A Novel Scheme for Discrete and Secure LoRa Communications

Clément Demeslay, Roland Gautier, Philippe Rostaing, Gilles Burel, Anthony Fiche

sensors

MDPI

Article

A Novel Scheme for Discrete and Secure LoRa Communications

Clément Demeslay *, Roland Gautier, Philippe Rostaing, Gilles Burel and Anthony Fiche

CNRS UMR 6285, Lab-STICC, Universuty Brest, CNRS, CS 93837, 6 Avenue Le Gorgeu, CEDEX 3, 29238 Brest, France

* Correspondence: clement.demeslay@univ-brest.fr

Abstract: In this paper, we present a new LoRa transceiver scheme to ensure discrete communications secure from potential eavesdroppers by leveraging a simple and elegant spread spectrum philosophy. The scheme modifies both preamble and payload waveforms by adapting a current state-of-the-art LoRa synchronization front-end. This scheme can also be seen as a self-jamming approach. Furthermore, we introduce a new payload demodulation method that avoids the adverse effects of the traditional cross-correlation solution that would otherwise be used. Our simulation results show that the self-jamming scheme exhibits very good symbol error rate (SER) performance with a loss of just 0.5 dB for a frequency spread factor of up to 10.

Keywords: self-jamming waveforms; synchronization scheme; cross-correlation receiver; LoRa enhanced transceiver; LoRa discrete communications; LoRa privacy

check for updates

Citation: Demeslay, C.; Gautier, R.; Rostaing, P.; Burel, G.; Fiche, A. A Novel Scheme for Discrete and Secure LoRa Communications. *Sensors* **2022**, *22*, 7947. https://doi.org/10.3390/s22207947

Academic Editor: Carles Gomez

Received: 9 September 2022

Accepted: 12 October 2022

Published: 18 October 2022

1. Introduction

In the past few years, LoRa has become a front-runner in low-power wide-area network (LPWAN) solutions applied to low-energy/low-cost Internet of Things (IoT) transceivers and is increasingly implemented to achieve practical solutions in areas such as agro-informatics [1], smart home design [2] and air-quality monitoring systems [3]. The increasing number of LoRa transceivers creates increased opportunities for malicious entities to disrupt or eavesdrop LoRa communications. Many studies have been conducted by the research community to evaluate the impact of jamming on performance and countermeasures have been proposed to tackle these threats. Below, we briefly review relevant studies that consider LoRa jamming schemes.

1.1. Previous Work on LoRa Jamming

In [4], the authors investigated the impact of traditional jammers, such as band and tone jamming, on the LoRa demodulation process and highlighted the sub-optimal energy efficiency of these jamming schemes. Other research has considered smarter and more efficient jammers involving jamming LoRa nodes with LoRa signals. In [5–8], LoRa reactive jammers (the jamming signal is only sent on detection of an incoming legitimate LoRa signal) and random jammers with a frequency hopping scheme were implemented and assessed on real-world devices. The authors concluded that jammer efficiency is obtained if the LoRa signal detection scheme is well-designed with good detection capability, and has a latency as low as possible to align the jamming signal in time with the signal of interest. In other studies, investigation of jamming where the jammer seeks to prevent a legitimate LoRa node to access the network was considered. In [9], a jammer was designed to reduce received signal strength indicator (RSSI) variations at the legitimate LoRa node, leading to an almost constantly obtained DevNonce key ID and preventing network access. The authors of [10] proposed a simple jammer detection scheme based on this philosophy, while [11,12] evaluated the jamming impact but on the global LoRa WAN network, with, for example, gateway occupancy or dropping probability metrics.

The eavesdropping case has, however, attracted less attention by the research community. To ensure secret communications, most of the proposed solutions rely on cryptographic schemes. For example, a frequency-hopping scheme was proposed in [13], while [14] introduced a reduced complexity advanced encryption system (AES) solution for the key management of LoRa WAN. Finally, recently in [15] a physical layer encryption method leveraging the randomness of the channel was presented to bypass the use of AES that imposes a burden on complexity for low-cost LoRa nodes.

1.2. Novelty and Contributions

In this paper, we propose a cooperative scheme between the transmitter and the receiver that further enhances [15] the scheme by improving the capacity for discrete LoRa transmission. The central notion is to leverage the well-known LoRa interference impact on demodulation but constructively by spreading the useful signal energy in the frequency space with a fixed power constraint. This can be seen as self-jamming with an added layer of spectrum spreading on top of LoRa. As the receiver is cooperative, the latter can then demodulate successfully. However, in realistic conditions, time and frequency synchronization between the transmitter and the receiver must be satisfied. We therefore propose a modified and adapted version of current state-of-the-art LoRa synchronization techniques as a solution.

The key contributions of the paper are as follows:

- Proposal of an enhanced scheme ensuring discrete and secure communication.
- A refined current LoRa synchronization front-end.
- Two variants of the scheme are proposed to adapt to power/complexity constraints of both uplinks and downlinks.

The remainder of the paper is organized as follows. In Section 2, we introduce the system model and some LoRa modulation basics. Section 3 presents a first approach to combatting an eavesdropper by modifying the preamble waveforms (introducing a self-jamming scheme). A modified synchronization front-end based on state-of-the-art techniques is proposed in Section 4. In Section 5, we investigate a possible threat where, in certain circumstances, an eavesdropper may synchronize itself. In Section 6, we enhance our initial self-jamming solution by proposing a modified payload demodulation scheme. Finally, we provide simulation results in Section 7 to evaluate the self-jamming method.

1.3. Notations

Table 1 lists the most relevant notations used throughout the paper.

Table 1. List of principal notations used in the paper.

Notation and Symbols	**Meaning**
global LoRa parameters	
SF	LoRa spreading factor
M	number of possible chirp waveforms per symbol: 2^{SF}
T	symbol period
F_s	sampling frequency
T_s	sampling period
B	LoRa bandwidth
F_c	carrier frequency
indexes	
k	time index
n	frequency index
i	symbol index
u	virtual path index
m	cross-correlation index

Table 1. *Cont.*

Notation and Symbols Meaning	
entities	
A	Alice
B	Bob
E	Eve
legacy LoRa frame parameters	
N_{up}	number of upchirp pilot symbols
N_{down}	number of downchirp pilot symbols
N_{pre}	number of pilot symbols: $N_{up} + N_{down}$
N_d	number of payload symbols
N_f	total number of symbols: $N_f = N_{pre} + N_d$
a	current transmitted symbol
$x_a[k]$	transmitted a-symbol waveform
modified LoRa frame parameters	
U	number of virtual channel paths
a_{up}	upchirp pilot symbol value
a_{down}	downchirp pilot symbol value
$a_{data}^{(d)}$	d-th payload symbol
m_{up}	vector of virtual channel delays of upchirp pilot symbols
m_{down}	vector of virtual channel delays of downchirp pilot symbols
ϵ	minimum DFT gap between virtual channel paths
P_s	total transmit power available
P_J	power of each virtual channel path: $P_J = P_s/U$
$S_{up}[k]$	modified upchirp preamble waveform
$S_{down}[k]$	modified downchirp preamble waveform
$S_{data}[k]$	modified data waveform
synchronization parameters	
τ	STO delay
Δ_f	baseband carrier residual
STO_{int}, STO_{frac}	integer and fractional STO part
CFO_{int}, CFO_{frac}	integer and fractional CFO part
L	number of preamble upchirps to detect for preamble detection
$\tilde{S}_{up}^{ref}[n]$	reference DFT upchirp for synchronization
$\tilde{S}_{down}^{ref}[n]$	reference DFT downchirp for synchronization
$\lambda_{STO_{frac}\approx 0.5}$	threshold for $STO_{frac} \approx 0.5$ case detection
R	oversampling factor for STO_{frac} mitigation
various notations	
$\langle x \rangle$	averaged x: $\langle x \rangle = \frac{1}{N}\sum_{i=0}^{N-1} x_i$

2. System Model

2.1. Eavesdropping Scenario

We consider the eavesdropping scenario presented in Figure 1. There are three entities, Alice, Bob and Eve, denoted with **A**, **B** and **E** characters, respectively. **A** and **B** communicate with each other (Alice–Bob direction in the figure) in a cooperative way and exchange sensitive data that must be kept secret from eavesdroppers such as **E**. **B** has the role of the gateway and both uplink and downlink links are taken into account, depending on the **A** role. If **A** is a pure LoRa sensor, the uplink is much more critical than the downlink as the latter mainly consists of signaling traffic. However, if **A** is an actuator driven by incoming commands from **B**, for example, the downlink must be protected from **E**. We are then interested in securing both up- and downlinks and also ensuring discrete communication, reducing the intercept capability of **E**. **E** is, in this context, a fully passive receiver located sufficiently close to **A** and **B** to be able to detect both **A** or **B** LoRa signals. In this scenario, all channels separating entities are flat with additive white Gaussian noise (AWGN) and they are assumed to be symmetric. Frequency-selective channels may be considered in the future as an extension of this study.

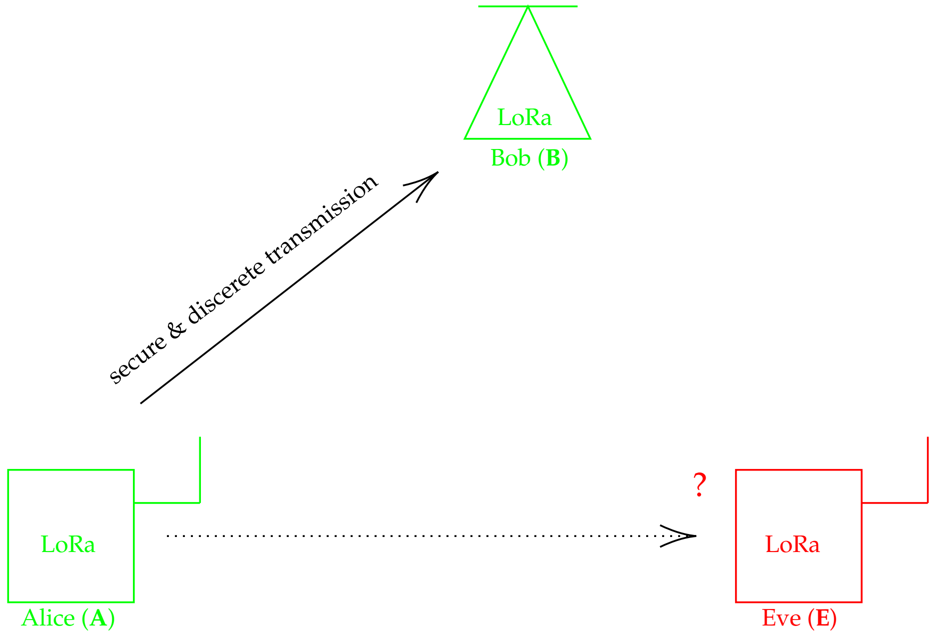

Figure 1. The eavesdropping scenario.

2.2. LoRa Modulation Overview

LoRa waveforms are a type of chirp spread spectrum (CSS) signal. These signals rely on sine waves with instantaneous frequency (IF) that vary linearly with time over the frequency range $f \in [-B/2; B/2]$ and the time range $t \in [0; T)$ (T, the symbol period). This basic signal is called an upchirp or downchirp when IF increases or decreases with time, respectively. A Lora waveform is an M-ary digital modulation, comprised of M possible chirp modulations where the IF of the upchirp is shifted by the M possible values. The modulo operation is applied to ensure that the frequency remains in the interval $[-B/2; B/2]$. The LoRa parameters are chosen such that $BT = M$ with $M = 2^{SF}$ and $SF \in \{7, 8, \ldots, 12\}$ is called the spreading factor, which also corresponds to the number of bits for a LoRa symbol. In the discrete-time signal model, the chip rate ($R_c = 1/T_c = M/T$) is usually used to sample the received signal, i.e., the sample period is $T_s = T_c = T/M = 1/B$. The signal then has M samples over one symbol period T. Each symbol $a \in \{0, 1, \ldots, M-1\}$ is mapped to an upchirp that is temporally shifted by $\tau_a = aT_c$ period. We note that a temporal shift results in a change in the initial IF.

This behavior is the heart of the M-ary chirp modulation. An expression of discrete LoRa waveforms sampled at $t = kT_s$ ($T_s = T_c$) has been derived by the authors in [16]:

$$x(kT_s; a) \triangleq x_a[k] = e^{2j\pi k\left(\frac{a}{M} - \frac{1}{2} + \frac{k}{2M}\right)} \quad k = 0, 1, \ldots, M-1. \tag{1}$$

The upchirp is the LoRa waveform with symbol index $a = 0$.

2.3. LoRa Demodulation Scheme

The authors of [17] derived a simple and efficient solution to demodulate LoRa signals. In an AWGN flat-fading channel, the demodulation process is based on the maximum likelihood (ML) detection scheme. The received signal is:

$$r[k] = \alpha x_a[k] + w[k] \tag{2}$$

with $\alpha = |\alpha|e^{j\phi}$, the complex gain of the channel and $w[k]$ an independent and identical distributed (i.i.d.) complex AWGN with zero-mean and variance $\sigma^2 = E[|w[k]|^2]$. The signal-to-noise ratio (SNR) is defined as: $SNR = |\alpha|^2 P_s/\sigma^2 = 1/\sigma^2$ with P_s the transmitted signal power and, without loss of generality, we assume $|\alpha|^2 = P_s = 1$. The ML detector aims to select the frequency index n that maximizes the scalar product $\langle r, x_n \rangle$ for $n \in \{0, 1, \ldots, M-1\}$, defined as:

$$\begin{aligned} \langle r, x_n \rangle &= \sum_{k=0}^{M-1} r[k] x_n^*[k] \\ &= \sum_{k=0}^{M-1} \underbrace{r[k] x_0^*[k]}_{\tilde{r}[k]} e^{-j2\pi \frac{n}{M} k} = \tilde{R}[n] \end{aligned} \tag{3}$$

The demodulation stage proceeds with two simple operations:

- multiply the received waveform by a downchirp $x_0^*[k]$ (also called dechirping),
- compute $\tilde{R}[n]$, the discrete Fourier transform (DFT) of $\tilde{r}[k]$, and select the discrete frequency index $\widehat{a}$ that maximizes $\tilde{R}[n]$.

in this way, the dechirp process merges all the signal energy into a unique frequency bin a that can be easily retrieved by taking the magnitude (non-coherent detection) of $\tilde{R}[n]$. The detected symbol is then:

$$\widehat{a} = \arg\max_n \left| \tilde{R}[n] \right| \tag{4}$$

2.4. LoRa Frame Structure

LoRa messages are transmitted in frames that follow the specific format depicted in Figure 2.

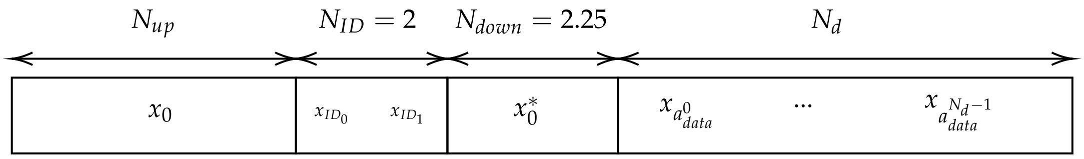

Figure 2. The legacy LoRa frame format.

The frame consists of a preamble followed by the payload symbols. The preamble is a critical component as it realizes the three following processes required to correctly demodulate the N_d payload symbols:

1. detecting the beginning of the frame by leveraging the N_{up} upchirps.
2. performing both frequency and time synchronization with the help of the N_{up} upchirps and N_{down} downchirps.
3. detecting if the received frame is dedicated to the receiver by checking if the $N_{ID} = 2$ consecutive network identification symbols correspond to its stored value.

LoRa transceivers generally use $N_{up} = 8$, a variable N_d value, and a fixed value $N_{down} = 2.25$. The number of symbols in the preamble and the entire frame are denoted, respectively, $N_{pre} = N_{up} + N_{down}$ and $N_{frame} = N_{pre} + N_{ID} + N_d$.

We choose to slightly change the frame format as depicted in Figure 3 with the following modifications:

1. Without loss of generality, the two identification symbols and the last quarter downchirp are ignored. The latter is not leveraged in the synchronization front-end. The symbol number in the frame then becomes $N_{frame} = N_{pre} + N_d$.
2. We also set the condition $N_{down} = N_{up}$. This enables a balanced noise immunity between the upchirps and downchirps as these are averaged during the synchronization procedure.

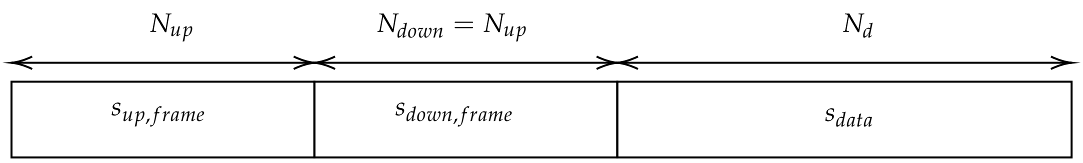

Figure 3. The modified self-jamming LoRa frame format.

The transmitted frame is then the concatenation of the upchirp, downchirp and payload symbol waveforms:

$$x[k] = s_{up,frame}[k] + s_{down,frame}[k - N_{up}M] + s_{data}[k - N_{pre}M] \tag{5}$$

3. Combat Basic LoRa Eavesdropper with Modified Preamble Waveform

A first approach to combat **E** is to only modify the preamble waveforms to disrupt its synchronization. A synchronization error will irredeemably lead to a demodulation error, preventing **E** from obtaining the critical data. The modified preamble waveforms are also designed to considerably increase the noise sensitivity for **E** and, thus, the discrete capacity of the scheme, while avoiding too much degradation of the performance of the link between **A** and **B**. The cooperative receiver leverages these modifications to improve its processing gain as much as possible.

The modified DFT preamble upchirp waveform in the preamble is illustrated in Figure 4. The green DFT bin depicts the legacy format. It consists of a unique DFT bin at known location $n = a_{up} = 0$, containing all the signal power $M\sqrt{P_s}$. The basic idea of the discrete scheme is to spread the power over several DFT bins with a uniform distribution in respect of a fixed power constraint. This is represented by the DFT bins with a dashed line in the figure. The modified preamble can be written as:

$$s_{up,frame}[k] = \sum_{i=0}^{N_{up}-1} s_{up}[k - iM] \tag{6}$$

$$s_{down,frame}[k] = \sum_{i=N_{up}}^{N_{pre}-1} s_{down}[k - iM] \tag{7}$$

with:

$$s_{up}[k] = \sqrt{P_J} \sum_{u=0}^{U-1} x_{(a_{up} - m_{up}^u) \mod M}[k] \tag{8}$$

$$s_{down}[k] = \sqrt{P_J} \sum_{u=0}^{U-1} x^*_{(a_{down} - m_{down}^u) \mod M}[k] \tag{9}$$

and U, the number of DFT bins present, P_J, the power level of each DFT bin with $P_J = P_s/U$, m_{up}^u and m_{down}^u, the u-th relative delay of the preamble upchirp and downchirp, respectively. We also note m_{up}, the associated delay vector that is sorted in ascending order, i.e., $m_{up}^0 = 0$ and $0 < m_{up}^{u>0} < M$. Each m_{up}^u delay must be unique to prevent a DFT bin overlapping issue, leading to adding DFT magnitudes and, thus, reducing the discrete capacity of the scheme. Note that $U = 1$ and $a_{up} = 0$ lead to the legacy format. The preamble downchirps follow the same structure but with a_{down} and m_{down} different from a_{up} and m_{up} to improve privacy.

Neglecting noise, the i-th received dechirped preamble upchirp or downchirp DFT is:

$$\tilde{R}_{up}[n] = \alpha M \sqrt{P_J} \sum_{u=0}^{U-1} \delta[n - (a_{up} - m_{up}^u) \mod M] \tag{10}$$

$$\tilde{R}_{down}[n] = \alpha M \sqrt{P_J} \sum_{u=0}^{U-1} \delta[n - (a_{down} - m_{down}^u) \mod M] \tag{11}$$

Note that each DFT bin has a null imaginary part. The DFT bin locations must remain secret from **E** to prevent its correct synchronization. a_{up}, m_{up}, a_{down} and m_{down} must then be random values that must be perfectly known by both **A** and **B**. That is, a specific procedure needs to be performed to satisfy this constraint. Possible solutions include the physical layer security schemes that leverage the randomness and reciprocity of the channel to enable both **A** and **B** to extract a pseudo-random bit sequence. These methods rely on the random received signal strength indicator (RSSI) variations, as LoRa transceivers have a built-in RSSI read-out feature, a solution chosen in [15], or using random channel path phase variation [18]. In practice, the **A** and **B** extracted sequences do not match perfectly and a reconciliation procedure is then necessary. This step requires the sequences exchange and may be vulnerable to eavesdroppers. The use of the Chinese remainder theorem (CRT), as in [15], or a code-word approach as in [19], are possible solutions to tackle this issue.

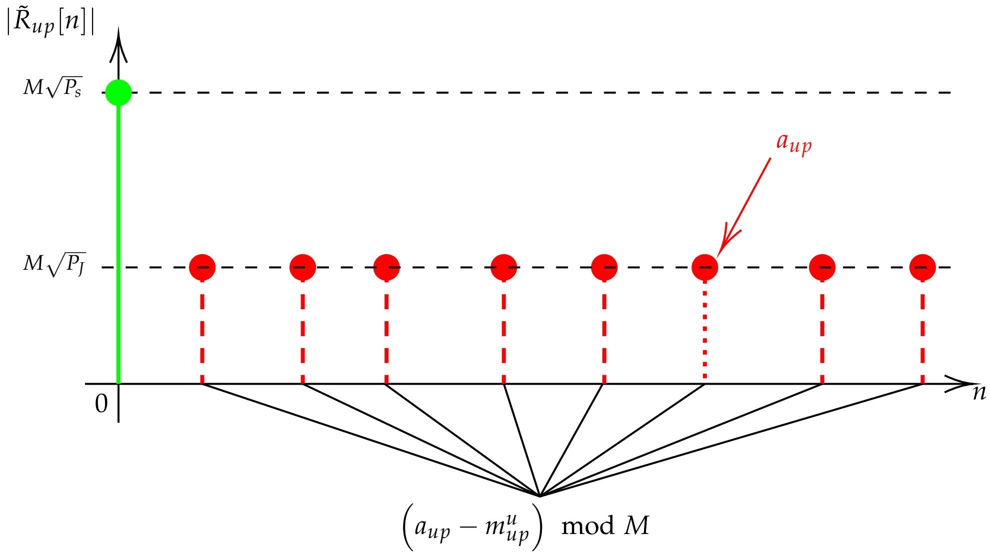

Figure 4. The modified preamble upchirp waveform.

4. Self-Jamming Synchronization Front-End

In this section, we introduce desynchronizations that a receiver undergoes in practice, their effects on the LoRa demodulation, and the synchronization front-end designed to address these issues.

4.1. Time Desynchronization Model—Sampling Time Offset (STO)

In real conditions, the receiver continuously collects chunks of M samples that are not necessarily aligned with the receiver, i.e., the sampling times are different between the transmitter and the receiver. This produces a temporal window shift τ up to a symbol period T, as depicted in Figure 5. This effect, referred to as the sampling time offset (STO), introduces inter-symbol interference (ISI) if the previous symbol is different from the current symbol, i.e., $a^- \neq a$ and $a \neq a^+$ in the figure. The higher the value of τ, the greater the ISI, with maximum signal deformation when $\tau \approx T/2$.

The preamble structure prevents ISI that could degrade synchronization performance, as consecutive upchirps and downchirps are identical (see Equations (8) and (9)). τ is modeled based on the LoRa sampling frequency $F_s = B$ and can then be converted to a certain number of sampling periods as:

$$\tau = \left(\underbrace{STO_{int} + STO_{frac}}_{STO} \right) \times T_s \tag{12}$$

with $STO_{int} = \lceil \tau/T_s \rfloor \in [0; M-1]$, the integer number of sampling periods plus a fraction of a sampling period $STO_{frac} = STO - STO_{int} \in [-0.5; 0.5)$. $\lfloor . \rceil$ denotes the rounding operation to the nearest integer.

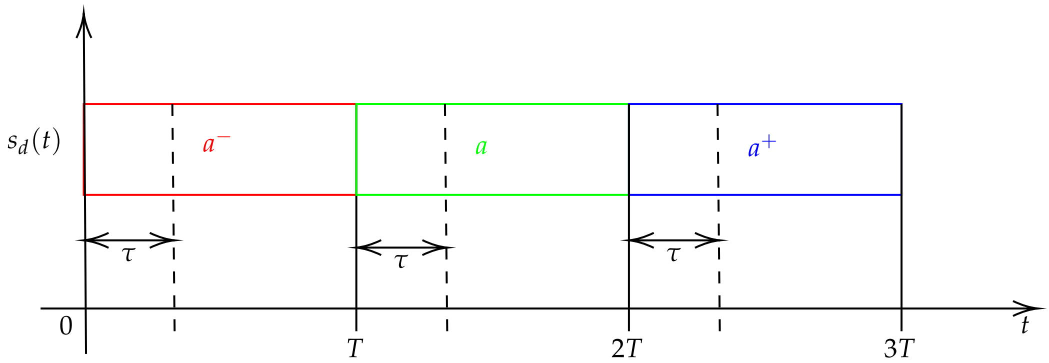

Figure 5. Illustration of the STO effect.

4.2. Frequency Desynchronization Model

Due to hardware imperfections, other desynchronizations may occur in the frequency domain, such as the carrier-frequency offset (CFO) and the sampling-frequency offset (SFO).

4.2.1. Carrier-Frequency Offset (CFO)

As a reminder, the CFO is the residual carrier frequency present in the base-band signal at the receiver side. The local oscillators of the transmitter and the receiver are not perfectly centered to the desired carrier frequency F_c. A residual frequency appears, then:

$$\Delta_f = F_c^t - F_c^r \tag{13}$$

with F_c^t (resp. F_c^r), the carrier frequency used by the transmitter (resp. the receiver). By analogy to the STO, Δ_f can be converted to a number of frequency bins:

$$\Delta_f = \left(\underbrace{CFO_{int} + CFO_{frac}}_{CFO} \right) \times \frac{B}{M} \tag{14}$$

with $CFO_{int} = \lfloor \Delta_f/(B/M) \rfloor \in [0; M-1]$, the integer number of DFT bins plus a fraction of a DFT bin $CFO_{frac} = CFO - CFO_{int} \in [0; 1)$. $\lfloor . \rfloor$ denotes the floor operation.

4.2.2. Sampling-Frequency Offset (*SFO*)

The *SFO* is a mismatch between the current and the desired sampling frequency at the receiver side:

$$F_s' = F_s + SFO \tag{15}$$

In hardware implementation, and especially for low-cost IoT transceivers, such as LoRa, the same oscillator is used to perform the sampling and the carrier transposition. That is, the CFO and *SFO* are generated from the same source and their relationship represented as follows [20]:

$$SFO = \frac{B}{F_c} \times \Delta_f \tag{16}$$

4.3. Time and Frequency Desynchronization Effects on LoRa

CFO_{int} and STO_{int} have the effect of shifting the DFT bin position (we consider $U = 1$ for the sake of simplicity) by a certain amount that is different when considering either upchirps: $\widehat{a}_{up} = (a_{up} + \lfloor CFO + STO \rceil) \mod M$ or downchirps: $\widehat{a}_{down} = (a_{down} + \lfloor CFO - STO \rceil) \mod M$. The fractional part CFO_{frac} and STO_{frac} progressively spread the DFT bin of interest energy to its neighbor as CFO_{frac} or STO_{frac} gets closer to 0.5: $n = a_{up} + 1$ and $n = a_{down} - 1$ for CFO; STO has the opposite behavior.

The SFO has the consequence, over time, of progressively distorting the received signal; a discrete model for LoRa is derived in [21] (considering upchirp symbols, for example, neglecting noise and channel path gains):

$$\tilde{r}_i[k] \approx \tilde{x}_{a_i}[k] e^{2j\pi ki\left[\left(\frac{B}{Fs'}\right)^2 - \frac{B}{F_s'}\right]} \tag{17}$$

with $\tilde{x}_{a_i}[k]$, the i-th received LoRa signal with symbol value a_i.

4.4. Synchronization Scheme

The adapted state-of-the-art LoRa synchronization front-end of our self-jamming scheme is presented in Figure 6. The front-end starts with a first pre-processing block which involves sampling the received signal at an over-sampled rate $R \times F_s$, dechirping N_{up} blocks of M samples (downsampled by R factor), estimating and correcting CFO_{frac} for these N_{up} blocks, and computing the N_{up} corrected DFTs. The receiver continues with the preamble detection as, in practice, the latter operates in real time.

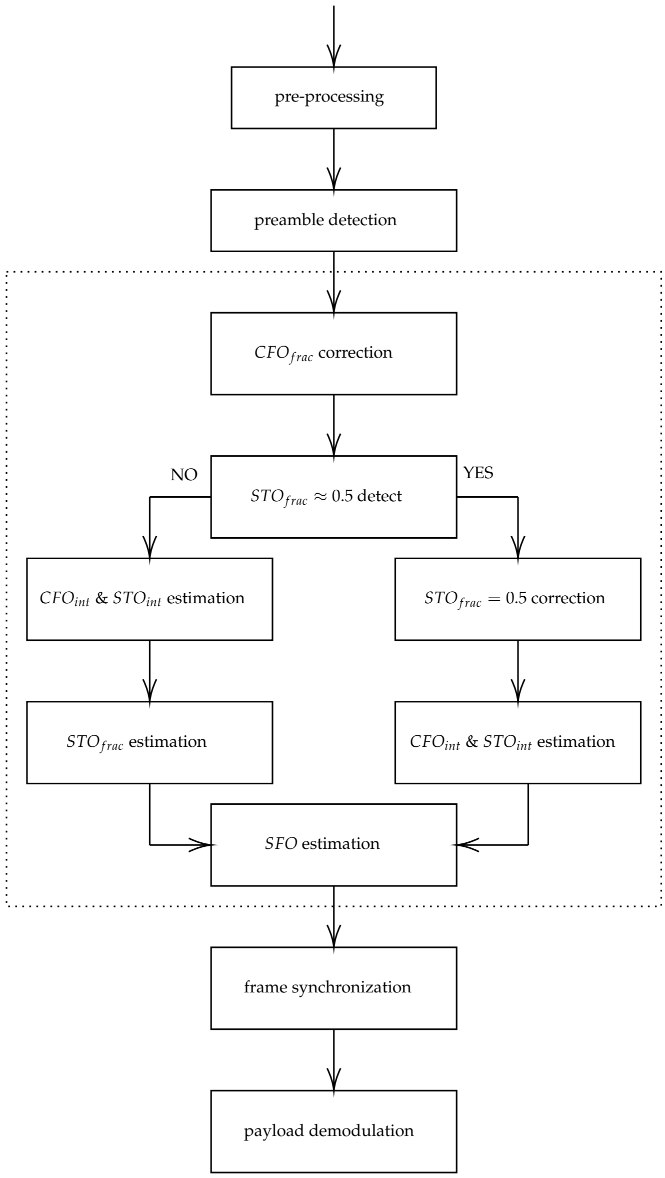

Figure 6. Illustration of the LoRa synchronization front-end adapted to the self-jamming scheme.

Once the preamble is detected, the receiver re-aligns the symbols in the detected frame by CFO_{frac} and estimates the other synchronization parameters, i.e., CFO_{int}, SFO, STO_{int} and STO_{frac}. The estimation of both CFO and STO is not trivial. As their effects are not independent of each other, the pipeline must then be designed wisely. It finally performs a frame correction to re-align itself in time and frequency. The over-sampling by the R rate is required to mitigate STO_{frac}.

4.4.1. Fractional CFO Correction and Preamble Detection

CFO_{frac} can be estimated and compensated in this step. As the CFO_{frac} estimator found in [22] has low sensitivity to the presence of multiple DFT peaks and operates blindly, we choose then to use this estimator. To ensure correct CFO_{frac} estimation, no energy other than AWGN must be present in the left and right adjacent DFT bins of each of the U DFT peaks. We set the constraint of choosing delays with a minimal gap of ϵ DFT positions between each. This is also valid for proper STO_{frac} estimation. Satisfying the constraint ϵ, the maximum number of virtual paths U value is:

$$U_{max} = \left\lfloor \frac{M}{\epsilon} - 1 \right\rceil \tag{18}$$

giving $U_{max} = 25$ for $\epsilon = 5$ and $SF = 7$, for example. In [22], the authors proposed an estimator that relies on the well-known three spectral lines (TSL) scheme by deriving $\widehat{CFO}_{frac}$ over N_{up} consecutive symbols. Each N_{up} received desynchronized symbol $y_i[k]$ is then corrected:

$$y'_i[k] = y_i[k]e^{-2j\pi k \frac{\widehat{CFO}_{frac}}{M}} \tag{19}$$

The preamble detection relies on detecting the presence of consecutive demodulated symbols. With very low AWGN and a well-aligned received signal, N_{up} identical and consecutive symbols should be detected but the noise progressively introduces errors and, in practice, it is very difficult to detect this specific pattern. To improve the detection performance at the cost of an increased false alarm rate, we set the constraint to detect at least L consecutive symbols having a maximum value difference of ± 1.

Due to the presence of multiple DFT peaks of the same magnitude, the classic demodulation scheme in (4) is not suitable as the detected DFT peak location will change over the N_{up} upchirps. To tackle this issue, we propose a cross-correlation approach. As the relative delays m_{up} are perfectly known by the receiver, the latter can rebuild locally the expected dechirped preamble upchirp with assumed transmitted power $P_s = 1$. This is denoted $\tilde{S}^{ref}_{up}[n]$. Then, for L consecutive received dechirped symbols, it computes the circular cross-correlation and extracts the maximum argument:

$$\begin{aligned} F'_{up,l}[m] &= \sum_{n=0}^{M-1} \left|\tilde{S}^{ref}_{up}[n]\right| \left|\tilde{Y}'_l[(n-m) \mod M]\right| & (20) \\ n_l &= \arg\max_m \; F'_{up,l}[m] & (21) \end{aligned}$$

with $p \le l \le p + (L-1)$, $p = \{0, 1, \dots, p_{max}\}$, $0 \le m \le M-1$ and $\tilde{Y}'_l[n]$, the DFT of $\tilde{y}'_l[k]$. Note that p_{max} is the last block of L demodulated symbols until preamble detection. Equation (20) can be efficiently computed with a fast Fourier transform (FFT) algorithm as:

$$F'_{up,l} = IFFT\left(FFT\left(\left|\tilde{S}^{ref}_{up}\right|\right) \times \left\{FFT\left(\left|\tilde{Y}'_l\right|\right)\right\}^*\right) \tag{22}$$

The preamble is detected if $(n_{p+i} + j) \mod M = n_p$ for $i = \{1, 2, \dots, L-1\}$ and $j = \{-1, 0, 1\}$. Once the preamble is detected, the rest of the symbols in the frame are corrected by $\widehat{CFO}_{frac}$.

4.4.2. Half Fractional STO Detection

As previously stated in Section 4.3, as STO_{frac} gets closer to 0.5, the neighbor DFT bin energy progressively increases, leading to higher noise sensitivity. When $STO_{frac} \approx 0.5$, two DFT peaks with almost the same magnitude are present, creating detection uncertainty and preventing correct CFO_{int} and STO_{int} estimation. That is, STO_{frac} must be mitigated before, independently from CFO_{int} and STO_{int}. The authors in [23] proposed a solution by performing an initial STO_{frac} mitigation, albeit partial, to remove this uncertainty.

We propose a different approach with a binary statistical test by detecting if $STO_{frac} \approx 0.5$. We define the hypotheses H_0, H_1 as $STO_{frac} \neq 0.5$ and $STO_{frac} = 0.5$, respectively. The basic idea is to evaluate the DFT magnitude difference between the peak of interest and its neighbor bin. The less the difference, the closer to 0.5 STO_{frac}. Below a certain difference threshold, the receiver decides H_1, otherwise H_0. The detector is designed as follows:

1. The N_{up} preamble upchirp DFTs are averaged to reduce noise sensitivity:

$$\left\langle \tilde{Y}'_{up}[n] \right\rangle = \frac{1}{N_{up}} \sum_{i=0}^{N_{up}-1} \tilde{Y}'_i[n] \tag{23}$$

2. The following cyclic cross-correlation is computed and normalized:

$$F'_{up}[m] = \sum_{n=0}^{M-1} \left|\tilde{S}^{ref}_{up}[n]\right| \left|\left\langle \tilde{Y}'_{up}[(n-m) \mod M] \right\rangle\right| \tag{24}$$

$$F'_{up}[m] = \frac{F'_{up}[m]}{\max\limits_{m} \quad F'_{up}[m]} \tag{25}$$

3. We extract the left and right neighbor DFT bin magnitudes of the maximum DFT peak and compute the criterion δ:

$$n^{up}_{max} = \arg\max_{m} \quad F'_{up}[m] \tag{26}$$

$$\begin{aligned} v^- &= F'_{up}[(n^{up}_{max} - 1) \mod M] \\ v^+ &= F'_{up}[(n^{up}_{max} + 1) \mod M] \end{aligned} \tag{27}$$

$$\delta = 1 - \max \quad (v^-, v^+) \tag{28}$$

4. $STO_{frac} \approx 0.5$ is finally detected as:

$$\delta \underset{H_1}{\overset{H_0}{\gtrless}} \lambda_{STO_{frac} \approx 0.5} \tag{29}$$

The frame contaminated by STO_{frac} is then corrected with $\widehat{STO}_{frac} = 0.5$ (if detected) by discarding the first $R \times (M - STO_{frac})$ samples. There are then $N_{up} - 1$ upchirp symbols in the preamble.

Figure 7 illustrates the evolution of averaged δ, denoted $\langle\delta\rangle$, as a function of $STO_{frac} = \{0, 0.1, \ldots, 0.9\}$ ($R = 10$) for several SNR values $SNR_{dB} = \{-15, -12, -9, -6\}$, $U = 4$ and $SF = 7$. The delays m_{up} are chosen randomly and uniformly in $[0; M-1]$ and satisfying the gap ϵ constraint.

We can see from the figure that $\langle\delta\rangle$ progressively decreases as STO_{frac} gets closer to 0.5 with the minimal point reached for $STO_{frac} = 0.5$. $\langle\delta\rangle$ has a symmetric pattern with $STO_{frac} = 0.5$. The noise has the effect of flattening the curve, reducing the contrast between STO_{frac} values. The threshold $\lambda_{STO_{frac} \approx 0.5}$ must be chosen wisely. A low value will increase the non-detection probability, a situation that must be avoided as far as possible. A very high value will lead to almost constant detection; the corrected frame will then have as many as STO_{frac} residuals with no $STO_{frac} \approx 0.5$ detection enabled.

In simulations, $\lambda_{STO_{frac}\approx 0.5} = 0.3$ is a balanced value for the LoRa SNR range of interest $SNR_{dB} = \{-15, -14, \ldots, -5\}$. We note that adjacent values $STO_{frac} = \{0.4, 0.6\}$ are almost constantly detected as $STO_{frac} = 0.5$, but the residual is ± 0.1, a value that has a negligible impact on demodulation performance.

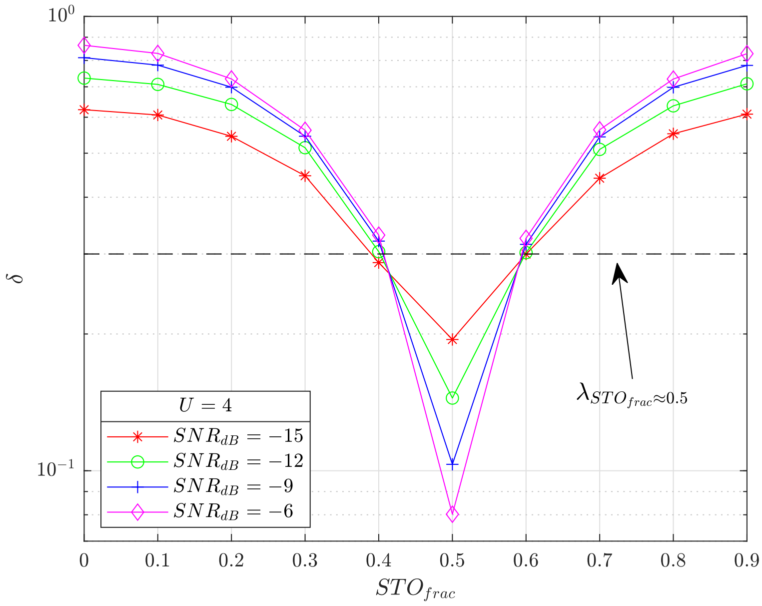

Figure 7. Evolution of the average value of the criterion $\langle \delta \rangle$ as a function of $STO_{frac} = \{0, 0.1, \ldots, 0.9\}$ for several SNR values $SNR_{dB} = \{-15, -12, -9, -6\}$, $U = 4$ and $SF = 7$.

Figure 8 illustrates the histograms of δ for $STO_{frac} = \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$, $U = 4$, $SNR_{dB} = -8$ and $SF = 7$. We note that the δ statistic follows a near-Gaussian distribution as the computed cross-correlation is a sum of Rayleigh random variables (RV). With extensive simulation results, we note that this distribution is slightly U dependent. Furthermore, increasing SF results in similar histograms but for lower SNRs, and the derived histogram for $STO_{frac}^{sym} = 1 - STO_{frac}$ is nearly the same as for STO_{frac} (symmetry).

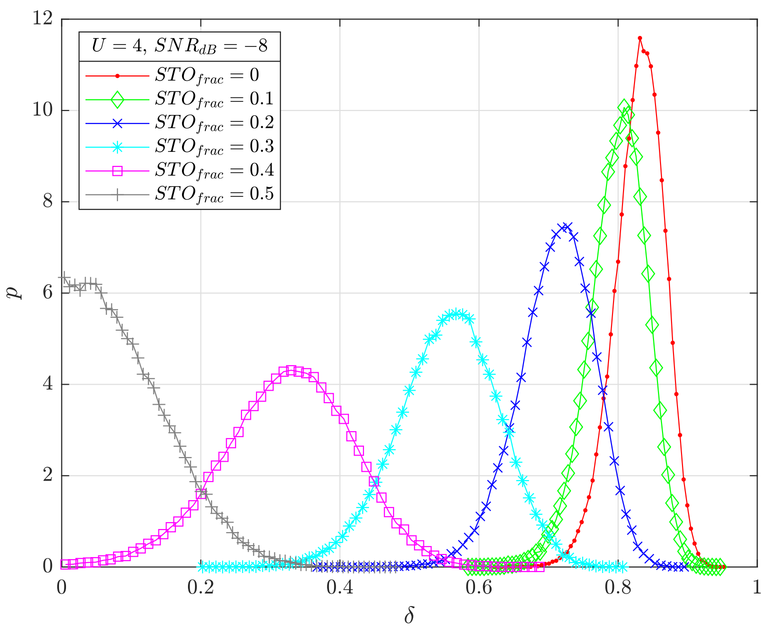

Figure 8. δ histograms as a function of $STO_{frac} = \{0, 0.1, \ldots, 0.5\}$ for $U = 4$, $SNR_{dB} = -8$ and $SF = 7$.

4.4.3. CFO and STO Integer Estimation

The next step in the synchronization front-end is to estimate CFO_{int} and STO_{int}. The process follows the same philosophy as so far applied to the cross-correlation approach. The receiver keeps the previously computed n^{up}_{max} in (26) and performs steps (23), (24), (26) for the preamble downchirps to derive n^{down}_{max}. CFO_{int} and STO_{int} are simply derived as:

$$\widehat{CFO}_{int} = \left\lfloor \frac{(n^{up}_{max} + n^{down}_{max}) \mod M}{2} \right\rfloor \tag{30}$$

$$\widehat{STO}_{int} = (a_{up} + n^{up}_{max} - \widehat{CFO}_{int}) \mod M \tag{31}$$

The SFO is simply derived as:

$$\widehat{SFO} = (\widehat{CFO}_{int} + \widehat{CFO}_{frac}) \times \frac{B^2}{M \times F_c} \tag{32}$$

As stated in [23], this synchronization scheme cannot correctly detect $CFO_{int} \geq M/4$ but, in practice, it is very unlikely to have such a high value.

4.4.4. Fractional STO Part Estimation

The final step is to estimate STO_{frac} in the case where $STO_{frac} \approx 0.5$ has not been detected earlier. The scheme is based on the TSL approach proposed in [23] but with slight modifications to be functional with our self-jamming scheme. The main steps are summarized in what follows:

1. The averaged preamble DFT upchirps $\left\langle \tilde{Y}'_{up}[n] \right\rangle$ are re-aligned by removing $\widehat{CFO}_{int}$ and $\widehat{STO}_{int}$ shifts. This is simply effected by performing a left circular permutation.
2. For each of the U DFT peaks in $\left\langle \tilde{Y}'_{up}[n] \right\rangle$, we extract its value and the left and right neighbor bins as:

$$w_{c,u} = \left\langle \tilde{Y}'_{up}[(a_{up} - m^u_{up} + c) \mod M] \right\rangle, \quad c \in \{-1, 0, 1\} \tag{33}$$

3. STO_{frac} is finally averaged over U estimates as:

$$\widehat{STO}_{frac} = \frac{1}{U} \sum_{u=0}^{U-1} -\Re\{\Pi_u\} \tag{34}$$

with:

$$\Pi_u = \frac{e(-h_u)w_{1,u} - e(h_u)w_{-1,u}}{2 \times w_{0,u} - e(-h_u)w_{1,u} - e(h_u)w_{-1,u}} \tag{35}$$

$$h_u = (\widehat{STO}_{int} + a_{up} - m^u_{up}) \mod M \tag{36}$$

$$e(x) = e^{2j\pi \frac{x}{M}} \tag{37}$$

5. EVE Blind Synchronization Threat

With this modified preamble structure, **E** cannot synchronize itself correctly without the knowledge of a_{up}, a_{down}, m_{up} and m_{down}. The synchronization error heavily impacts the payload demodulation stage and then prevents **E** from eavesdropping. In this section, we evaluate the ability of **E** to blindly estimate synchronization parameters that would possibly threaten the sustainability of our scheme.

As previously stated, CFO_{frac} can be blindly estimated by both **B** and **E**. However, **E** cannot synchronize itself if CFO is still present after CFO_{frac} correction, i.e., $CFO_{int} \neq 0$. That is, **E** has the ability to blindly estimate STO_{int} only if $CFO_{int} = 0$. This situation may happen if **E** is a higher-end device with low hardware impairments and, thus, $CFO < 1$.

In what follows, we present a blind method to extract STO_{int}. The basic idea is to leverage the fact that the STO introduces ISI only between the last upchirp and the first downchirp in the preamble. Then, **E** can use a STO_{int} candidate approach by computing an energy cost for each candidate and selecting the one that minimizes the cost function. We denote each STO_{int} candidate by $STO_{int}^{cand} \in \{0, 1, \ldots, M-1\}$. The blind extraction method is designed as follows:

1. **E** generates a temporary replica of the received frame and voluntarily simulates a STO with value STO_{int}^{cand} by discarding the first $R \times STO_{int}^{cand}$ samples, consequently modifying the time window process. It is denoted as $y'_{cand}[k]$.
2. It then dechirps, computes the DFT magnitude of the last preamble upchirp and the first preamble downchirp to derive the following quantities:

$$\gamma_{up}^{STO_{int}^{cand}} = \frac{1}{M}\sum_{n=0}^{M-1}\left|\tilde{Y}'_{cand,N_{up}-2}[n]\right| \tag{38}$$

$$\gamma_{down}^{STO_{int}^{cand}} = \frac{1}{M}\sum_{n=0}^{M-1}\left|\tilde{Y}'_{cand,N_{up}-1}[n]\right| \tag{39}$$

To construct the minimum cost function point at $STO_{int}^{cand} = STO_{int}$, **E** needs to add a left circular permutation of one position to $\gamma_{up}^{STO_{int}^{cand}}$. The cost function is simply derived as:

$$\begin{aligned} \gamma^{STO_{int}^{cand}} &= \gamma_{up}^{STO_{int}^{cand}} + \gamma_{down}^{STO_{int}^{cand}} \\ \gamma^{STO_{int}^{cand}=M-1} &= \max_{STO_{int}^{cand}} \gamma^{STO_{int}^{cand}} \end{aligned} \tag{40}$$

3. STO_{int} is finally estimated as:

$$\widehat{STO}_{int} = \underset{STO_{int}^{cand}}{\arg\min} \quad \gamma^{STO_{int}^{cand}} \tag{41}$$

This blind scheme has the drawback of being unable to correctly estimate $STO_{int} = M-1$ value, slightly increasing the STO_{int} estimation error. Moreover, STO_{frac} progressively increases the estimation error as it gets closer to 0.5, as highlighted in Section 7. If **E** has correctly estimated STO_{int}, it can easily estimate STO_{frac} even without a_{up} and m_{up} knowledge in (36). **E** can select the DFT bins that are above a given threshold ρ_E in $\left\langle \tilde{Y}'_{up}[n] \right\rangle$ (23) with:

$$\rho_E = \lambda_E \times \max_n \quad \left|\left\langle \tilde{Y}'_{up}[n] \right\rangle\right|, \quad \lambda_E \in \ ]0;1] \tag{42}$$

The derived DFT bin positions set $\mathcal{A}_E$ should correspond to $(a_{up} - m_{up}) \mod M$ and, thus, $|\mathcal{A}_E| = U$ in high SNR conditions, then enabling an identical STO_{frac} estimation performance to the legitimate receiver if $CFO < 1$. In such conditions, **E** successfully passes the synchronization front-end and can demodulate and retrieve the information in the payload.

We conclude that modification of the preamble only is necessary but not sufficient to ensure a discrete communication. A solution to tackle this more advanced **E** is then to also modify the payload waveform and is presented in the next section.

6. Combat Advanced LoRa Eavesdropper with Modified Payload Waveform

The payload waveform is modified with the same structure as for the preamble. This has the advantage of reducing scheme knowledge leaks, i.e., preamble symbols a_{up}, a_{down}, and delays m_{up} and m_{down}. The modified payload waveform is then:

$$s_{data}[k] = \sum_{d=0}^{N_d-1} s_{data}^{(d)}[k - (N_{pre} + d) \times M] \quad (43)$$

with:

$$s_{data}^{(d)}[k] = \sqrt{P_J} \sum_{u=0}^{U-1} x_{(a_{data}^{(d)} - l_d - m_{data}^{d,u}) \mod M}[k] \quad (44)$$

with l_d, a random shift (unknown by $\mathbf{E}$) applied to the d-th payload symbol, $m_{data}^{d,u}$ the u-th relative delay of the d-th payload symbol $a_{data}^{(d)}$. We note $m_{data}^{(d)}$ the delay vector of the d-th payload symbol. Each $m_{data}^{(d)}$ may be different between payload symbols to improve privacy. Again, the receiver may use the same legacy cross-correlation approach to demodulate the payload symbol. However, the latter has the drawback of increasing interference peak magnitudes in (20) as U grows. This reduces the AWGN immunity and degrades the symbol detection performance.

We propose a modified cross-correlation implementation, denoted as mod cross-corr, that considerably mitigates this detrimental effect. Considering perfect synchronization, it consists of dechirping the received symbol $r_{data}^{(d)}[k] = s_{data}^{(d)}[k] + w[k]$ over multiple downchirp symbols instead of the unique downchirp $x_0^*[k]$:

$$\tilde{r}_{data}^{(d)}[k] = \sum_{u=0}^{U-1} r_{data}^{(d)}[k] x^*_{(-m_{data}^{d,u} - l_d) \mod M}[k] \quad (45)$$

The symbol is still estimated in the frequency domain:

$$\hat{a}_{data}^{(d)} = \arg\max_n \left| \tilde{R}_{data}^{(d)}[n] \right| \quad (46)$$

To compare the legacy and the modified cross-correlation, we define the following criterion for the modified cross-correlation:

$$\eta_{mod\ cross-corr} = \frac{\left| \tilde{R}_{data}^{(d)}[a_{data}^{(d)}] \right|}{\frac{1}{M-1} \sum_{\substack{0 \le n \le M-1 \\ n \ne a_{data}^{(d)}}} \left| \tilde{R}_{data}^{(d)}[n] \right|} \quad (47)$$

and for the legacy cross-correlation:

$$\eta_{cross-corr} = \frac{F_{data}^{(d)}[a_{data}^{(d)}]}{\frac{1}{M-1} \sum_{\substack{0 \le m \le M-1 \\ m \ne a_{data}^{(d)}}} F_{data}^{(d)}[m]} \quad (48)$$

This represents the average magnitude difference between the DFT peak of interest and the interference peaks (AWGN plus cross-correlation peaks).

Figure 9 compares average η between the legacy and the modified cross-correlations as a function of $SNR_{dB} \in \{-15, -14, \ldots, -6\}$ for several $U = \{1, 2, \ldots, 10\}$. We assume perfect synchronization and delays chosen randomly, respecting the ϵ constraint.

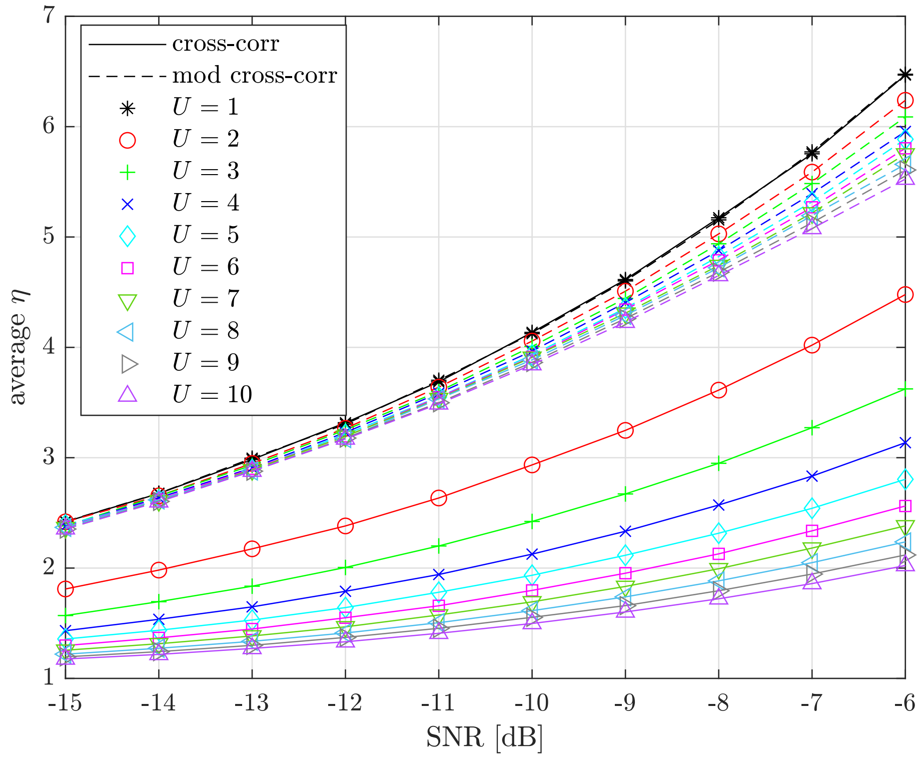

Figure 9. U sensitivity comparison between the legacy and the modified cross-correlation schemes, $SF = 7$.

We can see from the figure that $U = 1$ has a maximum and same average η between cross-corr and mod cross-corr as it is equivalent to the LoRa legacy demodulation scheme (4). It behaves as an upper limit as the higher average η, the higher the magnitude difference, and the better the performance. We also note that mod cross-corr has much lower U sensitivity. The loss between $U = 1$ and $U = 10$ is $\frac{6.475}{2.023} \approx 3.20$ for cross-corr against $\frac{6.475}{5.525} \approx 1.17$ for mod cross-corr at $SNR_{dB} = -6$. This solution is only sustainable if the STO has been correctly mitigated as would normally be the case when demodulating the payload. This modified cross-correlation is not suitable for synchronization parameter estimation as a candidate STO_{int} approach is required (similar to the blind STO_{int} estimation procedure) that gives poor synchronization performance.

Table 2 summarizes the parameters of our complete self-jamming scheme that the legitimate and eavesdropper receivers know, do not know, or must be kept secret from **E**, estimated with self-jamming scheme knowledge and blindly estimated. The symbols used in the table are described in Table 3. For conciseness, parameters which depend on others are not shown, e.g., $M = 2^{SF}$.

Note that, from the table, the only parameter that is identically estimated by the legitimate receiver and the eavesdropper is CFO_{frac}. Furthermore, **E** can blindly estimate the STO and retrieve U under the right conditions (see Section 5). However, the critical payload parameters $m_{data}^{(d)}$ and l_d are almost impossible to retrieve for **E** without using a brute-force approach, making proper demodulation very difficult.

Table 2. LoRa self-jamming scheme parameters supposed to be known, unknown, kept secret from **E**, estimated with self-jamming scheme knowledge and blindly estimated by the legitimate or eavesdropper receivers.

Self-Jamming Scheme Parameter	**A or B**	**E**
LoRa parameters		
SF	$\star$	$\star$
F_c, B	$\star$	$\star$
preamble waveform parameters		
N_{up}, N_{down}, N_d	$\star$	$\star$
a_{up}, a_{down}	$\star$	$\circ$
m_{up}, m_{down}	$\star$	$\circ$
payload waveform parameters		
$m_{data}^{(d)}$, l_d	$\star$	$\circ$
$a_{data}^{(d)}$	$\square$	$\circ$
global self-jamming parameters		
U	$\star$	$\triangle$
ϵ	$\star$	$\circ$
synchronization parameters		
L	$\star$	$\star$
$\lambda_{STO_{frac}\approx 0.5}$	$\star$	$\circ$
CFO_{int}	$\square$	$+$
CFO_{frac}	$\triangle$	$\triangle$
SFO	$\square$	$+$
STO_{int}, STO_{frac}	$\square$	$\triangle$

Table 3. Symbols meaning of symbols used in Table 2.

Symbol	**Symbol Meaning**
$\star$	known
$+$	unknown
$\circ$	kept secret from **E**
$\square$	unknown and estimated with self-jamming scheme knowledge
$\triangle$	unknown and blindly estimated

7. Simulation Results

In this section, we present several simulation results to assess the self-jamming scheme. The following parameters are used, if not stated:

- $SF = 7$, $M = 128$
- $N_{up} = N_{down} = 8$
- $L = 3$
- $R = 10$
- $F_c = 868$ MHz, $B = 125$ kHz
- $CFO \in \mathcal{U}[0.1; M/4 - 1 = 31]$
  We assume that $CFO < 0.1$ is very unlikely to happen in practice.
- $STO \in \mathcal{U}[0; M - 1]$
- $|\alpha| = 1$, $\phi \in \mathcal{U}[0; 2\pi]$
- $P_s = 1$, $P_J = P_s/U = 1/U$
- $\lambda_{STO_{frac}\approx 0.5} = 0.3$
- $\epsilon = 5$

7.1. Preamble Detection Performance

As **E** does not have a_{up} and m_{up} knowledge, the only possible preamble detection scheme for **E** is to compute the cross-correlation between two consecutive symbols as:

$$F'_{up,l,\mathbf{E}}[m] = \sum_{n=0}^{M-1} \left|\tilde{Y}'_l[n]\right| \left|\tilde{Y}'_{l+1}[(n-m) \mod M]\right| \quad (49)$$

$$n_{l,\mathbf{E}} = \arg\max_m \; F'_{up,l,\mathbf{E}}[m] \quad (50)$$

with $p \leq l \leq p + (L-1)$ and $p = \{0, 1, \ldots, p_{max}\}$. **E** also searches L consecutive symbols in $n_{l,\mathbf{E}}$ with value difference ± 1 to detect the preamble.

A and **B** also have the ability to use the modified cross-correlation to improve the preamble detection performance. However, as stated in Section 6, this approach does not demonstrate satisfactory performance if the STO is not mitigated. The preamble detection can only be performed in the presence of STO. That is, an STO_{int} candidate approach must be leveraged with the same philosophy as the blind STO_{int} estimation performed by **E** (see Section 5). To save computation resources, the candidate selection is only performed on the p-th received symbol and kept for the $L-1$ remaining symbols. The modified preamble detection scheme is:

1. **A** or **B** generates a temporary replica of the received frame and voluntarily simulates an STO with value STO_{int}^{cand} by discarding the first $R \times STO_{int}^{cand}$ samples, consequently modifying the time window process. It is denoted as $y'_{cand}[k]$.
2. It then computes the modified cross-correlation of the i-th received symbol and selects the maximum value for each STO_{int} candidate as:

$$\tilde{r}_{up,l=p}^{STO_{int}^{cand}}[k] = \sum_{u=0}^{U-1} y'_{cand,l=p}[k] x^*_{-m^u_{up}}[k] \quad (51)$$

$$v_{max,l=p}^{STO_{int}^{cand}} = \max_n \left|\tilde{R}_{up,l=p}^{STO_{int}^{cand}}[n]\right| \quad (52)$$

3. The candidate is selected as:

$$STO_{int}^{cand,sel} = \arg\max_{STO_{int}^{cand}} \; v_{max,l=p}^{STO_{int}^{cand}} \quad (53)$$

4. It then selects the maximum argument for each computed modified cross-correlation ($p \leq l \leq p + (L-1)$) associated with the chosen candidate:

$$n_l = \arg\max_n \left|\tilde{R}_{up,l}^{STO_{int}^{cand}=STO_{int}^{cand,sel}}[n]\right| \quad (54)$$

Figure 10 presents the preamble detection performance comparison between the legitimate receiver and **E** as a function of $SNR_{dB} = \{-15, -14, \ldots, 0\}$ for several $U = \{1, 2, 3, 4, 8, 10, 12\}$ and $SF = 7$. We also add the comparison between the legacy and the modified cross-correlation methods.

We can see from the figure that the preamble detection performance progressively decreases when U increases, even when using modified cross-correlation. This is because the same chosen STO_{int} candidate is used for all the symbols in the block of L received symbols. That is, increasing U increases the error probability to $STO_{int}^{cand,sel} \neq STO_{int}$. This error propagates on all symbols and the probability of detecting L consecutive symbols with value difference ± 1 then decreases.

For $U \leq 3$, the legacy and modified cross-correlation schemes have similar preamble detection performance, with a slight advantage for the modified cross-correlation method. However, for higher U, the modified cross-correlation scheme progressively outperforms the legacy cross-correlation scheme as U grows, with a performance difference of about 2 dB and a detection probability of 0.5 and $U = 12$. Note that the modified cross-correlation performance is almost the same for $U = \{8, 10, 12\}$.

E has much lower performance with a loss ≈4 dB between $U = 1$ and $U = 12$, with a detection probability of 0.5 and a loss ≥ 3 dB when compared to the legitimate receiver using the modified cross-correlation scheme, for a given U. **E** is much more prone to AWGN errors as the cross-correlation performed in (49) has two sources containing AWGN, while the reference upchirp in (20) is AWGN free.

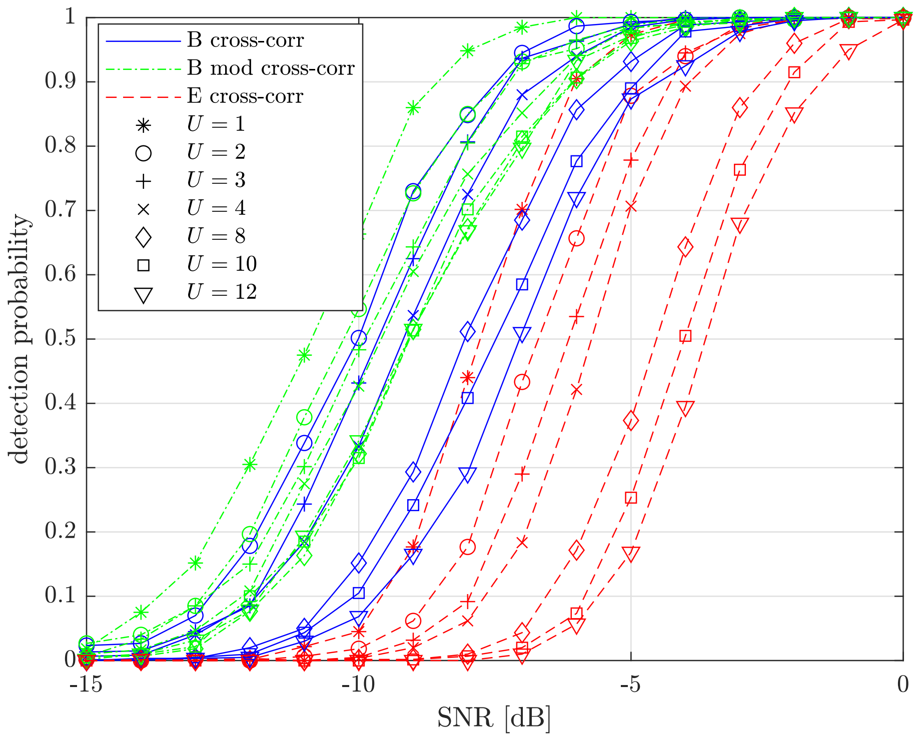

Figure 10. Preamble detection performance comparison between **B** and **E** for $U = \{1, 2, 3, 4, 8, 10, 12\}$, $SNR_{dB} = \{-15, -14, \ldots, 0\}$ and $SF = 7$. **B** can use both the legacy and the modified cross-correlation methods, while **E** is restricted to blindly detecting the preamble with the legacy cross-correlation scheme only.

7.2. Complexity Comparison between the Legacy and the Modified Cross-Correlation Methods

The considerably reduced U sensitivity of modified cross-correlation (see Section 6) is at the cost of increased complexity. The algorithms for both the legacy and the modified cross-correlation functions are provided in Algorithms 1 and 2.

Algorithm 1: Legacy cross-correlation algorithm

---

inputs : $\mathbf{r_i}$: the i-th received symbol vector
$\mathbf{m}$: the delays vector
$\mathbf{x_{ref}}$: the reference downchirp or upchirp vector
M: the constellation size
output: s: the maximum peak index of the legacy cross-correlation

$\tilde{\mathbf{R}}_\mathbf{i} := \mathrm{abs}(\mathrm{FFT}(\mathbf{r_i} \odot \mathbf{x_{ref}}))$
$\tilde{\mathbf{S}}_\mathbf{ref} := \mathbf{0}_M$ %init M-size vector
$\tilde{\mathbf{S}}_\mathbf{ref}[-\mathbf{m} \mod M] := M\sqrt{P_J}$
$\mathbf{F_i} := \mathrm{IFFT}(\mathrm{FFT}^*(\tilde{\mathbf{S}}_\mathbf{ref}) \odot \mathrm{FFT}(\tilde{\mathbf{R}}_\mathbf{i}))$
return $s = \arg\max(\mathbf{F_i})$

It is obvious that the legacy cross-correlation in Algorithm 1 does not depend on U; it then requires the same amount of operations irrespective of the U. However, in Algorithm 2, lines 2–4, U complex sums of M elements are required. That is, increasing U increases the complexity.

Algorithm 2: Modified cross-correlation algorithm

inputs : $\mathbf{r_i}$: the i-th received symbol vector
$\mathbf{m}$: the delays vector
M: the constellation size
output: s: the maximum peak index of the modified cross-correlation

$\mathbf{f_i} := \mathbf{0}_M$
for $u = 0$ *to* $U - 1$ **do**
  $\mathbf{z} := \mathbf{x}^*_{-\mathbf{m}[\mathbf{u}]}$ % LoRa downchirp with symbol value $-m[u]$
  $\mathbf{f_i} := \mathbf{f_i} + \{\mathbf{r_i} \odot \mathbf{z}\}$
$\mathbf{F_i} := \mathrm{abs}(\mathrm{FFT}(\mathbf{f_i}))$
return $s = \arg\max(\mathbf{F_i})$

This behavior is highlighted in Figure 11. We execute and report the execution times of C compiled versions of Algorithms 1 and 2 in a MATLAB environment, with $SF = 7$.

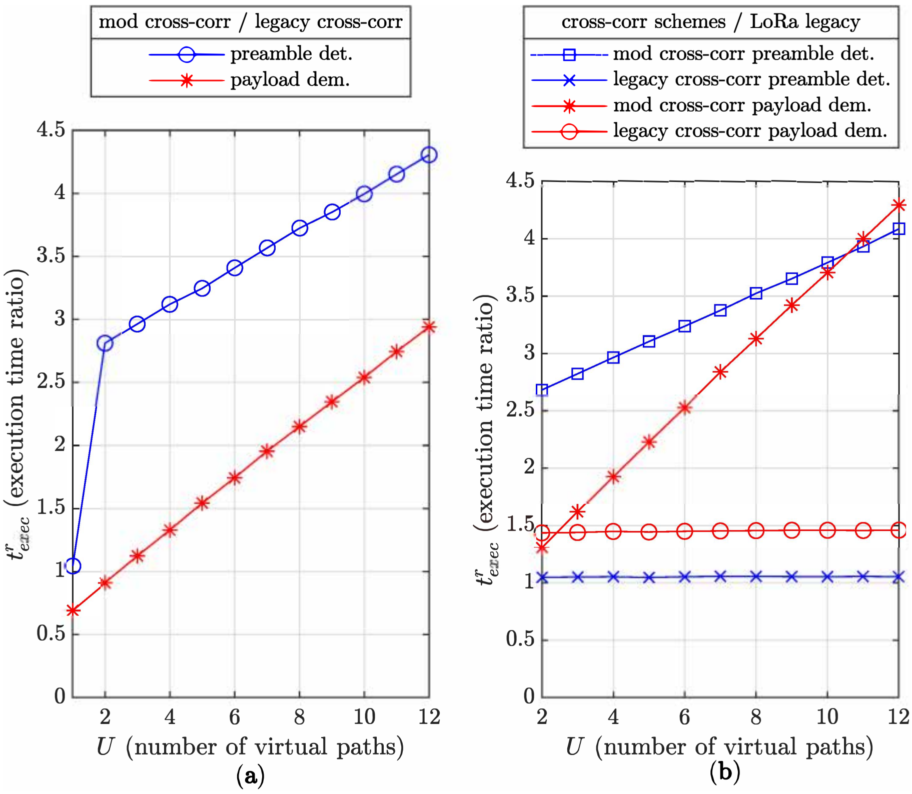

Figure 11. Complexity comparison for preamble detection and payload demodulation between: (**a**) mod cross-corr and legacy cross-corr. (**b**) mod cross-corr and LoRa legacy scheme, legacy cross-corr and LoRa legacy scheme.

In Figure 11a, the mod cross-corr/legacy cross-corr execution time ratios of the preamble detection and payload demodulation processes are presented for $U = \{1, 2, \ldots, 12\}$. We can see for $U = 1$ and the payload demodulation considered that mod cross-corr is about 30% faster than legacy cross-corr ($t^r_{exec} \approx 0.7$). Indeed, mod cross-corr with $U = 1$ is identical to the LoRa legacy demodulation scheme in (4). Then, computing the legacy cross-correlation for this case adds unnecessary complexity. Equally, when $U = 1$, the STO_{int} candidate procedure for preamble detection presented in Section 7.1 is useless, considerably decreasing the complexity, leading to a ratio ≈1.04. Activating the necessary STO_{int} candidate approach for $U > 1$ greatly increases the complexity cost, reflected in the high ratio transition from ≈0.7 to ≈2.8 between $U = 1$ to $U = 2$. Increasing

U progressively increases the mod cross-corr complexity to reach a complexity increase factor of about 3 at $U = 12$.

In Figure 11b, mod cross-corr and legacy cross-corr schemes are compared to the LoRa legacy demodulation when used for the payload demodulation and preamble detection processes. We note that the burden of mod cross-corr on preamble processing is much higher than that of the payload process for low U values but progressively reduces to reach a turnover point at $U = 11$ where the latter increases the advantage beyond this value. Again, the STO_{int} candidate approach is responsible for the high cost value at $U = 2$ but shows less increasing complexity with U. The complexity of mod cross-corr is progressively increased when U increases to reach a factor of about 4.3 at $U = 12$.

However, the cost of adding the legacy cross-correlation in the preamble section is very small with a constant ratio $\approx$1.05 as the legacy cross-correlation computation does not depend on U. We also note that using legacy cross-corr for the payload demodulation has higher relative complexity ($\approx$1.45) than for the preamble detection although its absolute complexity is much lower.

Tables 4 and 5 summarize the advantages and drawbacks of the legacy and mod cross-correlation schemes.

From Table 4, we can conclude that mod cross-corr almost completely removes U sensitivity and, thus, improves the frame detection and payload demodulation performances, but at the cost of increased complexity.

Table 5 shows the opposite behavior for legacy cross-corr, where it is more low-complexity compliant but has a high sensitivity with U which decreases the performances. That is, using mod cross-corr for the preamble detection mainly depends on performance–complexity trade-offs.

Table 4. Advantages and drawbacks of mod cross-corr.

Advantages
Mitigates U sensitivity Improves frame detection performance Improves payload demodulation performance
Drawbacks
Increases the complexity with U

Table 5. Advantages and drawbacks of legacy cross-corr.

Advantages
Adds low-complexity burden Does not increase the complexity with U
Drawbacks
Leads to high sensitivity with U Reduces frame-detection performance Reduces synchronization performance

7.3. Integer STO Part **E** *Blind Estimation Performance*

Figure 12 presents the blind STO_{int} estimation performance of **E** as the average estimation rate (ER) over Monte Carlo trials, defined as:

$$\langle ER \rangle = \frac{1}{N_{trials}} \sum_{t=0}^{N_{trials}-1} ER(t) \tag{55}$$

with:

$$ER(t) = \begin{cases} 1 & \text{if} \quad \widehat{STO}_{int}^{(t)} = STO_{int}^{(t)} \\ 0 & \text{else} \end{cases} \quad (56)$$

The figure plots the average ER as a function of $STO_{frac} = \{0, 0.1, \ldots, 0.9\}$ for random $STO_{int} \in \mathcal{U}[0; M-2]$, fixed $U = 8$, $CFO_{int} = 0$, two CFO_{frac} estimation residuals $CFO_r = \{0, 0.02\}$ in the cases of no AWGN and several $SNR_{dB} = \{-3, 0, 3, 6, 9\}$, $SF = 7$. We also add the legitimate receiver (**B** in the figure) performance as a comparison where the latter has the $STO_{frac} \approx 0.5$ case detection activated (see Section 4.4.2), for $SNR_{dB} = -3$ and $CFO_{frac} = 0.02$.

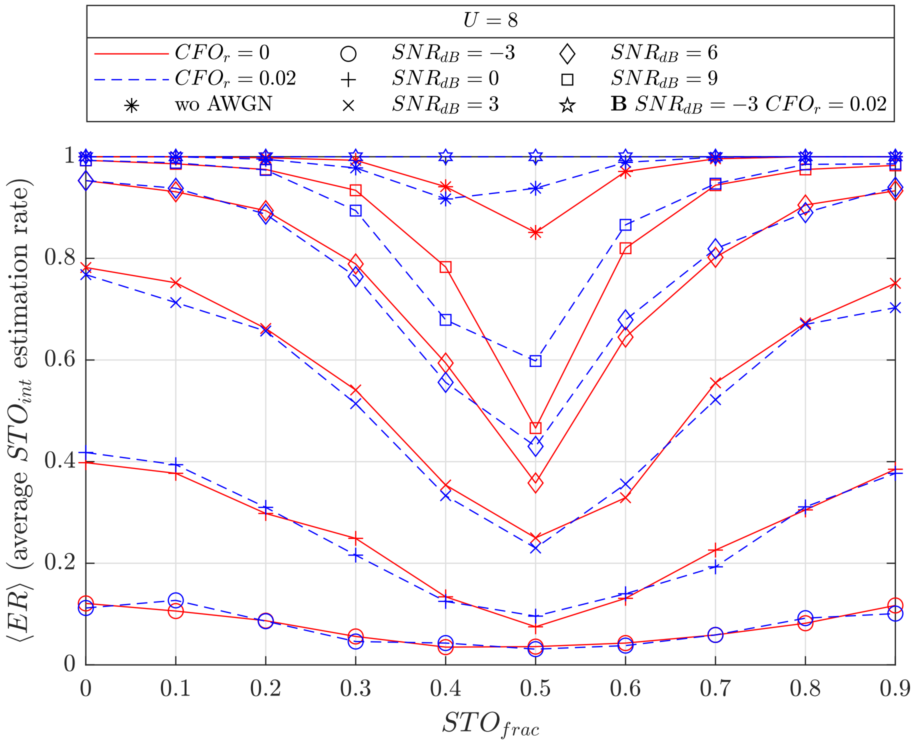

Figure 12. Blind STO_{int} estimation performance by **E** as a function of $STO_{frac} = \{0, 0.1, \ldots, 0.9\}$, $U = 8$, no AWGN and AWGN cases with $SNR_{dB} = \{-3, 0, 3, 6, 9\}$ for the latter and $SF = 7$. Legitimate receiver (**B**) performance is also considered for $SNR_{dB} = -3$ and $CFO_{frac} = 0.02$.

We can see from the figure that, in a perfect CFO_{frac} estimation scenario, i.e., $CFO_r = 0$, the average ER degrades progressively as STO_{frac} gets closer to 0.5. In the no AWGN case, $\langle ER \rangle$ is very good with $\langle ER \rangle \geq 0.87$ in the worst situation $STO_{frac} = 0.5$. Increasing the noise power progressively decreases $\langle ER \rangle$ performance with $\langle ER \rangle \leq 0.15$ at $SNR_{dB} = -3$.

We can conclude that **E** only has synchronization capability for very high SNR environments, i.e., located very close to **A** or **B** for uplinks and downlinks, respectively. Interestingly, the CFO_{frac} estimation residual produces a slightly better performance in no/very low AWGN conditions, i.e., $SNR_{dB} = \{\infty, 9, 6\}$. With sufficiently low SNR, the noise finally overtakes this effect. Note that higher U values slightly reduce $\langle ER \rangle$ performance.

We also see that **B** has a perfect ER of 1 as the SNR value considered here is high with respect to the traditional SNR range ($SNR_{dB} < -8$ usually for $SF = 7$) and then exhibits particularly good performance. Higher SNR values will exhibit identical performance and are not shown for the sake of figure clarity.

7.4. Legitimate Receiver SER Performance

Finally, we evaluate the legitimate receiver SER performance with a fully activated self-jamming scheme, i.e., modified preamble with complete synchronization and a modified cross-correlation method to demodulate payload symbols. The preamble is supposed to be detected already.

Figure 13 presents the SER performance of the legitimate receiver as a function of $SNR_{dB} = \{-15, -14, \ldots, -6\}$ for several $U = \{8, 10, 12, 14, 20\}$ and $SF = 7$. We also add the maximum performance reachable as the perfectly synchronized case with no self-jamming, i.e., $U = 1$.

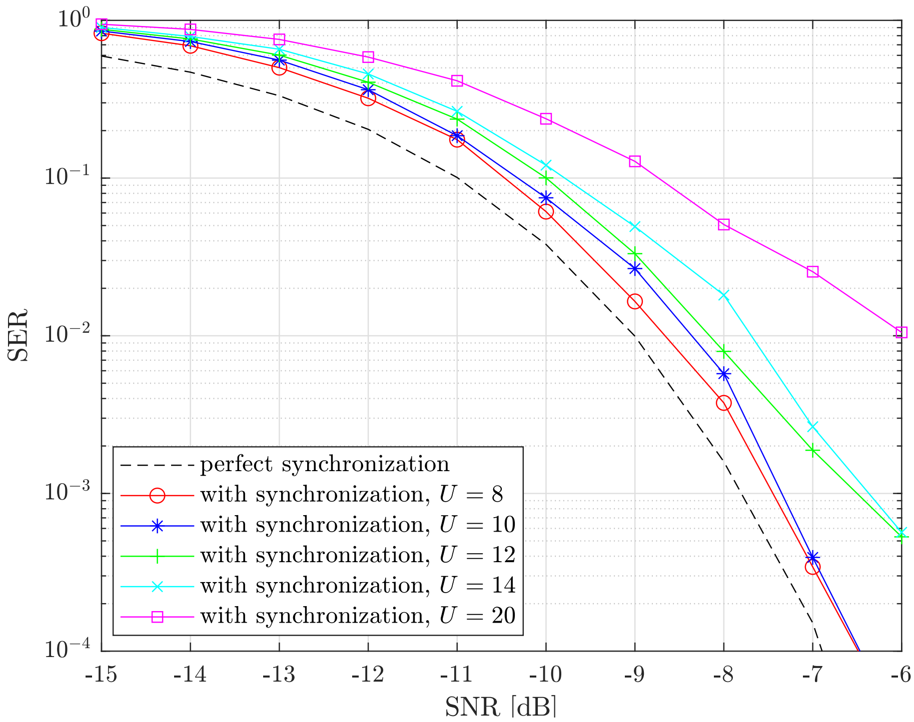

Figure 13. SER performance of **A** or **B** for $SNR_{dB} = \{-15, -14, \ldots, -6\}$, several self-jamming peaks number $U = \{8, 10, 12, 14, 20\}$ and $SF = 7$ with the synchronization front-end activated. The perfect synchronization case is also considered as an optimal performance bound.

We can see from the figure that $U = \{8, 10\}$ exhibit very good performance with a loss lower than 0.5 dB. Increasing U progressively degrades performance with a loss of about 3 dB for $U = 20$. This can be explained by the fact that the legacy cross-correlation is still used in the synchronization front-end with its U sensitivity (see Section 6), but also because of CFO_{frac} estimator limitation. If the preamble DFT peaks are too low, i.e., $U \geq 12$, CFO_{frac} will not be correctly estimated in a relatively high SNR. That is, the preamble DFT averaging performed straight afterwards will not perform well; CFO_{int} and STO_{int} will then be incorrectly estimated, leading to a payload demodulation error. However, the $U \leq 10$ value is more than sufficient to prevent **E** from correct demodulating, as explained in the next section.

7.5. **E** Blind Payload Demodulation Ability

In this subsection, we investigate the ability of **E** to blindly estimate the payload symbols with the modified payload waveform scheme (see Section 6). We assume that **E** passed the synchronization front-end successfully with the advantageous but restrained conditions $SNR_{dB} \geq 6$ and $CFO < 1$ with low CFO_{frac} residual, as seen in Section 7.3.

Since $m_{data}^{(d)}$ is unknown by **E**, the latter can only randomly choose one of the DFT magnitude bins that are above a given threshold $\rho_{data}^{(d)}$:

$$\rho_{data}^{(d)} = \lambda_{data} \times \max_n \left| \tilde{S}_{data}^{(d)}[n] \right|, \quad \lambda_{data} \in \ ]0;1] \tag{57}$$

with $|\tilde{S}_{data}^{(d)}[n]|$ the DFT magnitude of the d-th payload symbol $a_{data}^{(d)}$. The set of selected DFT bins and its length are denoted with $\mathcal{A}_{data}$ and $\widehat{U} = |\mathcal{A}_{data}|$, respectively. For a chance for **E** to detect correctly $a_{data}^{(d)}$, the latter must be in $\mathcal{A}_{data}$. We denote the probability that $a_{data}^{(d)} \notin \mathcal{A}_{data}$ as $p_{\mathcal{A}_{data}}$. This necessary condition depends on the λ_{data} value that also drives $\widehat{U}$. Then, λ_{data} must be chosen appropriately .

Figure 14 presents the impact of λ_{data} on average $\widehat{U}$ (denoted as $\langle\widehat{U}\rangle$) and $p_{\mathcal{A}_{data}}$, respectively. We consider $U = 8$ (a value giving very good SER performance for the legitimate receiver, as seen in Section 7.4), $SNR_{dB} = \{6,7,8,9\}$, $CFO < 1$ with CFO estimation residual $CFO_r = 0.02$ and random $STO_{frac} \in \{0, 0.1, 0.2, 0.8, 0.9\}$. These STO_{frac} values are the range in which **E** exhibits very good STO_{int} ER performance, as seen in Figure 12. In the simulation, **E** blindly estimates $STO_{int} \in [0; M-2]$ with the scheme presented in Section 5, and next performs the extraction of the DFT peaks with λ_E threshold to estimate STO_{frac}. The estimated STO is compensated and **E** can finally proceed to the payload section of the frame.

From Figure 14a,b, we can see that setting $\lambda_{data} = 0.1$ leads to very low $p_{\mathcal{A}_{data}}$ as most of the DFT bins are selected, leading to a very high $\langle\widehat{U}\rangle \approx 70$ at $SNR_{dB} = 6$. Increasing λ_{data} up to 0.3 decreases $\langle\widehat{U}\rangle$ a great deal to reach a floor level $\langle\widehat{U}\rangle \approx U = 8$. Interestingly, $0.2 \leq \lambda_{data} \leq 0.7$ does not impact $p_{\mathcal{A}_{data}}$ so much with $0.02 \leq p_{\mathcal{A}_{data}} < 0.1$. $\lambda_{data} > 0.7$ exhibits relatively high $p_{\mathcal{A}_{data}}$ up to ≈0.6 because of the benefit of a reduced $\langle\widehat{U}\rangle \approx 4.57$ at $\lambda_{data} = 0.9$ and $SNR_{dB} = 6$. In this example, $\lambda_{data} = 0.3$ is a good value to ensure high payload symbol capture in the DFT window of interest, i.e., $a_{data}^{(d)} \in \mathcal{A}_{data}$ and $\langle\widehat{U}\rangle \approx U$.

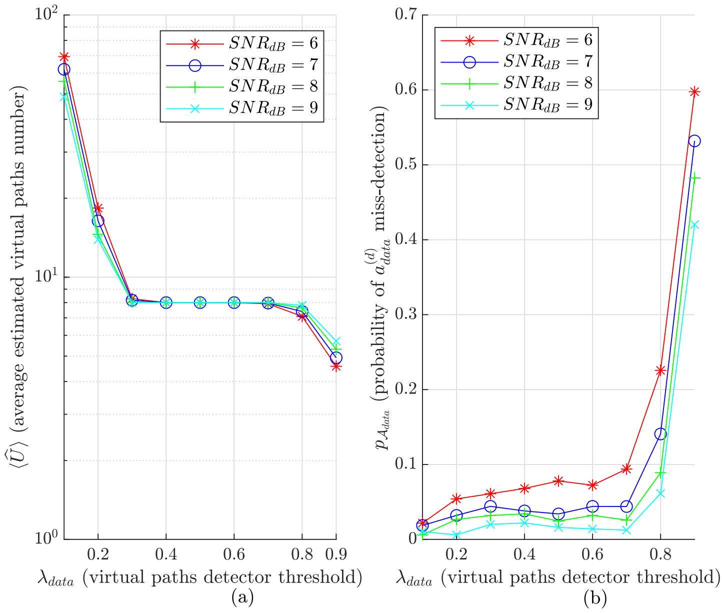

Figure 14. Eve blind payload demodulation performance as a function of λ_{data} for several SNR values and $SF = 7$. (**a**): Average estimated virtual paths number. (**b**): Probability of $a_{data}^{(d)}$ miss-detection.

Nevertheless, the demodulation brute-force complexity for **E** is still prohibitively high. If we consider $\langle\widehat{U}\rangle = U$, assuming that $a_{data}^{(d)}$ is always in $\mathcal{A}_{data}$, i.e., $p_{\mathcal{A}_{data}} = 0$, and payload symbols number N_d in the frame, this leads to the frame demodulation probability (FDP) of:

$$FDP = \frac{1}{U^{N_d}} \tag{58}$$

For $U = 8$ and $N_d = 100$, we have $U^{N_d} \approx 2.037 \times 10^{90}$ combinations and $FDP \approx 4.909 \times 10^{-90}$. At an optimistic speed of 10^9 combination trials per second, this would require 6.455×10^{73} years of trials. Therefore, it prevents **E** from efficient correct demodulation.

8. Conclusions

In this paper, we introduced an enhanced LoRa transceiver that ensures discrete and secure communications by leveraging a simple and elegant spread spectrum philosophy. This involved first modifying the preamble LoRa waveforms to prevent eavesdropper synchronization leading to incorrect payload demodulation.

We proposed a modified synchronization scheme based on current state-of-the-art techniques that estimates and mitigates the major synchronization impairments, such as the CFO, SFO and STO. We added a synchronization refinement by considering the pessimistic case $STO_{frac} \approx 0.5$, previously identified in [23], and proposed an approach based on a statistical test.

We also adopted the point of view of the eavesdropper by developing a blind STO_{int} estimation scheme. It exhibits good estimation performance provided that the SNR is much higher than the standard LoRa SNR range, the CFO is low and the received signal is well-aligned with sampling periods. Under these conditions, the eavesdropper is able to perform effective synchronization and finally retrieves the payload information. That is, modification of the preamble waveforms is necessary but not sufficient to ensure a discrete communication.

We then introduced the same modified waveform scheme to the payload but with a modified cross-correlation demodulation scheme to reduce the negative effects of the presence of multiple peaks in the LoRa DFT when using the LoRa legacy cross-correlation, at the cost of increased complexity for the legitimate receiver but much lower than that of the eavesdropper for an arbitrary small frame demodulation error. With the complete transmission scheme enabled, the SER performance loss for the legitimate receiver is less than 0.5 dB for a frequency spread factor up to $U = 10$ at $SF = 7$.

Table 6 summarizes the advantages and drawbacks of our LoRa self-jamming scheme. The main contribution of this scheme compared to other schemes described in the literature is the enablement of both discrete and private LoRa communications by considerably decreasing the eavesdropper's ability to correctly identify an outgoing LoRa transmission and preventing them from proper demodulation. The potential eavesdropper will also have great difficulty in blindly synchronizing itself and collectingthe most critical system design parameters, i.e., (U, $\mathbf{m}_U$, etc.) will only be possible with brute-force approaches. The proposed scheme is, however, not perfect and all of the advantages described are at the cost of higher implementation complexity and SER performance loss that is, however, reasonably small.

Table 6. Advantages and drawbacks of the LoRa self-jamming scheme.

Advantages
Enables more discrete LoRa communications
Hides sensitive information from eavesdroppers
Makes design parameter collection difficult for eavesdroppers
Drawbacks
Higher implementation complexity
Reasonably small SER performance loss
Software modifications required on existing LoRa transceivers

Note that this scheme does not interfere with other LoRa physical processing such as coding (e.g., Hamming and Gray coding), whitening and interleaving processes, or with the application layers, such as higher-level encryption mechanisms and LoRaWAN architecture.

From a practical implantation perspective, this scheme would require, at minimum, software modifications of existing LoRa transceivers having higher capabilities (higher computation and memory resources). This scheme may not be suitable for all applications but rather may be used for specific applications (e.g., securing a military area) where complexity constraints are not a priority but the preservation of good AWGN LoRa resilience is desired.

This analytic investigation has generated promising results for a LoRa self-jamming scheme with an adapted synchronization procedure that capitalizes on state-of-the-art LoRa synchronization algorithms. In [22], the authors evaluated the CFO_{frac}, CFO_{int} and STO_{int} estimators, as well as a variant of our STO_{frac} estimator with universal software radio peripheral (USRP) equipment, and obtained good synchronization performances.

However, this scheme needs to be assessed on real-world equipment. It will be of interest to evaluate the impact of this modified waveform on the different components of the hardware front-end. For example, as this scheme adds multiple LoRa waveforms that are not necessarily coherent with each other, it may result in an increase in the peak-to-average power ratio (PAPR) and, thus, lower the performance. This may be investigated, offering interesting research opportunities for the design of modified LoRa self-jamming waveforms that can mitigate potential PAPR increase.

Author Contributions: Conceptualization, C.D., R.G.; methodology, C.D., R.G. and P.R.; software, C.D.; validation, R.G., P.R.; formal analysis, C.D., R.G. and P.R.; investigation, C.D., R.G. and P.R.; resources, C.D.; data curation, C.D.; writing—original draft preparation, C.D.; writing—review and editing, C.D., R.G., P.R., G.B. and A.F.; visualization, C.D., R.G.; supervision, R.G. and P.R.; project administration, R.G.; funding acquisition, R.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research is jointly supported by the IBNM (Brest Institute of Computer Science and Mathematics), CyberIoT Chair of Excellence at the University of Brest (UBO), and the Brittany region—Pôle d'Excellence Cyber.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank to the University of Brest (UBO), the IBNM (Brest Institute of Computer Science and Mathematics) CyberIoT Chair of Excellence, and the Brittany region—Pôle d'Excellence Cyber for their funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gehani, A.; Harsha, S.; Raghav, R.; Sarkar, M.; Paolini, C. Application of 915 MHz Band LoRa for Agro-Informatics. In Proceedings of the 2021 Wireless Telecommunications Symposium (WTS), Virtual Event, 21–23 April 2021; pp. 1–4. [CrossRef]
2. Opipah, S.; Qodim, H.; Miharja, D.; Sarbini.; Hamidi, E.A.Z.; Juhana, T. Prototype Design of Smart Home System Base on LoRa. In Proceedings of the 2020 6th International Conference on Wireless and Telematics (ICWT), Yogyakarta, Indonesia, 3–4 September 2020; pp. 1–5. [CrossRef]
3. Od, S.; Huang, H.H.; Wei, J.B. Apply LoRa Technology to Construct an Air Quality Monitoring IoT System. In Proceedings of the 2021 IEEE 3rd Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability (ECBIOS), Tainan, Taiwan, 28–30 May 2021; pp. 88–91. [CrossRef]
4. Demeslay, C.; Gautier, R.; Fiche, A.; Burel, G. Band & Tone Jamming Analysis and Detection on LoRa signals. *arXiv* **2021**, arXiv:2107.07782.
5. Chin-Ya, H.; Ching-Wei, L.; Ray-Guang, C.; Jay, Y.S.; Shiann-Tsong, S. Experimental Evaluation of Jamming Threat in LoRaWAN. In Proceedings of the 2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring), Kuala Lumpur, Malaysia, 28 April–1 May 2019; pp. 1–6. [CrossRef]

6. Aras, E.; Small, N.; Ramachandran, G.S.; Delbruel, S.; Joosen, W.; Hughes, D. Selective Jamming of LoRaWAN Using Commodity Hardware. In Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, MobiQuitous 2017, Melbourne, Australia, 7–10 November 2017; Association for Computing Machinery: New York, NY, USA, 2017; pp. 363–372. [CrossRef]
7. Perković, T.; Rudeš, H.; Damjanović, S.; Nakić, A. Low-Cost Implementation of Reactive Jammer on LoRaWAN Network. *Electronics* **2021**, *10*, 864. [CrossRef]
8. Hou, N.; Xia, X.; Zheng, Y. Jamming of LoRa PHY and Countermeasure. In Proceedings of the IEEE INFOCOM 2021—IEEE Conference on Computer Communications, Virtual, 10–13 May 2021; pp. 1–10. [CrossRef]
9. Danish, S.M.; Qureshi, H.K.; Jangsher, S. Jamming Attack Analysis of Wireless Power Transfer on LoRaWAN Join Procedure. In Proceedings of the 2018 IEEE Globecom Workshops (GC Wkshps), Abu Dhabi, United Arab Emirates, 9–13 December 2018; pp. 1–6. [CrossRef]
10. Danish, S.M.; Nasir, A.; Qureshi, H.K.; Ashfaq, A.B.; Mumtaz, S.; Rodriguez, J. Network Intrusion Detection System for Jamming Attack in LoRaWAN Join Procedure. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6. [CrossRef]
11. Martinez, I.; Tanguy, P.; Nouvel, F. On the performance evaluation of LoRaWAN under Jamming. In Proceedings of the 2019 12th IFIP Wireless and Mobile Networking Conference (WMNC), Paris, France, 11–13 September 2019; pp. 141–145. [CrossRef]
12. Martinez, I.; Nouvel, F.; Lahoud, S.; Tanguy, P.; Helou, M.E. On the Performance Evaluation of LoRaWAN with Re-transmissions under Jamming. In Proceedings of the 2020 IEEE Symposium on Computers and Communications (ISCC), Rennes, France, 7–10 July 2020; pp. 1–7. [CrossRef]
13. Ahmar, A.U.H.; Aras, E.; Nguyen, D.T.; Michiels, S.; Joosen, W.; Hughes, D. CRAM: Robust Medium Access Control for LPWAN using Cryptographic Frequency Hopping. In Proceedings of the 2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS), Marina del Rey, CA, USA, 25–27 May 2020; pp. 95–102. [CrossRef]
14. Tsai, K.L.; Leu, F.Y.; You, I.; Chang, S.W.; Hu, S.J.; Park, H. Low-Power AES Data Encryption Architecture for a LoRaWAN. *IEEE Access* **2019**, *7*, 146348–146357. [CrossRef]
15. Zhang, C.; Yue, J.; Jiao, L.; Shi, J.; Wang, S. A Novel Physical Layer Encryption Algorithm for LoRa. *IEEE Commun. Lett.* **2021**, *25*, 2512–2516. [CrossRef]
16. Chiani, M.; Elzanaty, A. On the LoRa Modulation for IoT: Waveform Properties and Spectral Analysis. *IEEE Internet Things J.* **2019**, *6*, 8463–8470. [CrossRef]
17. Vangelista, L. Frequency Shift Chirp Modulation: The LoRa Modulation. *IEEE Signal Process. Lett.* **2017**, *24*, 1818–1821. [CrossRef]
18. Qian, W.; Hai, S.; Kui, R.; Kwangjo, K. Fast and scalable secret key generation exploiting channel phase randomness in wireless networks. In Proceedings of the 2011 Proceedings IEEE INFOCOM, Shanghai, China, 10–15 April 2011; pp. 1422–1430. [CrossRef]
19. Zhang, J.; Duong, T.Q.; Marshall, A.; Woods, R. Key Generation From Wireless Channels: A Review. *IEEE Access* **2016**, *4*, 614–626. [CrossRef]
20. Bernier, C.; Dehmas, F.; Deparis, N. Low Complexity LoRa Frame Synchronization for Ultra-Low Power Software-Defined Radios. *IEEE Trans. Commun.* **2020**, *68*, 3140–3152. [CrossRef]
21. Ghanaatian, R.; Afisiadis, O.; Cotting, M.; Burg, A. Lora Digital Receiver Analysis and Implementation. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 1498–1502. [CrossRef]
22. Tapparel, J.; Afisiadis, O.; Mayoraz, P.; Balatsoukas-Stimming, A.; Burg, A. An Open-Source LoRa Physical Layer Prototype on GNU Radio. In Proceedings of the 2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Atlanta, GA, USA, 26–29 May 2020; pp. 1–5. [CrossRef]
23. Xhonneux, M.; Orion, A.; Bol, D.; Louveaux, J. A Low-Complexity LoRa Synchronization Algorithm Robust to Sampling Time Offsets. *IEEE Internet Things J.* **2021**, *9*, 3756–3769. [CrossRef]