload seems weaker than that of collisions at preamble. Referring to Fig. 5, let us see how collisions at PHY header and payload part would influence the demodulation of legitimate chirps in demodulation windows. Suppose a legitimate chirp (①) collides with a jamming chirp (② and ③) as illustrated in the figure. Due to collision in the demodulation window, the demodulation results will have three spikes. Since jamming chirps are misaligned with legitimate chirps, the power of a jamming chirp will be divided into two adjacent demodulation windows and their corresponding spikes would be lower than that of a legitimate chirp. As such, LoRa nodes can tolerate collisions with jamming chirps at PHY header and payload with comparable or even slightly stronger signal strength. However, if jamming chirps and legitimate chirps are well aligned (*e.g.*, $< 10\%$ misalignment), spikes of jamming chirps within demodulation windows could become higher than those of legitimate chirps. In this case, legitimate packets will be jammed.

### 3.2 Anti-jamming techniques in other wireless networks

Previous countermeasures against jamming attacks in wireless networks have been studied in [6]. In general, frequency and channel hopping is the most commonly used countermeasure. However, as a LoRa packet has a quite long air-time, a jammer can easily track the transmitted LoRa packet during its long transmission time. Similarly, packet fragmentation also fails as LoRa chirps are long enough to be intercepted. Besides, frame masking [15], where a transmitter and a receiver agree on a secret pseudo-random sequence for the SFD in each packet, is proposed to protect packets from being detected by a jammer. However, as introduced in Section 2, a receiver needs to use SFD to lockon a LoRa packet and extract symbol edges. As such, this method cannot be applied to LoRa in practice. Redundant encoding is another commonly used countermeasure to improve the resilience of packets against jamming attacks. In fact, LoRa has already exploited certain levels of redundancy by configuring a *Code Rate (CR)* parameter. The code rates of LoRa control the ratio of actual data to forward-error-correcting capability that is added to the payload.

Another kind of anti-jamming technique exploits special antenna designs for spatial filtering (*e.g.*, directional antenna, antenna array, *etc.*) [16]. However, such methods are not suitable for LoRa. For example, directional antennas could help protect LoRa gateways from jamming attacks to some extent, but they may also cause packet loss, which is not desirable for LoRa communication. Since LoRa nodes typically communicate with gateways miles away, if we use directional antennas, any blockage of the line of sight path will lead to packet loss. Auxiliary antennas and self-adaption array antennas [17] are used in radar systems to cancel jamming signals. Such multiple antenna technologies can be helpful to separating legitimate LoRa signals from jamming signals. For example, spatial filtering is widely used in GPS [18] as an effective anti-jamming technique. However, these technologies require special hardware (*e.g.*, antennas, radio chains) and incur extra deployment costs, which is economically impractical for LoRaWANs.

### 3.3 Prior Collision Recovery Methods as Countermeasures

We can draw strength from recent advances in LoRa collision recovery and parallel transmissions to protect LoRa communication against jamming attacks.

For example, LoRa collision recovery schemes (*e.g.*, FTrack [10], mLoRa [12]) can resolve collisions of multiple LoRa nodes as long as their chirp boundaries are misaligned in time domain. FTrack detects the continuity of one chirp within a demodulation window to recover collisions. Referring to Fig. 5, we see the frequency of a legitimate chirp continuously increases while the frequency of jamming chirps are not continuous within a demodulation window due to chirp boundary misalignment. If jamming chirps and legitimate chirps are well-aligned in time domain, the heights of FFT spikes of jamming chirps and legitimate chirps will be very close to each other. In this case, if jamming chirps are slightly stronger than legitimate chirps, those collision recovery schemes will fail to resolve collisions.

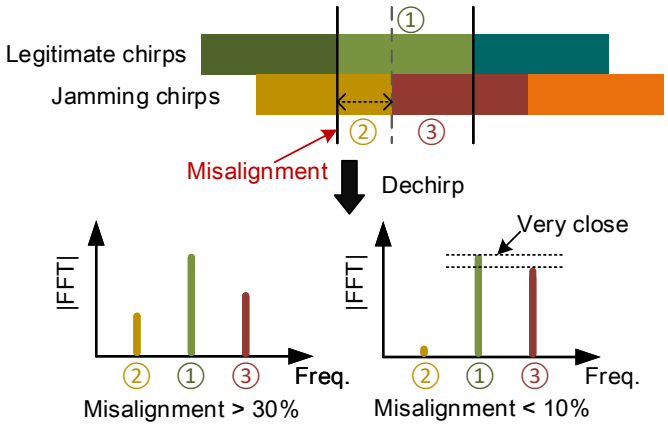



Fig. 5. Demodulation example: Chirps misaligned with a demodulation window will have part of its power split out.

Frequency domain collision recovery schemes (*e.g.*, Choir [11]) separate LoRa collisions by leveraging the frequency differences of colliding nodes due to their hardware imperfection. For example, Choir notices that the fractional part of initial frequencies of different LoRa nodes are unique, which can be used as physical layer fingerprints. As such, Choir can group different chirps according to fractional parts and thereby separate colliding LoRa chirps. If the frequencies of jamming chirps are synchronized with those of legitimate chirps (*e.g.*, emitting jamming chirps with the same fractional part of initial frequencies), those collision recovery schemes cannot separate legitimate chirps from jamming chirps.

In summary, prior collision recovery methods cannot separate legitimate LoRa chirps from jamming chirps if the jamming chirps are aligned with the legitimate chirps in time domain and frequency domain. In this case, if the power of a jamming chirp is higher than that of a legitimate chirp, a LoRa receiver will demodulate jamming chirps within demodulation windows rather than legitimate chirps.

## 4 DEFEATING PRIOR COUNTERMEASURES

### 4.1 Necessary Conditions of Jamming against Prior Countermeasures

To attack a legitimate LoRa node, an attacker needs to emit jamming chirps that satisfy the following three conditions.

*C-1: Jamming chirps should be well-aligned with legitimate LoRa chirps in time domain.*

*C-2: Jamming chirps should mimic legitimate LoRa chirps in frequency domain (e.g., central frequency).*

*C-3: Jamming chirps should have a higher power than legitimate LoRa chirps at a LoRa receiver.*

We note *C2* (*i.e.*, frequency condition) and *C3* (*i.e.*, power condition) are relatively easy to satisfy. For example, a jammer can measure the frequency of a legitimate preamble and extract the fractional part of frequency. After that, the jammer can emit jamming chirps with the same fractional part frequency, which can defeat frequency domain collision recovery scheme (*e.g.*, Choir [11]). To increase the power of jamming chirps at a receiver, a jammer can increase transmission power and get closer to the LoRa receiver.

However, *C1* (*i.e.*, timing condition) can be a bit challenging to satisfy because of signal processing delay caused by SDRs, different communication distance between a LoRa node and a LoRa receiver, *etc.* As such, jamming chirps may not be well-aligned with legitimate chirps in time domain. In this case, the power of jamming chirps will be divided into two adjacent demodulation windows. Moreover, the aforementioned time domain collision recovery schemes can separate legitimate chirps from misaligned jamming chirps.

### 4.2 Jamming with Synchronized Chirps

We illustrate a basic jamming workflow in Fig. 6. A LoRa jammer hears LoRa packets over the air. Upon detecting a valid LoRa preamble, it will attempt to lock on the packet by extracting synchronization information. After that, it checks whether the packet is transmitted by a targeted node. If so, the jammer then emits synchronized chirps to jam

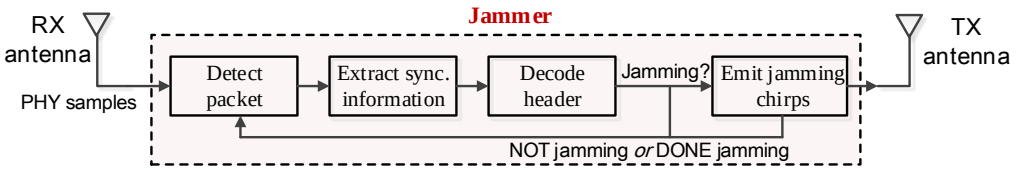



Fig. 6. The general workflow of LoRa jammer.

the legitimate packet. Specifically, to launch an effective jamming with well-synchronized chirps, a jammer needs to take all time/frequency offsets (*i.e.*, jamming conditions) into account and carefully compensate them before sending jamming chirps in real time. We then present several key steps to generate synchronized jamming chirps.

*4.2.1 Accounting for propagation delay.* Basically, jamming chirps are required to closely align with the chirps of a legitimate packet when received at a gateway. The communication distance between a jammer and a gateway and the corresponding propagation delay affects the arrival time of jamming chirps at a gateway. We notice that as LoRa typically adopts narrow bandwidths (*i.e.*, $\leq 500\ kHz$), the sampling interval of a LoRa receiver is relatively large (*e.g.*, $> 2\ \mu s$). Signals arrived within $2\ \mu s$ (which corresponds to a communication distance of 600 $m$) are aligned to the same PHY sample. In practice, a jammer can emit jamming chirps within 600 $m$ away from a gateway to mitigate the influence of propagation delay.

*4.2.2 Compensating carrier frequency offset (CFO).* In order to align jamming chirps with legitimate chirps, a jammer needs to detect chirp boundaries from the preamble of the targeted chirp. Intuitively, this can be done by correlating the received preamble with a locally generated base-chirp. However, the detected edges may not correspond to the correct chirp edges due to carrier frequency offset between the legitimate node and the jammer. As a result, the frequency offset translates into corresponding time offset for chirp signals [10, 19]. To be specific, let $\Delta f_{cfo}$ denotes the CFO. A received preamble chirp can be represented as

$$R_{pre}(t) = h \cdot e^{-j\Delta f_{cfo} t} \cdot C(t) \tag{1}$$

where $C(t)$ denotes a base up-chirp of preamble transmitted by a legitimate node, and $h$ is the channel between the node and a jammer. If we directly correlate $R_{pre}(t)$ with a local base chirp $C(t)$, the detected chirp edge would be $\Delta t = \frac{2^{SF}}{BW^2}\Delta f_{cfo}$ away from the real edge, as illustrated in Fig. 7(a). According to our measurements, this edge offset $\Delta t$ can be as large as ten samples in practice. As such, a jammer must compensate the timing offset caused by CFO and align jamming chirps to correct edges.

Firstly, a jammer needs to estimate CFO from the received signal. We exploit SFD that comes after preamble (see Fig. 1) for CFO estimation. In particular, a received SFD chirp can be represented as:

$$R_{sfd}(t) = h \cdot e^{-j\Delta f_{cfo} t} \cdot C^{-1}(t) \tag{2}$$

By multiplying Eq. (1) with Eq. (2), we obtain

$$R_{pre}(t) \cdot R_{sfd}(t) = h^2 \cdot e^{-j2\Delta f_{cfo} t} \tag{3}$$

We perform FFT (Fast Fourier Transform) on Eq. (3) and the resulting FFT peak indicates the value of $\Delta f_{cfo}$. We use $\Delta f_{cfo}$ to compute the corresponding chirp edge offset $\Delta t = \frac{2^{SF}}{BW^2}\Delta f_{cfo}$, which is finally used to infer the correct chirp edge from detected edges.

As we may detect incorrect chirp boundaries from the received preamble due to CFO, one may wonder how to extract the correct preamble chirp and SFD chirp for CFO estimation. As a matter of fact, we first perform correlation detection on the received preamble to coarsely detect the boundary timing of chirps with a time offset, as illustrated by red dashed lines in Fig. 7. We then use the coarsely detected timing to identify SFD chirps. Note that the extracted preamble base-chirp and SFD down-chirp have the same offset (*i.e.*, $\Delta t$) with their real edge timing, as shown in Fig. 7. Therefore, the extracted chirps in Eq. (3) for CFO estimation are actually $R_{pre}(t - \Delta t)$ and $R_{sfd}(t - \Delta t)$, rather than the ideal $R_{pre}(t)$ and $R_{sfd}(t)$. As the edge time offset ($\Delta t$) translates into frequency offset $\Delta f_{edge}$ for the up-chirp and an

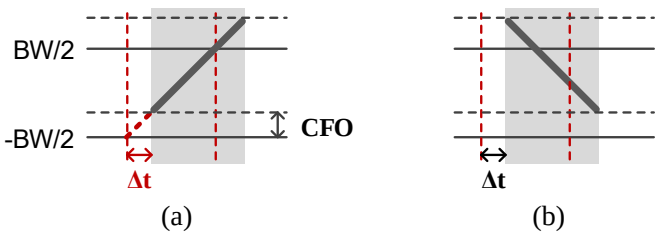



Fig. 7. CFO affects edge detection: (a)Detected edge vs. real edge of base up-chirp in preamble. (b)Extracted SFD down-chirp with edge offset $\Delta t$.

opposite frequency $-\Delta f_{edge}$ for the down-chirp, we have $R_{pre}(t-\Delta t)\cdot R_{sfd}(t-\Delta t) = R_{pre}(t)\cdot R_{sfd}(t)$. In summary, the above CFO estimation method (*i.e.*, Eq. (3)) still holds with the time offset in preamble and SFD detection.

*4.2.3 Compensating hardware and software delay.* A jammer also needs to process received signal and react in real time. This imposes a strict constraint on processing latency (termed *jamming delay*). We use an SDR (*i.e.*, USRP N210) as hardware and use an open-source GNU Radio (GR) as software to perform jamming on-line. In particular, we list the main contributors of jamming delay as follows.

- Data transfer: The delay of data transfers between different components, *e.g.*, from USRP Rx buffer to data processing blocks as well as from blocks to USRP Tx when emitting jamming chirps.
- Scheduling: The latency of OS (*i.e.*, operating system) and GR scheduling.
- Signal processing: The latency of signal processing including preamble detection, packet decoding, synchronization of jamming chirps, *etc.*

As signal processing is generally performed on PCs with powerful CPUs, the processing latency is relatively short (*e.g.*, tens of $\mu$s on our Intel i5 PC). In comparison, the air time of LoRa packet is of $2 \sim 3$ orders of magnitude longer. For instance, the transmission time of a typical LoRa chirp with $SF = 8, BW = 250$ kHz is about 1 $ms$ (*i.e.*, 100× longer than signal processing). Theoretically, this would leave a sufficient amount of time for a jammer to finish signal processing and generate jamming chirps in real time.

On the other hand, we observed that GR scheduling and data transfers exhibit time uncertainty in practice. The latency varies randomly from 100 $\mu$s to 10,000 $\mu$s in our measurements. We configure the GR scheduler with a Single-Thread-Scheduler mode (STS) to reduce the processing latency and time variation. We also configure the buffer size of inter-block data transfer to fit the size of LoRa chirps. The end-to-end jamming latency hence becomes rather stable (*e.g.*, 500 $\mu$s in our setting), which can be measured and compensated before sending jamming chirps.

In order to align a jamming chirp with a legitimate chirp, a jammer needs to infer which sample is currently transmitting in the air (*i.e.*, the front wave of a legitimate packet). To this end, the jammer continuously receives samples of a legitimate packet using USRP, which buffers the received samples and reports them when the buffer is full. In practice, the number of reported samples in every buffer and the corresponding timestamp can vary due to the uncertainly in GR scheduling. To address this problem, the jammer can estimate the current transmitting sample in the air with the latest received buffer size and its timestamp. By further counting in the processing latency, the jammer can determine the time compensation for precise alignment of jamming chirps with legitimate chirps.

### 4.3 Jamming with Identical Consecutive Chirps

The synchronized jamming approach satisfies all conditions listed in Section 4.1. However, the above jamming approach requires careful calibration and strict timing requirement to align jamming chirps with legitimate chirps. Next, we demonstrate that it is possible to jam in a lightweight manner without strict synchronization (*e.g.*, delay compensation).

If we perform jamming without synchronization, jamming chirps are likely to misalign with chirps of a legitimate packet. Suppose a gateway uses a time domain collision recovery scheme to protect legitimate packets from jamming attacks. Let us consider a demodulation window that is aligned with a legitimate chirp but not jamming chirps. As illustrated in Fig. 8(a), since the demodulation window spans across two adjacent jamming chirps, jamming signals

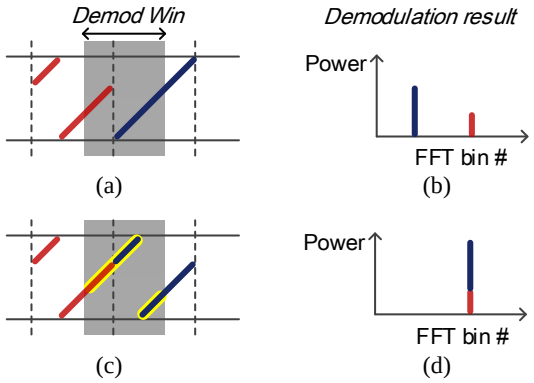



Fig. 8. Jamming without synchronization: (a-b) Non-identical jamming chirps and demodulation result vs. (c-d) Identical jamming chirps and the demodulation result. When consecutive jamming chirps are identical, the samples from adjacent chirps form a complete chirp in the demodulation window which well-aligns with legitimate chirp.

within this demodulation window would experience a sudden change in frequency at chirp boundary. As a result, after demodulation, there will be two FFT spikes at different FFT bins (Fig. 8(b)).

However, if the two adjacent jamming chirps are the same, their frequency would experience no sudden change at the jamming chirp boundary (see Figure 8(c)). As a result, both the jamming chirp and the legitimate chirp exhibit frequency continuity within the demodulation window, meaning that the power of consecutive jamming chirps will concentrate in the demodulation window, as if one jamming chirp is well-aligned with the window, as illustrated in Fig. 8(c) and (d). As such, a jammer can emit the same consecutive chirps to defeat existing countermeasures without synchronizing to legitimate chirps.

However, COTS LoRa radio interleaves the payload data to avoid the occurrence of consecutive identical symbols in PHY layer. Although we can observe two identical chirps in practice, we rarely observe more than three identical symbols appearing successively in the payload of packets transmitted by COTS LoRa nodes. As such, a jammer can emit two consecutive chirps with the same initial frequency as jamming chirps.

We note that the consecutive chirp pattern still differs from the random chirp pattern of a legitimate packet payload. Existing time domain collision recovery schemes can be adapted to discern a consecutive jamming attack by detecting chirps' consecutive patterns. As a result, the consecutive jamming approach may not be as effective as the synchronized jamming approach against existing countermeasures.

### 4.4 Jamming with Consecutive Down-chirps

Another lightweight jamming method without strict synchronization is to jam with consecutive down-chirps. As introduced in Section 2, a LoRa receiver needs to first detect 4 consecutive preamble chirps to pre-lock a packet and continuously listen to an SFD (*i.e.*, 2.25 down-chirps) to further lock on the packet. A LoRa receiver relies on the SFD down-chirps to determine the frame timing of a packet. Based on this fact, a jammer can mimic an SFD by sending consecutive down-chirps to block or interfere the lock-on process of a legitimate packet. If jamming down-chirps arrive in prior of the SFD of a legitimate packet, a normal receiver may lock on the false SFD or fail to lock on any SFD. As a result, a receiver cannot correctly detect the frame timing of a legitimate packet, leading to PHY-layer errors for the demodulation of legitimate packet.

We carry out an experiment to verify the effectiveness of the above jamming method. In this experiment, a jammer transmits consecutive down-chirps immediately after detecting a LoRa packet (*e.g.*, 4 consecutive preamble chirps). Generally, a LoRa receiver pre-locks a packet upon detecting 4 consecutive preamble chirps. It completes locking on the packet by detecting an SFD (*i.e.*, 2.25 down-chirps). Fig. 9 displays the spectrum of a legitimate LoRa packet jammed by consecutive down-chirps. As we can see, the jamming down-chirps collide with identical up-chirps, meaning that the collision happens at the preamble and amming down-chirps arrive earlier than the legitimate SFD. In this case, a

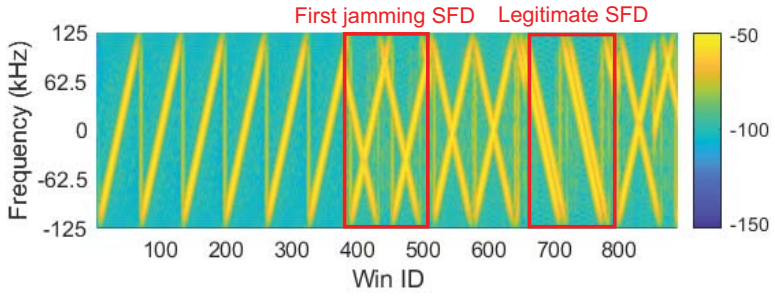
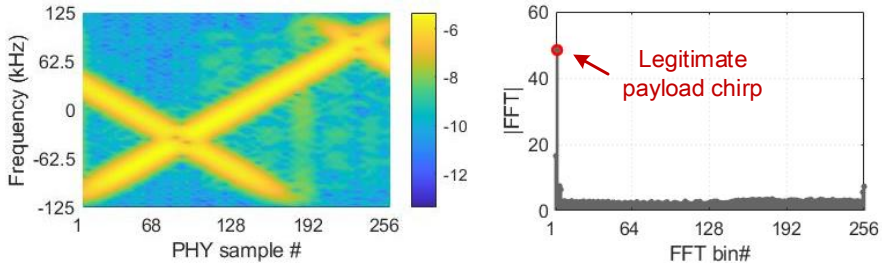



Fig. 9. Jamming with consecutive down-chirps at preamble part.




(a) (b)

Fig. 10. Jamming with down-chirps does not influence the payload demodulation. (a) Spectrum of one payload chirp jammed by a down-chirp. (b) FFT magnitude after dechirping. The legitimate chirp still achieves the highest FFT magnitude.

receiver would mistakenly detect jamming down-chirps as the SFD of a legitimate packet. As a result, the frame timing of a legitimate packet is detected incorrectly, which can lead to failures of packet demodulation.

However, this jamming method has specific requirements and limitations. First, jamming down-chirps need to arrive before a legitimate SFD. This requires the preamble length of a legitimate packet to be long enough for a jammer to catch up. In our experiment, we set the preamble length of a legitimate packet to be 12. We empirically observe that a packet is difficult to be jammed by consecutive down-chirps if the length of preamble is less than 12. Therefore, this jamming method is only applicable to packets with long preambles. Second, jamming with consecutive down-chirps can only work at the preamble part of a packet. The jamming down-chirps do not interfere with the up-chirps in the payload of a legitimate packet. A receiver can successfully demodulate a payload up-chirp in presence of jamming down-chirps, due to orthogonality of the two types of signals. For example, Fig. 10 shows the spectrum and FFT results of a payload chirp jammed by a down-chirp. After being dechirped, the legitimate up-chirp still yields the highest FFT magnitude while the energy of the down-chirp spans all frequencies. Besides, similar to identical consecutive chirps, this jamming attack can be easily detected using down-chirp correlation and detection.

## 5 COUNTERMEASURE

In the previous section, we reveal that current LoRaWAN suffers from the risk of synchronized jamming attacks. In this section, we present a new countermeasure to protect LoRaWAN against synchronized jamming attacks.

### 5.1 Rationale

Recall jamming conditions in Section 4.1 (*i.e.*, *C-3*), to jam a LoRa packet, the power of a jamming chirp should be higher than that of a legitimate chirp in a demodulation window, as illustrated in Fig. 11(a). We can expect a discrepancy in

**Algorithm 1** Legitimate Chirp Selection

**Input:** Jammed chirp $symb\_chirp$, RSS difference threshold $\Delta RSS_{TH}$, and RSS of legitimate preamble $RSS\_pkt$.
**Output:** Symbol id of a legitimate chirp $symb\_id$.

```
Initialize peak_num = 0 and peak_candidate = 0.
Dechirp the jammed chirp and perform FFT to get fft_pwr and corresponding peak index peak_idx(origin).
while (True) do
    [peak_pwr, peak_idx] = max(fft_pwr).
    if peak_pwr − RSS_pkt > ΔRSS_TH then
        peak_num = peak_num + 1.
        Store peak_idx to peak_candidate(peak_num) = peak_idx.
        Update fft_pwr by removing peak_pwr and nearby peaks.
    end if
end while
if peak_num > 0 then
    symb_id = peak_candidate(peak_num) − 1.
else
    symb_id = peak_idx(origin) − 1.
end if
```

FFT magnitude between a jamming chirp and a legitimate chirp after demodulation, as shown in Fig. 11(b). We base our countermeasure on this fact, and propose to differentiate a legitimate chirp from a jamming chirp by checking their received signal strength (*i.e.*, RSS) in *power domain*.

### 5.2 Hypothesis

(1) We assume that the transmit power of a LoRa node remains invariant during one packet transmission. In practice, the RSS of a LoRa packet can be influenced by many factors (*e.g.*, transmit power, communication distance, receiver gain, *etc.*), but most of these factors are generally invariant during one packet transmission. In our targeted scenarios, we assume that LoRa nodes generally remain stationary or move at a low speed. More importantly, since LoRa PHY (*i.e.*, CSS) does not modulate the amplitude of LoRa chirps, the power level of LoRa chirps from the same packet would remain pretty stable and share a high similarity. (2) We assume that a selective jammer starts jamming after detecting the header of a legitimate packet and leaves the packet preamble intact. As such, a receiver (*i.e.*, gateway) can measure the RSS of a legitimate packet from the preamble of the packet and use the measured RSS to separate legitimate chirps from jamming chirps.

### 5.3 Algorithm Design

In this subsection, we present the workflow and algorithm design of an RSS-based LoRa decoder. This RSS-based decoder can be used as a countermeasure to the synchronized jamming attack.

Figure 12 shows the countermeasure decoding process. A receiver first detects the preamble of a LoRa packet. In addition to extracting symbol timing (*i.e.*, chirp edges) from preamble as in a standard LoRa decoder, we also measure the RSS of preamble chirps. We then employ the same method of a standard decoder to locate and demodulate symbol chirps in the payload. In each demodulation window, we obtain the interleaved FFT results of demodulated legitimate and jamming chirps. Figure 11(b) shows the FFT magnitude of a demodulation window. There are two main peaks within this demodulation window. The magnitude of a jamming chirp is higher than that of a legitimate chirp. A standard decoder will select the highest FFT peak as demodulation result. Different from a standard decoder, we pick the FFT peak with a magnitude that can best match the RSS measured from the preamble as the demodulation result of a legitimate chirp.

Algorithm 1 presents how to select a legitimate chirp based on RSS of the preamble of a legitimate packet. The input of the algorithm includes raw signals of a jammed chirp, the RSS difference threshold, and the RSS of a legitimate preamble chirp. After dechirp and FFT operations, our algorithm selects an FFT peak which best matches the RSS of a legitimate preamble chirp, and considers it as the targeted legitimate chirp. We iteratively apply this method to

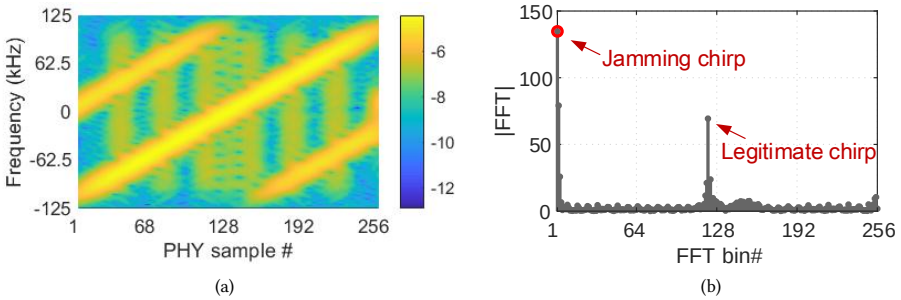



Fig. 11. Jamming power is higher than legitimate power: (a)Received signal strength of a jamming chirp vs. a legitimate chirp. (b)FFT magnitude of a demodulated jamming chirp vs. legitimate chirp.

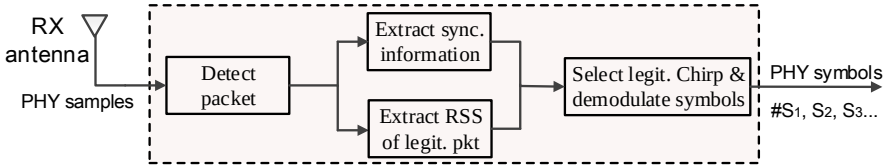


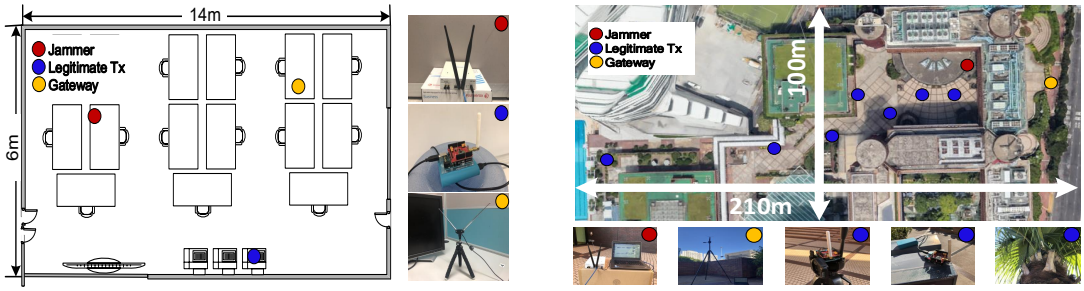

Fig. 12. Countermeasure workflow.

demodulate all legitimate chirps and feed the demodulated symbols into a standard decoder to produce the payload data of legitimate packets.

We note that if the RSS of jamming chirps and the RSS of legitimate chirps are very close, our RSS-based protection method alone cannot separate the legitimate chirps from the jamming chirps. There should be a minimum RSS difference ($\Delta RSS_{min}$) to separate legitimate chirps from jammed chirps. According to our empirical measurements, we find that when the power of jamming chirps is 20% higher than that of legitimate chirps, our countermeasure can separate them successfully with a high probability. As such, we set the minimum RSS difference as 20% of the power of a legitimate chirp in our implementation. We note that the minimum RSS difference differs in different experiment settings and may fluctuate as the environment changes. In practice, the minimum RSS can be affected by the power of background noise. In our jamming attack model, we consider jamming chirps as interference signals and consider all other signals coexisting in the same frequency band as background noise. Basically, the minimum RSS gap between jamming chirps and legitimate chirps needs to be higher than 2× of the power of background noise to separate them apart; otherwise, the peak value of a jamming chirp may be lower than that of a legitimate chirp when the noise adds up inversely with the jamming chirp while a legitimate chirp adds up coherently with noise.

In practice, it can be very challenging for a jammer to tune the transmission power of jamming chirps so that the RSS of jamming chirps received by a LoRa gateway can be of the similar RSS of legitimate chirps. Note that there is no feedback to the jammer from either the legitimate LoRa node or the LoRa gateway. Besides, in case of transmission failure because of jamming attacks, a LoRa node would retransmit at a different transmission power. Since our RSS-based protection method is orthogonal to the existing collision recovery methods that leverage time and frequency domain information, those existing methods can be used in parallel to enhance the protection method.

**Generality.** There are other wireless communication adopting CSS modulation in their physical layer. For example, radar systems adopt chirp signals for communication and FMCW systems uses chirp signals for wireless sensing. These systems may also suffer jamming attacks as in LoRa. We believe that our proposed countermeasure could be applied to other wireless systems as long as they use chirp signals in their physical layer.




(a) Indoor experiment map.

(b) Outdoor experiment map.

Fig. 13. Experiment layout.

### 5.4 Enhanced Countermeasure

In this subsection, we make one step further to enhance the protection at the gateway against synchronized jamming attacks. Recent work PCube [20] leverages channel phase domain information to differentiate packets transmitted from different legitimate nodes, inspiring us to improve RSS-based countermeasure to RSS-phase joint countermeasure.

According to our empirical study and jamming design, a jammer initiates jamming attacks by imitating the frequency and time characters of legitimate chirps and defeating legitimate chirps with higher power. However, phase information is ignored in our previous design. Signals transmitted by a legitimate node and a jammer propagate through different air channels and thus have different channel phase characters. These channel phase characters are relatively stable within one packet. In fact, it is impossible for a jammer to reproduce the channel phase information of a legitimate node. A jammer and a legitimate node are not co-located in attack scenarios; otherwise the location of the jammer is exposed easily. Therefore, channel phase can be used as a unique feature to separate legitimate chirps from jamming chirps. Thus, we propose an enhanced version of countermeasure combining both RSS and phase information.

We extract channel phase characters (*i.e.*, phase difference of air-channels, PDoA in PCube [20]) by reproducing methods in PCube. We skip the details of how to extract the phase characters for brevity. In our enhanced countermeasure, a gateway first extracts the RSS and PDoA from a legitimate preamble. It then utilizes RSS and PDoA information jointly to identify legitimate peaks in the payload part. With the assistance of channel phase information, the enhanced countermeasure can select legitimate chirps with higher accuracy. Since commodity LoRa gateways usually have two synchronized antennas [5, 21], we can obtain the phase information and implement this enhanced countermeasure on commodity gateways without incurring extra hardware costs. We evaluate the performance of the enhanced countermeasure in Section 6.2.3. Experimental results have demonstrated the effectiveness of the enhanced countermeasure in protecting gateways from synchronized jamming attacks.

## 6 IMPLEMENTATION AND EVALUATION

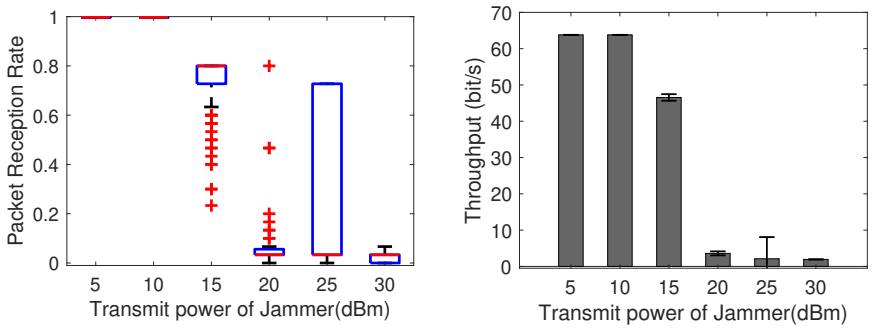
### 6.1 Implementation and Setup

We implement synchronized jamming attack and corresponding countermeasure with SDRs.

**Legitimate transmitter.** We use a COTS LoRa node (*i.e.*, LoRa shield, which consist of HopeRF's RFM96W transceiver module embedded with the Semtech SX1276 chip) as the legitimate transmitter.

**Jammer.** We a USRP N210 to work as a jammer (*i.e.*, red dot). Two antennas are mounted on the jammer. One is for overhearing the channel to detect targeted packets, and the other is for transmitting jamming signals. The signal processing algorithms for packet detection and jamming transmission are implemented in C++ with GNU Radio.

**Countermeasure gateway.** The prototype for our security-enhanced gateway is developed using an RTL-SDR dongle based on the `gr-lora` open-source project [22]. `gr-lora` is implemented with GNU Radio (C++ based). PHY samples received by an RTL-SDR dongle are sent to a laptop through a 100 Gigabit Ethernet for high computing efficiency. The laptop runs our RSS-based countermeasure implemented in MATLAB to process the raw PHY samples. As our countermeasure is software-based, in practice, it can be easily integrated into commodity LoRa gateways by updating the software of gateways. For comparison, this RTL-SDR without countermeasure algorithm is seen as a standard LoRa demodulator.




(a) (b)

Fig. 14. Jamming with different transmission power. Victim's (a) PPR and (b) Throughput.

We conduct experiments and evaluation in both indoor and outdoor environments. Specifically, as shown in Fig. 13, the indoor test bed spans $14 \times 6$ $m^2$ and it is a typical office room with rich multipaths. The outdoor test bed spans $210 \times 100$ $m^2$ and it is an urban outdoor environment with many skyscrapers. We put LoRa transmitters at different places (blue dots in Fig. 13(a) and Fig. 13(b)). Note that we do not need to modify the COTS LoRa node, neither hardware nor software. We set the central frequencies of all devices as 915 $MHz$. The default spreading factor, code rate, and bandwidth of the LoRa packets are 8, 4, and 250 $KHz$, respectively. We set the payload length of each packet to 30 with a default transmission power of 5 $dBm$. Since the duty cycle of standard LoRaWAN is too small (*e.g.*, 1% in Europe), we configure a legitimate transmitter to send LoRa packets every 2 seconds to facilitate our experiment. In fact, the packet inter-arrival rate would not affect the performance of jamming and countermeasure because our SDR-based jammer and gateway are always hearing the targeted channel. A jammer will jam every targeted packet and a gateway will receive raw samples of each packet.

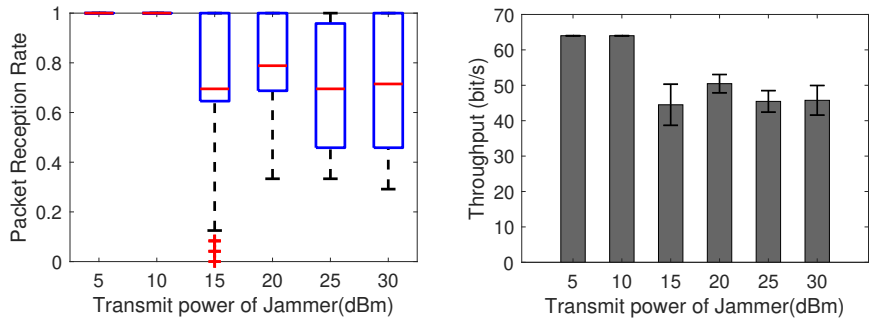
In the evaluation, we explore two main research questions. What's the impact of jamming attacks? And what's the effectiveness of our countermeasure? To answer these two questions, we implement the following two schemes: **1) Victim**: Legitimate LoRa communication (uses standard LoRa demodulation) under jamming attack, which is used to evaluate the impact of jamming attack; and **2) Countermeasure:** The victim protected by our countermeasure against jamming, which is used to demonstrate the effectiveness of our countermeasure.

We use the three metrics to evaluate the performance. **1) PRR:** Packet Reception Rate (PRR) is the ratio of correctly received packets over transmitted legitimate packets. **2) SER:** Symbol Error Rate (SER) is the ratio of incorrectly demodulated symbols over the total number of transmitted payload symbols; and **3) Throughput:** It quantifies the successfully received bits per unit time. We also compare our countermeasure with FTrack[10] and Choir[11] against jamming attack.

### 6.2 Basic Performance

*6.2.1 Impact of Jamming Attack.* We first evaluate the impact of jamming attacks in an indoor environment (*i.e.*, in Fig. 13(a)). In this experiment, we keep the three players static. We configure the transmission power of the legitimate transmitter to 5 $dBm$ while varying the transmission power of the jammer from 5 $dBm$ to 30 $dBm$. We conduct over 120 measurements under each transmission power setting.

Fig. 14 shows victim's PPR and throughput under jamming with different transmission power. We observe that when jamming power is relatively small ($5 \sim 10$ $dBm$), the PRR of Victim is almost 100%, meaning that this jamming attack has no impact on LoRa communication due to its low jamming power. With further increasing of jamming power (15 $dBm$), victim's PRR begins to decrease rapidly. When jamming power is above 20 $dBm$, the PRR decreases and almost all packets will be jammed by the attacker. A legitimate receiver can only achieve a median PRR of less than 10% when the jamming power is above 20 $dBm$. Accordingly, the throughput of victim is high when the transmission power of jammer is low. However, the throughput drops drastically when jamming power is $20 \sim 30$ $dBm$. Experiment




(a) (b)

Fig. 15. Countermeasure performance with different transmission power. Countermeasure's (a) PRR and (b) Throughput.

results reveal that the performance of LoRa communication can be severely affected by synchronized jamming attacks with a relatively high transmission power.

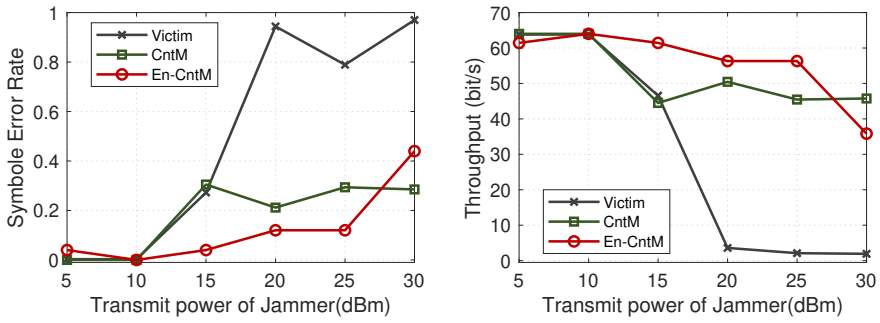
*6.2.2 Performance of Countermeasure.* We also propose countermeasure in Section 5 to protect legitimate LoRa communication. In this subsection, we conduct experiment to evaluate the performance of the proposed countermeasure. Experiment settings are same to settings in subsection 6.2.1. The only difference is that we use gateway equipped with countermeasure algorithm.

Fig. 15(a) shows countermeasure's PRR and Fig. 15(b) reports countermeasure's throughput. We can observe that our countermeasure can achieve a median PRR of higher than 70% even when the transmission power of jammer is higher than 15 $dBm$. We also notice that PRR of countermeasure is high when the transmit power of jammer is lower (*i.e.*, $5 \sim 10$ $dBm$). This experiment shows that our countermeasure can work well under different jamming power. The countermeasure's throughput in Fig. 15(b) presents similar trend as PRR in Fig. 15(a).

In comparison with Fig. 14, the overall PRR and throughput of countermeasure are much higher than those of victim, especially when jamming power is higher than 15 $dBm$. In specific, the throughput of countermeasure is 20× higher than that of victim when jamming power is 25 $dBm$, and 23× when jamming power is 30 $dBm$. This is because when transmission power is higher than 15 $dBm$, the SINR at receiver is low ($-10 \sim -5$ $dB$). In this case, the power of legitimate chirps is weaker than that of jamming chirps, leading to incorrect demodulation results of victim. In contrast, our countermeasures can separate legitimate chirps from interfering chirps by exploiting the difference in the strength of the received signals. Experimental results show that our countermeasures can protect LoRa gateways from such synchronized jamming attacks.

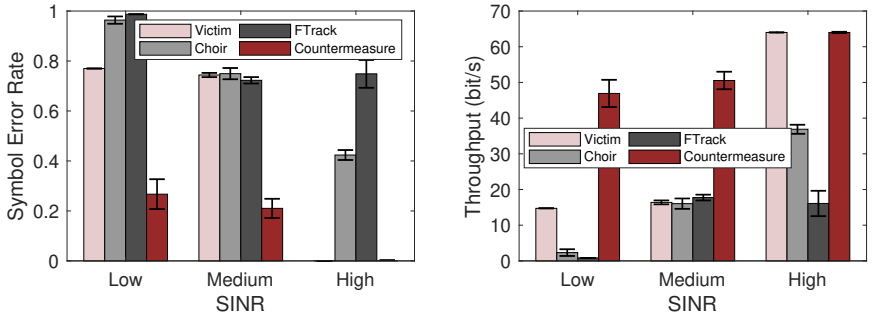
*6.2.3 Performance of Enhanced Countermeasure.* While the above approach yields encouraging results, the countermeasure misses to use channel phase information. In this subsection, we evaluate the performance of our enhanced countermeasure which utilizes both RSS and phase information jointly. Experiment settings are same to settings in the previous experiment.

We plot the average SER and throughput of the victim and two countermeasure methods in Fig. 16(a) and (b), respectively. We observe that the SERs of the victim and two countermeasures are very low when the jamming power is low. When the jamming power is 15 $dBm$, the victim and countermeasure have similar SER (30%) while the enhanced countermeasure's SER is 4%, which is much lower. These results indicate that phase information helps to improve the protection performance when the power of jamming chirps and legitimate chirps are comparable. As jamming power further increases, the SER of the victim climbs sharply to over 90%, and the average SERs of two countermeasures also increase. Yet, the overall performance of enhanced countermeasure is better than the RSS-based method. We notice that the average SER of the enhanced countermeasure is higher than that of the primary countermeasure when jamming power is 30 $dBm$. We interpret it as the phase information is sensitive to interference and noise in scenarios with high jamming power. If phase information is estimated poorly, it will be noisy for extracting legitimate peaks in the payload. Throughput results in Fig. 16(b) represent an inverse trend to the SERs in Fig. 16(a).




(a) (b)

Fig. 16. Enhanced countermeasure performance with different transmission power. Enhanced countermeasure's (a) PRR and (b) Throughput.

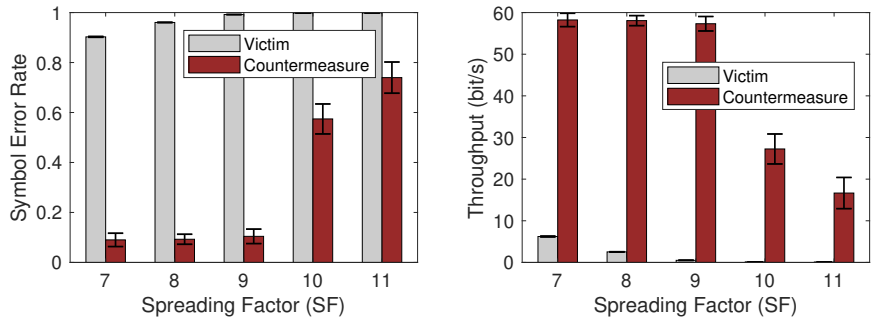



(a) (b)

Fig. 17. Performance comparison of Victim, Choir, FTrack, and countermeasure under different SINRs: (a) Symbol Error Rate (SER) and (b) Throughput.

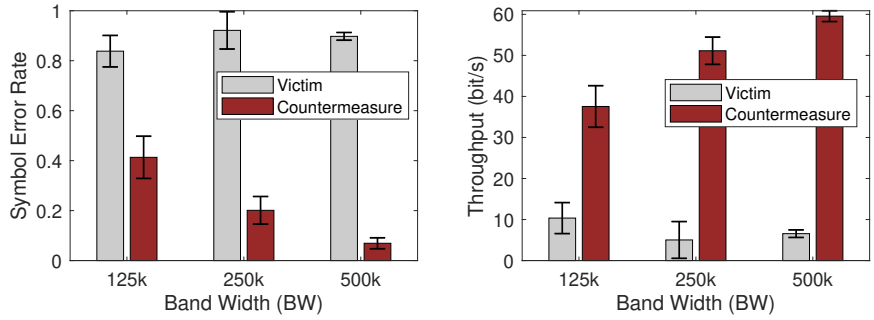
*6.2.4 Comparison with FTrack and Choir.* In this part, we compare victim and countermeasure with two typical collision recovery and parallel decoding methods, *i.e.*, FTrack and Choir. We compare these four methods in low ($-10 \sim -5\ dB$), medium ($-5 \sim 5\ dB$), and high ($5 \sim 10\ dB$) SINR scenarios. Each scenario includes more than 120 measurements. This experiment is also conducted in indoor environment.

We plot the SER and throughput in Fig. 17. We observe that victim, Choir and countermeasure have lower SERs as SINR becomes higher. However, FTrack has over 72% SER in all scenarios. This is because FTrack distinguishes colliding chirps by using frequency tracks caused by time misalignment of two chirps. However, jammer in this paper synchronizes jamming chirps with legitimate chirps, making it hard for collision recovery method which uses timing information to separate. Choir disentangles colliding chirps by leveraging the disparity in frequency domain, higher signal strength benefits its performance. However, since a synchronized jammer can mimic the fractional part of the a legitimate LoRa node, Choir also fails to distinguish collided chirps. As for victim, it achieves low SER and high throughput when the SINR is high while it suffers severe interference and achieves high SER and low throughput when the SINR is low. In contrast, our countermeasure has best performance in terms of SER and throughput in all SINR scenarios. Specifically, in low SINR scenario, our countermeasure only has 26% SER while FTrack and Choir have SER of 96.38% and 98.7% respectively, even higher than that of victim (77%) using standard LoRa demodulation. In high SINR scenario, Choir and FTrack still have very high SER and low throughput, while countermeasure and victim have almost 0 SER and 100% throughput. This experiment shows that our RSS-assisted countermeasure outperforms all existing countermeasures.




(a) (b)

Fig. 18. Impact of SF on (a) Symbol Error Rate (SER) and (b) Throughput of Victim and countermeasure.




(a) (b)

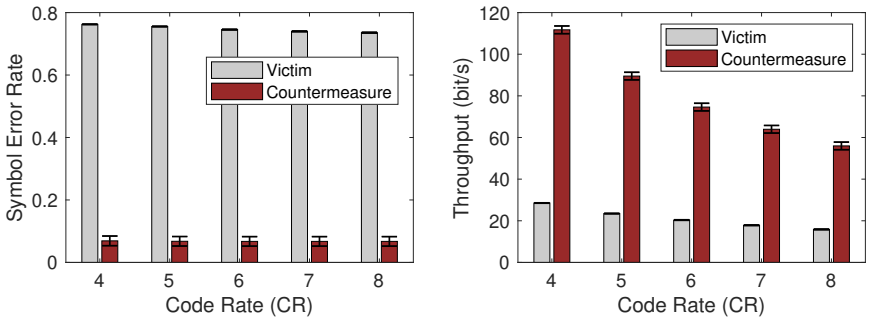
Fig. 19. Impact of BW on (a) Symbol Error Rate (SER) and (b) Throughput of Victim and Countermeasure.

### 6.3 Impact of LoRa Configuration

The symbol error rate and throughput of LoRa nodes are sensitive to LoRa configuration parameters, including spreading factors, bandwidths, and code rates. In this subsection, we investigate the impact of LoRa packet configuration on the performance of jamming attacks and our countermeasure strategy. We adopt the same experiment settings as in Section 6.2. Due to space limitation, we only present the results when a jammer emits chirps with high power ($\geq$ 20 $dBm$).

1) Impact of spreading factor (SF). In this experiment, we fix the bandwidth and code rate to 250 $kHz$ and 4, respectively. The parameter of SF is varied from 7 to 11 to see its impact on victim and countermeasure. We compare PHY layer symbol error rates of standard demodulation method (*i.e.*, victim) and our countermeasure strategy (*i.e.*, countermeasure) in Figure 18. As expected, the SER of victim stays at high level (*e.g.*, > 90%) for all SFs due to the high jamming power. The SER is almost 100% when the spreading factor is 9 $\sim$ 11. In contrast, the countermeasure can decode legitimate packets with SER lower than 10% when $SF = 7 \sim 9$. We observe that the SER of countermeasure increases dramatically to higher than 60% as SF increases to 10 and 11. This is because the frequency gap between LoRa symbols becomes narrower as SF increases, making it harder for countermeasure to demodulate symbols correctly. However, according to our observation, the countermeasure's decoding results (FFT bin locations) are close to the correct FFT bins, which, to some extent, can provide useful information to the receiver. We leave this for future exploration.

Figure 18(b) compares the throughput of victim and countermeasure. The throughput of victim is very low (nearly zero) as SF increases to 10. In contrast, the average throughput of countermeasure reaches more than 57 bit/s (90.0% of the ideal throughput) when SF is less than 10. However, when SF is greater than 9, the throughput of countermeasure decreases to less than 30 bit/s. For example, the average throughput of countermeasure is less than 20 bit/s when SF is 11. The above results demonstrate that packets with a larger SF are generally more vulnerable to jamming attacks. Countermeasure works better when SF is small.




(a) (b)

Fig. 20. Impact of CR on (a) Symbol Error Rate (SER) and (b)Throughput of Victim and Countermeasure.

2) Impact of bandwidth (BW). To explore the impact of bandwidth, we set $SF = 8$ and fix code rate to 4 while changing BW from 125 $kHz$ to 500 $kHz$ in this experiment. The SER and throughput of victim and countermeasure are shown in Figure 19(a). We can observe that the average SER of standard LoRa decoder is higher than 81% across all bandwidth settings. The average SER of victim even higher than 90% when BW is 250 $kHz$. In contrast, our countermeasure strategy can correctly demodulate legitimate chirps with $SER < 20\%$ when $BW \geq 250$ $kHz$. The countermeasure's SER decreases as bandwidth increases. This is reasonable because wider bandwidth can create larger frequency gap between LoRa symbols, making it easier to distinguish different symbols with different initial frequencies. As such, a wider bandwidth will generally make the demodulation more robust to jamming attack. This experiment demonstrates that our countermeasure can reduce symbol error rate in different BW settings.

Figure 19(b) illustrates the throughput of victim and countermeasure across different BW configurations. As expected, victim yields low throughput (*i.e.*, less than 15 bit/s) with all bandwidth settings. With the help of countermeasure, the average throughput reaches to 38 bit/s, 53 bit/s, and 59.6 bit/s, respectively. Our countermeasure performs better as bandwidth increases.

3) Impact of code rate (CR). CR is an important configuration parameter in LoRa networks. CR configures the encoding redundancy of LoRa packets to combat bit errors, thus largely impacting the error-correcting capability of a victim. To evaluate the impact of CR, we vary CR from 4 to 8, and fix SF to 8 and bandwidth to 250 $kHz$, respectively. In this experiment, a legitimate transmitter is set to send packets once per second. Figure 20 (a) and (b) show the evaluation results of SER and throughput, respectively. We can observe that the average SER of the standard LoRa decoder is high (*i.e.*, $> 73.6\%$) while the SER of our countermeasure is less than 6.9% across all CR settings. Note taht the SERs of victim and countermeasure remain stable across different code rates. However, as shown in Figure 20 (b), the throughput of both victim and countermeasure decrease as the code rate increases. In specific, victim and countermeasure achieve the highest throughput of 28.5 bit/s and 111.7 bit/s respectively when CR=4. This is reasonable because code rate represents the redundancy bits in encoding every 4-bit data. A larger CR indicates more redundancy bits in payload. Since the SER remains stable across different code rates, the overall valuable bits (*i.e.*, goodput) delivered by the transmitter decreases as code rate increases.

LoRa exploits redundancy to endure short and burst interference. However, increasing CR fails to protect LoRa signals against synchronized jamming. Besides, a larger CR also increases the transmission time and thus decreases the goodput. Our countermeasure can substantially reduce the symbol error rate and thus improve LoRa networks' throughput.

### 6.4 Impact of Jamming Distance

We investigate the impact of jamming distance in an outdoor testbed (Fig. 13(b)). We sue $SF = 8$, $BW = 250$ $kHz$ and CR=4 as default parameters unless otherwise specified. In the first experiment, we place a jammer at a fixed distance (15 $m$) to a gateway and keep them static. We vary the location of a legitimate transmitter to evaluate the effective jamming range. The transmission power of the jammer is fixed at 20 $dBm$ while the transmission power of the legitimate

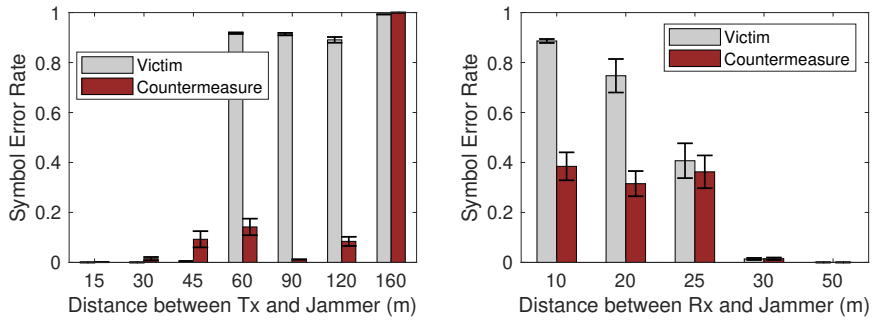



Fig. 21. Impact of (a) Distance between Tx and Jammer and (b) Distance between Rx and Jammer on SER of Victim and Countermeasure.

node is 23 *dBm* (which is the highest transmission power). The distance between the transmitter and the jammer varies from 15 *m* to 160 *m* with uneven steps.

We reports the SER of standard demodulation method (*i.e.*, victim) and our countermeasure strategy (*i.e.*, countermeasure) in Figure 21(a). We observe that both victim and countermeasure achieves low SER when the transmitter is within 45 *m* from the gateway. This is reasonable since the legitimate signal power is higher than that of jamming signal. When the distance is 60 $\sim$ 120 *m*, the SER of victim increases dramatically($\geq$ 80%), and thus a gateway cannot can correctly demodulate legitimate packets. This is because the signal power of legitimate packets falls below the jammer power. In contrast, gateways protected by our countermeasure can still demodulate packets correctly when the distance is 60 $\sim$ 120 *m*, achieving an average SER of no more than 16%. However, the SERs become almost 100% for both strategies as the distance further increases to 160 *m*. In this case, the received signal strength of legitimate packets is too weak to be demodulated.

In the second experiment, we keep a gateway and a legitimate node at fixed locations and move a jammer to different places to evaluate the jamming performance. Packets transmitted by the legitimate node follow the same configuration we adopted in the first experiment. We fix the distance between the legitimate node and the gateway at 20 *m*, and we vary the distance between the jammer and the gateway from 10 *m* to 50 *m* with uneven steps. We set the TX power gain of a jammer to 80 *dB*. The experiment results are presented in Fig. 21(b). We can observe that the overall SERs of the standard demodulation method (*i.e.*, victim) decrease as jamming distance increases. As the distance between gateway and jammer increases, the signal power of jamming chirps becomes weaker, making less interference to legitimate packets. The SERs of countermeasure are higher than 30% when jamming distance $\leq$ 25 *m* because of the comparable signal power between jamming chirps and legitimate chirps. The RSS-based countermeasure method has limited capability to separate legitimate chirps from jamming chirps when they have similar signal power. When the distance between gateway and jammer is long enough, *i.e.*, $\geq$ 30 *m*, both victim and countermeasure can correctly demodulate the symbols with nearly zero SERs since the power of jamming chirps becomes too weak in this range.

In summary, when a jammer is very close to a gateway receiver, the receiver's performance will be dramatically affected by the jammer. Gateways protected by countermeasure algorithm can help to demodulate some of the symbols correctly. Our countermeasure can work as a complementary therapy to jamming attacks and improve legitimate communication performance.

## 7 RELATED WORK

A variety of LPWAN technologies such as SigFox [23], NB-IoT [24], LTE-M [25] and LoRa [26] have been proposed to support wide area network connection for IoT devices. Prior efforts [1, 27–29] are devoted to the measurement study and performance analysis of LoRa, such as packet air-time [30, 31], power consumption [1, 32], coverage [33, 34], PHY security [35], *etc.* Based on these measurements, many strategies [3, 21, 36–47] have been proposed to optimize LoRa communications and applications.

Wireless jamming has been extensively studied in literature [48–50]. Recent works study the impact of jammer to LoRaWAN and propose countermeasures. LoRaTS [51] studies the attack-aware data timestamping in LoRaWAN, which can protect LoRaWAN against frame delay attack. Aras *et al.* [52] identify a few security vulnerabilities of LoRa including encryption key extraction, jamming attacks, and replay attacks. Aras *et al.* [7] use commodity LoRa nodes as jammers to selectively jam LoRa packets. Previous collision recovery and parallel decoding schemes can help mitigate the impact of those jammers. Unlike those works, we study the impact of synchronized jamming chirps and propose countermeasure to protect against such new jamming attacks. Radar and FMCW systems [53] also use chirp signals for communication or wireless sensing. They are also susceptible to jamming attacks like LoRa. We believe our proposed countermeasure can be applied to other wireless systems that use chirp signals in their physical layer.

Collision recovery and parallel demodulation schemes [54] can be used to solve collisions of LoRa chirps and jamming signals and thereby protect LoRa communication under jamming attacks. Choir [11] differentiates LoRa chirps by examining the frequency differences between different LoRa nodes. Choir groups different chirps according to different frequencies and separates colliding LoRa chirps. Other works leverage the misalignment of colliding packets in the time domain to separate colliding chirps. FTrack [55] detects the continuity of chirps within a demodulation window to recover collisions. mLoRa [12] derives the time offset between colliding packets based on preamble detection results and obtains collision-free PHY samples. CoLoRa [56] groups LoRa chirps to their corresponding LoRa nodes by examining the power level of the same frequency in different demodulation windows. Pyramid [54] enables real-time LoRa collision decoding with peak tracking. NScale [19] amplifies the time offsets between colliding packets with non-stationary signal scaling. CIC [57] exploits spectrum information of different parts within each symbol and uses such subsymbols to cancel out interference. Those previous works mainly resolve collisions by leveraging the time domain information, the frequency domain information, or both. Our protection mechanism complements and enhances the previous works. We extract legitimate chirps from jamming chirps by examining their corresponding received power in demodulation windows. Latest work PCube [20] leverages phase domain information to scale LoRa concurrent transmission. Inspired by PCube, we propose an enhanced countermeasure using RSS and phase information jointly to protect a gateway from synchronized jamming attacks.

## 8 CONCLUSION

In this paper, we reveal the vulnerability of LoRa PHY under the attack of synchronized jamming chirps. The insight of the jamming attack is that a well-synchronized jamming chirp cannot be separated from a legitimate LoRa chirp in the time domain. As a result, most existing protection methods cannot protect the LoRa PHY against such synchronized jamming chirps. To enhance the LoRa PHY, we propose countermeasures, which leverage both power and channel phase information to protect gateways. The protection method can complement and enhance existing collision recovery schemes which leverage the chirp misalignment in time domain or the frequency disparity in frequency domain.

## ACKNOWLEDGEMENT

This work is supported by the National Nature Science Foundation of China under grant 61702437 and Hong Kong GRF under grant PolyU 152165/19E. Yuanqing Zheng is the corresponding author.

## REFERENCES

[1] J. C. Liando, A. Gamage, A. W. Tengourtius, and M. Li, "Known and unknown facts of lora: Experiences from a large-scale measurement study," *ACM Trans. Sen. Netw.*, vol. 15, no. 2, Feb. 2019.
[2] J. P. S. Sundaram, W. Du, and Z. Zhao, "A survey on lora networking: Research problems, current solutions and open issues," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, 2019.
[3] X. Xia, Y. Zheng, and T. Gu, "Litenap: Downclocking lora reception," in *IEEE INFOCOM'20*, 2020.
[4] Semtech lora applications. [Online]. Available: https://www.semtech.com/lora/lora-applications
[5] F. Zhang, Z. Chang, K. Niu, J. Xiong, B. Jin, Q. Lv, and D. Zhang, "Exploring lora for long-range through-wall sensing," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 2, Jun. 2020. [Online]. Available: https://doi.org/10.1145/3397326
[6] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou, "A survey on jamming attacks and countermeasures in wsns," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 4, pp. 42–56, 2009.
[7] E. Aras, N. Small, G. S. Ramachandran, S. Delbruel, W. Joosen, and D. Hughes, "Selective jamming of lorawan using commodity hardware," in *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2017, pp. 363–372.
[8] A. Rahmadhani and F. Kuipers, "When lorawan frames collide," in *Proceedings of the 12th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, 2018, pp. 89–97.
[9] K. Mikhaylov, R. Fujdiak, A. Pouttu, V. Miroslav, L. Malina, and P. Mlynek, "Energy attack in lorawan: experimental validation," in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, 2019, pp. 1–6.

[10] X. Xia, Y. Zheng, and T. Gu, "Ftrack: Parallel decoding for lora transmissions," in *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*, 2019, pp. 192–204.

[11] R. Eletreby, D. Zhang, S. Kumar, and O. Yağan, "Empowering low-power wide area networks in urban settings," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, 2017, pp. 309–321.

[12] X. Wang, L. Kong, L. He, and G. Chen, "mlora: A multi-packet reception protocol in lora networks," in *2019 IEEE 27th International Conference on Network Protocols (ICNP)*. IEEE, 2019, pp. 1–11.

[13] N. Hou, X. Xia, and Y. Zheng, "Jamming of lora phy and countermeasure," in *IEEE INFOCOM'21*, 2021.

[14] L. Shield. (2019, Jan.) Lora shield. [Online]. Available: https://wiki.dragino.com/

[15] A. D. Wood, J. A. Stankovic, and G. Zhou, "Deejam: Defeating energy-efficient jamming in ieee 802.15. 4-based wireless networks," in *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. IEEE, 2007, pp. 60–69.

[16] L. Neng-Jing and Z. Yi-Ting, "A survey of radar ecm and eccm," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 3, pp. 1110–1120, 1995.

[17] D. Chapman, "Adaptive arrays and sidelobe cancellers-a perspective," *Microwave Journal*, vol. 20, pp. 43–46, 1977.

[18] S. Wdowinski, Y. Bock, J. Zhang, P. Fang, and J. Genrich, "Southern california permanent gps geodetic array: Spatial filtering of daily positions for estimating coseismic and postseismic displacements induced by the 1992 landers earthquake," *Journal of Geophysical Research: Solid Earth*, vol. 102, no. B8, pp. 18 057–18 070, 1997.

[19] S. Tong, J. Wang, and Y. Liu, "Combating packet collisions using non-stationary signal scaling in lpwans," in *ACM MobiSys'20*, 2020.

[20] X. Xia, N. Hou, Y. Zheng, and T. Gu, "Pcube: scaling lora concurrent transmissions with reception diversities," in *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, 2021, pp. 670–683.

[21] B. Xie and J. Xiong, "Combating interference for long range lora sensing," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 69–81.

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

[23] SigFox. (2019, Jan.) Sigfox overview. [Online]. Available: https://www.sigfox.com/en/sigfox-iot-technology-overview

[24] R. Ratasuk, B. Vejlgaard, N. Mangalvedhe, and A. Ghosh, "Nb-iot system for m2m communication," in *Proceedings of IEEE Wireless Communications and Networking Conference*, ser. (WCNC 2016), Apr 2016, pp. 1–5.

[25] M. Lauridsen, I. Z. Kovacs, P. Mogensen, M. Sorensen, and S. Holst, "Coverage and capacity analysis of lte-m and nb-iot in a rural area," in *Proceedings of IEEE 84th Vehicular Technology Conference*, ser. (VTC-Fall 2016), Sep 2016, pp. 1–5.

[26] L. Alliance. (2019, Jan.) Lorawan for developer. [Online]. Available: https://lora-alliance.org/lorawan-for-developers

[27] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of lorawan," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, Sep. 2017.

[28] D. Bankov, E. Khorov, and A. Lyakhov, "On the limits of lorawan channel access," in *Proceedings of the 2016 International Conference on Engineering and Telecommunication (EnT)*, Nov 2016, pp. 10–14.

[29] M. C. Bor, U. Roedig, T. Voigt, and J. M. Alonso, "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)*, Nov 2016, pp. 59–67.

[30] U. Noreen, A. Bounceur, and L. Clavier, "A study of lora low power and wide area network technology," in *Proceedings of the 2017 International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, Oct 2017, pp. 1–6.

[31] A. Lavric and V. Popa, "A lorawan: Long range wide area networks study," in *Proceedings of the 2017 International Conference on Electromechanical and Power Systems (SIELMEN)*, Oct 2017, pp. 417–420.

[32] T. Bouguera, J.-F. Diouris, J.-J. Chaillout, R. Jaouadi, and G. Andrieux, "Energy consumption model for sensor nodes based on lora and lorawan," *Sensors*, vol. 18, no. 7, p. 2104, Jul 2018.

[33] J. Petajajarvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the coverage of lpwans: range evaluation and channel attenuation model for lora technology," in *Proceedings of the 2015 14th International Conference on ITS Telecommunications (ITST)*, Dec 2015, pp. 55–59.

[34] J. Petäjäjärvi, K. Mikhaylov, M. Pettissalo, J. Janhunen, and J. Iinatti, "Performance of a low-power wide-area network based on lora technology: Doppler robustness, scalability, and coverage," *International Journal of Distributed Sensor Networks*, vol. 13, no. 3, pp. 1–16, Mar 2017.

[35] N. Hou and Y. Zheng, "Cloaklora: A covert channel over lora phy," in *The 28th IEEE International Conference on Network Protocols (ICNP'20)*, 2020.

[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 *Proceedings of the ACM/IEEE IPSN'18*, 2018, p. 60–71.

[37] 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 *Proceedings of the USENIX NSDI'20*, 2020, pp. 339–352.

[38] A. Gadre, F. Yi, A. Rowe, B. Iannucci, and S. Kumar, "Quick (and dirty) aggregate queries on low-power wans," in *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2020, pp. 277–288.

[39] A. Balanuta, N. Pereira, S. Kumar, and A. Rowe, "A cloud-optimized link layer for low-power wide-area networks," in *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 247–259.

[40] A. Gamage, J. C. Liando, C. Gu, R. Tan, and M. Li, "Lmac: Efficient carrier-sense multiple access for lora," in *The 26th Annual International Conference on Mobile Computing and Networking (MobiCom'20)*, 2020.

[41] M. Hessar, A. Najafi, and S. Gollakota, "Netscatter: Enabling large-scale backscatter networks," in *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*, 2019, pp. 271–284.

[42] R. Nandakumar, V. Iyer, and S. Gollakota, "3d localization for sub-centimeter sized devices," in *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, 2018, pp. 108–119.

[43] G. Chen, W. Dong, and J. Lv, "Lofi: Enabling 2.4ghz lora and wifi coexistence by detecting extremely weak signals," in *IEEE INFOCOM'21*, 2021.

[44] L. Liu, Y. Yao, Z. Cao, and M. Zhang, "Deeplora: Learning accurate path loss model for long distance links in lpwan," in *IEEE INFOCOM'21*, 2021.

[45] Y. Wang, X. Zheng, L. Liu, and H. Ma, "Polartracker: Attitude-aware channel access for floating low power wide area networks," in *IEEE INFOCOM'21*, 2021.

[46] X. Guo, L. Shangguan, Y. He, J. Zhang, H. Jiang, A. A. Siddiqi, and Y. Liu, "Aloba: rethinking on-off keying modulation for ambient lora backscatter," in *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, 2020, pp. 192–204.

[47] J. Liu, J. Gao, S. Jha, and W. Hu, "Seirios: leveraging multiple channels for lorawan indoor and outdoor localization," in *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking*, 2021, pp. 656–669.

[48] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, 2005, pp. 46–57.
[49] Y. Zou, J. Zhu, X. Wang, and L. Hanzo, "A survey on wireless security: Technical challenges, recent advances, and future trends," *Proceedings of the IEEE*, vol. 104, no. 9, pp. 1727–1765, 2016.
[50] Y. Liu, H.-H. Chen, and L. Wang, "Physical layer security for next generation wireless networks: Theories, technologies, and challenges," *IEEE Communications Surveys and Tutorials*, vol. 19, no. 1, pp. 347–376, 2017.
[51] C. Gu, L. Jiang, R. Tan, M. Li, and J. Huang, "Attack-aware data timestamping in low-power synchronization-free lorawan," in *IEEE ICDCS'20*, 2020.
[52] E. Aras, G. S. Ramachandran, P. Lawrence, and D. Hughes, "Exploring the security vulnerabilities of lora," in *CYBCONF'17*, 2017.
[53] Q. Huang, Z. Luo, J. Zhang, W. Wang, and Q. Zhang, "Loradar: Enabling concurrent radar sensing and lora communication," *IEEE Transactions on Mobile Computing*, 2020.
[54] Z. Xu, P. Xie, and J. Wang, "Pyramid: Real-time lora collision decoding with peak tracking," in *IEEE INFOCOM'21*, 2021.
[55] X. Xia, Y. Zheng, and T. Gu, "Ftrack: Parallel decoding for lora transmissions," *IEEE/ACM Transactions on Networking*, vol. 28, no. 6, pp. 2573–2586, 2020.
[56] S. Tong, Z. Xu, and J. Wang, "Colora: Enable muti-packet reception in lora," in *IEEE INFOCOM'20*, 2020.
[57] M. O. Shahid, M. Philipose, K. Chintalapudi, S. Banerjee, and B. Krishnaswamy, "Concurrent interference cancellation: decoding multi-packet collisions in lora," in *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, 2021, pp. 503–515.