Spreading Factor Detection for low-cost Adaptive Data Rate in LoRaWAN Gateways

Daniel Jay Koch
University of Wisconsin-Madison
dkoch5@wisc.edu

Muhammad Osama Shahid
University of Wisconsin-Madison
mshahid2@wisc.edu

Bhuvana Krishnaswamy
University of Wisconsin-Madison
bhuvana@ece.wisc.edu

ABSTRACT

In order to meet the capacity needs of LoRa networks, Adaptive Data Rate (ADR) has been proposed and implemented in LoRaWANs. The network server running ADR determines the optimum datarate and hence spreading factor setting for each LoRa device in a network. This in turn requires the gateway to be capable of receiving all possible spreading factors. Existing gateways achieve this by using multiple RF front ends, increasing their overall cost and complexity. In this work, we propose a Discrete Wavelet Transform based spreading factor detection algorithm that is agnostic to transmitter settings. This computationally light-weight algorithm can be implemented on any off-the-shelf SDR, bringing down the cost and ease of LoRaWAN gateway implementations. Using experimental, real-world datasets, we show that the proposed algorithm can detect the spreading factor of over 99.5% of the received packets at SNRs down to -10dB.

CCS CONCEPTS

• **Networks → Wireless access points, base stations and infrastructure**; **Packet classification**.

KEYWORDS

LPWAN, LoRa, Discrete Wavelet Transform, Adaptive Data Rate

ACM Reference Format:
Daniel Jay Koch, Muhammad Osama Shahid, and Bhuvana Krishnaswamy. 2022. Spreading Factor Detection for low-cost Adaptive Data Rate in LoRaWAN Gateways. In *The 20th ACM Conference on Embedded Networked Sensor Systems (SenSys '22), November 6–9, 2022, Boston, MA, USA*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3560905.3568112

1 INTRODUCTION

The popularity of LoRa [1] has highlighted the need to support large-scale deployments of end devices spread across a wide area [2–4]. LoRaWAN [5] was proposed as the Medium Access Control (MAC) layer to address the growing capacity needs of LoRa networks. LoRaWAN architecture connects the end devices to a network server through gateways. The end devices implement an ALOHA-based MAC protocol to communicate with the gateway, and are hence not associated with a specific gateway. The gateway demodulates LoRa samples, converts to IP packet, and relays them to a network server through the internet backhaul.

To meet the connectivity and capacity needs of networks, multiple LoRa gateways have been proposed as a solution [6]. If multiple copies of the same message are received by the network server, only one of the duplicates are forwarded to applications. Thus, the network server keeps track of messages from the same end devices through multiple gateways. The benefits of multiple gateways were experimentally shown in a case study by LoRa [7]. However, simply adding more gateways is a costly solution. This led to the development of Adaptive Data Rate (ADR) by LoRa alliance [8]. ADR optimizes network capacity and energy consumption of end devices by allowing gateways to dictate the transmit power, SF, bandwidth and coding rates of end nodes based on the history of their received Signal to Noise Ratios (SNR). ADR is implemented at the network server; this allows gateways to intelligently reduce transmit powers of end devices to only be heard by the closest gateway, while also setting SF to improve overall network throughput. [9].

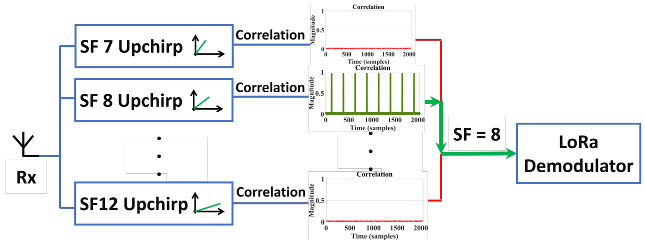
The data-rate of a LoRa end device can be set by three key factors: 1) Spreading Factor (SF) 2) Bandwidth (BW) and 3) Coding Rate (CR). LoRaWAN standards allow SF to takes values from [7, 8, 9, 10, 11, 12], a BW from [125kHz, 250kHz, 500kHz] and a CR from [4/5, 4/6, 4/7, 4/8]. Of these parameters, SF is unique to LoRa modulation. By changing their SF, LoRa devices can directly trade-off their range for their data-rate. Therefore, to successfully implement ADR in a LoRaWAN network, the end devices will often vary their data-rate through their individual SFs. However, unlike WiFi, where the data-rate is indicated in the physical layer header of every packet, a LoRa end device transmits the entire packet, including its preamble using a predetermined SF. Therefore, in an ADR network, the gateway must be capable of receiving LoRa packets with any of the six SFs. Current LoRa gateways achieve this by implementing multiple, parallel radio frontends [10], each continually demodulating using a specific SF. Thus a LoRa gateway capable of ADR correlates raw I/Q samples with the preamble of each SF (7 through 12) to detect the SF of the transmitter.

This expensive requirement has led to the development of LoRa gateways costing over 100s and even 1000s of dollars, depending on the number of parallel SFs and channels that can be processed [11, 12]. The focus of this work is to connect LoRa devices to a network server with ADR while only using a single low-cost LoRa gateway implemented on relatively inexpensive software defined radios (SDR) [3, 13, 14]. **Particularly, we focus on the problem of detecting the spreading factor of a single received LoRa packet at the gateway without the prior knowledge of the transmitter settings in real-time using a single RF front end.**

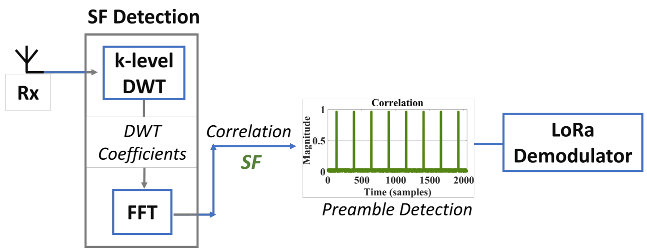
We propose a Discrete Wavelet Transform (DWT) based SF detector that can estimate the SF of a received packet without any prior

ENSsys '22, November 6, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9886-2/22/11…$15.00
https://doi.org/10.1145/3560905.3568112



(a)



(b)

Figure 1: (a) Existing Technique for SF detection (b) Proposed Technique for SF detection

knowledge of the transmitter settings. The proposed DWT-based SF detection can be implemented using off-the-shelf SDR in conjunction with software defined LoRa demodulators and decoders. Thus, a low-cost SDR with the ability to detect SF can open the research community to explore more challenges and opportunities in ADR for LoRaWAN without the need for expensive ADR-capable base stations.

Towards our goal of low-cost LoRaWAN gateways, we make the following contributions:

- We propose a low complexity, Discrete Wavelet Transform (DWT) based spreading factor detector to estimate the data-rate settings of LoRa end devices without prior knowledge of the transmitter settings.
- We perform practical experiments and show that the proposed algorithm can detect all SFs making it suitable for practical deployments with ADR.
- We show that the proposed algorithm can correctly detect all SFs for 99.5% of received packets at or above -10dB of SNR, displaying its potential for use in long-range, wide area networks.

2 AN OVERVIEW OF LORAWAN AND ADR

CSS Modulation. LoRa uses Chirp Spread Spectrum (CSS) scheme to modulate data. Data symbol is derived by cyclically shifting a base chirp by a frequency f_{sym}. Base chirp's frequency increases linearly from $\frac{-BW}{2}$ to $\frac{BW}{2}$ over a symbol duration T_s where BW is the bandwidth of transmission and T_s can be defined as $T_s = \frac{2^{SF}}{BW}$. $SF \in \{7, 8, 9, 10, 11, 12\}$ defines a packet's Spreading Factor, a value that dictates both data-rate and resilience to interference.

In order to demodulate LoRa's symbols, the receiver starts by determining the boundaries of symbols within a packet by searching for its preamble. The LoRa preamble comprises of a sequence of $N = 6$ to 65535 consecutive base chirps, followed by two SYNC

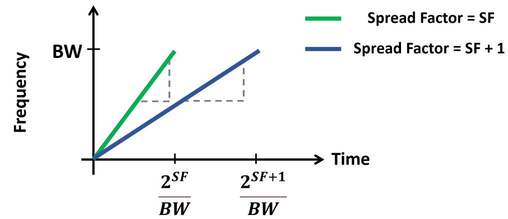


Figure 2: Time-Frequency Slope of different SFs

symbols, 2.25 down-chirps and then the data symbols. LoRa packet structure does not contain Spreading factor information explicitly. To detect a new packet, the receiver has to continuously de-chirp the received buffer with downchirps of S= 7,8, . . . , 12 and performs an FFT to each SF stream until it finds N consecutive peaks with the same frequency. Alternatively, recent works [15–17] rely on correlating received buffer with base chirp to detect preamble sequence. The SYNC words and down-chirps then help locate the symbol boundaries, followed by de-chirping and FFT.

Adaptive Data Rate (ADR): LoRaWAN supports ADR [5], through which it assigns or changes the data rates of end nodes based on their proximity to the gateway and their link quality. As mentioned earlier, end nodes can transmit at different SFs, which in turn determines the data rate. If an end node is closer to the gateway, it is generally assigned a lower SF which promises low time-on-air for packets i.e. higher data rates. Whereas the end nodes not enjoying a good link quality are given higher SF, which ensures connectivity at the cost of more airtime, i.e. low data rates.

Existing SF Detection in LoRaWAN Gateways: To support ADR in a network, LoRaWAN will distribute SFs across multiple end nodes to optimize throughput and minimize airtime. Upon receiving any arbitrary SF packet, gateways will lack prior knowledge of what SF to demodulate. As discussed earlier, to detect every possible SF, ADR enabled gateways must correlate the received buffer with Upchirps of SF= 7, 8, . . ., 12 as shown in Fig. 1(a), or dechirp the buffer with Downchirps of each SF. The upchirp whose SF matches the SF of the packet in the received buffer yields 8 consecutive correlation peaks that are 2^{SF} samples apart. The gateway looks for such peaks across all the branches and upon detecting these peaks finalizes the packet start time and the corresponding SF. The packet start time is then used to demodulate the packet.

3 DISCRETE WAVELET TRANSFORM TO DETECT SPREADING FACTOR

In this work, we propose to use Discrete Wavelet Transform (DWT) followed by Fast Fourier Transform (FFT) to detect the SF and hence the data-rate settings of a LoRa packet, without correlating the buffer with downchirps of all SFs. We leverage the periodicity of DWT coefficients that is unique to each SF to determine the data-rate settings of a given packet. Since LoRa uses CSS modulation, the frequency of a symbol increases linearly with time. SFs define the slope of this linear increase. For example in Fig. 2, given a BW, packets of different SFs have different slopes. This slope governs the rate at which the frequency varies, and correspondingly varies the duration of individual chirps. Given the preamble of LoRa packets contain repeated chirps, we can infer SF from the unique frequency-duration signature of any given packet. We extract this information using a light-weight, time-frequency analysis algorithm and estimate the SF without relying on dechirping or correlation.

3.1 A Primer on DWT

The Wavelet Transform is a multi-resolution technique that decomposes an input signal using a set of mutually orthogonal basis functions which are shifted and scaled versions of a 'mother wavelet'. Each shifting and scaling of the mother wavelet provides a unique band-pass spectrum that can correlate with the input signal's spectral content at that time and frequency. These various scaled wavelets can be converted to high and low-pass filters that can efficiently decompose a signal's frequency and time content. Different mother wavelets have been designed for specific applications ranging from computational efficiency to specific input signal correlation.

The Discrete Wavelet Transform (DWT) is a popular, low complexity variant of the Wavelet Transform, that is used for time-frequency analysis and compression of digital signals [18]. We propose to use DWT to infer the spectral content and inturn the SF of a LoRa packet. DWT extracts spectral content by dyadically applying a discretized mother wavelet's low and high-pass filters on a signal. Fig. 3 shows a k-level DWT that uses a series of high and low pass filters $h[n]$ and $g[n]$ respectively to decompose a signal to fewer and fewer coefficients at each level, where the low-pass coefficients are downsampled and iterated upon again. Initial decomposition levels provide high time-resolution but poor frequency resolution, whereas later levels trade time resolution for higher frequency resolution.

Given an input signal of length N, the above described filter bank implementation of DWT could be implemented in $O(N)$ time complexity. DWT's lower computational complexity and its ability to provide time-frequency resolution makes it an excellent choice for extracting SF from chirp signals in real-time.

3.2 Periodicity of DWT coefficients

Typically, the raw samples at the LoRa gateways are oversampled in order to capture hardware based frequency offsets and time offsets. Therefore, the DWT coefficients corresponding to lower levels are mostly attributed to high frequency noise. However, as the order of the level increases, we begin to observe a periodicity in the DWT coefficients. Fig. 4 shows the DWT coefficients of an SF 10 and an SF 11 signal respectively, particularly over a preamble portion. A periodic pattern in the DWT coefficients can be clearly seen in the levels 3 through 5+ in Fig. 4(a) and (c) for both the SFs. While the pattern looks similar for both SFs, the time period over which the parabolic pattern repeats is different for both SFs. **This periodicity corresponds to the symbol duration of the packet and hence is unique to each SF.** More precisely, the period of an SF11's single level DWT coefficients are twice that of an SF10 signal's coefficients. We then leverage an FFT to capture this periodicity and uniquely identify the SF of any received packet without prior knowledge of the transmitter settings.

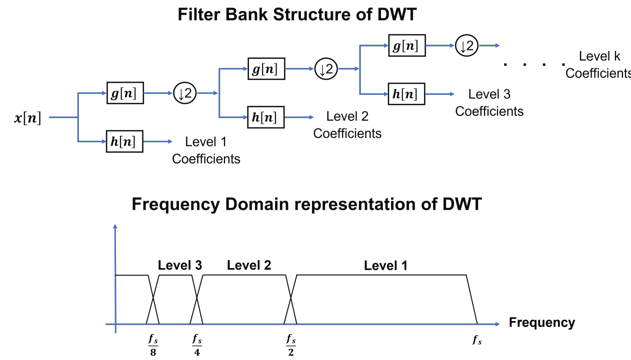


Figure 3: DWT Filter Bank Representation

In this work, we focus on DWT-based SF detection for LoRaWAN Class B and Class C devices, where the gateway transmits a synchronization beacon to schedule uplink data transmission. Class A LoRaWAN devices follows ALOHA-based MAC protocol where end nodes initiate the transmission, while class B and C utilize a slotted ALOHA protocol. Because class B and C end devices are synchronized to the base station our algorithm can easily run on the portion of a received buffer corresponding to any received packet's preamble. Our algorithm assumes that there exists some packet in the receive buffer at these time slots and only acts as an SF estimate, not a packet detector. The SF estimate can then be provided to any LoRa demodulation and decoding technique running on the base station before being sent to the cloud.

Fig. 1(b) illustrates the components of our proposed SF detection algorithm. On receiving raw I/Q samples in the receive buffer, we compute an N-level DWT. The DWT level of interest is determined by the oversampling factor. An oversampling factor of 4X would render meaningful time-frequency resolution in Level 3 and a factor of 8X would lead to meaningful information in Level 4. For a given bandwidth, the oversampling factor is known at the gateway and hence the corresponding level is determined. Let us consider an oversampling factor of 4X in the rest of this discussion; Level-3 DWT coefficients convey the time-frequency components of the received signal. Therefore, performing an FFT on Level-3 coefficients will reveal the periodicity of chirp signals, and correspondingly the SF. However, the length of this FFT window will determine the accuracy of detecting the above mentioned periodicity.

We identify the optimum FFT window size to be equal to 8 symbols of SF10 (this corresponds to the preamble length of an SF10 packet. This choice is experimentally justified in section 4.2). Within the FFT, we compute the index of the maximum peak which tells us about the SF of the underlying packet (assuming there always is one). For example, the Level-3 DWT Coefficients and their corresponding FFT of an SF10 packet is shown in Fig. 4(a) and Fig. 4(b) respectively. The index of the maximum is 8Hz, i.e, the periodicity of DWT coefficients is 8Hz. On the other hand, Level-3 DWT Coefficients and the corresponding FFT of an SF11 packet is shown Fig. 4(c) and Fig. 4(d) respectively; here, the index of maximum peak is 4Hz. Since SF11 chirps have twice the symbol duration as that of SF10, its corresponding DWT coefficients display a frequency half that of SF10. By comparing Figs 4(b) and (d), the frequency peak

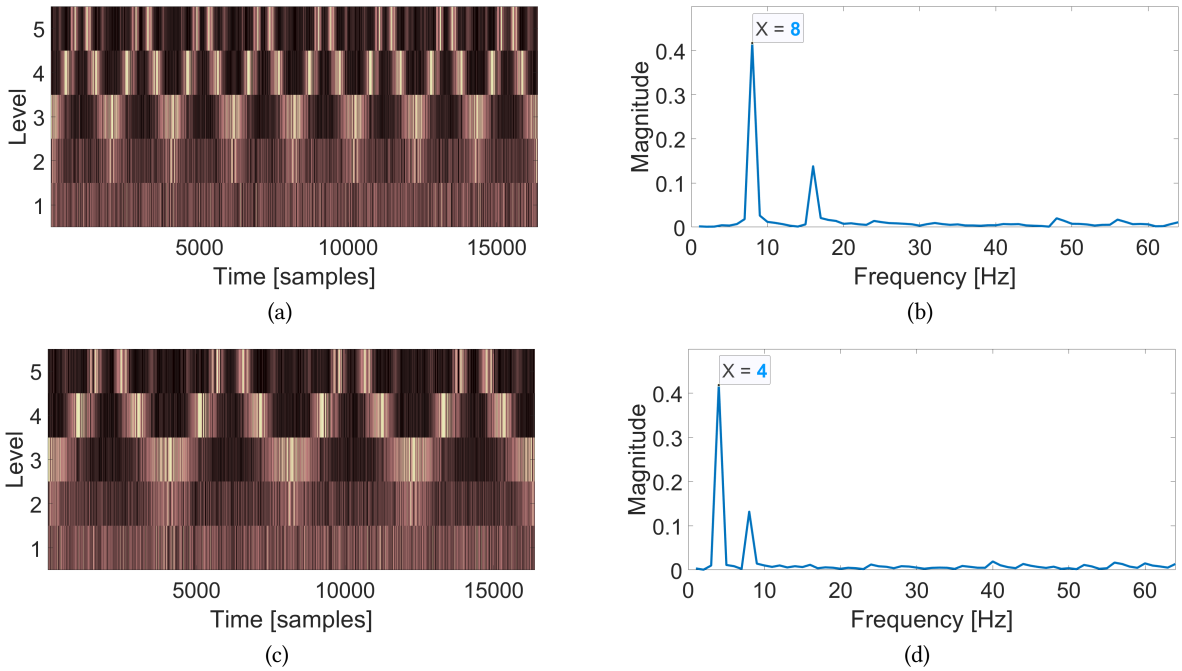


Figure 4: 5-Level DWT Coefficients of LoRa Signal with different SF (a) SF = 10, 5-Level DWT Coeff. (b) SF = 10, FFT of 3rd level Coeff. (c) SF = 11, 5-Level DWT Coeff. (d) SF = 11, FFT of 3rd level Coeff.

for (d) is exactly half of that of (b). [1] Thus, we uniquely map the FFT peak of DWT coefficients to the SF of the received packet. Our results in Section 4 show that the proposed algorithm extracts the maximum peak index which identifies accurate SF of LoRa signal at SNRs lower than -20dB.

FFT window selection algorithm: In order to track the periodicity of DWT coefficients, we need at least 2 symbols for all SFs to be lying in our window, so that FFT is able to capture the frequency with high resolution. A preamble of SF10 contains at least 2 symbols of SF12 and 64 symbols of SF7 (ref. Fig. 2). Our choice of 8 SF10 symbols ensures that at least 2 symbols from the preamble for each SF would lie within our window and it's periodicity be captured by FFT.

3.3 Computational Complexity

Our proposed framework significantly reduces the number of operations performed to detect SF unlike in current LoRa gateways, as evident in Fig. 1s (a) and (b). Existing gateways perform correlation for each individual SF to detect incoming packets. Given M SFs and a received buffer of length N, it takes $O(MN^2)$ computations to accurately detect every incoming packet. Whereas our proposed method consists of only a single operation for all SFs, and identifies the SF with only $O(N + NlogN)$ run-time.

[1] We also observe some harmonics in high SNR packets, but these harmonics are always less in magnitude than the chirp frequency and showed no hindrance to our results in practice.

4 EVALUATION

We evaluate the accuracy of DWT-based SF detection using real-world traces of LoRa packets across all SFs and over a wide range of SNRs. We show that we can detect the SFs of close to 100% of all packets received at SNRs greater than -10 dB. We further evaluate the impact of DWT wavelet family selection, and the length of FFT window on the accuracy of SF detection.

Experimental Setup. We collected samples using USRP B200 [19], sampling at 1 MSamples per second in an outdoor setup, with COTS LoRa transceivers [20] sending packets with BW = 125kHz and SF ranging form 7 through 12 to emulate an ADR driven network. Each trace lasts 5 minutes. The distance between the transmitter and the receiver was varied between 2 meters and 100 meters, both line-of-sight and non-line-of-sight to emulate SNR variations. To simulate longer network distances, the LoRa node's transmit power was restricted to the hardware minimum of 2dBm. To determine the ground truth, the captured trace was passed through six standard LoRa demodulators and decoders (one of each spreading factor). The six demodulators were able to successfully detect and decode a total of 1694 total packets. As can be seen in Fig. 5, these packets span an SNR range from -30dB to +30dB with an average of 0dB.

Implementation. The proposed SF detector is implemented as a python function utilizing the pyWavelets library for the initial DWT, and Numpy for the chirp frequency calculation. Overall this simple-yet-efficient function consists of only 5 lines of python code and can run on any low-cost SDR I/Q stream in real-time. While in post-processing, the algorithm qualitatively can more than keep up with our collected 8x over-sampled data. Thus promising real-time

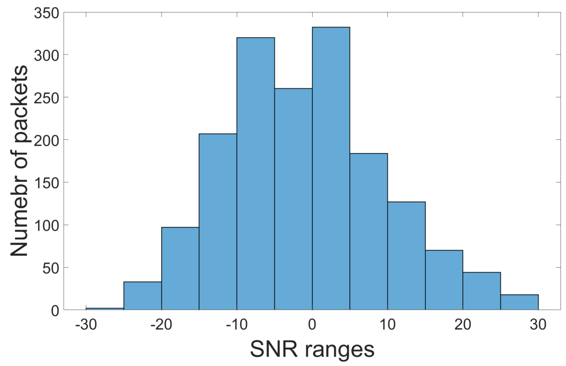


Figure 5: SNR distribution of collected LoRa packets

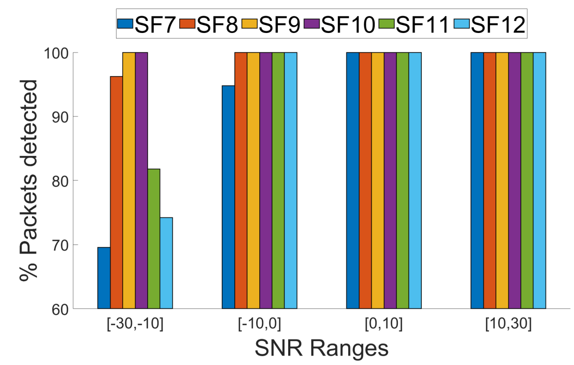


Figure 6: SF predictor accuracy over various ranges of SNRs

operation in practical deployments even at 1 MSamples per second sampling frequency. We choose biorthogonal wavelets in the rest of our implementation, as well as an SF10 8-chirp preamble window size for our FFT. These configuration parameters are discussed below in 4.2 and 4.3 respectively.

4.1 Overall SF Detector Accuracy

We perform DWT-based SF detection on the 1694 packets collected from our real-world trace. The SNR of these 1694 packets varied from -30dB to +30dB while SF sweeps from 7 through 12. The overall accuracy of our proposed SF detection against standard LoRa's parallel front-end detection is presented in Fig. 6 for various SNR ranges and SFs. These results show the excellent performance in the SF detector in accurately determining a single packet's SF without any prior knowledge about the transmitter. Overall our design showed 99.5% accuracy across all spreading factors with an SNR over -10dB, and 100% accuracy across all spreading factors over -5dB. As can be seen in Fig. 6, the detection accuracy for SF9-10 packets was 100% even down to -20dB of SNR. Due to the limitations of the window size (further evaluated below), we do witness a drop in accuracy for the largest and the smallest SF packets at the moment. We plan to explore this further in our future work to improve the accuracy of all SFs to 100% even in these ultra-low SNR ranges. It must be noted the achieved accuracy in detecting SFs was done at the low computation cost of one DWT+FFT, and low hardware cost of one SDR front-end. As opposed to multi-channel LoRa gateways with parallel front-ends, we are able to detect majority of the packets using COTS software defined radios.

4.2 Impact of FFT Window Size

As discussed in §3.2, the FFT window size determines the resolution of periodicity, in turn the accuracy of SF detection. While the ideal window size would be equal to a packet's preamble length, this length will vary depending on the SF, creating a chicken-egg problem. For example: if a selected window is only as long as an SF7 preamble, packets of SF10 will only fit a single chirp, and packets of SF11 and 12 will not even fit in this window (thus they are undetectable). Conversely, if a window is the length of an SF12 preamble, an SF7 preamble will only make up at most 1/32nd of the overall FFT energy, and will be buried in the noise of its own

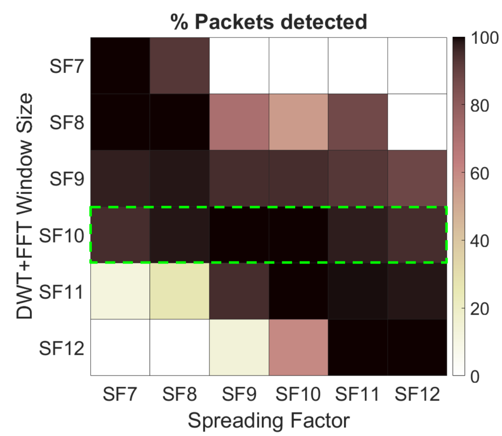


Figure 7: Impact of FFT window size on SF detector accuracy. The optimum window size is highlighted.

data symbols. With these issues in mind, we decided an optimal window would be around 8 SF10 chirps long.

We verify the correctness of our design choice in Fig 7. Here, we present a heat map of the percentage of packets detected for each SF in x-axis, for varying DWT+FFT window sizes in y-axis. It can be seen that SF7 packets are detected with the highest accuracy when the window length is equal to SF7 and similarly for other SFs. We can also clearly see that when a small window such as an SF7 preamble is used, larger packets are almost never properly detected. Conversely, SF11-12 sized windows detect <50% of SF7-8 packets while detecting >95% of SF11-12 packets. As can be seen in the darkened diagonal line in Fig. 7, a window with the same length as a preamble will yield the most accurate results.

We selected the window length that provided the most accuracy across all 6 SFs which, as assumed, happened to be an SF10 preamble (dashed box). This window size is able to still accumulate two chirps of SF12 preambles, while SF7 preambles will at least account for 1/8th of the overall window's energy.

4.3 Effect of Wavelet Families

DWT allows users to pick and choose among various families of "Mother-Wavelets". The choice of mother wavelet changes the frequency response of the low pass and high pass filters in the filter

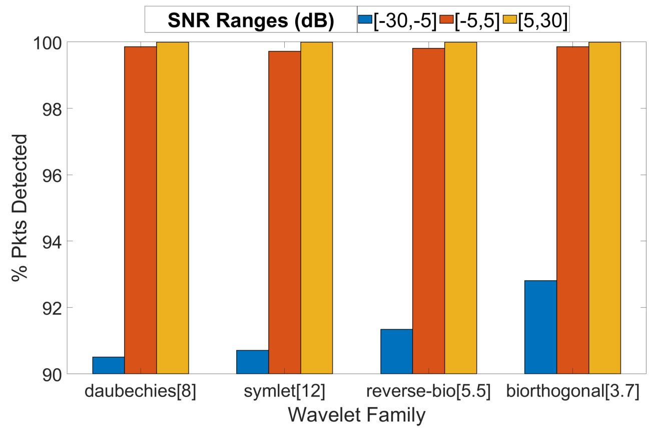


Figure 8: SF detector accuracy as a function of Wavelet-Families over various SNR ranges.

bank implementation(Fig. 3). In this section, we evaluate the impact of various wavelet families on the accuracy of SF detection. Fig. 8 plots the SF detection accuracy over all SFs for various SNR ranges as a function of four wavelet families viz., Daubechies, Symlet, Bi-orthogonal, and Reverse Bi-orthogonal. While all of them have comparable detection accuracy at high SNRs, at extremely low SNR, biorthogonal has the highest accuracy. Based on our exhaustive search over all available wavelet families, Bi-orthogonal performs the best in majority of the scenarios. We believe further analysis on the theoretical and experimental evaluation of the impact of wavelet families will further improve the detection accuracy.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we present a DWT-based SF detection algorithm that enables an inexpensive SDR to replace high-end LoRa gateways. The low complexity of the DWT significantly reduces the number of computations compared to existing solutions. We hope that this work will allow researchers to use SDRs to implement and improve Adaptive Data Rate algorithms in actual LoRaWAN deployments, as there is plenty of room to improve LoRa network throughput. We also believe that there is room for further research to improve the proposed algorithm and the achieved results. We plan on addressing the following as part of our future work.

- At SNRs < -10dB, SF detection suffers due to the choice of FFT window sizing. Our current FFT window encompasses only 2 symbols of SF12 packets, whereas a larger window would improve the detection accuracy of these packets. This would lead to higher FFT peaks and thus further improve SF detection at lower SNRs. We plan on working with multiple FFT sub-windows to accumulate larger SFs energy over a long period of time, while not forfeiting the accuracy of smaller FFT windows on smaller SFs.
- Due to hardware inaccuracies as well as DWT filter transition-bands, chirp energy leaks across adjacent DWT levels as shown in Fig. 4(a), resulting in harmonics. Further work needs to be done on either suppressing these harmonics to obtain clean FFT peaks, or accumulating signal energy from these harmonics to potentially boost SNR robustness.
- We currently use standard signal processing to infer SF from the distribution of FFT peaks. Simple machine-learning algorithms could be trained on these FFT peaks to potentially further boost performance.
- The current design assumes course synchronization i.e. it detects SF on a received buffer in time slots followed by beacons (per LoRaWAN Class B and C). We plan on presenting a similar framework for LoRaWAN Class A devices as well where end nodes are completely un-synchronized with the base station. Initial results show that as long as a preamble is contained within the bounds of the FFT window, accuracy is not hindered.
- Our current implementation assumes prior knowledge of the bandwidth of operation. Due to DWT's dyadic filter decomposition, incoming packet bandwidths can also be estimated through simple level-wise energy comparisons, allowing us to expand on the range of variables ADR can change.. We have estimated signal bandwidth by simply comparing average energy at each potential bandwidth's corresponding DWT decomposition level.
- ADR allows multiple end devices to transmit simultaneously. We plan to expand this work to detect in the presence of inter-SF packet collisions.

6 ACKNOWLEDGMENTS

We would like to thank our anonymous reviewers for the valuable comments and helping us improve the paper. The authors are partially supported through the following NSF grants : CCSS-2034415, CNS - 2107060, CNS-2142978, CNS-2213688, CNS-2112562, and Wisconsin Alumni Research Foundation awards.

REFERENCES

[1] LoRa. https://www.semtech.com/lora.
[2] Branden Ghena, Joshua Adkins, Longfei Shangguan, Kyle Jamieson, Philip Levis, and Prabal Dutta. Challenge: Unlicensed lpwans are not yet the path to ubiquitous connectivity. In *The 25th Annual International Conference on Mobile Computing and Networking*, pages 1–12, 2019.
[3] Adwait Dongare, Revathy Narayanan, Akshay Gadre, Anh Luong, Artur Balanuta, Swarun Kumar, Bob Iannucci, and Anthony Rowe. Charm: exploiting geographical diversity through coherent combining in low-power wide-area networks. In *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 60–71. IEEE, 2018.
[4] Artur Balanuta, Nuno Pereira, Swarun Kumar, and Anthony Rowe. A cloud-optimized link layer for low-power wide-area networks. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, pages 247–259, 2020.
[5] What is LoRaWAN. https://lora-alliance.org/about-lorawan/.
[6] LoRa and LoRaWAN overview. https://lora-developers.semtech.com/documentation/tech-papers-and-guides/lora-and-lorawan/.
[7] Ross Gilson and Michael Grudsky. Lorawantm capacity trial in dense urban environment. 2017.
[8] Adaptive Data Rate. https://www.thethingsnetwork.org/docs/lorawan/adaptive-data-rate/.
[9] Understanding Adaptive Data Rate. https://lora-developers.semtech.com/documentation/tech-papers-and-guides/understanding-adr/.
[10] SX1301 Datasheet. https://www.semtech.com/products/wireless-rf/lora-core/sx1301#datasheets.
[11] 8 Channel LoRa Gateway. https://www.adafruit.com/product/4327.
[12] Multi Channel LoRa Gateway. https://connectedthings.store/gb/?SubmitCurrency=1&id_currency=3.
[13] Moein Khazraee, Yeswanth Guddeti, Sam Crow, Alex C Snoeren, Kirill Levchenko, Dinesh Bharadia, and Aaron Schulman. Sparsdr: Sparsity-proportional backhaul and compute for sdrs. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, pages 391–403, 2019.
[14] Revathy Narayanan and Swarun Kumar. Revisiting software defined radios in the iot era. In *Proceedings of the 17th ACM Workshop on Hot Topics in Networks*, pages 43–49, 2018.

[15] Shuai Tong, Jiliang Wang, and Yunhao Liu. Combating packet collisions using non-stationary signal scaling in lpwans. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, pages 234–246, 2020.

[16] Muhammad Osama Shahid, Millan Philipose, Krishna Chintalapudi, Suman Banerjee, and Bhuvana Krishnaswamy. Concurrent interference cancellation : Decoding multi-packet collisions in lora. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, SIGCOMM '21. Association for Computing Machinery, 2021.

[17] Yinghui Li, Jing Yang, and Jiliang Wang. Dylora: Towards energy efficient dynamic lora transmission control. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, pages 2312–2320, 2020.

[18] Ingrid Daubechies. *Ten lectures on wavelets*. SIAM, 1992.

[19] USRP B200. https://www.ettus.com/all-products/ub200-kit/.

[20] Adafruit Feather M0 with RFM95 LoRa Radio. https://www.adafruit.com/product/3178.