LiteNap: Downclocking LoRa Reception

Xianjin Xia
The Hong Kong Polytechnic University, Hong Kong, China
csxxia@comp.polyu.edu.hk

Yuanqing Zheng
The Hong Kong Polytechnic University, Hong Kong, China
csyqzheng@comp.polyu.edu.hk

Tao Gu
RMIT University, Melbourne, Australia
tao.gu@rmit.edu.au

***Abstract*—This paper presents LiteNap which improves the energy efficiency of LoRa by enabling LoRa nodes to operate in a downclocked 'light sleep' mode for packet reception. A fundamental limit that prevents radio downclocking is the Nyquist sampling theorem which demands the clock-rate being at least twice the bandwidth of LoRa chirps. Our study reveals under-sampled LoRa chirps suffer frequency aliasing and cause ambiguity in symbol demodulation. LiteNap addresses the problem by leveraging an empirical observation that the hardware of LoRa radio can cause phase jitters on modulated chirps, which result in frequency leakage in the time domain. The timing information of phase jitters and frequency leakages can serve as physical fingerprints to uniquely identify modulated chirps. We propose a scheme to reliably extract the fingerprints from under-sampled chirps and resolve ambiguities in symbol demodulation. We implement LiteNap on a software defined radio platform and conduct trace-driven evaluation. Experiment results show that LiteNap can downclock LoRa nodes to sub-Nyquist rates for energy savings (*e.g.*, 1/8 of Nyquist rate), without substantially affecting packet reception performance (*e.g.*, >95% packet reception rate).**

I. Introduction

The recent advances of LP-WANs (*i.e.*, Low Power Wide Area Networks) enable wireless network access for long-term operated IoT devices. Among many LP-WAN technologies (*e.g.*, SigFox [1], NB-IoT [2]), LoRaWAN has received wide attention from industry and academia due to its capability in terms of long range, low power, *etc* [3]. The PHY layer of LoRaWAN adopts Chirp Spread Spectrum (CSS) modulation, which is robust against noise, multi-path and Doppler effects.

Despite the excellent communication performance of LoRa, energy efficiency remains a major concern since LoRa nodes are typically battery-powered and expected to operate over a long time (*e.g.*, $\geq$ 10 years) without replacing the battery. However, measurement study [4] reports that in practice the lifetime of LoRa nodes can be much shorter in case of frequent packet transmission and reception [5]. As the on-air time of LoRa packet is much longer than that of other wireless technologies (*e.g.*, WiFi, ZigBee) [6] [7], LoRa nodes must stay awake for a longer time to send and receive a packet during packet transmission. As a result, the per-packet power consumption of LoRa can be much higher than those of conventional radios. The most power-hungry components of LoRa radio are Analog-to-Digital Converter (*i.e.*, ADC) and the processing unit (*i.e.*, MCU).

Orthogonal to the conventional energy saving schemes such as duty-cycling, our work explores an alternative approach to reduce the per-packet power consumption of LoRa using downclocking. The power consumption of MCU and ADC is generally proportional to the operating clock rates [8]. By decreasing the clock rates, we expect to proportionally reduce the power consumption of LoRa radio.

However, it is challenging to downclock LoRa radio to reduce power consumption without affecting communication performance. It is well known that the clock-rate of ADC is fundamentally limited by the Nyquist's theorem, which requires the sampling rate to be at least twice the signal bandwidth. If the clock-rate of a receiver falls below the Nyquist rate (*i.e.*, twice the maximum frequency of LoRa chirps), the frequency aliasing may lead to incorrect demodulation. As LoRa spreads the chirp frequency across the entire bandwidth, it leaves little room for downclocking LoRa receiver without causing frequency aliasing. Aiming at reducing the power consumption of LoRa receiver, this paper asks a question: *can we decode LoRa packets sampled at sub-Nyquist rate?*

We conduct extensive measurement study and theoretical analysis on LoRa packet reception at sub-Nyquist rates. Our study yields two observations: (1) The frequency aliasing may fold two LoRa chirps separated by the sampling frequency into the same aliased frequency. As a result, the two LoRa chirps measured at a sub-Nyquist rate may resemble each other and cause ambiguity in demodulation. Hence, *the key to demodulate under-sampled chirps lies in how to resolve the ambiguity caused by frequency aliasing due to under-sampling.* (2) As a LoRa chirp continuously sweeps across a pre-configured LoRa band, *frequency leakage* inevitably occurs when the instantaneous frequency suddenly changes from its maximum to minimum. Since LoRa varies the starting frequency of chirp to encode different data and the increasing rate of frequency remains constant, frequency leakage happens at distinct time in different LoRa chirps. Such frequency leakage in the time domain can thus be used as a *physical fingerprint* to uniquely identify a LoRa chirp and help resolve the ambiguity caused by frequency aliasing. More importantly, as the frequency leakage spans across the full frequency band, the timing information can still be reserved even sampled at sub-Nyquist rates.

Based on the above observations, we propose a decoding method named LiteNap for down-clocked LoRa reception. We first exploit the prior knowledge of LoRa preamble to detect chirp boundaries. Then, we demodulate under-sampled chirps and measure the initial frequency of LoRa chirps with

frequency aliasing. We finally disambiguate the aliasing with the timing information of frequency leakages. Intuitively, we use the timing information of frequency leakage to determine the missing bits caused by under-sampling.

It entails tremendous challenges to extract timing information of frequency leakage from under-sampled chirps. Firstly, as frequency leakage occurs in a rather short time, a down-clocked radio with low sampling rates may miss detecting frequency leakage. Secondly, as the power strength of frequency leakage is weak, which can be buried below noise, it is non-trivial to reliably detect from the frequency domain. Our study reveals that frequency leakages are essentially caused by the phase jitters of chirp samples, which are introduced by the hardware of LoRa modem. We find that the hardware phase jitters will add constant phase shifts to all modulated samples transmitted after the jitters. We leverage the finding to extract symbol fingerprint in the phase domain. Specifically, we check the phase of received chirp samples to detect phase shift and extract the timing information as fingerprint. We experimentally demonstrate that the phase-based approach can correctly extract unique fingerprints from under-sampled chirps in various SNR conditions.

We implement and evaluate LiteNap on a software define radio platform. We present a strategy to adaptively schedule the downclocking of LoRa radio. Evaluation results show that LiteNap can reduce half the energy consumption of LoRa communication by downclocking a receiver to 1/8 the Nyquist-rate with high packet decoding accuracy.

II. Background and Motivation

A. LoRa Modulation

The PHY modulation of LoRa adopts Chirp Spread Spectrum (CSS), where the frequency of a LoRa chirp increases linearly with time and sweeps through a predefined bandwidth. A *base chirp* is represented as $C(t) = e^{j2\pi(\alpha t+f_0)t}$, where f_0 and $(\alpha t + f_0)$ denote the initial frequency and instantaneous frequency at time t, respectively. The frequency increasing rate (α) and time duration (T_{chirp}) of a chirp is determined by two parameters of CSS modulation: spreading factor (SF) and frequency bandwidth (BW), *i.e.*, $T_{chirp} = \frac{2^{SF}}{BW}$, $\alpha = \frac{BW}{T_{chirp}}$.

In modulation, LoRa varies the initial frequency of chirp signal to represent different data. The modulation procedure can be represented as follows.

$$S(t, f_{sym}) = C(t) \cdot e^{j2\pi f_{sym} t} \tag{1}$$

where f_{sym} is the starting frequency of chirp $S(t, f_{sym})$ and $C(t)$ represents a base chirp. We see that the starting frequency f_{sym} carries the information of modulated data. To demodulate a received symbol from a chirp $S(t, f_{sym})$, we multiply the received chirp with the conjugate of a base chirp (denoted as $C^{-1}(t)$) as below:

$$S(t, f_{sym}) \cdot C^{-1}(t) = e^{j2\pi f_{sym} t}. \tag{2}$$

Then, we perform FFT (*i.e.*, Fast Fourier Transform) to derive the starting frequency (*i.e.*, f_{sym}) and demodulate the chirp.

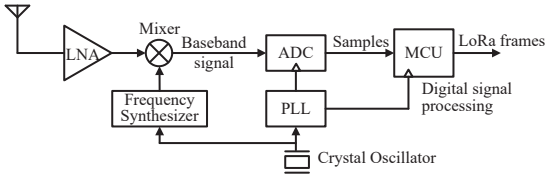


Fig. 1. Simplified receiver chain of LoRa node [9].

TABLE I
Power profiles of a LoRa node in various operation modes.

Modes		Enabled Components	Current Draw
Active	Transmit	MCU + TX Chains	20∼120 mA*
	Receive	MCU + RX Chains	11.5 mA
	Standby	MCU (disabled RF & PLL)	1.6∼1.8 mA
Sleep		Circuit	0.2∼1.5 μA

*The current draw varies with respect to the transmission power of 7∼20 dBm.

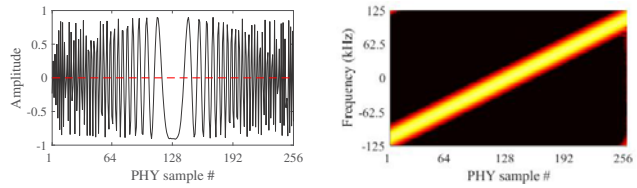
B. Radio Power vs. Clock Rate

Fig.1 shows the receiver chain of a LoRa radio [9]. The RF frontend first down-converts the received signal to intermediate frequency using a mixer. Next, an Analog-to-Digital Converter (ADC) samples the baseband signal, and passes the samples to a MCU for further processing. The radio is typically driven by a 32 MHz crystal oscillator, which feeds both the frequency synthesizer and Phase-Locked-Loop (PLL). The output of PLL controls the sampling rate of ADC and the clock-rate of MCU.

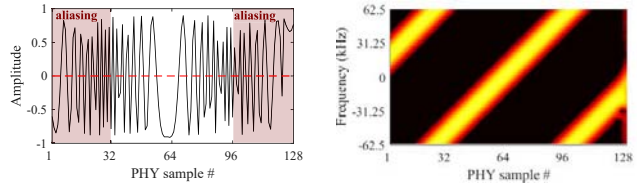
When a LoRa radio is awake in scheduled TX/RX windows, it operates in one of the three modes, *i.e.*, Transmit, Receive or Standby. Table I compares the power profiles of various operation modes for a LoRa node based on the Semtech SX1276 datasheet. When a radio is in the Receive mode, ADC continuously samples the channel to detect, receive and decode incoming packets. As such, ADC and MCU are the most power-hungry components in the Receive mode. When a radio turns to Standby, most components on the receiver chain (*e.g.*, ADC) except MCU are powered off [4], [9]. The power consumption of MCU and ADC generally follows $P \propto V^2 f$ [8], [10], where V is the supply voltage, f denotes the clock-rate for MCU and the sampling rate for ADC, respectively.

We aim to reduce the active power consumption of LoRa radio by downclocking the power-hungry components. This can be realized by changing the PLL configuration of desired output frequency, which is programmable in practice. As a LoRa node typically stays in Standby for most of the time [4], we expect significant reduction in power consumption if the clock-rate of MCU can be decreased in the Standby mode.

However, a key problem arises: When a radio changes the PLL frequency to downclock MCU, the sampling rate of ADC also decreases, since the two components share the same clock as illustrated in Fig.1. The sampling rate of ADC is restricted fundamentally by the Nyquist theorem. If the sampling rate falls below the Nyquist rate (*i.e.*, twice the signal bandwidth), a LoRa node may not be able to demodulate incoming LoRa chirps because of frequency aliasing. Our work tackles the problem by exploring feasible approaches to decode LoRa



(a) Samples, F_s=250 ksps (b) Frequency, F_s=250 ksps



(c) Samples, F_s=125 ksps (d) Frequency, F_s=125 ksps

Fig. 2. The time-domain and frequency-domain presentations of a LoRa chirp when sampling at Nyquist-rate (a,b) and sub-Nyquist rate (c,d).

packets at sub-Nyquist sampling-rates.

III. Downclocked Reception

In this section, we investigate the frequency aliasing effect of LoRa chirps when sampling at sub-Nyquist rates. Consider the transmitted chirp of a single symbol, *i.e.*, $S(t, f_{sym})$. We denote the received signal as

$$R(t, f_{sym}) = h(t)e^{j2\pi\Delta f_{cfo}t} \cdot S(t, f_{sym}) + n(t),\ t \in [0, T_{chirp})$$

where, $n(t)$ is noise, $h(t)$ denotes the changes of amplitude and phase induced by wireless channel, and Δf_{cfo} denotes the carrier frequency offset between transmitter and receiver.

The received signal (*i.e.*, $R(t, f_{sym})$) is sampled by ADC. A normal receiver shall sample the chirp above Nyquist rate, which is $F_s = BW$ as chirp frequency varies within $-\frac{BW}{2} \sim \frac{BW}{2}$. We denote the discrete signal samples with

$$R(k) = R(\frac{k}{BW}, f_{sym}), \quad k = 0, 1, \cdots, 2^{SF} - 1.$$

If the receiver reduces the sampling rate by a factor of D (*i.e.*, with a sampling-rate $F_s = \frac{BW}{D}$), the signal samples become

$$R(Dk) = R(\frac{Dk}{BW} + \Delta t, f_{sym}), \quad k = 0, 1, \cdots, \lfloor \frac{2^{SF}}{D} \rfloor - 1. \tag{3}$$

where, Δt represents the time offset between the first sample and the arrival time of the chirp (*i.e.*, chirp boundary).

To illustrate how frequency aliasing affects LoRa demodulation, we use a USRP to receive LoRa packets transmitted by a Semtech SX1276 based LoRa node with $SF = 8$ and $BW = 250$ kHz. The sampling rate of USRP is 250 ksps (*i.e.*, the Nyquist rate). We emulate a down-sampling factor of D by drawing one sample from every D samples. Fig.2 shows the time-domain and frequency-domain results of the same LoRa chirp under different sampling-rates. In Fig.2(a,c), we see that as the sampling-rate decreases below the Nyquist rate, the obtained signal samples become sparse. As

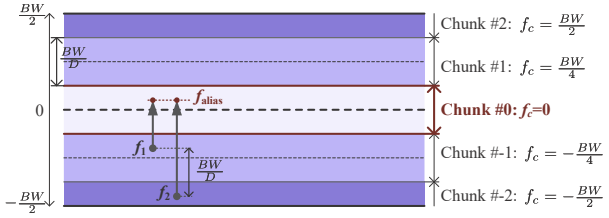


Fig. 3. Frequency aliasing model (down-clocking factor $D = 4$): f_1, f_2 are folded to the same aliased frequency (f_{alias}), resulting in ambiguity.

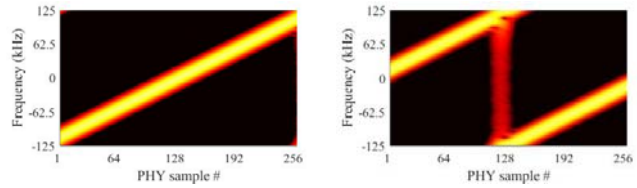
expected, the frequency of the signal reconstructed from sub-Nyquist samples are distorted from that of the original signal, especially for the high-frequency components (see Fig.2(b,d)).

According to the Nyquist-Shannon sampling theorem, when sampling at the rate of F_s, we can only reconstruct the signal of frequency ranging from $-\frac{F_s}{2}$ to $\frac{F_s}{2}$. If the receiver radio samples LoRa chirps at a sub-Nyquist rate $F_s = \frac{BW}{D}$, the original frequency of chirp signal which spreads across the entire LoRa bandwidth (*i.e.*, $[-\frac{BW}{2}, \frac{BW}{2}]$) will be folded into $[-\frac{BW}{2D}, \frac{BW}{2D}]$. Specifically, we can characterize the frequency aliasing of LoRa chirps as follows.

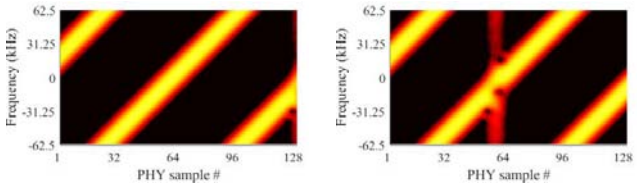
As illustrated in Fig.3, we divide the entire frequency band into $D+1$ chunks. We represent each chunk as $[f_c - \frac{BW}{2D}, f_c + \frac{BW}{2D})$, where $f_c = n_c\frac{BW}{D}$ denotes the central frequency of chunk n_c, and n_c denotes the chunk ID. Because of the frequency aliasing effect, a frequency $f = n_c\frac{BW}{D} + f_{alias}$ of outer chunks (*i.e.*, $|n_c| > 0$) will be folded into f_{alias} of chunk 0, where $f_{alias} \in [-\frac{BW}{2D}, \frac{BW}{2D})$.

The frequency aliasing effect of downsampling can cause *symbol ambiguity problem* in LoRa demodulation. As shown in Fig.4, the chirps of symbol #0 (f_{sym0}=-125kHz) and #127 (f_{sym127}=0kHz), when sampled at half the Nyquist rate, will be folded into the same chunk. More importantly, the starting frequencies of both aliased symbols become the same (*i.e.*, 0kHz). As a result, we cannot distinguish the two aliased symbols by examining their starting frequencies. Generally, with a downsampling factor of D, such ambiguity would happen between the two symbols f_1 and f_2, if $|f_1 - f_2| = n\frac{BW}{D},\ n = 1, 2, \cdots, D-1$.

Therefore, the key problem in downclocked LoRa reception is *how to resolve the ambiguity of under-sampled chirps and correctly demodulate an aliased chirp*. We observe from Fig.4 that although the starting frequencies of aliased symbols are the same, comparing Fig.4 (c,d), we see that the two aliased symbol exhibit different patterns. In particular, while there is a vertical frequency leakage band appearing at PHY sample 64 in Fig.4 (d), there is no such a vertical band at the location in Fig.4 (c). As a matter of fact, the vertical band appears at PHY sample 128 in Fig.4 (c). That is because when a chirp frequency suddenly jumps from $\frac{BW}{2}$ to $-\frac{BW}{2}$, there is frequency leakage spanning the entire bandwidth. Comparing Fig.4 (b,d), we also observe that the timing information of frequency leakage is reserved even in the aliased chirp. It motivates us to exploit the timing information of frequency leakage to resolve symbol ambiguity.



(a) Original chirp of symbol #0 (b) Original chirp of symbol #127



(c) Aliased chirp of symbol #0 (d) Aliased chirp of symbol #127

Fig. 4. (a,b) The original chirps of symbol #0 and #127 (sampling above the Nyquist-rate); (c,d) The reconstructed signals of symbol #0 and #127 when sampling at half the Nyquist-rate.

IV. Hardware-assisted Demodulation

A. Demodulating Aliased Symbol

We adapt the standard method of Eq.(2) to demodulate an under-sampled LoRa chirp (*i.e.*, Eq. (3)). Specifically, we replace $C^{-1}(t)$ in Eq.(2) with a discrete under-sampled down-chirp, denoted by $C^{-1}(\frac{Dk}{BW}),\ k=0,1,\cdots,\lfloor\frac{2^{SF}}{D}\rfloor-1$. The operation is represented as follows.

$$\begin{aligned} R(Dk)\cdot C^{-1}(\tfrac{Dk}{BW}) &= R(\tfrac{Dk}{BW}+\Delta t, f_{sym})\cdot C^{-1}(\tfrac{Dk}{BW}) \\ &\approx h(\tfrac{Dk}{BW}+\Delta t)e^{j2\pi\Delta f_{cfo}(\frac{Dk}{BW}+\Delta t)}\cdot \\ &\quad S(\tfrac{Dk}{BW}+\Delta t, f_{sym})\cdot C^{-1}(\tfrac{Dk}{BW}) \end{aligned}$$

We omit noise $n(t)$ due to the fact that the signal strength, while being added up across the whole chirp, is usually much higher than noise. Note that the time offset of under-sampling (Δt) can transform to a frequency offset of $\Delta f_{sample}=\frac{\Delta t}{2^{SF}}$ in LoRa chirp. We have $S(\frac{Dk}{BW}+\Delta t, f_{sym})=S(\frac{Dk}{BW}, f_{sym}+\Delta f_{sample})$. Substituting into the above equations, we get

$$\begin{aligned} & R(Dk)\cdot C^{-1}(\tfrac{Dk}{BW}) \\ \approx\ & h(\tfrac{Dk}{BW}+\Delta t)e^{j2\pi\Delta f_{cfo}(\frac{Dk}{BW}+\Delta t)}\cdot \\ & e^{j2\pi(f_{sym}+\Delta f_{sample})\cdot\frac{Dk}{BW}} \\ \approx\ & e^{j2\pi(f_{sym}+\Delta f_{sample}+\Delta f_{cfo})\cdot\frac{Dk}{BW}} \end{aligned} \tag{4}$$

We exclude $h(t)$ from the equations because $h(t)$ can be measured and cancelled by leveraging the prior knowledge of chirps in preamble. Next, we expect to perform FFT on $R(Dk)\cdot C^{-1}(\frac{Dk}{BW})$ to derive the encoded symbol (*i.e.*, f_{sym}). However, two issues still need to be handled:

Frequency offset cancellation. To obtain the correct symbol (*i.e.*, f_{sym}), we must remove $(\Delta f_{sample}+\Delta f_{cfo})$ from Eq. (4). As the frequency offsets of radio hardware (Δf_{cfo}) and sample timing (Δf_{sample}) remain relatively stable across all chirps in the same packet, we can detect $(\Delta f_{sample}+\Delta f_{cfo})$ from the preamble of LoRa packet (by leveraging the fact that $f_{sym}=0$ in preamble) and subtract $(\Delta f_{sample}+\Delta f_{cfo})$ from the FFT results of symbols in the payload.

Ambiguity mitigation. As the chirp signal is received below the Nyquist sampling-rate, by performing FFT on $R(Dk)\cdot C^{-1}(\frac{Dk}{BW})$, we can only obtain an aliased frequency f_{alias}. According to the aliasing model, the real symbol can be any of $f_{alias}+n\frac{BW}{D},\ n=0,1,\cdots,D-1$. We resolve the ambiguity by exploiting additional information in the following.

B. Frequency-based Ambiguity Mitigation

In this subsection, we propose a frequency-based approach to detect frequency leakage and mitigate the symbol ambiguities caused by frequency aliasing.

Symbol fingerprint. Fig.5 presents the spectrogram of a portion of a LoRa packet sent by a COTS LoRa node. We see from Fig.5 that there are frequency leakages from the main frequency of chirp signal (which indicates the instability of chirp frequency) as LoRa chirps experience sudden changes of frequency (*e.g.*, when a chirp shifts from the maximum frequency to the minimum or at chirp boundaries). The frequency leakage spans across the whole bandwidth, as displayed in Fig.5. We observe the same phenomenon in various types of COTS LoRa nodes (*e.g.*, Adafruit Feather 32u4, Dragino LoRa shield, *etc.*).

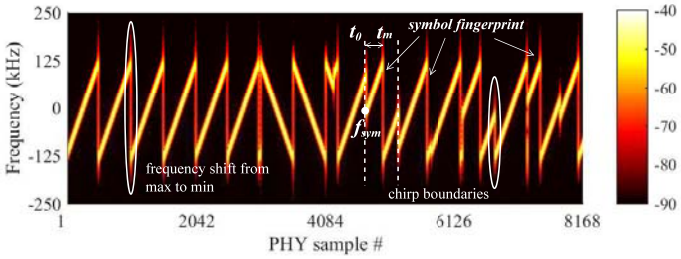


Fig. 5. Illustration of frequency leakage in the received LoRa chirps.

Generally, the manufacture imperfection of radio electronics (*e.g.*, oscillator) can add random phase jitters to the modulated signal [11]. Based on the study of hardware implementations of CSS [12] [13] and Semtech datasheet [9] [14], we know that LoRa radio employs angle modulation (*i.e.*, phase-based modulation) to generate CSS samples. We present the hardware architecture of CSS modem in Fig.6. To handle the frequency shifts when a new symbol starts and when a chirp reaches the maximum frequency, the modem requires switching to a new modulation parameter or enabling additional blocks to amend the modulated phases. During such changes, LoRa radio will experience hardware instabilities that lead to phase jitters. The phase jitters of modulated signal would consequently exhibit as frequency leakage as shown in Fig.5.

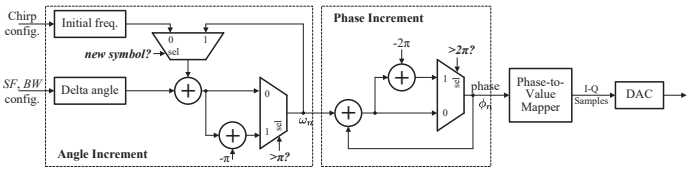
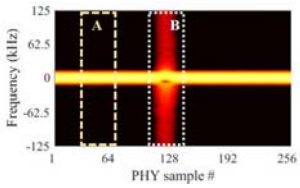
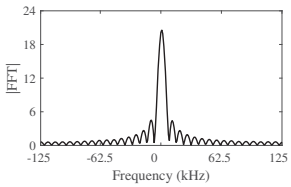


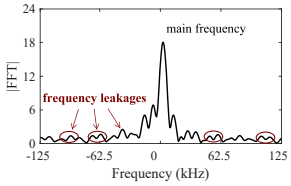
Fig. 6. Hardware implementation of CSS based on angle modulation.



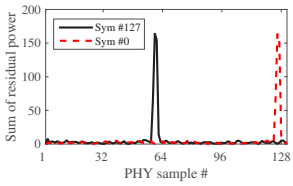
(a) Dechirped signal, F_s=250 ksps



(b) FFT of window A

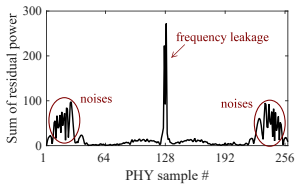


(c) FFT of window B

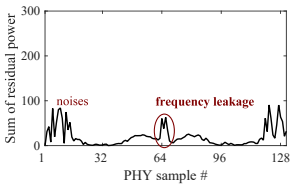


(d) Detected Loc. of freq. leakage

Fig. 7. Frequency-based fingerprint detection: (a) Dechirping the received chirp signal of Fig.4(b); (b,c) FFT results of the signal in window A and B; and (d) Frequency leakages detected from the under-sampled chirps of symbol #0 and #127, ambiguity symbols have different fingerprints.



(a) Detected freq. leakage (D=1)



(b) Detected freq. leakage (D=2)

Fig. 8. Impacts of noise on the detection of frequency leakage: Detecting results of (a) fully-sampled and (b) under-sampled chirps. Frequency leakage is buried within noise in the case of down-sampling (*e.g.*, D=2).

The frequency leakage appearing around the time when a chirp shifts from the maximum frequency to the minimum can serve as a PHY fingerprint to uniquely identify an aliased symbol. As illustrated in Fig.5, let t_0 denote the starting time of the chirp, t_m the time of chirp frequency shifting from $\frac{BW}{2}$ to $-\frac{BW}{2}$. Since chirp frequency increases linearly with time, we can infer the starting frequency of chirp as $f_{sym} = \frac{BW}{2} - \alpha \cdot (t_m - t_0)$. Therefore, we can detect frequency leakage within a chirp and extract the timing information to resolve the ambiguity caused by frequency aliasing.

Fingerprint extraction. We detect the frequency leakage as follows. As in the standard demodulation process, we first perform a dechirp operation by multiplying the received chirp with $C^{-1}(t)$ (see Eq. (2)), which accumulates power into a main frequency (*i.e.*, f_{sym}). Next, we perform FFT on the results to detect if there is frequency leakage spanning across the whole bandwidth. Fig.7(a) presents the dechirped results of the chirp shown in Fig.4(b). We take two different segments (with/without frequency leakage) of the dechirped signal with the same segment length. We compare the FFT of two segments in Fig.7(b) and (c). As displayed in Fig.7(c), the irregular tiny spikes apart from the main peak indicate frequency leakage, yet the spikes in Fig.7(b) correspond to the sinc side lobes of the main frequency, which are introduced by FFT. To remove the influence of main frequency, we subtract the FFT result of Fig.7(b) from that of Fig.7(c), then sum up the residual power in all FFT bins to get the power of frequency leakage.

The timing of frequency leakage can be detected with a sliding-window method: We move an FFT window across all samples of the chirp, and extract the power of frequency leakage in each window by subtracting the FFT of adjacent windows. Fig.7(d) presents the detected power of frequency leakage at different offsets of the chirp in Fig.7(a). We can adopt a threshold to detect the time of frequency leakage (*i.e.*, t_m). For the example shown in Fig.4, we can detect different t_m from the chirps of symbol #0 and #127, as displayed in Fig.7(d). We exploit the detected fingerprint information to resolve ambiguity of the two symbols.

Remarks. The frequency-based approach relies on detecting the power of frequency leakage. However, as frequency leakage appears in short time, it can be missed under lower sampling-rates. In addition, the method of power detecting is vulnerable to noise. Fig.8 presents the detecting results on a noisy chirp of symbol #127. We see that the noise power can cause distortions to the frequency leakage detection. The power of frequency leakage is buried within noise in the case of down-sampling (see Fig.8(b)), from which we may not be able to reliably extract the fingerprint and determine the timing of frequency leakage. Therefore, a natural question arises: *how can we reliably extract the fingerprint of symbol under lower sampling-rates?* We answer the question in the following.

C. Phase-based Ambiguity Mitigation

In this subsection, we explore the opportunity to enhance the robustness of fingerprint extraction under lower sampling-rates. We propose a phase-based approach and demonstrate the advantages in comparison with the frequency-based approach.

Opportunity. We explore opportunity for fingerprint extraction by investigating the phase characteristic of LoRa chirps. Recall that LoRa modem employs angle modulation, as shown in Fig.6, the modem controls the phase of each sample to modulate a chirp signal. Denote the phase of the k'th sample by ϕ_k. The value of ϕ_k is determined by the phase of the former sample (*i.e.*, ϕ_{k-1}) and the instantaneous frequency to be modulated (denoted by ω_k). Specifically, $\phi_k = \phi_{k-1} + \omega_k$. Taking phase noise into accounts, we have

$$\phi_k - \phi_{k-1} = \omega_k + J(k), \quad k = 0, 1, \cdots, 2^{SF} - 1 \qquad (5)$$

where $J(k)$ denotes the phase jitters added by LoRa modem.

Fig.9(a) presents the raw phase data of symbol #127 when sampling at full clock-rate. According to Eq. (5), the phase difference of adjacent samples represents the instantaneous angular frequency (*i.e.*, ω_n) of chirp modulation (see similarities between Fig.9(b) and Fig.4(b)). The phase difference (*i.e.*, instantaneous frequency) would also suffer from aliasing in the case of under-sampling, as displayed in Fig.9(c).

Based on Eq.(5), we can get the phase of any sample in the chirp as follows.

$$\phi_k = \phi_0 + \sum_{n=0}^{k} \omega_n + \sum_{n=0}^{k} J(n). \quad (6)$$

Eq.(6) indicates that the phase jitters of former samples will be accumulated in the phases of later samples. Suppose that the hardware of LoRa modem only causes phase jitters at a specific time (*i.e.*, denoted by t_{jit}), all samples after t_{jit} then will carry phase jitters yet the samples before t_{jit} will not. As a result, we can anticipate a phase shift of $\Delta\phi_{jit} = \sum_n J(n)$ around the time of t_{jit} (see Fig.9(d)). It enables us to extract fingerprint (*i.e.*, timing of phase jitters) by comparing the phases of all samples in a received chirp, rather than detecting the samples of frequency leakage in short time duration.

Solution. To formally define the problem, we amend the signal model in Eq. (1) to incorporate the phase jitters of LoRa modem. We denote the modulated chirp of symbol f_{sym} as

$$S_{jit}(t, f_{sym}) = S(t, f_{sym}) \cdot e^{j\varphi(t)},\ t \in [0, T_{chirp}) \quad (7)$$

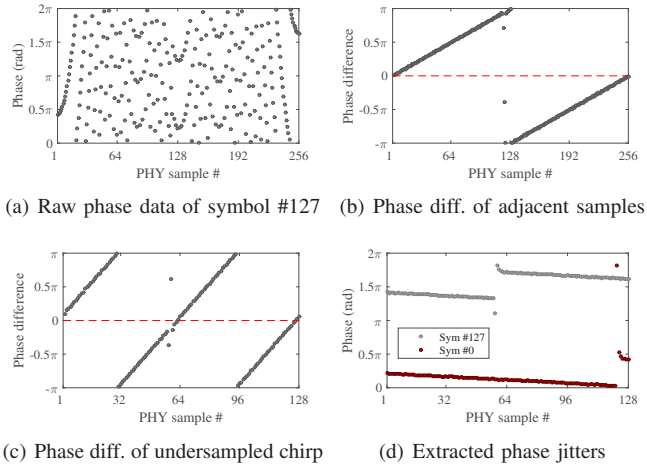
where $\varphi(t)$ denotes the sum of phase jitters accumulated from time 0 to t (*i.e.*, $\varphi(t) = \sum_{n=0}^{t} J(n)$). Specifically, $\varphi(t) = 0$ if $t < t_{jit}$, otherwise $\varphi(t) = \Delta\phi_{jit}$. The received signal (*i.e.*, $R(t, f_{sym})$) can be modified correspondingly to include phase jitters (*i.e.*, denoted by $R_{jit}(t, f_{sym})$). Similar to Eq. (3), $R_{jit}(Dk)$ represents the samples of a chirp signal received at the down-clocking factor of D. Our goal is to detect time t_{jit} from $R_{jit}(Dk)$, using the phase of received samples.

To extract phase jitters (*i.e.*, $e^{j\varphi(t)}$) from a received chirp (*i.e.*, $R_{jit}(t, f_{sym})$), we can ideally multiply the conjugate of $S(t, f_{sym})$ with the chirp signal based on Eq. (7). However, it requires $S(t, f_{sym})$ which is not available since f_{sym} is still unknown. Recall that f_{alias}, *i.e.*, the aliased frequency of f_{sym}, has been derived using the method in Section IV-A. We can locally generate the chirp signal of $S(t, f_{alias})$ with Eq. (1). Since $f_{sym} = f_{alias} + n\frac{BW}{D},\ n = 0, \pm1, \cdots, \pm\frac{D}{2}$, we multiply the conjugate of $S(t, f_{alias})$ with the received chirp signal, which produces the following.

$$\begin{aligned} & R_{jit}(t, f_{sym}) \cdot S^{-1}(t, f_{alias}) \\ \approx\ & h(t)e^{j2\pi\Delta f_{cfo}t} S(t, f_{sym}) e^{j\varphi(t)} \cdot S^{-1}(t, f_{alias}) \\ =\ & h(t)e^{j2\pi\Delta f_{cfo}t} \cdot e^{j2\pi(n\frac{BW}{D})t} \cdot e^{j\varphi(t)} \end{aligned} \quad (8)$$

In practice, we select downclocking factor D to ensure $n\frac{BW}{D}$ is an integer. Consequently, Eq. (8) is composed of only phase jitters (*i.e.*, $e^{j\varphi(t)}$) and the item of carrier frequency offsets (*i.e.*, $e^{j2\pi\Delta f_{cfo}t}$). Fig.9(d) shows the results of applying Eq. (8) to the under-sampled chirps of two ambiguity symbols #127 and #0. We observe that phase shifts (which indicates the phase jitters of LoRa modem) happen at different time for the two symbols. We can detect the continuity of obtained phase results to extract the timing of phase jitters (*i.e.*, t_{jit}), which forms the fingerprint of symbol encoded on the chirp.

Remarks. The phase-based approach has two advantages over the frequency-based approach: (1) The phase-based approach makes use of all samples in a received chirp, rather than only the samples of short frequency leakage. It improves fingerprint extraction from 'point' detection to 'line' detection, which is more robust in the case of under-sampling. (2) The frequency-based approach performs FFT at different offsets to detect frequency leakage. The computational overhead is $O(N_s \cdot n \log n)$, where n denotes the size of FFT window and N_s the number of samples in a received chirp. In contrast, the phase-based approach is more lightweight (*i.e.*, $O(N_s)$) as it performs no FFTs.



(a) Raw phase data of symbol #127 (b) Phase diff. of adjacent samples

(c) Phase diff. of undersampled chirp (d) Extracted phase jitters

Fig. 9. (a) Raw phase data of a chirp; (b,c) The difference of phases between adjacent samples in fully-sampled and under-sampled chirps; (d) Phase jitters extracted from under-sampled chirps: ambiguity symbols experience phase shifts at different timing.

Algorithm 1 Downclocked LoRa Reception

Input: SF, BW, and downclocking factor D.
Output: The payload data of a LoRa packet.
1: Detect and receive packet at the sampling-rate of $\frac{BW}{D}$.
2: Synchronize the symbol timing of LoRa packet by searching for chirp boundary in the preamble.
3: **for** the chirp of each symbol in the payloads **do**
4: Demodulate f_{alias} from the chirp signal using Eq. (4).
5: Detect t_{jit} from the phase of received signal.
6: Determine the chunk # of f_{sym} as $n_c = \lfloor \frac{\frac{BW}{2} - \alpha \cdot t_{jit}}{\frac{BW}{D}} \rfloor$.
7: Demodulate the symbol as $f_{sym} = f_{alias} + n_c \frac{BW}{D}$.
8: **end for**
9: Decode payload data from $\{f_{sym}\}$.

D. Decoding Below the Nyquist

A under-sampled chirp can get decoded as follows. First, we employ Eq. (4) to demodulate the under-sampled chirp, which produces an aliased frequency (*i.e.*, f_{alias}). We next check the phase of received samples to extract the embedded fingerprint of symbol (*i.e.*, t_{jit}, timing of phase jitters). We exploit t_{jit} to estimate the chunk ID (*i.e.*, n_c) of symbol f_{sym} (see the aliasing model in Fig.3). Lastly, by combining f_{alias} and n_c, we recover the encoded symbol as $f_{sym} = n_c \frac{BW}{D} + f_{alias}$.

We iteratively apply the above operations to every chirp of a received packet. The demodulated symbols will be fed into a conventional LoRa decoder, which interprets the data transmitted in the packet. We present the detailed scheme of downclocked LoRa reception in Algorithm 1.

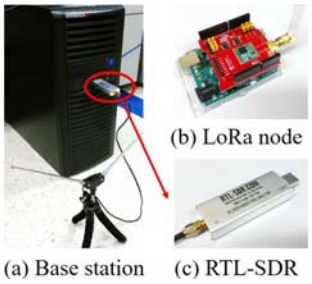


Fig. 10. Experiment devices.

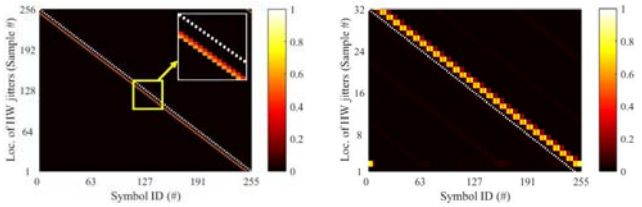


Fig. 11. The fingerprints of all 256 symbols extracted from (a) fully-sampled chirps and (b) under-sampled chirps (downclocking factor D=8), respectively. The color indicates the probability of phase jitters detected at different sample locations within the chirps of different symbols.

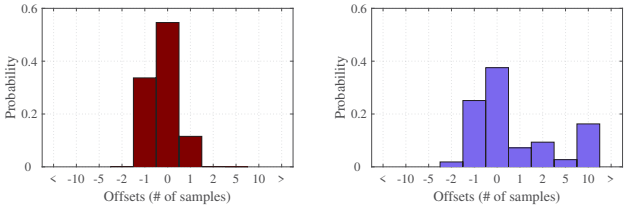


Fig. 12. Histogram of offsets (in # of PHY samples) between the extracted fingerprints and the ideal locations under the downclocking factor of D=8.

The computation overheads of symbol demodulation mainly come from FFT operations. Let N_s denote the number of samples in a normally received LoRa symbol (*i.e.*, at full clock-rate). The overhead of demodulating a fully-sampled symbol would be $O(N_s \log(N_s))$. In contrast, when the radio down-clocks by a factor of D, the sample number of a symbol decreases to $\frac{N_s}{D}$. The overhead of FFT thus becomes $O(\frac{N_s}{D} \log(\frac{N_s}{D}))$. Although demodulating an under-sampled symbol requires additional computations for fingerprint extraction, they can be completed within $O(\frac{N_s}{D})$ time. The overall computational complexity of Algorithm 1 is $O(\frac{N_s}{D} \log(\frac{N_s}{D}))$. The algorithm is lightweight and suitable for MCU operated in downclocking mode that has weaker computation capability.

V. Evaluation

We implement *LiteNap* on GNURadio based on `gr-lora` projects [15] and build a testbed using COTS LoRa nodes and software radio base station to evaluate the proposed schemes, as displayed in Fig.10. The LoRa nodes are composed of HopeRF's RFM96W transceiver with Semtech SX1276 radio chip. As the clock rate of SX1276 is fixed at 32 MHz, we use a low-cost software defined radio (*i.e.*, RTL-SDR dongle) to receive the packets transmitted by COTS LoRa nodes. The low-cost SDR is used to only receive but not transmit LoRa packets. We re-sample the received signal to emulate the downclocked reception at specific clock-rates. We implement our LoRa decoding scheme based on the GNURadio library and develop MATLAB program to process the PHY samples. We operate SDRs and LoRa nodes at the 915MHz frequency. The measured noise level is around -90dBm throughout the experiments. Unless otherwise specified, we configure LoRa nodes with SF=8, BW=250 kHz and coding rate CR=4/5.

A. Fingerprint Extraction

In this experiment, we study the characteristics of fingerprint and evaluate the performance of the proposed approaches for fingerprint extraction. We use a USRP to receive the packets transmitted by COTS LoRa nodes. We configure LoRa nodes to send packets with a duty-cycle of 2%. The packet payload consists of 37 PHY symbols. We randomly choose the payload contents and collect more than 10,000 packets to obtain sample data for all 256 symbols.

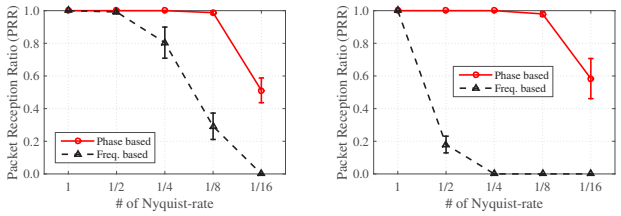
We use the phase-based approach to extract fingerprints (*i.e.*, locations of hardware phase jitters) from received chirps and associate them with corresponding symbol IDs. Fig.11 shows the distribution of the jitter locations of all 256 symbols. As expected, the jitter locations change linearly with symbol IDs. It verifies that the location of phase jitters can be used as fingerprints to uniquely identify the symbols.

Fig.11 also compares the extracted fingerprints of symbols with their expected locations as marked by the white dashed line. We observe a tiny offset between the obtained results and ideal locations. This offset varies with different LoRa nodes. But it remains constant across symbols of the same device. In practice, we can detect the offset from the preamble of LoRa packets and subtract it from the extracted jitter locations of payload symbols to obtain the correct fingerprints.

Fig.12 presents the histogram of offsets between the extracted fingerprints after being corrected with the preambles and the ideal locations. We compare the phase-based and frequency-based approaches across ten LoRa nodes with the downclocking factor of 8. As the frequency-based approach is vulnerable to noise, lots of detected fingerprints deviate away from ideal locations. In contrast, the phase-based approach is more robust. All detected fingerprints are within one sample of the ideal locations and can be used to resolve ambiguity, since aliased symbols are separated by more than one sample.

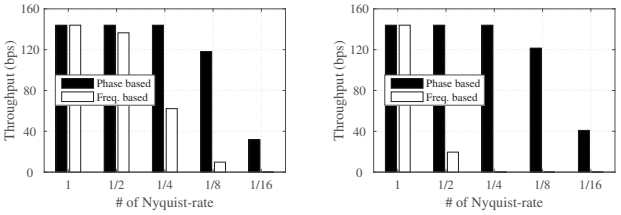
B. Packet Reception Performance

This experiment examines the communication performance of *LiteNap*. We setup a COTS LoRa node as the transmitter and a RTL-SDR dongle as the receiver. We configure LoRa node to transmits 1,000 packets (payload: 22 Bytes). RTL dongle receives at the physical sampling-rate of 1 Msps. We re-sample the received signal to emulate downclocked reception. We vary the re-sampling rate across 1, 1/2, 1/4, 1/8 and 1/16



(a) Good SNRs (b) Poor SNRs

Fig. 13. Impacts of downclocking on Packet Reception Ratio (PRR).



(a) Good SNRs (b) Poor SNRs

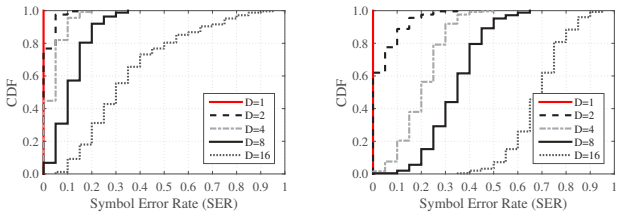
Fig. 14. Impacts of downclocking on throughput.

of the Nyquist-rate. In particular, we repeat the experiments across five LoRa nodes and two RTL-SDR dongles. We change the locations of transmitter within a 8-floor building, where the maximum transmission range can be about 150m in Non-Line-Of-Sight (NLOS), and classify received packets into two SNR regimes, *i.e.*, Poor (<5dB) and Good (≥5dB) SNRs.

We first evaluate the Packet Reception Ratio (PRR), *i.e.*, ratio of packets being successfully decoded. Fig.13 presents the PRR of *LiteNap* in different SNRs. We see that the frequency based approach produces lower PRRs. The performance of frequency based approach degrades fast when receiver downclocks to lower rates. As the SNRs become poor, the frequency-based approach cannot decode any packet when clock-rate falls below 1/4 the Nyquist's, as shown in Fig.13(b). In comparison, the phase-based approach outperforms the frequency-based approach in both good and poor SNRs. The phase-based approach can correctly decode all packets as the receiver downclocks to 1/4 the Nyquist-rate. The PRR remains above 95% when the clock-rate is 1/8 the Nyquist-rate.

The throughput of *LiteNap* exhibits a similar trend with that of PRR. As shown in Fig.14, the throughput of the phase-based approach does not change as the receiver downclocks from 1 to 1/4 the Nyquist-rate. It still achieves about 85% the throughput of full clock-rate when the clock-rate is 1/8 the Nyquist-rate. The throughput of the frequency-based approach decreases fast as clock-rate reduces to sub-Nyquist rates or when the SNRs become poor (*e.g.*, <5dB).

Fig.15 examines the link-layer symbol demodulation errors of *LiteNap* in good SNRs (we omit the results of poor SNRs due to page limit). We see that the Symbol Error Rates (SERs) of both approaches become higher as the downclocking factor (*i.e.*, D) increases. It is because demodulation errors stem mainly from the detection error of symbol fingerprint, which can cause *LiteNap* to incorrectly demodulate aliased



(a) Phase-based approach (b) Frequency-based approach

Fig. 15. Performance of downclocked symbol demodulation in good SNRs.

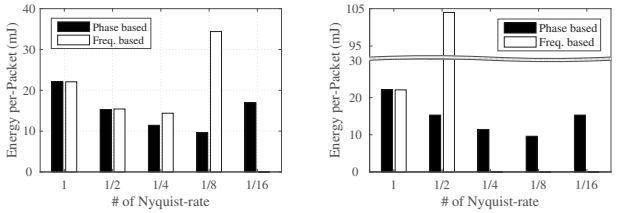
symbols. As the downclocking factor increases, the number of ambiguity symbols would increase, which in turn increases the SERs of *LiteNap*. Nevertheless, *LiteNap* can use the phase-based approach to recover 99% symbols with less than 0.1 Symbol Error Rates (SERs) when downclocking factor D=2. The ratio of SER≤0.1 decreases to 80% and 60% when D=4 and 8. Despite that, most demodulation errors can be corrected by the error correction scheme of LoRa (*i.e.*, Hamming coding [16]). This explains why the PRRs of phase-based approach are constantly above 95% when the receiver downclocks to 1/2, 1/4 and 1/8 of the Nyquist-rate (see Fig.13(a)).

Discussion: Our evaluations show that downclocking may abate the sensitivity of packet reception, which can affect long-range communications of LoRa. We note that IoT networks do not always prefer long-range communications in practice, as it may cause issues of interference and reduced capacity [17] [18]. *LiteNap* can adaptively schedule the downclocking factor to strike a balance between low-power and long-range.

C. Energy Saving

We evaluate the energy efficiency of downclocked reception through trace-driven simulations. We characterize the power consumption of LoRa radio based on the model proposed in [4]. The basic energy profiles are obtained from the data sheet of Semtech SX1276 [9]. We use the power of SX1276 as an estimate for the power consumption of fully-clocked radio. The downclocked power consumption can be estimated by proportionally scaling the power of full-clocking with respect to the downclocking factor (*i.e.*, D). We summarize the adopted configurations in Table II.

We collect traffic traces from a realistic LoRaWAN link composed of one transmitter (*i.e.*, COTS LoRa node) and one receiver (*i.e.*, RTL-SDR base station). Both transmitter and receiver run the LoRaWAN MAC protocol with Class A. We configure the LoRa node to periodically send packets (payload: 22 Bytes) with a duty-cycle of 2%. The RTL-SDR dongle employs *LiteNap* to decode received packets. If a packet is successfully decoded, the receiver will reply with an ACK. A packet would be re-transmitted if no ACK is received. We vary downclocking factor D across 1, 2, 4, 8, 16, and receive 1,000 packets under each downclocking setting. We replay the collected traffic to simulate the energy drains of both transmitter and receiver. We compute *the total energy consumption of transmitter and receiver*, then average it out to the number of received packets. We use this per-packet energy



(a) Good SNRs (b) Poor SNRs

Fig. 16. Per-packet energy consumption under different downclocking factors. For the frequency-based approach, since no packets are correctly received in the cases of D=16 in good SNRs and D=4, 8, 16 in poor SNRs (see Fig.13), the corresponding results are absent.

TABLE II
POWER CHARACTERISTICS OF DOWNCLOCKED LORA RECEPTION.

Downclocking factors	***D*=1**	***D*=2**	***D*=4**	***D*=8**	***D*=16**
Transmit power (mW)	66.00	66.00	66.00	66.00	66.00
Receive power (mW)	37.95	24.67	17.08	13.28	11.39
Standby power (mW)	5.94	3.86	2.67	2.08	1.78
Packet on-air time	35.84 $\sim$ 46.08 (ms)				
LoRaWAN on-duty time*	3 (s)				

*on-duty time = TX Win + 2×RX Win + RX Delays(idle waiting).

consumption as a metric to evaluate the energy-efficiency of downclocked LoRa reception.

Fig.16 presents the energy performance of *LiteNap*. The phase-based approach exhibits similar performance regardless of the SNR conditions. The best performance is achieved when D=8. In the case of good SNRs (see Fig.16(a)), as receiver downclocks from the Nyquist-rate to 1/8 Nyquist, the per-packet energy consumption decreases from 22.1 to 9.6 mJ (*i.e.*, reduced by 56.6%). However, the energy increases when the clock-rate decreases to 1/16 the Nyquist-rate, due to an increasing number of packet re-transmissions. On the other hand, the frequency based approach produces the optimal performance when D=4 in good SNRs, with an energy reduction of 36.5% as compared to that of D=1. However, the frequency based approach cannot reduce power consumption in poor SNRs due to re-transmissions caused by packet errors, while the phase based approach can still reduce power in poor SNRs.

Discussion: *LiteNap* can be used on both LoRaWAN base stations and regular nodes. Although LoRaWAN nodes perform more packet sending than receiving in certain IoT scenarios, *LiteNap* still benefits such LoRa nodes by downclocking radio during the states of idly-standby and channel detecting.

VI. RELATED WORK

Energy efficiency for LoRa. Many prior efforts [4], [19]–[22] had been devoted to characterize LoRa power consumption. They empirically measure the power consumption of COTS LoRa radio in various operation modes and LoRaWAN classes. Based on the measurements, researchers propose energy models for LoRa and analytically study the relationships between energy consumption and various impacting factors, such as network topology [19], duty-cycle ratio [4], configuration of communication parameters [22], *etc*. These works conclude that the energy performance of current LoRa platforms are far from optimal [4], [20], [23].

Existing works study resource scheduling [24]–[27] and parameter allocations to reduce power consumption [28]. Liando et al. [4] employ prediction models to allocate spreading factor and transmission power for LoRa. Bor et al. [29] propose a link probing scheme to select the optimal LoRa parameters. uLoRa [30] presents an ultra low-power hardware and software design. [31]–[34] employ backscatter communications to reduce the power consumption of LoRa hardware to μW level. Orthogonal to these works, this paper aims to reduce the power consumption by downclocking LoRa radio.

Downclocked communication. E-MiLi [8] downclocks WiFi radio during idle listening to reduce power consumption. SASD [35] extends E-MiLi by conveying data on preamble that can be decoded by a down-clock radio. SloMo [36] applies compressed sensing to DSSS to enable downclocked sending and receiving. Enfold [37] exploits the aliasing structure of OFDM to decode under-sampled packets. Sampless WiFi [38] combats the aliasing of OFDM by combining multiple retransmissions to recover packet data. Recently, LongBee [39] employs downclocking to concentrate signal power at narrower bandwidth for long-range communications across WiFi and ZigBee. PLoRa [33] detects LoRa packets at sub-Nyquist sampling-rates, but it does not support the decoding of under-sampled packets. To the best of our knowledge, this paper is the first study on downclocked LoRa packet reception.

VII. CONCLUSION

This paper aims to improve the energy efficiency of LoRa by enabling sub-Nyquist sampling and packet decoding. To this end, we study the frequency aliasing of under-sampled LoRa packets, which reveals that the frequency aliasing can cause ambiguity in demodulation. Fortunately, our empirical study also discovers that the timing of frequency leakage within a chirp can serve as a fingerprint to uniquely identify a symbol. More importantly, the timing information can be well reserved even when under-sampled below the Nyquist sampling rate. Based on this observation, we propose two fingerprint extraction methods to reliably detect the timing information and resolve the ambiguity caused by frequency aliasing. We evaluate the proposed methods through testbed experiments and trace-driven simulations. Results show that a down-clock receiver can reduce power consumption by up to 50%, while achieving comparable packet reception performance of a full-clock receiver in good channel conditions.

VIII. ACKNOWLEDGEMENTS

We thank anonymous reviewers for their helpful comments. This work is supported in part by the National Nature Science Foundation of China under grant 61702437 and Hong Kong GRF under grant PolyU 152165/19E, and in part by the Australian Research Council (ARC) Discovery Project Grants DP190101888 and DP180103932. Yuanqing Zheng is the corresponding author.

REFERENCES

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

[2] 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). New York, USA: IEEE, Apr 2016, pp. 1–5.

[3] 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*, 2019.

[4] 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, pp. 16:1–16:35, Feb. 2019. [Online]. Available: http://doi.acm.org/10.1145/3293534

[5] Semtech. (2019, Jun.) Semtech SX1276: 137MHz to 1020MHz long range low power transceiver. [Online]. Available: https://www.semtech.com/products/wireless-rf/lora-transceivers/sx1276

[6] M. Bor, J. Vidler, and U. Roedig, "LoRa for the Internet of Things," in *Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks*, ser. EWSN'16. USA: Junction Publishing, 2016, pp. 361–366. [Online]. Available: http://dl.acm.org/citation.cfm?id=2893711.2893802

[7] J. M. Marais, R. Malekian, and A. M. Abu-Mahfouz, "Evaluating the LoRaWAN protocol using a permanent outdoor testbed," *IEEE Sensors Journal*, vol. 19, no. 12, pp. 4726–4733, June 2019.

[8] X. Zhang and K. G. Shin, "E-MiLi: Energy-Minimizing Idle Listening in Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1441–1454, Sep. 2012.

[9] Semtech. (2019, Jun.) SX1276/77/78/79 Datasheet. [Online]. Available: https://www.semtech.com/products/wireless-rf/lora-transceivers/sx1276

[10] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl, "A case for adapting channel width in wireless networks," in *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication*, ser. SIGCOMM'08. New York, NY, USA: ACM, 2008, pp. 135–146. [Online]. Available: http://doi.acm.org/10.1145/1402958.1402975

[11] A. Demir, A. Mehrotra, and J. Roychowdhury, "Phase noise in oscillators: a unifying theory and numerical methods for characterization," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 5, pp. 655–674, May 2000.

[12] F. A. P. Figueiredo, F. S. Mathilde, F. L. Figueiredo, and F. A. C. M. Cardoso, "An FPGA-based time-domain frequency shifter with application to LTE and LTE-A systems," in *2015 IEEE 6th Latin American Symposium on Circuits Systems (LASCAS)*, Feb 2015, pp. 1–4.

[13] S. Kim and J.-W. Chong, "Chirp spread spectrum transceiver design and implementation for real time locating system," *Int. J. Distrib. Sen. Netw.*, vol. 2015, pp. 8:8–8:8, Jan. 2015. [Online]. Available: https://doi.org/10.1155/2015/572861

[14] Semtech. (2019, Jul.) SX1272/3/6/7/8: LoRa Modem Designer's Guide AN1200.13. [Online]. Available: https://www.semtech.com/uploads/documents

[15] Gr-LoRa GitHub community. (2019, Jul) gr-lora projects. [Online]. Available: https://github.com/rpp0/gr-lora

[16] P. Robyns, P. Quax, W. Lamotte, and W. Thenaers, "A multi-channel software decoder for the LoRa modulation scheme," in *Proceedings of the 3rd International Conference on Internet of Things, Big Data and Security*, ser. IoTBDS, Mar 2018, pp. 1–11.

[17] A. Rahmadhani and F. Kuipers, "When lorawan frames collide," in *Proceedings of the 12th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization (WiNTECH'18)*. New York, USA: ACM, Nov 2018, pp. 89–97.

[18] R. Eletreby, D. Zhang, S. Kumar, and O. Yagan, "Empowering low-power wide area networks in urban settings," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM'17. New York, USA: ACM, Aug 2017, pp. 309–321.

[19] M. N. Ochoa, A. Guizar, M. Maman, and A. Duda, "Evaluating LoRa energy efficiency for adaptive networks: From star to mesh topologies," in *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Oct 2017, pp. 1–8.

[20] L. Casals, B. Mir, R. Vidal, and C. Gomez, "Modeling the energy performance of LoRaWAN," *Sensors*, vol. 17, no. 10, p. 2364, Oct 2017. [Online]. Available: http://dx.doi.org/10.3390/s17102364

[21] P. S. Cheong, J. Bergs, C. Hawinkel, and J. Famaey, "Comparison of LoRaWAN classes and their power consumption," in *2017 IEEE Symposium on Communications and Vehicular Technology (SCVT)*, Nov 2017, pp. 1–6.

[22] 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, Jun 2018. [Online]. Available: http://dx.doi.org/10.3390/s18072104

[23] M. Costa, T. Farrell, and L. Doyle, "On energy efficiency and lifetime in low power wide area network for the internet of things," in *2017 IEEE Conference on Standards for Communications and Networking (CSCN)*, Sep. 2017, pp. 258–263.

[24] Z. Qin and J. A. McCann, "Resource efficiency in low-power wide-area networks for iot applications," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Dec 2017, pp. 1–7.

[25] J. Haxhibeqiri, I. Moerman, and J. Hoebeke, "Low overhead scheduling of LoRa transmissions for improved scalability," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3097–3109, April 2019.

[26] W. Gao, W. Du, Z. Zhao, G. Min, and M. Singhal, "Towards energy-fairness in LoRa networks," in *Proceedings of the IEEE ICDCS'19*, July 2019, pp. 788–798.

[27] W. Du, J. C. Liando, H. Zhang, and M. Li, "When pipelines meet fountain: Fast data dissemination in wireless sensor networks," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, 2015, pp. 365–378.

[28] B. Reynders, W. Meert, and S. Pollin, "Power and spreading factor control in low power wide area networks," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.

[29] M. Bor and U. Roedig, "LoRa transmission parameter selection," in *2017 13th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, June 2017, pp. 27–34.

[30] H. Sallouha, B. Van den Bergh, Q. Wang, and S. Pollin, "uLoRa: Ultra low-power, low-cost and open platform for the LoRa networks," in *Proceedings of the 4th ACM Workshop on Hot Topics in Wireless*, ser. HotWireless'17. New York, NY, USA: ACM, 2017, pp. 43–47. [Online]. Available: http://doi.acm.org/10.1145/3127882.3127890

[31] V. Talla, M. Hessar, B. Kellogg, A. Najafi, J. R. Smith, and S. Gollakota, "LoRa Backscatter: Enabling the vision of ubiquitous connectivity," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. UbiComp*, vol. 1, no. 3, pp. 105:1–105:24, Sep. 2017. [Online]. Available: http://doi.acm.org/10.1145/3130970

[32] A. Varshney, C. P. Penichet, C. Rohner, and T. Voigt, "Towards wide-area backscatter networks," in *Proceedings of the 4th ACM Workshop on Hot Topics in Wireless*, ser. HotWireless'17. New York, NY, USA: ACM, 2017, pp. 49–53. [Online]. Available: http://doi.acm.org/10.1145/3127882.3127888

[33] Y. Peng, L. Shangguan, Y. Hu, Y. Qian, X. Lin, X. Chen, D. Fang, and K. Jamieson, "PLoRa: A passive long-range data network from ambient LoRa transmissions," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM'18. New York, NY, USA: ACM, 2018, pp. 147–160. [Online]. Available: http://doi.acm.org/10.1145/3230543.3230567

[34] 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)*. Boston, MA: USENIX Association, 2019, pp. 271–284. [Online]. Available: https://www.usenix.org/conference/nsdi19/presentation/hessar

[35] T. Xiong, J. Yao, J. Zhang, and W. Lou, "It can drain out your energy: An energy-saving mechanism against packet overhearing in high traffic wireless LANs," *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1911–1925, July 2017.

[36] F. Lu, G. M. Voelker, and A. C. Snoeren, "SloMo: Downclocking WiFi Communication," in *Proc. of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI'13)*, 2013, pp. 255–258.

[37] F. Lu, P. Ling, G. M. Voelker, and A. C. Snoeren, "Enfold: Downclocking OFDM in WiFi," in *Proc. of the 20th Annual International Conference on Mobile Computing and Networking (MobiCom'14)*, 2014, pp. 129–140.

[38] W. Wang, Y. Chen, L. Wang, and Q. Zhang, "Sampleless Wi-Fi: Bringing low power to Wi-Fi communications," *IEEE/ACM Transactions on Networking*, vol. 25, no. 3, pp. 1663–1672, June 2017.

[39] Z. Li and T. He, "Longbee: Enabling long-range cross-technology communication," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, April 2018, pp. 162–170.