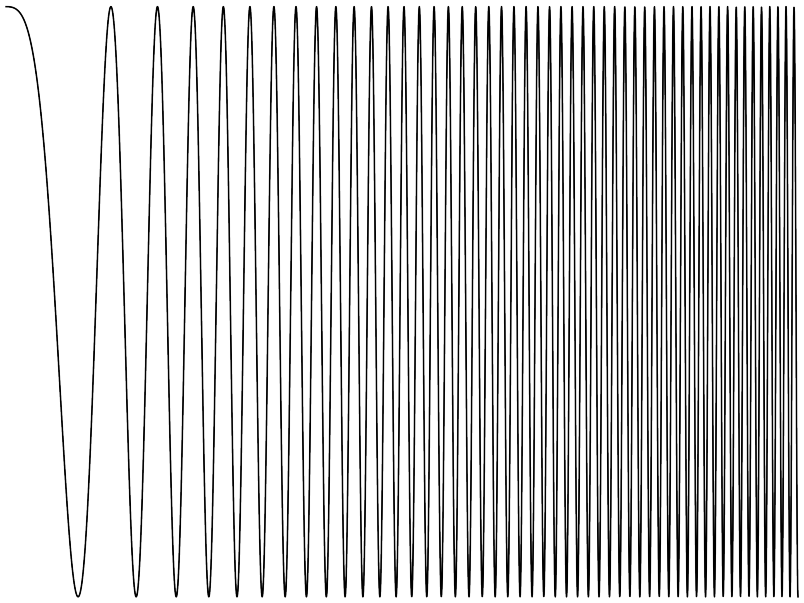
CHALMERS
UNIVERSITY OF TECHNOLOGY



Exploring LoRa and LoRaWAN

A suitable protocol for IoT weather stations?

Master's thesis in Communication Engineering

Kristoffer Olsson & Sveinn Finnsson

Department of Electrical Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2017

Master's thesis 2017:09

Exploring LoRa and LoRaWAN

A suitable protocol for IoT weather stations?

Kristoffer Olsson & Sveinn Finnsson

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Division of Communication and Antenna Systems
Chalmers University of Technology
Gothenburg, Sweden 2017

Exploring LoRa and LoRaWAN
A suitable protocol for IoT weather stations?
Kristoffer Olsson & Sveinn Finnsson

Supervisors: Árni Alfreðsson, Electrical Engineering & Anders Olsson, ALTEN Sverige
Examiner: Erik Ström, Electrical Engineering

Master's Thesis 2017:09
Department of Electrical Engineering
Division of Communication and Antenna Systems
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2017

Exploring LoRa and LoRaWAN
A suitable protocol for IoT Weather stations?
Kristoffer Olsson & Sveinn Finnsson
Department of Electrical Engineering
Chalmers University of Technology

Abstract

Svenska Sjöräddningssällskapet (SSRS) maintains a mobile-phone application that provides up-to-date weather information to seafarers in Sweden. In order to increase the granularity of the weather data that powers the application, they wish to place simple weather stations at popular sailing destinations in the archipelagos surrounding Sweden.

In this thesis we examine a new radio protocol called LoRa and the accompanying low power wide area network protocol LoRaWAN. The aim of the thesis is to evaluate if and how these protocols can be used for the purpose of transmitting weather data from simple IoT weather stations. Furthermore, we wish to discuss and present a specification to extend the effective range of the network.

The LoRa protocol is examined, along with the theory behind the chirp spread spectrum modulation, which LoRa exploits. The network layer protocol LoRaWAN and its structure is presented and shortly explained. We discuss how this structure can be utilized for testing of the protocol and for our use-case. Furthermore, packet error rate testing is performed between an RN2483 transceiver and a Kerlink gateway. Utilizing the results from this testing, we discuss and create a specification for network range extending intermediate-nodes. In addition to the specification, we provide insight into suitable placement of the IoT weather stations and intermediate-nodes for good network coverage.

The LoRa protocol and the accompanying LoRaWAN network protocol is found to be useful for the intended IoT weather stations. Furthermore, we find that our suggested network range extending specification is a good fit for the intended weather station network, but the intermediate-nodes introduce some limitations to the network when compared to gateways.

Keywords: LoRa, LoRaWAN, IoT, LPWAN, Weather, Station, Network, Extension.

Acknowledgements

We thank ALTEN Sverige for supplying us with the problem presented in this thesis as well as the resources necessary for its fulfillment. Furthermore, we thank Anders Olsson at ALTEN Sverige for his guidance and support during the thesis. In addition we thank our supervisor Árni Alfreðsson at Chalmers for providing us guidance and assistance during our thesis project. We would also like to thank Patrik Särenfors at Indesmatech (Semtech representative Nordic) for his support on making the gateway (Kerlink IoT station) function properly in mobile mode. Finally, we thank Þórhildur Hafsteinsdóttir for proof-reading the report and giving suggestions for improving it, and Karin Nylinder for her continuous support during the work on the thesis.

Kristoffer Olsson &
Sveinn Finnsson, Gothenburg, September 2017

Contents

List of Figures

List of Tables

1
Introduction

1.1 SSRS & Alten

Svenska Sjöräddningssälskapet (SSRS) have entered into a collaboration with Alten in developing weather stations to be placed in the archipelagos surrounding Sweden. The weather stations will be used to provide more localized weather information about popular destinations in the archipelagos. The information from these weather stations will then be made publicly available along with additional information, making it easier for both inexperienced and experienced seafarers to understand the current sea conditions. According to both SSRS [1] and Sjöfartsverket (Swedish Maritime Administration) [2], the number of rescue operations at sea have seen a large increase the last couple of years. The additional information provided by the weather stations can hopefully minimize the number of seafarers setting sails and heading out to sea during questionable conditions and which then might have to call SSRS for assistance or rescuing. If severe accidents at sea can be successfully avoided due to intelligent application of the data, then in the long run this could lead to lives being saved without the need to perform additional rescue operations. Alten's part of the project is to design complete weather stations for SSRS that will be ready for placement at desired locations in the archipelagos.

1.2 Problem description

Svenska Sjöräddningssällskapet (SSRS) has a mobile-phone application that provides up-to-date weather information along with additional information to seafarers around Sweden. One of the improvements SSRS would like to see is increased granularity of their weather information by adding additional weather stations at popular locations in the archipelagos surrounding Sweden. As the weather stations will be located in remote locations they should preferably be very low maintenance and self-sufficient for a long time (>1 year). The weather stations also need to report their information back to a central server or application for further processing and displaying. To fulfill these requirements a low-cost, long-range and low power protocol with an Internet connected backbone is necessary. An additional problem to take into consideration is that some weather stations might be out of range from a central gateway, so either a mesh-network protocol or a star-network protocol with some range extending feature is necessary.

1.3 Thesis description

In this thesis the new wireless protocol LoRa and the network protocol LoRaWAN is evaluated with respect to its usability as a wireless transmission protocol for SSRS weather stations placed in the Gothenburg archipelago. The aim of this thesis is to provide a range extending protocol for LoRaWAN suitable for weather stations/node network in the Gothenburg archipelago. Firstly, the basics of the protocol along with its suitability as a communication protocol for IoT weather stations is evaluated by review of the protocol specification along with tests of hardware and protocol where real world performance is evaluated. The test evaluates the packet-error-rates (PER) for different spreading factors (data rates) of the protocol at various distances. Evaluation and analysis of the test results will server as the basis for the design of the range extending protocol. The proposed range extending protocol will allow devices located outside of a central gateway's range to do a hop to an intermediate node that forwards the message to the central gateway. The main motivation for a range extending protocol based on the LoRa and LoRaWAN protocols is that it could potentially reduce the number of gateways necessary for a network, thus minimizing network costs.

In this thesis the range extending specification will be presented ready for software implementation, but neither a software or hardware implementation will be done. Furthermore, we will provide simple guidelines for placement of intermediate-nodes and end-devices in a range extended network. These guidelines will be based on the results of the real world test and protocol evaluation. After presenting the network extending specification, the advantages and drawbacks of the specification are discussed and compared to the option of adding additional gateway capacity.

In Chapter 2 the LoRa and LoRaWAN protocols are introduced. In Chapter 3 the theory behind the LoRa modulation and its benefits to our use case is explored. In Chapter 4 the chip to gateway test of LoRa is presented. Chapter 5 contains discussion and reasoning behind an possible network extension protocol for IoT weather stations, followed by a suggested extension specification. In Chapter 6 a final discussion is had about LoRa, LoRaWAN and the proposed network extension specification before concluding the report in Chapter 7. Chapter 8 lists possible future work.

2
LoRa and LoRaWAN

With a rising interest in Internet of Things (IoT) devices, requirements for a new communication standard to suit their needs has arisen. The main requirements for these protocols are simplicity and low power, as the devices that implement these protocols should be cheap and be able to operate for a long time on battery-power. Several new communication protocols and corresponding hardware have been developed to meet these criteria. One of these protocols is LoRa, developed by Semtech and the LoRa-Alliance [3]. It can be said that LoRa consists of two parts, LoRaWAN and LoRa modulation. The former is a network architecture and the latter is a protocol for the physical layer in the OSI model [4].

2.1 Other IoT protocols

LoRa and LoRaWAN are not the only IoT protocols out there, and other protocols worth exploring are SigFox and DASH7. The SigFox protocol is an ultra narrow-band protocol, with little overhead and low data rates. Like LoRa, SigFox is also able to transmit over long distances. However, the SigFox protocol limits transmission to 140 messages with a 12 byte payload per day per unit. This effectively removes the capability of creating any useful intermediate-nodes. Furthermore, SigFox requires that all end-devices connect to their infrastructure, this limits connection points for end-devices and limits choice of infrastructure. DASH7 is another protocol which might be useful for our network, as it is a low energy protocol. DASH7 allows for packet sizes of up to 256 bytes and can transmit at data rates up to 166.67 kbit/s depending on channel width. However, the main drawback is that it is a medium range protocol with a significantly smaller link budget than SigFox and LoRa, which are both long range protocols. As one of the main components that is being investigated in this project is range, we feel that LoRa offers the best trade-off between data rate and range. We therefore choose to focus on LoRa and LoRaWAN and explore its usability for our use case.

2.2 LoRa

LoRa is the physical layer protocol often used in conjunction with the LoRaWAN MAC-layer protocol. Unlike the LoRaWAN protocol, which is open source, the LoRa protocol is a proprietary protocol developed by Semtech. Due to LoRa being a proprietary protocol, information about the design and implementation is not readily available from Semtech. However, some information about the protocol has

Code rate	Error Correction [bits]	Error detection [bits]
4/5	0	0
4/6	0	1
4/7	1	2
4/8	1	3

Table 2.1: Error correction and detection capabilities of LoRa

been released by Semtech and subsequently the protocol has been reverse engineered to a point where the implementation of the protocol is considered well understood.

2.2.1 Basics of LoRa

2.2.2 LoRa - Chirp Spread Spectrum

LoRa utilizes a spread spectrum technique called Chirp Spread Spectrum (CSS) that was initially developed for radar applications in the 1940's [5]. In LoRa the spreading of the spectrum is achieved by generating a chirp signal that continuously varies in frequency [5]. These chirps are often referred to as up-chirps, if they are continuously increasing in frequency, or down-chirps if they are continuously decreasing in frequency [6]. A theoretical description of the CSS technique is presented in chapter 3.

2.2.2.1 Coding scheme

LoRa makes use of Hamming codes for forward error correction (FEC). This is a simple linear block code algorithm that is easy to implement. LoRa offers code rates of 4/5, 4/6, 4/7 and 4/8. If we assume that the code blocks are well defined such that the minimum hamming distance is 1, 2, 3 and 4 for code rates 4/5, 4/6, 4/7 and 4/8 respectively, the error correction and error detection capabilities are as shown in table 2.1 [7]. As can be seen from table 2.1, error correcting is only introduced by the 4/7 code rate. Furthermore, code rate 4/8 does not add to the error correction capabilities, only detection capabilities. code rate 4/5 offers no clear advantage over no coding and code rate 4/6 only adds error detection, but no correction capabilities. Therefore, in order to have actual error correcting capabilities, at least code rate 4/7 must be used. However, introducing coding and utilizing code rate 4/7 increases the payload length by 75% compared to no coding.

2.2.2.2 Achievable data rates

The LoRa specification has defined its chirp rates as SPREADING FACTORS (SF), ranging from 6-12, although use of spreading factor 6 is currently not enabled by Semtech. The spreading factors, in conjunction with coding-rates dictate the achievable data rates for the LoRa protocol. The nominal bit rate can be calculated as [5]:

$$R_b = SF \frac{\left\lfloor \frac{4}{4+CR} \right\rfloor}{\left\lfloor \frac{2^{SF}}{BW} \right\rfloor}$$

Where SF is the chosen spreading factor between 7 and 12, CR is the code rate and BW is the bandwidth.

2.2.3 Key properties of LoRa

Some of the key properties and selling points of LoRa according to Semtech [5] are:

Bandwidth Scalable
LoRa modulation can easily be adapted for either narrowband frequency hopping and wideband direct sequence applications as it is both bandwidth and frequency scalable.

Constant Envelope / Low-power
LoRa modulation is a constant envelope modulation scheme. Therefore low-cost, low-power and high-efficiency power amplifier stages can be used. This reduces hardware costs.

High Robustness
LoRa is highly resistant to both in-band and out-of-band interference due to its high bandwidth-time product (>1) and asynchronous nature.

Multipath and Fading Resistant
Due to the relatively broadband nature of the chirp pulse, the LoRa modulation is robust against multipath and fading. These properties are well suited for urban and sub-urban environments where multipath and fading are dominant.

Long Range Capability
Compared to conventional FSK, for fixed output power and throughput, LoRa's link budget is improved. This in conjunction with other properties of LoRa can translate into significant improvements in range.

2.3 LoRaWAN

The LoRa-Alliance describes LoRaWAN [3] as:

> LoRaWAN™ is a Low Power Wide Area Network (LPWAN) specification intended for wireless battery operated Things in a regional, national or global network. LoRaWAN targets key requirements of Internet of Things such as secure bi-directional communication, mobility and localization services. The LoRaWAN specification provides seamless inter-

> operability among smart Things without the need of complex local installations and gives back the freedom to the user, developer, businesses enabling the roll out of Internet of Things.

As can be seen from the above quote, the main focus of LoRaWAN is to be a simple network protocol that is easy to deploy and fulfills all the basic requirements for wireless battery operated IoT devices.

2.3.1 Network topology

LoRaWAN is a Low Power Wide Area Network specification [8]. The specifications targets wireless battery operated devices and allows for easy setup of devices wishing to connect to a network server. A LoRaWAN network consists of at least a network server, gateway and an end-device. End-devices might be some sensor or other entity producing data that it wishes to relay to a network server. A gateway receives data from one or multiple end-devices connected to it over LoRa and forwards it to the network server, acting as a transparent relay between the end-device and network server. A single end-device can also be connected to several gateways. The network server then makes the data available to an end-user/application. Communication between an end-device and a gateway is over the LoRa protocol (see chapter 2.2), whilst the communication between a gateway and a network server is over TCP/IP, meaning a gateway has to be connected to the Internet in some way. In order to increase spectral efficiency, battery life and range, a LoRaWAN gateway can negotiate data rate, RF output power and which frequency-channels to use with end-devices using an adaptive data rate scheme. Furthermore, LoRaWAN supports broadcasts from gateways and bi-directional communication, although with limitations. These limitations reflect the use cases for the end-devices, resulting in three classes of end-devices. These classes are described in section 2.3.2.

LoRaWAN networks have a star-of-stars network topology, where a central server is the root or center of the network. One or multiple gateways are then connected to the central server, creating a network with a star layout. Furthermore, each gateway then has its own star-network, where the gateway is the central node and end-devices connect to it. This results in a star-of-stars topology.

As mentioned previously, LoRaWAN uses a star-of-stars topology. This has some advantages and disadvantages compared to a mesh-network topology as used by some other wireless sensor networks, such as ZigBee. One of the main advantages of having a star topology is that it makes it unnecessary for end-devices to listen for incoming messages and forward them, which draws a significant amount of power. Furthermore, a star-topology does not require the end-devices to contain any routing logic, resulting in simpler end-devices. However, using a star-topology has several drawbacks compared to a mesh-topology, mainly star-topologies rely on a central node, which means that for example a gateway failure will take several end-devices with it offline. Furthermore, a star-topology network will have no way to recover from that failure until the gateway is back up again, meanwhile a mesh-topology

Spreading Factor	Bit rate [bits/s]
7	5469
8	3125
9	1758
10	977
11	537
12	293

Table 2.2: LoRa Data Rates

network could re-route, perhaps losing some throughput but maintaining a usable network.

2.3.2 Device classes

Class A devices have the most limited bi-directional communication capabilities intended for devices that rarely need to receive down-link transmissions. All down-link transmissions to a class A device must be performed after an up-link transmission from the class A device. This is due to the fact that a class A device only opens up two short receive windows within a set time limit from its up-link transmission. Down-link transmission is not possible outside of those two receive windows, if down-link transmission is required at any other time, the gateway simply has to wait until the next up-link transmission from the class A device before transmitting its message on the down-link.

Class B devices are similar to Class A devices and are required to implement all the functionality of the Class A devices. In addition, Class B devices also allow for more receive slots by opening up receive windows at scheduled time slots. Class B end-devices are synced with the gateway by reception of a time synchronization beacon transmitted by the gateway.

Class C devices are best suited when significant down-link transmission is expected. Devices in class C are constantly listening for incoming messages, that is, their receive window is always open except when transmitting data.

2.3.3 Data rate and duty cycles

Currently the LoRa protocol is limited to six different data rates, commonly referred to as spreading factors (SF) 7-12. The lower SF numbers offer higher data rates, but shorter distances, whilst the higher spreading rates offer lower data rates but increased transmission robustness. In general one can assume that the data rate is halved when increasing the SF by one. The indicative physical bit rate for a 125 KHz channel with different SF is given in table 2.2 [9]. The bit rates shown in table 2.2 are calculated for a code rate of 4/5.
As can be seen from table 2.2, LoRa is a low data rate protocol. However, as LoRa's spreading factors are all orthogonal to each other, it is in theory possible to transmit

Edge Freq.-	Edge Freq.+	Field/Power	Spect. Access	Bandwidth
865 MHz	868 MHz	+6.2dBm/100 KHz	1% or LBT AFA	3 MHz
865 MHz	870 MHz	−0.8dBm/100 KHz	0.1% or LBT AFA	5 MHz
868 MHz	868.6 MHz	14 dBm	1% or LBT AFA	600 KHz
868.7 MHz	869.2 MHz	14 dBm	0.1% or LBT AFA	500 KHz
869.4 MHz	869.65 MHz	27 dBm	10 % or LBT AFA	250 KHz
869.7 MHz	870 MHz	7 dBm	No Requirement	300 KHz
869.7 MHz	870 MHz	14 dBm	1% or LBT AFA	300 KHz

Table 2.3: LoRa Bands, Sub-Bands and applicable regulations, reproduced from [11]

using all six spreading factors simultaneously on the same channel.

In Europe end-devices operate in the open 868 MHz ISM band and have to comply with the ETSI regulations [10] for wideband modulation. This allows the LoRa devices to operate on frequencies between 863 MHz to 870 MHz, but with restrictions on output effective radiated power (ERP) and transmission. From sx1272's (a LoRa Modem) ETSI compliance sheet [11] we find the regulatory bands that support wideband modulation along with their applicable limitations. This information is listed in table 2.3.
As can be seen in the fourth column of table 2.3, the max duty cycle requirements for spectrum access are very stringent and can vary greatly between bands. According to the European regional parameters for LoRa, all units must implement at least the three following frequency channels of 125 KHz width with center frequencies at 868.1 MHz, 868.3 MHz and 868.5 MHz. These channels all allow for a duty cycle of $< 1\%$ or 36 sec/hour and an output of 14 dBm ERP. If any other frequency channels are to be used, caution must be used so that all regulatory requirements are met.

2.3.4 PHY and MAC layer structure

The LoRa and LoRaWAN protocols both make use of headers for data transmission. In the following sections we will explain the PHY and MAC layer formats.

2.3.4.1 PHY Message Formats

LoRa the radio protocol utilizes the PHY headers to make radio-transmission and reception possible. There exists two PHY formats, one for up-link and one for down-link messages. The difference between those formats is that the up-link format contains an optional cyclic redundancy check (CRC) field. The PHY uplink message format is structured as can be seen in figure 2.1. The preamble length

Figure 2.1: Uplink PHY structure

can vary between regions, but in Europe the LoRa protocol uses 8 symbols of the

sync word 0x34 [9]. According to application note 1200.18 [12], the PHDR should contain a length and an address field, each a byte long. Unfortunately, as LoRa is a proprietary protocol, the specification does not provide further information about the PHDR and PHDR_CRC . The PHYPayload is of variable length, from 0 bytes to a maximum of 255 bytes. Section 2.3.4.2 expands on the layout and functionality of the PHYPayload.

2.3.4.2 MAC Message Formats

LoRaWAN's MAC messages are contained within the radio PHY payload of the LoRa protocol. The structure of a PHY payload is illustrated in figure 2.2. Furthermore, the MAC payload field can alternatively be exchanged for a network join-request or a join-response, if necessary. We will not expand further on the network join-requests and responses in this thesis. The MAC header (MHDR) and message

Figure 2.2: PHY Payload

integrity check (MIC) are fixed to a length of 1 octet and 4 octet respectively. The MAC payload is however of dynamic size with a variable max-length depending on which data rate is in use. The structure of a MAC payload is illustrated in figure 2.3 and contains a frame header (FHDR), frame port (FPort) and a frame payload (FRMPayload). Furthermore, the FHDR of the MAC payload contains four fields

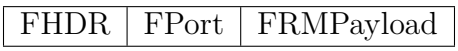


Figure 2.3: MAC Payload

which are utilized by the LoRaWAN protocol. As pictured in figure 2.4 these fields are the device address (DevAddr), frame control (FCtrl), frame counter (FCnt) and frame options (FOpts). In total the FHDR is 7-22 bytes long depending on whether any frame options are used. The minimum length of 7 bytes is due to the fixed length of the device address, frame control and frame counter of 4, 1 and 2 bytes each. The frame port is a single byte number ranging from 0 to 255, where port 0

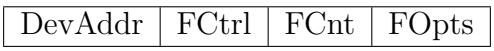


Figure 2.4: Frame header

indicates that the frame payload only contains MAC commands. Ports 1 to 223 are application specific and are free to be used by any application. Port 224 is reserved for the LoRaWAN MAC layer test protocol. The rest of the ports, from 225 to 255 are reserved for future standardized application extensions.

The length of the frame payload is variable and is dependent on the amount of data to be transmitted. Furthermore, depending on region and data rate the maximum frame payload length differs. For the European region the maximum application

payload length is 51 bytes for data rates 0-2 (SF10-12), 115 bytes for data rate 3 (SF9) and 222 bytes for data rates 4 and 5 (SF8 and SF7) [9]. This payload length assumes that the frame options field is empty.

For each transmitted message within a LoRaWAN network in Europe, we require at least 8 symbols for synchronization and then we have a MHDR of 1 byte and MIC of 4 bytes. The frame header within the MAC payload has a minimum length of 7 bytes, this gives us a minimum transmission of 8 symbols and 12 bytes for an empty message. However, some additional data has to be accounted for within the PHY header and PHY header CRC.

3
Theory

This chapter aims to provide the reader with a basic understanding of one of the foundations on which LoRa is built; CHIRP SPREAD SPECTRUM (CSS). First, the (possibly) familiar topic of SPREAD SPECTRUM and closely associated terms such as FADING, SHADOWING and MULTIPATH PROPAGATION. After that, modern varieties of spread spectrum techniques are discussed, before the theory of PULSE COMPRESSION is investigated and the idea behind CSS is revealed.

A short description of line-of-sight (LOS) and Fresnel zone clearance will also be covered, since knowledge of these topics could prove important in order to successfully deploy a LoRa network as intended in this project.

3.1 Spread Spectrum

3.1.1 Spread spectrum and fading channel behavior

Spread spectrum is a term that encompasses several (similar) techniques that are used (mainly when dealing with wireless communications) to combat the problem of fading channel behaviour. The varying attenuation of a radio frequency (RF) signal, fading, is often divided into two categories: signal multipath propagation and objects blocking the signal's path (shadowing). While both multipath and shadowing are dependent on parameters such as transmitter/receiver positioning and surrounding geometry, spread spectrum techniques are mainly used to relieve interference due to multipath reflections (although the techniques will also help solve some of the problems associated with shadowing).

When an information carrying signal traverses a channel from a transmitting source towards the receiving end, it can travel many different paths. The (if there is one) signal with a direct line-of-sight (LOS) will reach the destination first, and shortly afterwards (one or several) reflected versions of the same signal will arrive. The difference in distance will produce a change in phase between the arriving copies of the same signal. When these different phases add up to distort the combined signal, it is said that the receiver side experiences multipath fading.

If a receiver sees many reflected versions of a signal, a larger amount of time is needed in order for all the echoes (of significant amplitude) to arrive, thus widening the channel's impulse response. Another name for this lag is DELAY SPREAD (τ_d)

and it is an important characteristic used when describing the wireless channel. If a new signal is sent before the channel has settled from the previous signal, the symbols will cut into each other, causing inter-symbol interference (ISI). Thus, the delay spread of a channel will limit its symbol rate.

The channel's delay spread is linked to the COHERENCE BANDWIDTH (B_c) through $\tau_d \approx \frac{1}{B_c}$. The coherence bandwidth can be seen as the frequency spread over which the channel's fading stays constant. When the bandwidth of a signal fits within the channel's coherence bandwidth, it is said to experience FLAT FADING. On the other hand, if the signal occupies a frequency band significantly larger than the coherence bandwidth, it will encounter regions of varying attenuation. It is said to be subject to FREQUENCY SELECTIVE FADING.

By raising the bandwidth of a signal (by the use of spread spectrum techniques) to be large compared to the coherence bandwidth, the probability that the signal echoes can be effectivly resolved (by using appropriate recombination techniques, e.g. a receiver that employs multipath-assigned correlators) is raised when compared to a narrowband signal experiencing flat fading. The frequency-selective behavior is then utilized as a means of FREQUENCY DIVERSITY [6, Chapter 1.1].

3.1.2 Spread spectrum: frequency hopping and direct sequence

As mentioned, the spread spectrum effect can be realized using several different techniques. The most readily used techniques today are FREQUENCY HOPPING SPREAD SPECTRUM (FHSS) and DIRECT SEQUENCE SPREAD SPECTRUM (DSSS).

In FHSS, the data carrying signal is spread over a large band in the frequency domain, where each frequency chunk equals the bandwidth of the original signal. The order in which the signal jumps, the SPREADING CODE, is decided by a PSEUDO-RANDOM NUMBER (PN) sequence. For an outsider, without knowledge of the PN sequence, the spread signal would look like noise, and this low probability of intercept was one of the main reasons for inventing FHSS. A well-known technique that uses an implementation of FHSS is the communications protocol Bluetooth.

Direct sequence spread spectrum differs from FHSS in such that it directly modulates the information carrying bits with PN sequence. The high rate of the PN sequence corresponds to the total bandwidth of the DSSS system, which usually is much larger than the bandwidth of the information carrying signal. The PN sequences used in DSSS are commonly designed to have low autocorrelation except at zero delay, making it possible to find the start of a signal seemingly drowned out in noise. An example of a system using DSSS in such a way (for processing gain) is the global positioning system (GPS).

3.2 Chirp Spread Spectrum

While FH and DS are the most commonly used spread spectrum techniques today, there are other techniques. One such is LINEAR FREQUENCY MODULATION, or chirp spread spectrum. As opposed to both FHSS and DSSS, CSS does not use any PN sequence for the frequency spreading. Instead it sweeps the whole (allotted, not infinite) frequency band in linear-ramp behaviour. This linear frequency sweep has a clear advantage over both FHSS and DSSS in that it can be realizable without (expensive) digital signal processors (DSP), which was a deciding factor back at the time of its invention.

The theory of linear frequency scaling is nothing new. While the technical terms and applications were not explicitly mentioned until 1962, the fundamentals have been actively researched since the era of the second World War, and the invention of the radar (radio detection and ranging) [6, Chapter 1.5]. The main idea that underpins it all is called PULSE COMPRESSION.

As mentioned previously, the essential idea behind CSS can be derived from the early days of radar enhancing techniques. One of the fundamental problems that all radar systems encounter is the inevitable trade-off between range (transmitted power) and resolution (signal duration). Consider the outgoing sinusoidal pulse $s(t)$, with unity amplitude, carrier frequency f_0 and duration T_c:

$$s(t) = \begin{cases} e^{j2\pi f_0 t}, & 0 \leq t < T_c \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

The received signal $r(t)$ is the reflected and attenuated (A) versions of $s(t)$, arriving at the site of the transmitter delayed according to t_r:

$$r(t) = \begin{cases} Ae^{j2\pi f_0(t-t_r)} + n(t), & t_r \leq t < t_r + T_c \\ n(t) & \text{otherwise} \end{cases} \tag{3.2}$$

where n(t) is zero-mean additive white Gaussian noise (AWGN), with variance σ^2. The most efficient way of mitigating the influence of noise in an AWGN channel is to convolve the received waveform with the matched filter output of the original signal. If we define the matched filter $h(t)$ of the signal $s(t)$ in equation (3.1) as:

$$h(t) = s^*(-t) \tag{3.3}$$

the aforementioned convolution becomes:

$$(h \star r)(\tau) = \int_{-\infty}^{+\infty} s^*(t) r(t+\tau) dt \tag{3.4}$$

Inserting $h(t)$ and $r(t)$, as given in equations (3.3) and (3.2) respectively, into equation (3.4) will result in the matched filter output given by:

$$(h \star r)(\tau) = A \cdot \text{tri}\left(\frac{t - t_r}{T_c}\right) e^{j2\pi f_0(t-t_r)} + N(t) \tag{3.5}$$

where $N(t)$ is the correlated noise (to the sent signal) and $\text{tri}\left(\frac{t-t_r}{T}\right)$ is the time-shifted and scaled triangle function, the convolution of two rectangular pulses. An illustration depicting the sent signal s(t), and the received signal r(t), consisting of several noisy reflections, can be seen in figure 3.1a, while the result from the correlations can be seen in figure 3.1b.

As can clearly be seen in figure 3.1, if the echoing signals are separated in time with at least one pulse width (T_c), the individual reflections can be recreated. However, if the distance becomes less than T_c (figure 3.2a), the reflections will no longer be distinguishable (as illustrated in figure 3.2b). This dependency on the pulse width to successfully resolve echoes is called the RANGE RESOLUTION of a radar system. Given the propagation velocity of an electromagnetic (EM) wave is c, along with the fact that the total distance covered during a pulse period T_c is twice that of the range of the reflecting target, the range resolution can be specified as

$$\frac{cT_c}{2} \tag{3.6}$$

It is plain to see that in order to get higher range resolution, the pulse duration T_c must be minimized.
Reducing the pulse duration has a major drawback, the energy of the received pulse, E_r, will also be lowered (unless the power is increased to compensate accordingly). Remembering equation (3.2), the energy of the signal component in $r(t)$ is given by:

$$E_r = \int_{T_c} |r(t)|^2 dt = A^2 T_c \tag{3.7}$$

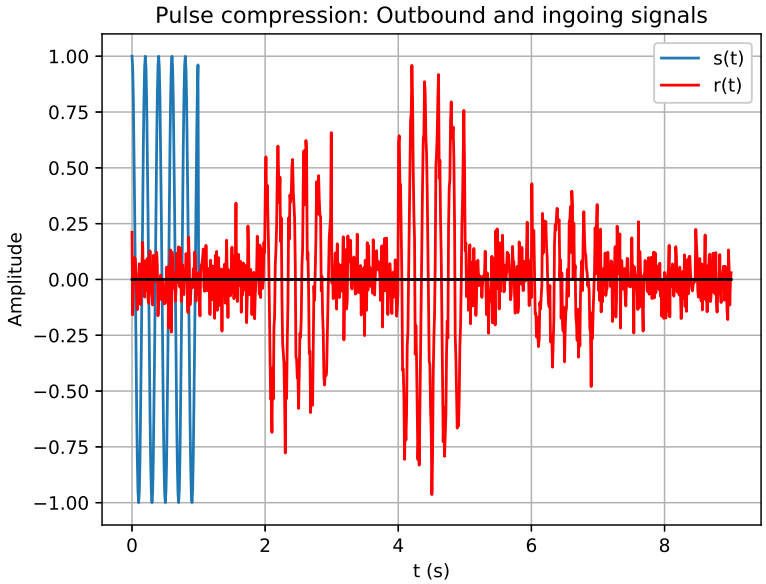
With the noise variance of the AWGN channel defined as σ^2, the signal-to-noise-ratio (SNR) for the echo at the receiver becomes

$$\text{SNR} = \frac{E_r}{\sigma^2} = \frac{A^2 T_c}{\sigma^2} \tag{3.8}$$

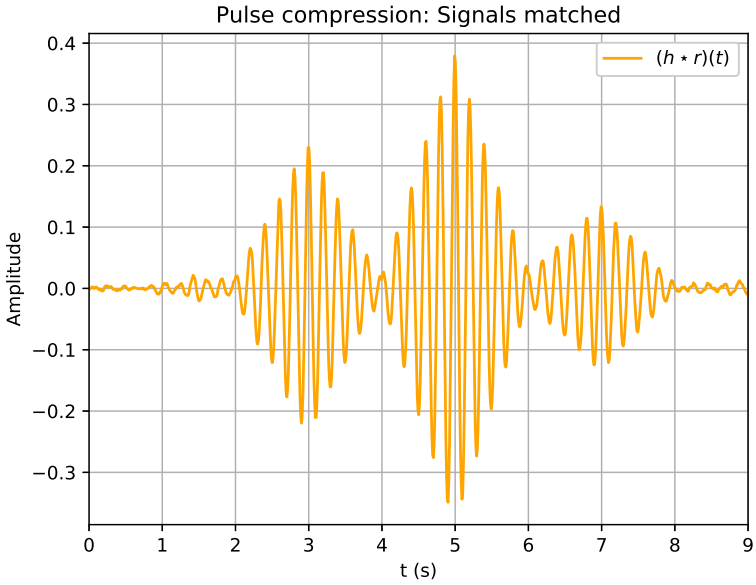
Comparing equations (3.8) and (3.6), it is clear that a compromise is necessary. Lowering the pulse duration will improve the resolution, thus increasing the ranging capability. At the same time, the lowered duration will deteriorate the SNR, eventually drowning the sought signal in the channel's noise. One way to compensate for the decreased pulse duration is to raise the power of the outbound pulse. In the limiting form that would constitute a Dirac delta function. However, even long before approaching that point, such a solution would become unrealistic in terms of necessary power.

How can the aforementioned trade-off (between duration versus resolution) be solved without putting excessive amount of energy into a transmitted pulse? One solution would be to look at the relationship between a signal's representations in both time- and frequency domains. Remember Parceval's relation [13, Chapter 4.3.7]

$$\int_{-\infty}^{+\infty} |x(t)|^2 dt = \int_{-\infty}^{+\infty} |X(2\pi f)|^2 df \tag{3.9}$$



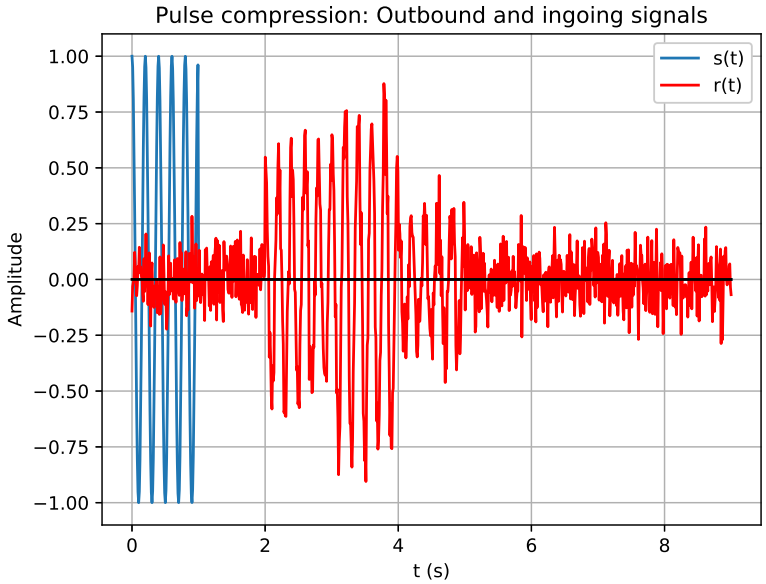
(a) An outgoing sinusoidal pulse $s(t)$ and returning echos $r(t)$ separated by at least pulse duration T_c



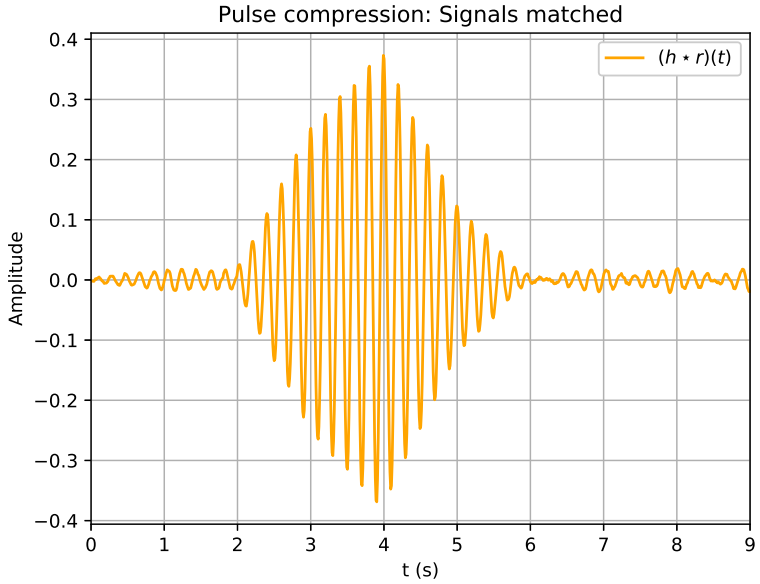
(b) Each om the returning echoes can clearly be distinguished after the matched filter

Figure 3.1: Illustration showing that when echoes from a sinusoidal pulse are properly spaced in time, then each individual peak is clearly discernible after matched filtering

which states that the total energy of a signal $x(t)$, assuming Fourier transform $X(2\pi f)$, can be found by either integrating in the time plane, or by the corresponding computation in the frequency plane. This theorem can be combined with



(a) Outgoing sinusoidal pulse $s(t)$ and returning echos $r(t)$, this time separated by less than pulse duration T_c and interfering with each other



(b) After the matched filter, the individual echoes are no longer discernible

Figure 3.2: Illustration showing that when echoes from a sinusoidal pulse are interfering (i.e. not sufficiently apart in time), then the individual peaks can not be distinguished after the matched filter output

another familiar fact, the scaling property of Fourier transforms [13, Chapter 4.3.5]

$$x(at) \overset{\mathrm{F}}{\leftrightarrow} \frac{1}{|a|} X\left(\frac{j\omega}{a}\right) \tag{3.10}$$

In equation (3.10), $\overset{\mathrm{F}}{\leftrightarrow}$ denotes the Fourier transform, again assuming the transformation can be applied to $x(t)$, and the inverse transformation on $X(j\omega)$. For $X(j\omega)$, ω is the angular frequency ($\omega = 2\pi f$ [radians/s]), and with an additional amplitude correction of 2π it is fully interchangeable with f in the equation. Equation (3.10) that a scaling in the frequency domain will result in an inversely proportional scaling in time. Thus, combining equations (3.9) and (3.10), people concerned with the range vs. duration problem of radar pulses had found a possible solution. By expanding a pulse in frequency, a proportional compression in time could (theoretically) be achieved without any loss in signal energy.

One simple way of producing the frequency scaling of a pulse is to let it sweep through a band of frequencies, B_w, for its duration. The method of linear frequency modulation is commonly known as CHIRPING (as in Chirp Spread Spectrum), possibly due to similarities shared with the sound produced by birds and certain insects. A regular way to define a chirped pulse, denoted c_h, is

$$c_h(t) = \begin{cases} \cos\left(2\pi\left(f_0 t \pm \mu\frac{t^2}{2}\right)\right), & -\frac{T_c}{2} \leq t \leq \frac{T_c}{2} \\ 0 & \text{otherwise} \end{cases} \tag{3.11}$$

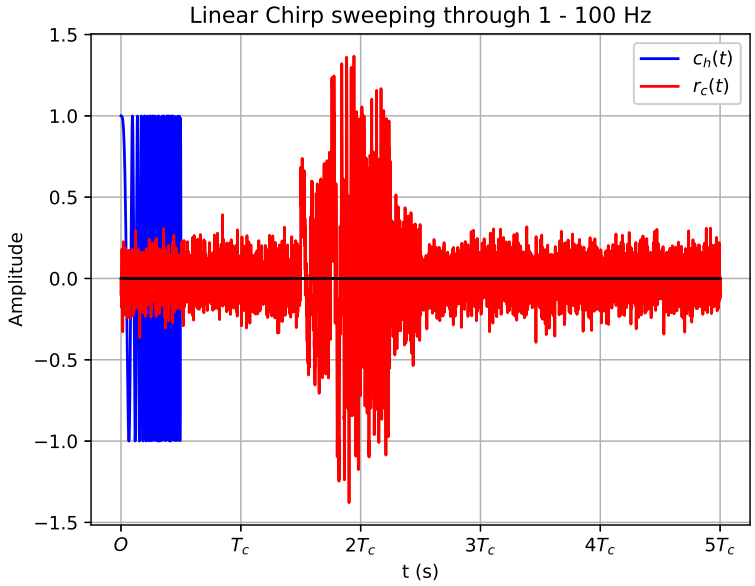
where f_0 is the carrier frequency, μ is the rate of the sweep (in Hz/s), and T_c is the pulse duration. The sweep rate μ is usually defined as $\mu = \frac{B_w}{T_c}$, where B_w is the frequency band that is swept and T_c is the pulse duration. As for the non-compressed pulse in eq. (3.1), the returning echo of the scaled pulse c_h can be considered a delayed and attenuated version of the one given in equation (3.11). An illustration of the pulse(s) is given in figure 3.3a, where, for the sake of visibility, the carrier frequency has been set to 0.

In a fashion closely resembling that for the non-compressed pulse, a matched filter $h(t)$ is applied to the echo to best deal with the added noise of the AWGN channel:

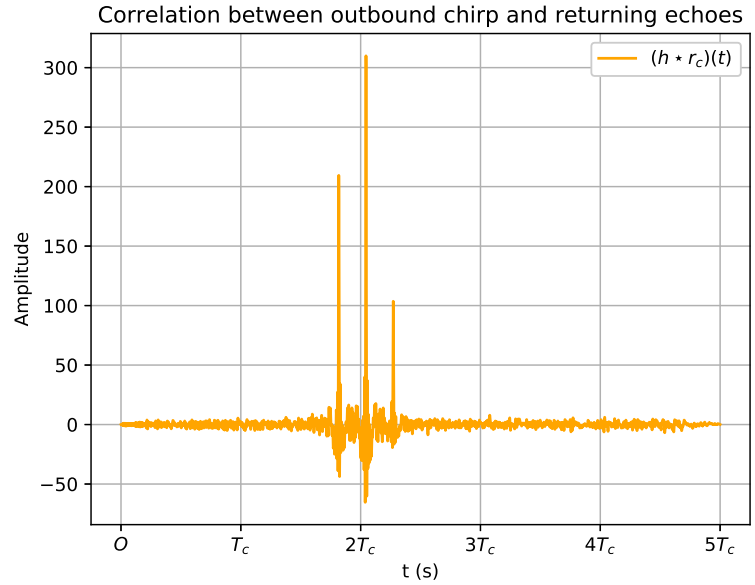
$$h(t) = \sqrt{4\mu}\cos\left(2\pi\left(f_0 t \mp \mu\frac{t^2}{2}\right)\right), \quad -\frac{T_c}{2} \leq t \leq \frac{T_c}{2} \tag{3.12}$$

If the sweep rate μ in equation (3.11) has a positive sign, the chirp signal sweeps up through the frequency band B_w and $c_h(t)$ is called an UP-CHIRP. From the inverted $\mp$ sign in equation (3.12) it then follows that the matched filter $h(t)$ will have a negative sweep rate, producing a DOWN-CHIRP. Thus, the matched filter of an up-chirped signal is a down-chirped (and scaled) version of said signal.

When matching the signals described in equations (3.11) and (3.12), it can be shown [6, Chapter 1.4-2.3] [14, Chapter 2.1.2.3] that the filter output ($g(t) = (h \star r_c)(t)$), where $r_c(t)$ is the returning, delayed version of $c_h(t)$ defined in equation (3.11), takes the form of



(a) Outbound upchirp $c_h(t)$ and three interfering echoes $r_c(t)$ returning



(b) Each echo is solvable even though clearly interfering with one another

Figure 3.3: Chirped signal and echo (a) and correlation of the two (b)

$$g(t) = \sqrt{4\mu}\cos\left(2\pi f_0 t\right)\frac{\sin\left(\pi\mu t\left(T_c - |t|\right)\right)}{2\pi\mu t}, \quad -T_c \leq t \leq T_c \tag{3.13}$$

The resulting output g(t) behaves very much like a scaled cardinal sine (sinc) function, with peak amplitude ($\sqrt{T_c B_w}$) and the majority of its energy found in

$-\frac{1}{B_w} \leq t \leq \frac{1}{B_w}$, where again T_c is the pulse duration and B_w is the swept frequency band. An illustration of the correlated result described above can be seen in figure 3.3b. It is this concentration of the pulse's energy in the time domain (going from a duration of T_c to approximately $\frac{2}{B_w}$) that has given rise to the name PULSE COMPRESSION.

While the benefits of pulse compression is clear for radar applications, it can also be of merit when used in communications systems. As discussed in section 3.1, for frequency selective channels, the ability of a receiver to recombine several multipath components could prove decisive when recovering the transmitted signal. The term $T_c B_w$, commonly known as the TIME-BANDWIDTH PRODUCT, that dictates the power amplification, and thereby improving the resolution in a radar system, could in similar fashion be used to improve the multipath resolution of the (multipath) channel [6, Chapter 2] in a communications system.

Furthermore, by increasing the pulse duration T_c, while keeping signal peak-power and bandwidth B_w unchanged, allows for increased signal energy without compromising multipath solveability. With the chirp-rate μ defined as $\mu = \frac{B_w}{T_c}$, this time expansion corresponds to raising the spreading factor introduced in section 2.2.2.2. This additional power could be interpreted as a PROCESSING GAIN, which permits the system to use low peak-power, which in turn admits the power amplifier of a transmitting circuit to operate exclusively in its highly efficient linear region. For power-limited (mainly battery-driven) devices, operating on low data rates (thus being able to afford the necessary bandwidth) in fading channels, this makes techniques employing pulse compression (i.e. CSS) interesting alternatives.

3.3 Line-of-sight and Fresnel zone clearance

The multipath behavior of the fading channel was briefly discussed in section 3.1.1. In section 3.3.1 it will be shown that even though line-of-sight can quite easily be achieved for a communications link, it will not make the problem of destructive interference vanish. In section 3.3.2, a way of determining the effect of multipath components stemming from different regions along the path of propagation is introduced, along with a discussion on how this knowledge can be used minimize multipath contribution to destructive interference.

3.3.1 Line-of-sight

When deciding where to locate the antennas in a communications link, visibility is of utmost importance. In a system where many transmitting nodes need to reach a specific receiver, the positioning of said receiver should be dealt with carefully. For short distance communication links, free line-of-sight between transmitter and receiver (antennas) poses no problem. However, when the distance starts to grow past a few kilometers, one must take Earth's curvature into account.

From basic trigonometry it can be shown [15] that the distance d to the horizon is given by

$$d = \sqrt{2Rh + h^2} \quad \text{(m)} \tag{3.14}$$

where R is Earths radius (6.371×10^6 m) and h is the height above R.

Suppose a transmitter is to be located 15 km away from the receiver. Now, assume that the height of the transmitter antenna for some reason is limited to two meters. From equation (3.14) it is seen that the distance to the horizon from the transmitter antenna is 5.04 km. In order for the radio link to achieve line-of-sight, the receiver antenna must be able to see at least 9.96 km in the direction of the transmitter. Setting the distance d to 10 km, and solving equation (3.14) for the antenna height, gives $h = 7.85$ m. Thus, it can be seen that for even relatively short distances (in the kilometer range), the feasibility of line-of-sight must be take into account.

3.3.2 Fresnel zones

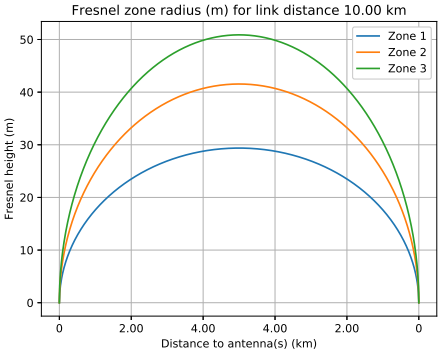
At first glance, it would seem that if free line-of-sight for a radio link is fulfilled, then optimal signal strength at the receiver would be achieved. However, from Huygens-Fresnel's theory it can be shown that the behavior of electromagnetic (EM) wave propagation is more complicated.

From an omnidirectional antenna, the transmitted RF power propagates in all directions (at least in theory), creating a spherical wavefront that moves away from the transmitting antenna. On the wavefront, the signal is all in-phase (given constant distance and speed of propagation/phase velocity). Suppose a receiving antenna is stationed a distance D_{direct} apart from the transmitter antenna, with a clear line-of-sight. Huygens-Fresnel states that the EM field at the location of the receiving antenna is the summation of infinitesimally small fields re-radiating from the wavefront [16, Chapter 1.4].

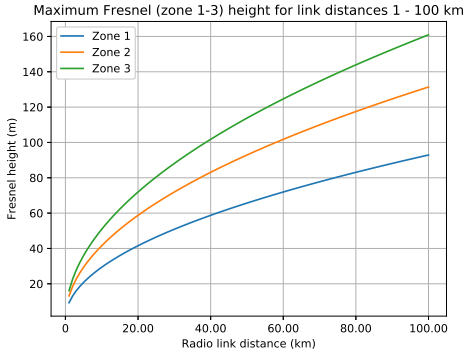
Now, assume a wavefront somewhere along the antennas' line-of-sight (distance d_1 from the transmitter and d_2 from the receiver, where $d_1 + d_2 = D_{direct}$). At any point P on the surface of the wavefront (except in the direct line-of-sight), the distance from transmitter (r_1) and the receiver (r_2) will add to a difference (be further away) from the direct path. As long as the field components add in coherent fashion (i.e., either constructive OR destructive, but not both) at the receiver, a closed surface on the wavefront is considered a Fresnel zone, F_n. Over the distance of the radio link, these cross sectional surfaces create a prolate ellipsoid shape, where the radius, or height R_n of the cross section is given by

$$R_n \approx \sqrt{n\lambda \frac{d_1 d_2}{d_1 + d_2}} \quad \text{(m)} \tag{3.15}$$

where n, is the zone number (1, 2, 3, ...), λ is the carrier wavelength and distances d_1 and d_2 given in meters.



(a) First three Fresnel zone heights for link (10 km) operating at 868.5 MHz



(b) Maximum Fresnel zone height for radio link distances between 1 - 100 km

Figure 3.4: Fresnel zone height for different positions along 10 km communications link (a) and maximum Fresnel height for different link distances (b)

An illustration of the Fresnel zone height for an RF link of distance 10 km, using a carrier operating at 868.5 MHz can be seen in figure 3.4a. In the first Fresnel zone, the different multipath components can be considered to add constructively at the receiver (without further consideration of the effects of RF wave polarization one might add). In the second zone however, the opposite is true, only to change sign again in the third zone etc..

From figure 3.4a it can readily be seen that the maximum radius of the Fresnel zones is reached at half the distance, and for the example given, this height is close to 30 meters. An example of how the zone height grows with the distance of the RF link is given in figure 3.4b, for the same carrier frequency.

Since multipath components from zone one add to the received signal strength, it is important to keep the Fresnel height in mind when designing radio links operating over longer distances. If the cross section of the first Fresnel zone is heavily impaired somewhere along the path of propagation, it could prove devastating for the receiver's ability to recombine the multipath components. As a rule of thumb, at no point should the clearance be less than 60% of the Fresnel height plus three meter [16, Chapter 1.4].

4

Chip To Gateway Test

4.1 Purpose of Test

In order to successfully design a system that relies on the RN2483 chips, we need to understand the limitations of both the technology and its implementation on the RN2483 chips. Currently, only a handful of studies have been done on LoRa and LoRaWAN and reliable information about its performance is therefore hard to find. Furthermore, as LoRa is a proprietary protocol developed by Semtech, most of the information that exists in their literature and white-papers has a tendency to highlight the protocols advantages, but seldom mention its drawbacks. To us, who were designing a system based on this technology, we felt that we needed to have a good understanding of the protocol and its limitations before continuing with our design.

In order to gain a better understanding of the protocol, tests were performed to better map the usable transmission range of different spreading factors of the LoRa protocol in a setting that closely resembled the final installation environment. The metric used to determine the usability of each spreading factor at a certain distance was the PER. The PER metric was chosen as it is of big concern when designing multi-hop systems where a packet might have to traverse several links on its way to its final destination. A high PER might not be problematic in a point-to-point connection, however, having a packet that traverses multiple high PER links, the PER will magnify and soon make the system unusable. These tests also allows us to explore the trade-off between data-rate and PER.

4.2 Related work

There have been previous, related studies looking at the robustness of the LoRa protocol. The focus has been on both theoretical performance (Orestis and Usman [17]), as well as testing retail hardware in the field (Petäjäjärvi et al. [18]).

4.2.1 Theoretical performance

Using the technique from the study by Orestis and Usman, approximations for the expected performance of LoRa can be computed. By defining the chirped signal $s(t)$ as

$$s\left(t\right)=\sqrt{\frac{2E_s}{T_s}}\cos\left[2\pi f_c t\pm\pi\left(u\left(\frac{t}{T_s}\right)-w\left(\frac{t}{T_s}\right)^2\right)\right] \tag{4.1}$$

it can be seen from equation (4.1) that $s\left(t\right)$ is essentially the same as given in equation (3.11), save for the normalized energy and different notation for the sweep rate (u and w versus μ).

Suppose $s(t)$ is transmitted over a flat fading channel, $h(t)$, described as a complex (i.e. two-dimensional) zero-mean independent Gaussian random variable. Given symmetry (equal variances) it can be shown that the channel is Rayleigh distributed [19, Chap. 3.2.2]. These properties of $h(t)$ can be used to calculate the outage probability due to path loss (distance), shadowing (obstacles) and fading (reflections). The path loss $g(d)$ is a deterministic function depending on the distance d (m), and is defined as

$$g\left(d\right)=\left(\frac{\lambda}{4\pi d}\right)^{\eta}=\eta\log_{10}\left(\frac{\lambda}{4\pi d}\right)\quad\text{dB} \tag{4.2}$$

(following from Friis' transmission equation). Here λ is the carrier frequency wave length (from f_c in equation (4.1)) and η is the path loss exponent ($\eta\geq 2$). It is assumed that both transmitting and receiving antennas are isotropic (i.e. have gains of 1), hence they are omitted in equation (4.2).

Shadowing adds zero-mean AWGN to the path loss, and the noise variance σ^2 is given by

$$\sigma^2=-174+10\cdot\log_{10}\left(BW\right)+NF\quad\text{dBm} \tag{4.3}$$

where BW is the bandwidth of $s(t)$ and -174 (dBm) is the thermal noise in one Hertz of bandwidth. The noise figure of the receiver, NF, can be considered to have a value of 6 dB in the intended hardware implementations [5].

Finally, to calculate the probability of outage in the Rayleigh channel due to fading, the impact on the SNR from path loss and shadowing should be included. If the complement to the outage probability, coverage, is defined as the probability of the receiver SNR being equal to or larger than some threshold value q_{SF} this gives

$$P\left[SNR\geq q_{SF}\right] \tag{4.4}$$

Letting $\mathcal{P}$ be the transmitted power (W), and

$$|h|^2\sim\exp\left(1\right)$$

then equation (4.4) can be re-written as (using equations (4.2) and (4.3) and rearranging the terms)

$$P\left[|h|^2\geq\frac{\sigma^2\cdot q_{SF}}{\mathcal{P}\cdot g(d)}\right]=\exp\left(\frac{\sigma^2\cdot q_{SF}}{\mathcal{P}\cdot g(d)}\right) \tag{4.5}$$

Equation (4.5) calculates the probability that the SNR of $s(t)$, as defined in equation (4.1), at a distance d from the source of radiation, is larger than or equal to the receiver SNR threshold q_{SF} (see figure 4.1). The SNR threshold q_{SF} is depending on the receiver's sensitivity S according to [20]

$$S = k_B \left(T_a + T_{rx}\right) BW \cdot q_{SF} \quad [\mathrm{W}] \tag{4.6}$$

where $T_a = T_0 = 290$ [K] is the receiver antenna noise temperature (290 [K] is considered standard room temperature), and $T_{rx} = T_0 \left(NF - 1\right)$ is the receiver's equivalent noise temperature. The konstant k_B in equation (4.6) is the Boltzmann constant ($k_B = 1.38{\cdot}10^{-23}$ [$^{\mathrm{J}}/_{\mathrm{K}}$]). With BW equal to the signal bandwidth, equation (4.6) could also be written as $S = \sigma^2 \cdot q_{SF}$, where σ^2 is given in equation (4.3). With sensitivity values given in [5] and recited in table 4.1 for different spreading factors, the corresponding SNR thresholds q_{SF} have been calculated and given as well.

Table 4.1: Receiver sensitivity for different spreading factors

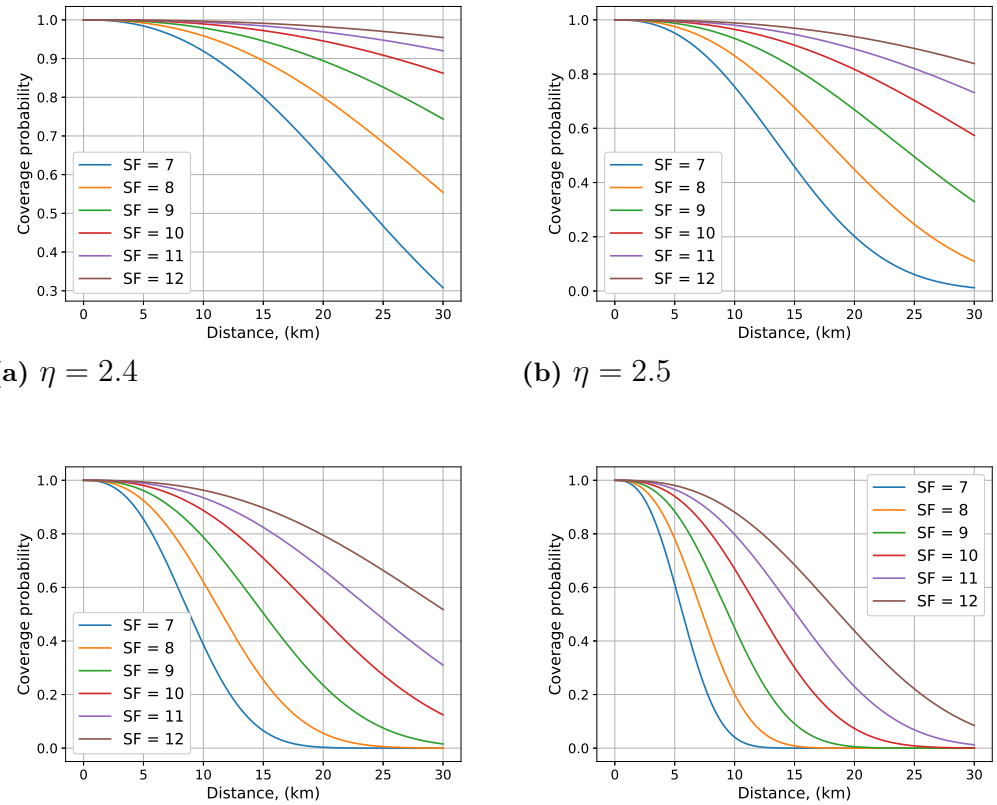
Spreading Factor	Sensitivity (dBm)	q_{SF} (dBm)
7	-123	-6
8	-126	-9
9	-129	-12
10	-132	-15
11	-134.5	-17.5
12	-137	-20

Observant readers may notice that there is approximately 3 dBm difference in sensitivity between each spreading factor and its closest neighbor in table 4.1. For each increment in spreading factor, LoRa practically halves the sweep rate of the chirp signal, meaning the signal duration redoubles. Recall from section 3.2 that the processing gain seen in a compressed sinusoidal pulse was approximately proportional to its altered time-bandwidth product ($T_c B_w$). With the bandwidth B_w fixed, a doubling of the signal's duration T_c effectively results in redoubling of the processing gain. This can be seen in the varying sensitivity levels of the different spreading factors.

With the values from table 4.1 inserted into the probability given in equation (4.5), the range versus coverage probabilities for Lora communication over different spreading factors can now be calculated (using constant transmit power $\mathcal{P} = 14$ dBm).

Remembering how Friis' transmission equation (4.2) depends on the path loss exponent η, and that guidelines generally put its value in the range of $2.4 - 2.7$ (although for suburban areas $\eta = 2.7$ is preferrable [17]), it is worth pointing out that the resulting coverage probabilities varies largely. In figure 4.1, the coverage has been calculated for $\eta = [2.4, 2.5, 2.6, 2.7]$.

As is clearly illustrated in figures 4.1a through 4.1d, finding an appropriate value



(a) $\eta = 2.4$ **(b)** $\eta = 2.5$

(c) $\eta = 2.6$ **(d)** $\eta = 2.7$

Figure 4.1: Coverage probabilities for path loss exponents 2.4 through 2.7 are given in (a) - (d) for different spreading factors on a carrier frequency of 868.5 MHz. Radio link distances varies from $0 - 30$ km

for the path loss exponent is critical if the illustrated coverage probabilities are to be useful.

4.2.2 Measured performance

As mentioned, the paper by Petäjäjärvi et al [18] focused on the performance of available hardware, as opposed to the more theoretical approach reviewed above. The hardware used for measuring closely resembles the one tested in this report. In said paper, the receiver/gateway employed was the LoRa IoT station from Kerlink [21], [22], which is identical to the one utilized for this report. The end-node transmitting was a LoRaMote [23], a device that relies on Semtechs SX1272 chip to handle the LoRa modulation. While mainly used to illustrate the capabilities of LoRaWAN, the LoRaMote is quite customizable and lets the user tweak a few parameters (e.g. spreading factor and duty cycle) before sending the data. Since the authors of this report have had the opportunity to test such a device and compare it to the RN2483 module, it can be verified that the two devices perform in similar fashion.

In addition to the equipment similarities, the environment in which the measurements were carried out closely resembles that of Gothenburg and its archipelago. Thus, the conditions in [18] should quite accurately mirror those in this report. Hence, even though the measurements were carried out in different style (non-stationary transmitter with only occasional line-of-sight in the paper), the results from the paper should be usable as reference for the tests executed in this report. The results are presented in tables 4.2 and 4.3, for measurements over land and water, respectively.

Table 4.2: LoRaMote (SX1272) measurements from moving car, SF12 used. Results reproduced from [18]

Range	Transmitted packets	Received packets	Packet loss ratio
0 - 2 km	894	788	12 %
2 - 5 km	1215	1030	15 %
5 - 10 km	3898	2625	33 %
10 - 15 km	932	238	74 %
Total	6813	4506	34 %

Table 4.3: LoRaMote (SX1272) measurements from moving boat, SF12 used. Results reproduced from [18]

Range	Transmitted packets	Received packets	Packet loss ratio
5 - 15 km	2998	2076	31 %
15 - 30 km	690	430	38 %
Total	3688	2506	32 %

While direct comparison between measurements taken while travelling in car (table 4.2) and in boat (table 4.3) may not be fully representative (due to the relative vagueness of the results), a few hints can be seen none-the-less. For longer distances, there is a clear favor in sending the RF signals over water compared to over land. The resaons for this might be numerous, e.g. better line-of-sight, lower velocities or superior reflectivity coefficient [16, Table 2.3, Chapter 2.4] to name a few.

4.3 Test parameters

The tests were performed at lake Lygnern, located a few kilometers south of Gothenburg. This location was chosen as it closely resembles the end system's intended environment and it also allowed us to do comprehensive line of sight testing of up to 15 km without needing access to a boat. The location is also within a short distance from Gothenburg, which made carrying out the tests easier. Another benefit

of this location is that little to no other LoRa traffic was encountered during testing.

We collected data for 4 locations in total, the locations were chosen such that they were 2 km apart from 1 to 7 km. All locations were chosen such that the transmission link experienced roughly the same conditions, that is, there was always line of sight, the gateway and transmitters were placed at heights such that the effects of the curvature of earth and Fresnel zones (see chapters 3.3.1 and 3.3.2 for more details) had minimal affect on the result. Furthermore, we tried performing all the testing during similar weather conditions. A photograph of the lake and the surrounding nature is shown in figure 4.2.

Figure 4.2: The Kerlink LoRa IoT station positioned at lake Lygnern

At each location we transmitted the current GPS-position of the transmitter in hexadecimal format, resulting in a message that resembled the final weather data message length. In order to be able to distinguish between what spreading factor was used for transmission, the message "Port" numbers were set to the corresponding spreading factor. Furthermore, each message sent by the transmitter contains a frame counter that increases for each new message transmitted. We collected this frame counter number and used it for calculating the PER. Alongside the previously mentioned information, the gateway we used provided us with additional information, such as frequency channel, data rate, signal to noise ratio and received signal strength indicator (RSSI).

The gateway was connected to the internet through a 3G connection and communicated with a network server and application residing on server EU1 at www.loriot.io. The application at loriot.io forwarded the data to an IBM Bluemix IoT hub that was connected with a cloud-based database application which automatically stored the collected data. We considered this the easiest and best way to store the collected

data.

In order to gain statistically significant results about the PER of LoRa we transmitted 1000 messages at each location. In general, when simulating bit error rates you wish to transmit at least 10^2 bits more than the reciprocal of the corresponding bit error rate you are aiming for, that is if you wish to have a results for a bit error rate of 10^{-5} you have to simulate transmission of at least 10^7 bits. Following this logic we would have liked to increase the number of transmitted messages at each location. However, due to the duty cycle limitations of the ETSI regulations, which the LoRaWAN protocol adheres to, increasing transmitted messages at each location by an order of magnitude or more would have made the testing prohibitively slow. The low amount of transmitted messages has to be taken into account during evaluation of the PER. However, we also have to consider that each packet contains multiple bits and that the PER does not distinguish between completely lost packets and single bit errors in a packet. Therefore there is not a one to one mapping between bit error rate and PER and the resulting PER might be substantially higher than the bit error rate.

4.4 Results

The results from the measurements (taken at distances [1, 3, 5, 7] km) are given in tables 4.4 - 4.7. The tables are quite self-explanatory, and what needs to be known about the PER was discussed in section 4.3. However, there are a couple of other things worth mentioning before delving into the results.

The RSSI values are calculated at the receiver, which in this case is Kerlink's LoRa IoT station 868 [21], [22]. As possibly suggested by its denotation, RSSI is a measured value. In general, the precision of the measurements degrade when the signal strength is either far above the receiver's sensitivity (above -100 dBm), or when the SNR (which is calculated using the RSSI) is below zero [24, Chapter 5.5.5]. This means that the SNR and RSSI values, as given in tables 4.4 - 4.7, should advisably be seen with a somewhat skeptical view.

While at the subject of the RSSI measurements, it is worth mentioning although the values given in tables 4.4 - 4.7 are averaged, in figures A.5 - A.8 and A.13 - A.16 (see appendix A) all the reported values for RSSI and SNR are illustrated, respectively.

As a final note on the subject it must be noted that while the RSSI values can (and indeed are [24, Chapter 5.5.5]) be calculated continuously (i.e. even when a signal is not being received), the IoT station (and its accompanying Internet service Loriot) will only report a value when a received package has successfully been decoded.
The measurements taken when the transmitters were at a distance of one km from the receiver are given in table 4.4. For spreading factors 7, 8 and 10, the results look pretty much the same both in terms of RSSI/SNR and PER, performing in the same range (slightly worse) was spreading factor 11. Only 894 messages were

Spreading Factor	Received msgs.	PER %	Avg. SNR [dB]	Avg. RSSI [dBm]
7	994/1000	0.6	8.861	-77.744
8	997/1000	0.3	9.83	-74.554
9	963/1000	3.7	9.387	-96.339
10	993/1000	0.7	9.078	-74.392
11	884/894	1.1	8.578	-74.392
12	962/1000	3.8	7.103	-95.468

Table 4.4: Test results at transmitter-receiver-distance one km

Spreading Factor	Received msgs.	PER %	Avg. SNR [dB]	Avg. RSSI [dBm]
7	997/1000	0.3	8.808	-86.533
8	982/1000	1.8	8.202	-97.662
9	971/1000	2.9	10.120	-93.920
10	994/1000	0.6	8.421	-92.891
11	995/1000	0.5	8.066	-92.542
12	987/1000	1.3	8.694	-91.726

Table 4.5: Test results at transmitter-receiver-distance three km

transmitted on spreading factor 11 due to human errors during testing.

What can clearly be seen is that the measurements taken when using spreading factors nine and 12 perform quite a bit worse compared to the other four. The RSSI column hints at much lower signal strength for those two spreading factors. Curiously, the SNR reported for SF9 is the second highest, while the RSSI is the lowest one.

To get a better overview (compared to the somewhat cluttered illustrations in figures A.5 - A.8 and A.13 - A.16) of the fluctuations of both SNR and RSSI, histograms are given in figures A.9 and A.1. In similar fashion, histograms for distances three, five and seven kilometers, are given in figures A.10 - A.12 and A.2 - A.4, for SNR and RSSI respectively.
For the transmitter-receiver distance of three km, the problem of seemingly optimistic SNR values (given the relative PER) surfaced again, for SF9 (see appropriate row in table 4.5).

Spreading Factor	Received msgs.	PER %	Avg. SNR [dB]	Avg. RSSI [dBm]
7	901/1000	9.9	-0.180	-116.173
8	991/1000	0.9	5.489	-111.942
9	994/1000	0.6	6.887	-111.815
10	993/1000	0.7	5.944	-107.196
11	965/1000	3.5	3.302	-111.267
12	945/1000	5.5	-3.254	-117.157

Table 4.6: Test results at transmitter-receiver-distance five km

Spreading Factor	Received msgs.	PER %	Avg. SNR [dB]	Avg. RSSI [dBm]
7	783/800	2.215	6.777	-104.586
8	797/800	0.375	8.280	-101.920
9	787/800	1.625	7.464	-107.839
10	0	0	6.795	-107.180
11	0	0	7.624	-98.333
12	0	0	4.299	-106.117

Table 4.7: Test results at transmitter-receiver-distance seven km

At five km distance between transmitters and receiver, the RSSI values start to approach the sensitivity of the receiver when SF7 is used (compare the RSSI in table 4.6 with the ones given 4.1). Thus, for SF7, the PER sees a clear increase due to low signal strength. Somewhat surprisingly, after SF7, the worst performers were the two spreading factors with the most delicate (i.e. best) sensitivity, SF11 and SF12.

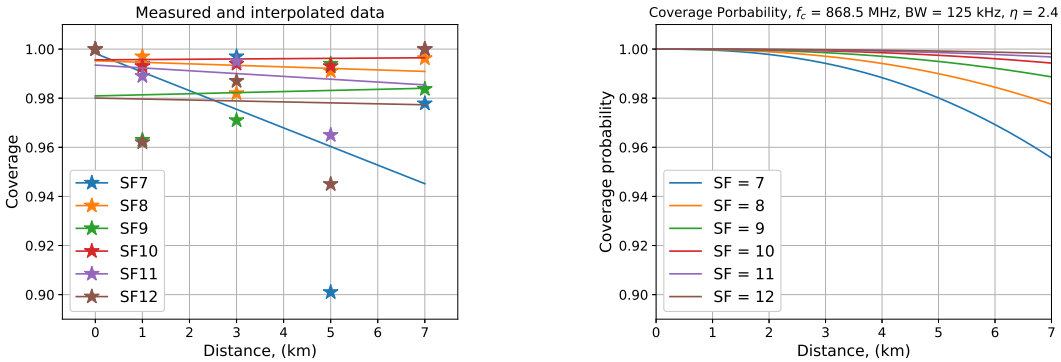
In order to find a spot with line-of-sight at transmitter-receiver distance of seven km, it was necessary to move the receiver to a location situated at higher ground. The receiver was moved to a position approximately 20 meter above the surface of the lake, compared to about five meter during the other ranges. The extra height would then help mitigate some of the problems associated with failing the Fresnel zone clearance recommendations for LOS radio links (see chapter 3.3.2). That would help to explain the increased RSSI when comparing seven km (table 4.7) to the shorter five km distance (table 4.6).

Regrettably, the measurements were not complete. Due to connectivity problems between the gateway (Kerlink LoRa IoT station) and Loriot (the service handling all the Internet-bound traffic from the gateway), a large part of the payloads (PHYPayload) on SF12, SF11 and SF10 were reported as dropped, while the headers (PHDR, see table 2.1 for reference) were still correctly decoded. The service resumed normal operation for the last 800 packages sent on SF7, SF8 and SF9. Thus, while we choose to only include the PER for spreading factors seven, eight and nine (see table 4.7), we decided on keeping all the SNR/RSSI (also histogram and linear plots in figures A.12, A.4, A.16 and A.8).

With the results from our measurements available, it is tempting to compare them with the results from the related studies (as presented in section 4.2). Starting with the paper that performed similar measurements (Petäjäjärvi et al., see section 4.2.2), the importance of LOS is plain to see. Comparing PER between tables 4.2 - 4.3 and 4.4 - 4.7, a decisive advantage can be seen when a direct LOS is kept between transmitter and receiver. Admittedly, the LoraMote used for transmitting in the other study had a PCB antenna of 0 dBi gain [23], compared to the 3 dBi gain of the standard 868 MHz antenna used on the RN2483 modules. Also, the transmitter was moving around, causing Doppler shift in the received signals in the other study not seen in our case. However, given one of LoRa modulations selling

points is its high Doppler resistance [5, Section 4.1.5], and given the (comparatively) small impact that 3 dB has on the RSSI values in tables 4.4 - 4.7, we still believe that the lion's part of the PER difference is due to the direct line-of-sight. Of course, other parameters will also affect the performance of the radio link, e.g. adequate (receiver) antenna height to properly combat problems related to Fresnel clearance (see section 3.3.2 for explanation and general guidelines).

If the results in tables 4.4 - 4.7 were to be interpolated over the whole distance covered, the end-result would be similar to what is given in figure 4.3a.



(a) The measured data (stars), interpolated using simple linear regression

(b) The theoretical probability of coverage using path loss exponent $\eta = 2.4$

Figure 4.3: Comparison between the measured data (tables 4.4 - 4.7), fitted using linear regression and the coverage probability from section 4.2.1 (using path loss exponent $\eta = 2.4$ and zoomed in accordingly)

The linear regressions in figure 4.3a is only intended to serve as a rough outline, usable in a comparison with the theoretical coverage probabilities discussed in section 4.2. Given the scarceness of points measured from and the inconsistent results (due both to the small number of messages sent and the varying conditions presented at the different locations), a linear fit was the only viable option.

Comparing figure 4.3a with to the coverage probabilities for different values of the path loss exponent (figure 4.1), it can be seen that the measured data perform in the region of what could be expected when the path loss exponent of 2.4 is chosen in Friis' transmission equation (see equation (4.2)). A properly zoomed in version of figure 4.1a is given in figure 4.3b, alongside figure 4.3a for easy comparison.

While far from an exact fit, figure 4.3 goes some way in affirming that the theoretical model from section 4.2 [17] could be useful for determining what spreading factor to aim for, given a certain range and a threshold for acceptable PER. However, the usage of the model is conditioned on a good approximation of the current channel's path loss exponent. In our case, it turned out that $\eta \approx 2.4$, which should be considered a conservative estimate. Looking at figure 4.3a, it can be seen that

the linear approximations are deflated considerably due to the measurements at the five-km-distance. With proper antenna heights for all tests (not only the ones taken at seven km), the approximation of the path loss exponent would be even lower than 2.4. This makes sence, given that $2.4 \leq \eta \leq 2.7$ assumes suburban environments (i.e. only partial LOS), and that the conditions provided during the measurements can be considered direct LOS, with generally favorable surroundings.

It must be stressed that the results from the measurements are somewhat inconsistent. To be more specific, the PER did not uniformly increase as a function of distance between receiver and transmitter. For some spreading factors, the PER even seemed to decrease when the distance grew (see figure 4.3a). Like previously said, some of the inconsistencies could partly be explained by the varying conditions (in terms of antenna height and surroundings) presented at each test spot. Another factor surely contributing is the relatively low number of messages sent (see section 4.3 for details).

There is also the possibility of hardware deficiencies, in either the transmitters (RN2483 modules) or the receiver (Kerlink LoRa IoT station). Looking at the RSSI figures (A.5 - A.8), drops of varying magnitude (averaging at approximately 15 dB) can be seen in the strength of the signals. It is not unusual for these drops to remain for long periods (several hundred messages in a row). Seeing how the drops are spread randomly over both time and spreading factors, it made us question if perhaps faulty hardware could be the cause of this. However, since there is no clear fit between these RSSI/SNR disturbances and increased PER, it is possible that the anomalies are simply faulty RSSI values being reported by the receiver. Again, given that this information determines what data rate the gateway finds appropriate (if adaptive data rate is operating), this issue could benefit from further investigation.

4.5 Discussion of test results

The reason for performing these tests was (as described in section 4.1) to see how the different data rates (i.e. employing different spreading factors) affects the PER. At the relatively short distances where PER were measured, the use of higher spreading factors (especially SF11-SF12) did not pay off. The main reason being that from a performance perspective, not much set them apart. As has already been pointed out, the tests are far from complete. A few more measurement spots at further distances would have been preferable. However, due to hardware associated problems detected after the initial round of tests, the measurements had to be carried out from scratch. Given the time needed on each location, tests at distances above seven kilometers had to be omitted.

One might ask: why not simply use the highest spreading factor and maximize the sensitivity? According to the theory from one of the related studies [17], even the model best approximating the test results shows that higher spreading factors will reduce the PER, also for distances below 10 km. A higher reliability (although theoretical) would also somewhat compensate for the incompleteness of the mea-

surement tests (i.e. the absence of results for longer distances).

While the reasoning above is sound, there are several problems associated with exclusive usage of high (11+) spreading factors. The first one is power consumption. Assuming a fixed bandwidth then enhancing the sensitivity (increased spreading factor) means that the chirped signal's duration needs to be extended, thus increasing the time-bandwidth product $T_m B_w$ (details on how the chirp's sensitivity/processing gain is related to $T_m B_w$ can be found in section 3.2). LoRa spreads its chirps by altering the number of chips (bits carrying no information) needed to represent a symbol. Since each incremental step practically doubles the number of chips used to represent the payload information, it also means that the bit rate of said information is essentially cut in half (for details, see the nominal bit rate equation given in section 2.2.2.2). Of course, a symbol rate cut in half means double the transmission time. Given a constant power envelope, double the transmission time equals double the power consumption. Thus, by each increment in spreading factor, the battery life could be halved.

Even if power consumption is not the main concern, the problem of lowered data rates still persists. Since no listen-before-talk is used in LoRa, channel access time is restricted by duty cycles. A 1% duty cycle means time on air is a precious commodity. By essentially doubling the time of each message, each unnecessary increase in spreading factor will be expensive.

While it was seen in section 4.4 that the theoretical coverage model seemed to (approximately) fit the measured data when $\eta \approx 2.4$, it must be mentioned that the model used comes with limitations. The most acute one being the fact that it does not take package collision into account. The LoRa network's best-effort-setup, in absence of tight node-synchronization, could be seen as employing the Aloha protocol. For a fixed time frame, the maximal efficiency is approximately $\frac{1}{2e}$ of said frame (see section 5.1.6). If the frame is optimized for smaller spreading factors, then the probability of collision will rise if the spreading factor increases. It is worth noting that Aloha networks are not node-constrained for networks consisting of small number of nodes. Thus, for a LoRa network made up of few nodes, collisions will not be a big concern. However, once the Aloha (i.e. LoRa) network reaches it maximum efficiency, the node-constraint will rapidly turn communication impossible due to collisions. This effect will be further amplified when higher spreading factors are used.

Another reason for advocating lower spreading factors instead of higher ones can be found in the first receive window (RX1) of LoRa nodes. When an uplink message has been sent, LoRaWAN mandates that the end-node waits for a (possible) response from the receiver (gateway). Depending on the spreading factor (SF7 - SF12) utilized for the upstream, the downlink message (e.g. message confirmation from the gateway) will have a varying data rate offset. What this data rate offset (i.e. spreading factor offset) between the transmitted upstream message and the response message from the gateway in the first receive window (RX1DROffset) is

RX1DROffset	0	1	2	3	4	5
Upstream data rate	Downstream data rate in RX1 slot					
SF12	SF12	SF12	SF12	SF12	SF12	SF12
SF11	SF11	SF12	SF12	SF12	SF12	SF12
SF10	SF10	SF11	SF12	SF12	SF12	SF12
SF9	SF9	SF10	SF11	SF12	SF12	SF12
SF8	SF8	SF9	SF10	SF11	SF12	SF12
SF7	SF7	SF8	SF9	SF10	SF11	SF12

Table 4.8: The gateway may find it necessary to send repeated downstream messages to an end node. An would be if a message confirmation does not have the intended effect on an end node. Columns 0 - 5 indicates how downstream messages will cycle through different SFs depending on the SF used in the original upstream message. Table reproduced from [9, Chapter 2.1.7]

depends on the preceding upstream data rate and is shown in table 4.8.

It can be seen in table 4.8 that the downstream data going in the RX1 slot has a heavy bias towards SF12. For example, if employing the highest SF is the rule, then RX1DROffset will not shift. Instead, all the repeated messages going in the downstream direction will be stuck at SF12, without the ability to cycle through the higher data rates. For small LoRaWAN networks, this will not pose a big problem. When a network is crowded however, this could cause unnecessary packet collisions.

5
Network design

The design of a specification for intermediate-nodes is discussed and presented in this chapter. The use case for intermediate nodes is to reliably extend the network range without having to add additional, expensive gateways. A successful specification of intermediate nodes would allow for reliable transmission over at least a double-hop link, making the installation of weather stations feasible in more locations.

5.1 Considerations

When designing the protocol for the intermediate nodes the weather station use case along with the limitations of the LoRa protocol have to be considered. In this section some several of the, what we consider, most important considerations when designing a range extending network specification are listed and shortly discussed.

5.1.1 Range of LoRa

As stated in LoRaWAN marketing material nodes can have a range of up to 15 km or further when in line of sight. However, the range of the LoRa protocol depends on multiple factors, such as data rate, weather and line of sight. Given steady weather conditions and data rate, the further a transmitter moves from a receiver, the higher the PER will become. To combat the increasing PER the data rate can be lowered. Therefore, in this network specification, a trade-off between data rate and range has to be made. The results of the chip to gateway test, presented in section 4.4, can be used to make an informed decision about a suitable data rate which balances both the range and data rate criteria.

5.1.2 Frequency Channel

The ETSI regulations and LoRa regional parameters specification [9] specify that frequencies between 863 MHz and 870 MHz can be used. Furthermore, the specification requires each chip to implement at least three different channels, 868.10 MHz, 868.30 MHz and 868.50 MHz. These channels are used to guarantee a minimal common channel set between end-devices and gateways. However, for the intermediate-nodes, the possibility of deviating away from this common set of frequency channels exists and it might even be wise to do so in order to avoid collisions. One of the main considerations when deciding on which frequency channel to use is whether all units should communicate on the same channel, simplifying setup

and operation of the nodes, or whether each intermediate node should form its own small cluster communicating on its own frequency channel. The advantages and disadvantages of both methods are listed below.

Single frequency
The main advantage of using a single frequency for communication between the end-devices and intermediate nodes is that it simplifies setup and operation of the nodes. A new end-device or intermediate-node can be inserted anywhere in the network and it can start communicating right away without any additional setup. Furthermore, if there are multiple intermediate nodes in range of an end-device, all intermediate nodes will receive and have the ability to forward the message, increasing redundancy and possibly minimizing the need for re-transmissions by the end-device (if re-transmission is used). If one intermediate-node goes down, some end-devices might still be able to reach other intermediate nodes and stay online. However, this simplicity and redundancy comes at a cost of an increased chance of collisions and lower network throughput. Assuming a simple Aloha protocol [25] is used for multiple access, the risk of collisions will grow with an increasing number of nodes, increasing the risk of dropped frames.

Multiple frequencies / cells
Creating smaller cells around each intermediate-node, where each cell communicates on its own frequency channel has some considerable advantages. A cell implementation where each cell utilizes a frequency channel that does not intersect with neighbouring cells removes the risk of inter-cell collisions. Assuming a simple Aloha protocol approach for multiple access, in-cell collisions are still possible. However, if the cells are kept small the collision risk will be minimal compared to a single frequency setup. The main drawback of a frequency cell setup is that it requires extra steps during setup/commissioning of the system. Furthermore, when adding additional end-devices each end-device must be configured to match its intended intermediate node. When adding a new intermediate-node a new cell needs to be created and the end-devices which connects to it need to be configured or re-configured to match the settings of the new intermediate-node. Another big drawback for the cell layout is that it has no redundancy built in if an intermediate-node goes down, meaning all end-devices connected to an intermediate node go down with it.

It is clear that both the single frequency and multiple frequencies approach have their advantages and disadvantages. Both approaches described above can be seen as extremes and a compromise might be a valid idea. A possible compromise would be to limit the multiple frequencies method in a way, such that it would easier to add new intermediate-nodes and additional end-devices without a cumbersome setup. A possible approach is described below.

Mixed approach
Instead of configuring each intermediate-node and all end-devices belonging

to the cell, we only configure the intermediate-node to respond to different frequencies. The end-devices are then configured, such that they contain a list of legal frequency channels. When a new end-device is added to the network or a cell, it transmits a join message, cycling through the frequency channel list until it receives a response. This way you get a simple setup but most of the benefits of a multiple cell system. However, if an intermediate-node goes down, the system still does not have the redundancy of a single frequency system. Furthermore, this solution requires bi-directional communication while the other solutions only require an uplink connection from the end-devices to the intermediate node.

As can be seen from the methods above, no clear cut solution exists. The aforementioned methods all have their advantages and drawbacks which have to be considered when creating the network specification.

5.1.3 Spreading Factor

An additional limitation imposed by using an intermediate-node compared to a gateway is that in general, the transceivers in the end-devices/intermediate-nodes only support listening to a single channel and a single spreading factor at the same time. This requires the intermediate-nodes and end-devices to agree on using a single spreading factor for communication. Like multiple frequency channels can be used to differentiate between network cells if a multiple- or mixed frequency approach from section 5.1.2 is used, the spreading factor can be used to differentiate between different cells if a single frequency is shared by all cells. However, using multiple spreading factors has some obvious drawbacks compared to using multiple frequencies to differentiate between cells. The main drawbacks are that there are fewer spreading factors available for use compared to frequency channels and that spreading factors drastically affect the range and data rate of the network. The fact that the spreading factor dictates the range and data rate is also our main consideration when choosing which factor to use, so creating different cells by use of different spreading factors might be counterproductive.
Due to the effects that the spreading factor has on the data rate, it can severely impact the network usability. Accounting for LoRa's duty cycle restrictions and low data rates, choosing a high spreading factor, which has a low data rate, will have a negative effect on transmission collisions. Furthermore, if a collision occurs for these higher spreading factors, the transmitter will have used up a fair bit of its duty cycle and might not be able to re-transmit its message due to duty cycle limitations. For example, according to LoRa Modem Calculator Tool a 16 byte message with a 10.25 symbol preamble length has a time on air of 926 ms.

This limits a transmitter to a maximum of around 36 messages per hour, including re-transmissions. This perhaps is no issue for a small network with few nodes, but can become cumbersome when the network grows. Furthermore, the longer range of the higher spreading factors (SF10, SF11, SF12) compared to the lower spreading factors (SF7, SF8, SF9) increases the likelihood of collisions. Therefore a lower

spreading factor is preferred, as it allows for less interference between nodes, higher data rates and reduces likelihood of collisions compared to higher spreading factors.

5.1.4 Message and Node Identification

In a regular LoRaWAN, a sender and a message can easily be identified by the network due to the built in headers. However, when using intermediate nodes, this becomes less straight forward due to fact that no MAC-layer information is sent when the devices utilize the LoRa protocol for point-to-point communication between themselves outside of LoRaWAN. Thus, the intermediate node has to wrap the message it is forwarding in its own header. Due to this wrapping, the message appears to originate from the intermediate-node. Therefore it will be hard to tell where a message is originating from on a network level, instead extra logic has to be added to the receiving application, such that it can distinguish the received messages. In order to distinguish the messages, each sender and each message needs to include a unique identifier that an application can make use of. As the data rates supported by LoRa are very low and we have strict duty-cycle restrictions, the main consideration for identifiers is to keep them small. Due to the fact that the intermediate-nodes has to forward these identifiers, keeping them small could allow for higher throughput and for more messages to be bundled together when they are being forwarded by the intermediate-node.

5.1.5 Acknowledgement of reception by intermediate node

Ideally it would be good to have the intermediate node send an acknowledgement that it has received a message by an end-device. This can be used to ensure that the message reaches its destination as the end-device can re-transmit until it receives an acknowledgement from the intermediate node. If the network consists of radio cells, the use of acknowledgements might increase the networks ability to self-heal in case of outage of an intermediate node. If an end-device does not receive an acknowledgement in a pre-determined time, it tries to join another cell. Acknowledgements, however, have some drawbacks too. One of the main drawbacks is that there is no built-in acknowledgement message for the radio protocol, meaning that this functionality would have to be implemented in software. Furthermore, due to duty cycle limitations the end-device may inadvertently use up its duty cycle allocation during re-transmission or the intermediate-node might use up all its duty cycle for acknowledgements, leaving it unable to forward the received messages. This could lead to further loss of messages. As weather data is generally slow varying with time, it is not critical that each and every message reaches its destination. Therefore, it might be wiser to simply increase the update frequency of the weather data instead of using acknowledgements.

5.1.6 Transmission protocol

One of the simplest transmission protocols is known as Aloha [25]. In Aloha, each end-device that has any data to transmit, transmits its data whenever it is ready

to do so. For our network, this simple approach is desirable. As the Aloha protocol does not require time synchronization, it becomes extremely easy to implement. The main drawback of the Aloha protocol is that is has a max throughput of only $1/2e \approx 0.18 = 18\%$ [25]. However, due to the small number of nodes in our network and low message frequency from each node, network congestion is not of huge concern. Another notable transmission protocol to consider would be Slotted Aloha, this is an improved version of Aloha, where end-devices are only allowed to transmit during predefined transmission slots. Slotted Aloha has double the maximum throughput of the simple Aloha protocol, which gives a maximum throughput of $1/e \approx 0.36 = 36\%$. However, slotted aloha requires time synchronization between end-devices such that all transmission slots align correctly. The time synchronization can however be very hard to achieve and we therefore wish to avoid implementing it if possible. The simple Aloha protocol is thus considered to be best suited for our needs.

5.1.7 Data transmission frequency

The end-devices and intermediate-nodes in this network will be transmitting weather data to an end application. Although weather often seems to change from minute to minute, the weather is rather slow varying and we do not need such high granularity data for the intended application. Each end-device might not necessarily need to transmit aggregated weather data more often than once every 5 or 10 minutes. This allows for long intervals where the radio is powered of and therefore also saves battery, prolonging the lifetime of the node. We also see that even at high spreading factors, we will not be able to fully utilize the duty-cycle if our transmission frequency is kept below a single message every 5 minutes. However, we have to consider that if all nodes transmit their updates at an equal update frequency, this might create continuing message collisions, resulting in lost data. This is due to the fact that if two devices experience a transmission collision, but neither device gets any feedback about the collision, they will keep transmitting messages with the same frequency, continuing the message collisions in perpetuity. Therefore, introducing some randomness to the update frequency of the weather data might be necessary to avoid this scenario.

5.1.8 Packet size from intermediate-node to Gateway

Since the information the nodes transmits is of fixed length, there is no reason to have variable packet lengths between end-devices and intermediate-nodes. However, the intermediate-nodes may have received several packets from end-devices before they re-connect to the LoRaWAN network for forwarding of the stored messages. Due to the short message format used to transmit the weather information, the MAC headers in the LoRaWAN protocol can be a significant part of the total transmitted message. It would therefore be beneficial to aggregate the data and transmit several of the stored messages as a single message from the intermediate-node to the gateway.

5.1.9 Security

The PHY layer LoRa protocol does not offer any encryption options like the MAC layer LoRaWAN protocol does. The LoRa protocol only offers the option of a cyclic-redundancy-check, which checks the message integrity, but provides no encryption. Therefore, unless implemented by the end-devices and intermediate-nodes, all messages between those two parties will be sent in the clear. Although the weather data itself perhaps is not very sensitive information that needs to be protected, the lack of encryption and lack of a secure way for end-devices to identify themselves and intermediate-nodes to verify this identification means that the system is very vulnerable. A possible vulnerability would be an injection attack where an attacker poses as an end-devices and transmits false data to the intermediate-node. This could inject false values into the end application. In a long-term implementation of this kind of network, the security needs to be taken into account.

5.1.10 Over The Air Updates

As is known, engineers don't always get things right on the first try and devices need to be maintained, the need to update the device software arises. Due to the spread out nature of the devices it would be beneficial to have the means of updating the end-devices and intermediate-nodes over the air (OTA). However, doing OTA updates requires bi-directional communication. As mentionend in chapter 2, LoRaWAN supports bi-directional communication, but it is very limited. Furthermore, OTA updates of end-devices also requires bi-directional communication between the intermediate-nodes and end-devices, adding additional complexity to the protocol. On top of the communication limitations, OTA updates will require additional means to verify that the received update is correct and initiated by a trusted party. We therefore suggest that OTA updates should not be made available.

5.2 Network Extending Specification

Having listed and discussed considerations for a network extending specification for intermediate-nodes in the previous section, an implementable specification is proposed in this section. The specification is intended to be used in conjunction with a LoRaWAN as intermediate-nodes need to intermittently connect to a LoRaWAN for message forwarding to a centralized network server. The aim of this specification is thus to define how end-devices and intermediate-nodes communicate.

5.2.1 Range of LoRa - Placement of nodes

The placement of a node depends on a large number of parameters. While each location needs to be evaluated, there are a few guide lines worth mentioning here. The first thing to consider is the surroundings of the communications link. It is assumed that all nodes will be located at sufficient height to compensate for Earth's curvature, in accordance with the quick guide given in chapter 3.3.1).

Distance	1 km	3 km	5 km	7 km	9 km
Fresnel zone 1 maximum radii	9.3 m	16.1 m	20.8 m	24.6 m	27.9 m
Recommended clearance	8.6 m	12.7 m	15.5 m	17.8 m	19.7 m

Table 5.1: Maximum Fresnel radii and the accompanying recommended clearances for possible transmitter-receiver distances in intermediate-node connections

Supposing then that the scenery will be similar to the archipelago outside of Gothenburg, a large part of the environment will consist of water. Although calm water can reflect a signal favorably from the perspective of a receiver, it is probably unwise to assume open water to always behave. Even when relatively calm, highly reflective surfaces can cause unwanted effects. These effects were mentioned in 4.4 when discussing the RSSI variations between measurements taken at transmitter-receiver distance of five and seven kilometers. Although two kilometers further apart, the extra antenna-heights provided at the seven kilometer distance resulted in a sizable RSSI increase compared to the five kilometer distance.

Recalling the Fresnel zone clearance, mentioned in section 3.3.2), equation (3.15) can be used to calculate the first Fresnel zone radius for probable distances in intermediate-node communication links. The maximum radii can be seen in table 5.1.
Using the maximum heights as references, the recommended clearances, utilizing the 60% plus an additional three meters rule, are given in table 5.1.

5.2.2 Frequency Channel

For the intended application of weather stations, we propose to utilize a single frequency channel for communication between end-devices and intermediate-nodes. Preferably we would recommend the making use of a frequency channel between 868 and 868.6 MHz or 869.7 and 870 MHz to as it allows for a duty cycle of 1% in compliance with ETSI regulations. The communication between the intermediate-nodes and the gateway would be relegated to the pre-defined default frequency channels of 868.1 MHz, 868.3 MHz and 868.5 MHz. Therefore, in order to minimize the collision risk, the upper frequency band of 869.7 to 870 MHz might be better suited for end-device to intermediate-node communication. As the weather station network is relatively sparse, the benefits of a simple single frequency network outweigh the drawbacks. Furthermore, due to the sparse nature of the weather stations, it is unlikely that they will run into network congestion problems.

5.2.3 Spreading Factor

In order to decrease the time on air for the end-devices we wish to keep the spreading factor as low as possible. Our testing showed that with good placement of the

gateway and end-devices, such that line of sight was achievable, there was very little difference in PER between SF7, SF8 and SF9 even as distances grew to 7 km. However, these tests represent a best case scenario where we have a gateway with a higher gain antenna and possibly a more sensitive receiver compared to an RN2483 chip. Therefore we might experience a performance drop when going from end-device to gateway to end-device to intermediate-node, which both utilize the same RN2483 chip. With well placed intermediate-nodes and end-devices this should not have a considerable negative effect of the performance. We therefore suggest utilizing SF8 for communication between end-devices and intermediate-nodes. As the data the weather stations are transmitting is fixed and of a limited nature, SF8 should easily provide the necessary data rate and range, while minimizing time on air for the end-devices. Another benefit of choosing a low spreading factor is that it allows for either an increased update frequency of weather data or increased number of end-devices.

For the communication between an intermediate-node and a gateway we suggest using SF7 for data transmission. In our testing we saw that the performance of SF7 with regards to PER was stable even with increasing range. In addition, for this project the gateway would be placed on a mast on top of SSRS office in Långedrag. With smart placement of the intermediate-nodes this allows for line of sight communication between the intermediate-nodes and gateway, mirroring the environment of our testing and as such we can expect similar performance. Another reason for recommending SF7 or a lower spreading factor than the end-devices use, is that the intermediate-node must be able to aggregate and transmit all the data received from the end-devices, within its own duty cycle. Of course, if there are few end-devices or/and the update frequency of the weather data is limited, the intermediate-node will be able to aggregate and transmit the received data even if using the same or possibly a lower spreading factor. However, choosing a lower spreading factor, allows for either more devices in the network or a higher update frequency, which both are desirable traits.

5.2.4 Message and Node Identification

Each node should be given a unique two byte node identification number during commissioning, this number is then always included when the node transmits a frame of data. This allows for 65536 unique nodes in the network. For message identification, a two byte number is included in and incremented with each transmitted frame. When the frame counter reaches 65535 it is reset to 0 and resumes ordinary operation.

5.2.5 Acknowledgement of reception by intermediate node

There will be no acknowledgement procedure implemented in this network protocol. The main reason for this is simplicity, battery-life and duty cycle limitations. In addition, the data to be transmitted by the end-devices is weather data, which is usually slow varying, meaning the end application does not suffer from intermittent

Node ID	Frame Counter	Payload	CRC
2 Bytes	2 Bytes	10 Bytes	2 Bytes

Figure 5.1: Message format

loss of messages. Instead of using acknowledgements, it is suggested to increase the update frequency of weather data such that a higher transmission loss can be tolerated. Although, it should be noted that increasing the update frequency can have a significant negative effect on battery-life.

5.2.6 Transmission protocol and Data transmission frequency

The network will utilize the Aloha protocol, due to its simplicity and ease of implementation. Thus, when a device has any data that it wishes to transmit, it transmits it. However, update rate of data, also referred to as the data transmission frequency, must be taken into account here. In order to avoid continued collisions between end-device transmissions, some randomness should be introduced into the update frequency of the data. The data transmission frequency is to be set at 300 s with a random factor of $\pm 10\%$. If deemed necessary, this data transmission frequency can be updated to better suit the requirements of the network. That is, it can be increased if the focus is more on battery-life and less on data granularity or decreased (within duty cycle limitations) if data granularity is more important than battery-life.

5.2.6.1 End-device to intermediate-node

The packet size between end-devices and intermediate-nodes will be limited to 16-bytes. The first two bytes are a unique node identifier and the third and fourth bytes combine as a frame counter. The trailing 12-bytes are used for the weather data/payload and a cyclic redundancy check. The resulting message format can be found in figure 5.1.

The implementation of each field in table 5.1 is expanded on below.

Node ID:
The node id is a unique identifier which should be set independently for each node. As the field is two bytes long, valid ID's are between 0 and 65535. The network is therefore limited to 65535 unique devices.

Frame Counter:
The frame counter is used to identify messages from end-devices. The frame counter is increased with each transmitted message until it reaches its upper limit of 65535. When the frame counter is at its upper limit it starts again at 0. In addition to providing identification for messages the frame counter provides a way to calculate PER.

Temp.	Rain	Pressure	Wind Speed	Humidity	Wind Dir.	CRC
2 Bytes	2 Bytes	2 Bytes	2 Bytes	1 Byte	1 Byte	2 Bytes

Figure 5.2: Payload format

Byte	Encoding
5	MSB of measured temperature
6	LSB of measured temperature
7	MSB of measured rain
8	LSB of measured rain
9	MSB of measured air pressure
10	LSB of measured air pressure
11	MSB of wind speed
12	LSB of wind speed
13	MSB of measured humidity
14	MSB of wind direction
15	MSB CRC
16	LSB CRC

Table 5.2: Byte order of payload

Payload:
The payload format is designed to carry the sensor values from 6 sensors. Figure 5.2 details the payload format and which data is collected and transmitted. To fit into this payload format, the data shall be encoded as seen in table 5.2. As the hardware design of the weather station is not yet complete, we can not make final recommendations on how the data bytes should be used to represent the data. However, we will give general recommendations based on a format used for the LoRaMOTE, which is a multi-purpose test device with similar data recording capabilities. The LoRaMOTE uses the MPL3115A2 chip [26], we therefore base our recommendations on the data format used by that chip as it is described in the LoRaMOTE users guide [23].
Byte 5 and 6 represent a signed value of the measured temperature (x100) by the MPL3115A2 chip. The value can then be divided by 100 to get the temperature with decimal values.
Byte 7 and 8 represent the measured rainfall as an unsigned value. For the rainfall we have no reference sensor, therefore we leave it up to implementation to decide how these two bytes should represent the value.
Byte 9 and 10 represent the measured atmospheric pressure in deci-Pascal (dPa) as measured by the MPL3115A2 sensor. The number is therefore divided by 10 to get the hekto-Pascal number (hPa).
Byte 11 and 12 are used to represent the measured wind speed. We have no reference sensor, but we suggest that the data should be measured and transmitted in mm/s.
Byte 13 is used to represent humidity in percent. As this field is a single byte long, it does not support decimal values, only whole numbers. If the chosen

sensor gives decimal values, they should be rounded to nearest integer.
Byte 14 is used to represent wind direction. The wind direction as measured by the sensor should be converted into an unsigned integer value between 0 and 255. The measured degree value is converted into the integer by dividing the measured value by $360/256 \approx 1.40625$. The integer value can then be converted to a degree value by multiplying it by $360/256 \approx 1.40625$.
Byte 15 and 16 are used for a cyclic redundancy check. See section 5.2.7 for additional information.

5.2.6.2 Intermediate-node to Gateway

The payload format used between the intermediate-node and gateway is very similar to the format described in section 5.2.6.1. The only difference is that the intermediate-node has a variable payload length and can aggregate several received messages into a single message for forwarding. Therefore, the intermediate-node frame format contains a single byte header which informs the application about the number of forwarded messages the incoming message contains. That is, if the intermediate-node is forwarding three messages, it will set the first byte value to 3 and then concatenate the 42 message bytes (3 x 14 Bytes) before transmission. This is illustrated in figure 5.3.

No. Messages	Message 1	Message ...	Message N
1 Byte	14 Bytes	14 Bytes	14 Bytes

Figure 5.3: Intermediate-node frame format

This allows the intermediate-node to forward as many messages as it deems necessary, within the rules of the regional parameters as described in section 2.3.4.2.
In addition to forwarding messages, the intermediate-nodes can themselves be end-devices/weather stations. Therefore, each intermediate node also contains its own unique identifier and frame counter, as described earlier, and forwards its own messages in the same way as it would with any message received from an end-device.

5.2.7 Security

There is no built in encryption in the LoRa physical layer protocol. However, we strongly recommend that some encryption is used for the payload. We suggest making use of symmetric encryption, where the end-devices and intermediate-nodes share the same key. The key therefore has to be known by the intermediate-nodes and end-devices when they are being deployed. The reason for choosing symmetric encryption over asymmetric encryption, is that symmetric encryption usually requires shorter keys than asymmetric encryption. This makes the encryption easier to use on the very limited hardware that the weather stations will contain. Furthermore, as we have the option of distributing the keys during deployment, the advantage of asymmetric encryption is little to none.
We also wish to add a cyclic redundancy check to the end of the message, which gives additional error detection capabilities. We assume that the hardware we use

will be based on Arduino, therefore we suggest making use of AES_128 encryption. This is the same encryption standard that is being used by LoRaWAN. Furthermore, there already exists a good implementation called Arduino AESlib [27]. This simplifies usage considerably. For the cyclic redundancy check, there exists a good Arduino library called crc-16 [28]. The CRC should be calculated and added to the end of the payload before encryption. This both ensures that the CRC is immutable and furthermore creates a message of 16 bytes, which is the necessary input length for the encryption functions in the AES_128 library. The encryption key used between the end-devices and intermediate-nodes should differ from the Network- and Application Keys used for the LoRaWAN setup. This is to ensure that even in the event that the encryption key is leaked, devices will not be able to use it to connect to the LoRaWAN network. Furthermore, the AES_128 encryption key should be randomly generated, not chosen by the implementer.

5.2.8 Over The Air Updates

No mechanism for OTA updates shall be implemented. As OTA updates require bi-directional communication, the additional complexit and drawbacks of implementing it outweigh the advantages.

5.2.9 Connecting to and leaving LoRaWAN

The intermediate nodes must regularly connect to the LoRaWAN to forward received data and transmit collected data. When leaving the LoRaWAN, the RN2483 chips issue a mac-pause command and get a response telling them the amount of milliseconds they can leave the network for. The intermediate-nodes must therefore re-join the LoRaWAN before this time is exceeded. They should, however, limit their time on the LoRaWAN as much as possible as their intermediate-node functionality is none whilst they are connected to the LoRaWAN. Therefore, the intermediate-nodes should only connect to the LoRaWAN when they have data to transmit or when they are about to exit the out-of-network time limit and then leave the LoRaWAN again as soon as possible.

6
Discussion

6.1 LoRa and LoRaWAN

In general the LoRa and LoRaWAN protocols seem well thought out and cater nicely to the IoT use case. Having said that, both LoRa and LoRaWAN have some quirks.

6.1.1 LoRa

The LoRa modulation is a well suited for low-power and long-range, but suffers from a low data rate and low spectral efficiency. When used in the ISM bands, where duty cycles are strictly regulated, the use cases for LoRa are bound to non data-heavy applications. Furthermore, the use of Hamming codes for FEC feels somewhat counterproductive as the coding rate can be set to both 4/5 and 4/6, which introduces a 25-50 % increase in packet length, without introducing any error correction capabilities.

From the tests performed in this project, it was concluded that LoRa modulation can uphold low PER over the intended distances of the weather stations application. In order for the PER to stay sufficiently low however, the location of the transmitters had to be carefully planned. For long-distance end-nodes and intermediate-nodes in particular, appropriate clearance had to be meticulously considered. Such requirements make the deployment of LoRa capable nodes a more time consuming measure than it might appear at first glance, epsecially for projects similar to this in scope.

6.1.2 LoRaWAN

The LoRaWAN specification feels suitable for smaller deployments, where a network operator has control over most of the end-devices in the network. However, selling or giving unrestricted access to a network to multiple different users can have crippling effects on the network. Assuming every user wishes to fully utilize the duty cycles its devices, the Aloha protocol quickly breaks down due to network usage. Furthermore, the LoRaWAN specification allows end-devices to send confirmed packets, meaning packets should be acknowledged upon reception by a gateway. As the gateway too is bound by the same duty cycle regulations imposed by ETSI in the ISM bands, the gateway's duty cycle can be fully consumed by these acknowledgements. This leaves the gateway unable to respond to new incoming messages that requires acknowledgement, creating additional network congestion as devices will re-transmit already received messages due to the missing confirmation packets from the gateway.

Therefore, a more complex protocol than Aloha or stricter duty cycle limitations might be required in the future as LoRaWANs grow bigger.

6.2 Network extension specification vs. additional gateways

The network extension specification detailed in section 5.2 arises from the wish of minimizing the number of gateways necessary in a network. Here, the advantages and limitations of this approach are discussed. Suggestions are also made about how the proposed specification might be altered to better suit larger network deployments.

6.2.1 Advantages

The suggested network specification for extending networks with intermediate nodes has some clear advantages compared to extending the network with additional gateways. They are listed below:

Low cost hardware and easy setup.
The intermediate-nodes can be produced cheaply and their setup to the LoRaWAN is straightforward, just like any other LoRaWAN device. Compared to an intermediate-node, an additional gateway is several times more expensive, both in hardware and running costs. The monthly subscription costs of a gateway, for back-end and back-bone services, can easily surpass the cost of an intermediate-node.

Homogeneous hardware.
The specification is lightweight and is designed such that it can be easily implemented on the same hardware as the end-devices. Therefore, the only difference between a intermediate-node and an end-device is whether the device contains software for acting as an intermediate-node or not. This makes it easy to re-purpose end-devices into intermediate-nodes or vice-verse if necessary. The main suggested hardware addition for intermediate-nodes is a larger battery in order to accommodate the increased energy consumption due to the always listening nature and increased transmission rate, compared to end-devices.

No additional backbone connection necessary.
Unlike a gateway, an intermediate-node does not require any additional backbone connections, as itself is a relay node towards the gateway that has a steady backbone connection. The gateway's backbone connection can either be directly to the Internet through an Ethernet cable with network access or through a 3G network. These backbone connections can either be hard to find or costly to implement at the desired placement of an gateway making the

intermediate-node a better choice.

Increased battery-life.
As the intermediate-nodes contain far simpler hardware and only listens to a single frequency and SF, they draw less power than a gateway. This is an important consideration as the intermediate-nodes are likely to be located in hard to reach locations, with limited power sources.

Friendlier to the environment.
As the intermediate nodes consists of less hardware, they consume less resources and create less waste than a gateway. In addition, as mentioned above, during operation they consume less power than a gateway.

6.2.2 Limitations

A network utilizing the network extending specification, which uses intermediate-nodes, has several drawbacks and limitations compared to a network only utilizing gateways.

Single frequency and spreading factor
Communication between intermediate-nodes and end-devices are bound to a single frequency channel and a single spreading factor. Meanwhile, the gateways can listen to multiple frequency channels and spreading factors simultaneously. Therefore, the gateways introduce much more agility to the network. This also allows more end-devices to connect to each gateway compared to intermediate-nodes.

Simple hardware
The intermediate-nodes contains far simpler hardware than the gateways. Therefore, using intermediate nodes instead of gateways can lower the available link budget drastically, raising PER or limiting transmission range and placement of end-devices.

One-way communication vs. Bi-directional communication
The gateways offer bi-directional communication compared to the one-way communication of the intermediate-nodes. Although the bi-directional communication between gateways and end-devices is limited, it can be very useful. For example the bi-directional communication allows for adaptive data rates and over-the-air activation (OTAA) of new nodes.

Low capacity of intermediate-nodes
Due to the single frequency and single frequency setup of the intermediate-nodes, the network capacity is quicker to reach its limits compared to the multi-channel and multi-SF gateways.

No OTAA capability

The intermediate-nodes do not have the ability of conducting OTAA for end-devices. Therefore, all new additions to the network have to be pre-configured for the network. Possibly slowing down roll-out of new devices. However, this is perhaps not of great concern for a small network, such as the weather station network the protocol is designed for.

Lower data rate and fixed payload length

Due to the fact that the intermediate-nodes implement a double-hop network, compared to the single-hop network that the gateway connections provide, the total throughput from end-device to gateway will be significantly lower than from an end-device to gateway. Furthermore, as the intermediate-node has to forward all its received data within its duty cycle limitations, it creates even stricter duty cycle requirements for the end-devices that connect to it. The effective duty cycle of each end-device in a network of N end-devices and an intermediate node becomes $1/(N+1)$, assuming both intermediate-nodes and end-devices use the same spreading factor and that the end-devices fully utilize their packets. However, as the packets that the end-devices transmit are limited to 14 bytes of information, the effective duty cycle can be slightly improved by aggregating the data in the intermediate node and then forwarding it. This allows several end-device frames to share the overhead of a single intermediate to gateway message. If the communication between intermediate-node and gateway utilizes a higher data rate than the end-device and intermediate-nodes do, the effective duty cycle is less affected. However, utilizing only gateways, all end-devices can make full use of its duty cycle.

Back-end application logic necessary

Using the intermediate-nodes requires an additional processing step to decode the incoming message and attribute the data to the correct node. Utilizing gateways, the data can only originate from a single end-device. Therefore, in a gateway scenario, it is possible to make use of the device id that accompanies each LoRaWAN message instead of having to create new unique device identifiers.

Lowered battery life of devices.

Intermediate-nodes will have a significantly lower battery life than the end-devices. Compared to end-devices, the always listening nature of the intermediate-nodes, along with the message forwarding requirements which increases their transmission rates lowers the intermediate-nodes battery-life considerably. However, compared to a gateway the intermediate-nodes will still consume less power. Though, in general, the gateways would be connected to a fixed power-source, lowering the importance of energy consumption considerations.

No adaptive data rate.

Unlike the gateways, the intermediate-nodes offer no adaptive data rate. This necessitates additional compromises on range and data rate between intermediate-

nodes and end-devices as they must all use the same spreading factor. Therefore, end-devices located close to the intermediate node cannot utilize its location advantage compared to an end-device located further away, limiting the data rate from end-devices to intermediate-nodes to the lowest commonly usable data rate. Furthermore, forcing devices to use a higher SF than necessary lowers the battery life of the device as the higher SF requires longer transmit time.

6.2.3 Suggested network extension specification changes for larger networks

The network specification detailed in chapter 5.2 is well suited for smaller, less dense networks where the end-device count is low enough that we never reach the limits of the Aloha protocol. However, for larger and denser networks this network specification might break down due to some of the limitations mentioned above. Therefore for larger deployments where the risk of running into the limits of the Aloha protocol is significant, some modifications to the protocol are therefore proposed. Perhaps the first thing that should be re-visited is the payload format. The format works well for the intended weather station application but might not be optimal for other use-cases. It is still suggested that the payload format is kept to a fixed length multiple of 16, such that AES_128 encryption can be utilized. Furthermore, the continued use of a CRC is encouraged. If a network is reaching the capacity limitations of the Aloha protocol, it is suggested the specification should introduce multiple frequencies and possibly multiple SF. This allows the network operator to set up cells, allowing for co-existence of cells due to frequency multiplexing, significantly improving network capacity.

For a larger network implementation all these changes might be considered worthwhile. However, for the relatively small network that the weather stations will compose, these changes add unnecessary complexity. Another considerations is that for larger networks with dense end-device deployments the benefits of extending the network with intermediate-nodes becomes less as each additional gateway that is deployed will be able to serve a significant number of nodes. For these conditions, the advantages of a gateway might outweigh the advantages of the intermediate-node approach. Therefore, we believe that the network extensions specification is best suited for small extensions to a LoRaWAN, with use cases such as the weather stations or similar low density networks.

7
Conclusion

In this report the LoRa modulation and protocol, along with the LoRaWAN specification has been explored. We found the LoRa and LoRaWAN protocols to be competent and useful for IoT applications such as an IoT weather station network.

The FEC used in LoRa is found to be somewhat lacking as it offers four coding rates, where two rates don't offer any error correction capabilities and one rate only introduces an extra coding bit, but no additional error correction capabilities. Disregarding the FEC of the protocol, the LoRa modulation itself is sound, with linear frequency modulation being a tried and tested approach to spread spectrum.

In our testing we found that LoRa and LoRaWAN both perform well with low PERs at distances of up to 7 km when transmitting with line of sight. However, although the higher SF theoretically should outperform the lower SF, this was not the case during our testing. A probable reason is that for the distances tested, the measured RSSI values stayed within the sensitivity thresholds provided by each SF.

Another reason for the lack of performance differentiation between various SF might be due to the way that gateways in a LoRaWAN chooses which SF is used when responding to end-devices. Although our testings was performed with unconfirmed packets such that the gateway should not transmit acknowledgement responses to each received packet, the gateway still responded to one out of approximately 20 packets. Due to us performing the testing with multiple different transmitters transmitting on different SF simultaneously and with regular intervals, the gateway sent responses to several end-devices within a short time frame. As seen in Table 4.8, the gateway response messages are biased towards the higher SF. This bias towards the higher SF in conjunction with only three channels being used for up- and downlink transmission, long transmission time of SF12 messages and multiple well synced transmitters used for testing creates a higher collision risk between up-link messages transmitted from end-devices with high SF and response messages sent from the gateway, compared to up-link messages sent on lower SF. When a collision occurs, we are likely to experience a packet error, leading to an undue increase in packet errors on higher SF compared to lower SF. It should be noted that in normal operation, where end-devices are not as well synced and can not utilize the network as well as in our tests, the response messages from the gateways are less likely to affect the PER as substantially. It is more likely that the LoRawWANs transmission protocol would lead to high packet losses before the response routine of the LoRaWAN specification becomes highly disruptive.

We have discussed and specified a network extending protocol for LoRaWANs. The network extending protocol offers the possibility of extending a LoRaWAN in a cheap and simple manner. The specification is found to be well suited for deployment of smaller networks, where the extension only needs to handle a relatively small amount of units. When deploying larger networks the duty cycle limitations and Aloha medium access protocol become prohibitive. For the use case of simple IoT weather stations, the network extending specification is considered successful, although rather case-specific.

8

Future work

This report has investigated LoRa and LoRaWAN, and explored how the combination of modulation technique and network protocol might be useful for a weather station application. A specification that would allow for the weather stations to be placed outside of the range of the gateway has been designed. However, the design specifications have not yet been implemented. Thus, if permitted continued work on this project, implementation would be the logical next step. This would allow for the design to be verified. In addition, it means that performance testing could be carried out as well as exploring how the protocol behaves in the real world. It would also be interesting to see how the protocols/networks throughput is affected for different data rates and for larger deployments. Furthermore, implementing bi-directional communication into the extension would make new use cases possible and should also be worth exploring.

In addition to the weather data, future revisions of the weather stations might wish to transmit static images or even a live stream of current weather conditions. Since LoRa is a low data-rate protocol, future upgrades to the weather stations might need to make use of some other protocol if they wish to fulfill those transmission requirements. Therefore it might be beneficial to look into other IoT-protocols operating in the ISM bands which might be able to offer higher data-rates. Of course, this would come at the cost of range and/or battery-life.

While the tests performed in this report has shown that low PER can indeed be maintained over comparatively long distances, the performance is conditioned on clear LOS and adequately positioned transmitters. Another prerequisite that needs to be fulfilled is that the network must not run the risk of becoming congested. For the intended weather station application of this project, these requirements should be realizable. However, should the employment of LoRaWAN networks see continued increase, then the risk of interference from neighboring LoRa nodes will also rise. Thus, an interesting subject for future development of this project is that of interference mitigation. A paper that has looked at said topic is [29]. The paper found that for networks experiencing varying levels of congestion, three closely spaced gateways considerably improved the PER compared to employing a single gateway, even if highly directional antennas were utilized. For future development of this project, further monitoring the findings of [29] could prove beneficial.

Lastly, one subject probably worth further investigating is what kind of applications that LoRa and LoRaWAN are truly suited for. A study performed in the

paper [30] confirms the main drawback of LoRaWAN that has been already noted in this paper. That is, due to its restrictive duty cycles and low data rates, LoRaWAN is not a solution that caters all connectivity needs in the LPWAN space. Instead, every setup must be carefully designed to meet the requirements of each use case. Thus, a future development of this project could be to try and specify applications wherein LoRaWAN would have a high probability of performing well.

Bibliography

[1] P. Weyde. (May 31, 2017). Antalet sjöräddningsinsatser ökar - P4 Göteborg | Sveriges Radio, [Online]. Available: http://sverigesradio.se/sida/artikel.aspx?programid=104&artikel=6707458 (visited on 06/11/2017).

[2] ——, (Apr. 2, 2016). Kraftig ökning av sjöräddningsinsatser - Nyheter (Ekot) | Sveriges Radio, [Online]. Available: http://sverigesradio.se/sida/artikel.aspx?programid=83&artikel=6392915 (visited on 06/11/2017).

[3] LoRa Alliance. (Feb. 6, 2017). Lora-Alliance, [Online]. Available: https://www.lora-alliance.org (visited on 02/06/2017).

[4] Unkown. (Feb. 13, 2017). OSI model, [Online]. Available: en.wikipedia.org/wiki/OSI_model (visited on 02/13/2017).

[5] Semtech. (Mar. 15, 2017). AN1200.22 - LORA Modulation Basics, [Online]. Available: www.semtech.com/images/datasheet/an1200.22.pdf (visited on 03/15/2017).

[6] J. Pinkney, "Low Complexity Indoor Wireless Data Links Using Chirp Spread Spectrum", PhD thesis, Univerity of Calgary, 280 Discovery Place One, 3553-31 St. N.W., Calgary, Alberta, Dec. 2003, 243 pp.

[7] A. G. i Amat, *SSY125 Digital Communications, Lecture Notes*, Department of Signals and Systems, Chalmers University of Technology, Nov. 2015, 121 pp. eprint: PDF.

[8] LoRa Alliance. (Feb. 6, 2017). Lora-Techonology, [Online]. Available: https://www.lora-alliance.org/What-Is-LoRa/Technology (visited on 02/06/2017).

[9] LoRa Alliance Technical committee, *LoRaWAN - Regional Parameters*, 1.0, LoRa Alliance, Inc., Jul. 2016, 45 pp.

[10] ETSI. (May 2012). ETSI EN 300 220-1 V2.4.1 (2012-05), [Online]. Available: www.etsi.org/deliver/etsi_en/300200_300299/30022001/02.04.01_60/en_30022001v020401p.pdf (visited on 05/18/2017).

[11] Semtech. (Jul. 2013). ETSI Compliance of the SX1272/3 LoRa Modem, [Online]. Available: http://www.semtech.com/images/datasheet/etsi-compliance-sx1272-lora-modem.pdf (visited on 05/18/2017).

[12] ——, (Oct. 2013). Implementing Data Whitening and CRC Calculation in Software on SX12xx Devices, [Online]. Available: http://www.semtech.com/images/datasheet/AN1200.18_STD.pdf (visited on 06/01/2017).

[13] A. Oppenheim and A. Willsky, *Signals & Systems, International Edition*, 2nd ed. Prentice-Hall International, 1997, 957 pp., ISBN: 0-13-651175-9.

[14] A. Hein, *Processing of SAR Data, Fundamentals, Signal Processing, Interferometry*. Springer Berlin Heidelberg, 2004, ISBN: 9783662094570.

[15] Wikipedia. (Mar. 28, 2017). Horizon - Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Horizon (visited on 05/20/2017).

[16] R. Freeman, *Radio System Design for Telecommunications*, 3rd ed. IEEE Press (John Wiley & Sons), 2007, 880 pp., ISBN: 9780471757139.

[17] G. Orestis and R. Usman, "Low Power Wide Area Network Analysis: Can LoRa Scale?", in *IEEE Wireless Communications Letters*, vol. 6, IEEE, 2017, pp. 162–165.

[18] J. Petäjäjärvi, K. Mikhaylov, A. Roivainen, T. Hanninen, and M. Pettissalo, "On the coverage of LPWANs: Range evaluation and channel attenuation model for LoRa technology", in *2015 14th International Conference on ITS Telecommunications (ITST)*, IEEE, 2015, pp. 55–59.

[19] A. Goldsmith, *Wireless Communications*, 1st ed. Cambridge University Press, 2005, 674 pp., ISBN: 9780511841224.

[20] Semtech. (Jul. 2013). LoRa Modem Design Guide, [Online]. Available: http://www.semtech.com/images/datasheet/LoraDesignGuide_STD.pdf (visited on 05/24/2017).

[21] F. Zandbergen. (Dec. 16, 2016). Kerlink LoRa IoT Station - Specifications - The Things Network, [Online]. Available: https://www.thethingsnetwork.org/docs/gateways/kerlink/specs.html (visited on 05/24/2017).

[22] Kerlink. (May 2017). WIRNET STATION 868 MHZ - Kerlink, [Online]. Available: http://www.kerlink.fr/en/products/lora-iot-station-2/wirnet-station-868 (visited on 05/24/2017).

[23] Semtech. (Jul. 2014). LoRaMote - USER GUIDE, [Online]. Available: http://www.semtech.com/images/datasheet/User_Guide_LoRaMote_STD.pdf (visited on 05/23/2017).

[24] ——, (Mar. 2015). Semtech LoRa SX1272/73 - Datasheet, [Online]. Available: http://www.semtech.com/images/datasheet/sx1272.pdf (visited on 05/28/2017).

[25] E. Modiano. (Apr. 25, 2017). Lecture 10/11: Packet Multiple Access: The Aloha protocol, [Online]. Available: http://web.mit.edu/modiano/www/6.263/lec10.pdf (visited on 04/25/2017).

[26] NXP B.V. (Sep. 13, 2016). MPL3115A2 datasheet, [Online]. Available: http://cache.freescale.com/files/sensors/doc/data_sheet/MPL3115A2.pdf (visited on 05/12/2017).

[27] D. Landman. (May 13, 2017). Arduino AESlib, [Online]. Available: https://github.com/DavyLandman/AESLib (visited on 05/13/2017).

[28] V. Mennella. (May 2017). Crc16 a simple crc-16 library for Arduino, [Online]. Available: https://github.com/vinmenn/Crc16 (visited on 05/19/2017).

[29] T. Voigt, M. Bor, U. Roedig, and J. Alonso, "Mitigating Inter-network Interference in LoRa Networks", in *Proceedings of the 2017 International Conference on Embedded Wireless Systems and Networks*, Arxiv.org - Cornell University, Feb. 20, 2017.

[30] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia, and T. Watteyne, "Understanding the limits of LoRaWAN", Jul. 2016.

A

Appendix 1

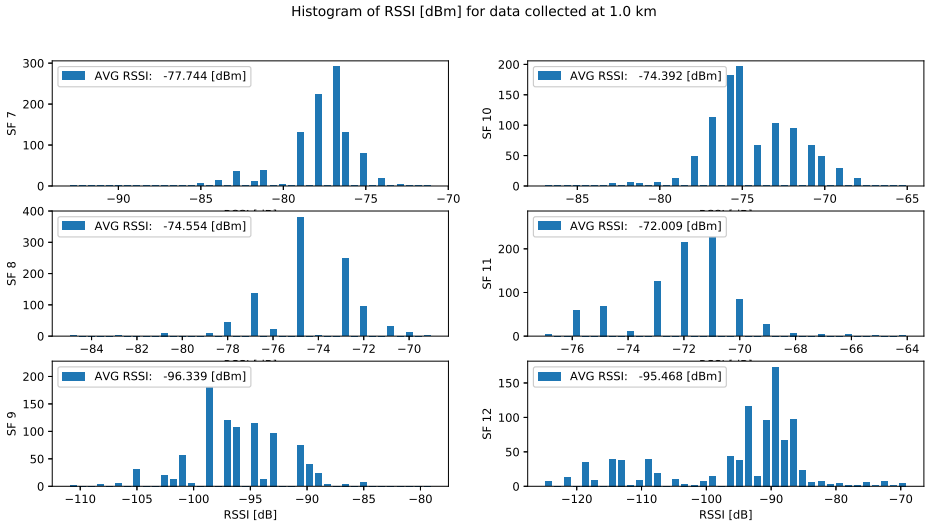


Figure A.1: Histogram of RSSI for data collected at 1.0 km.

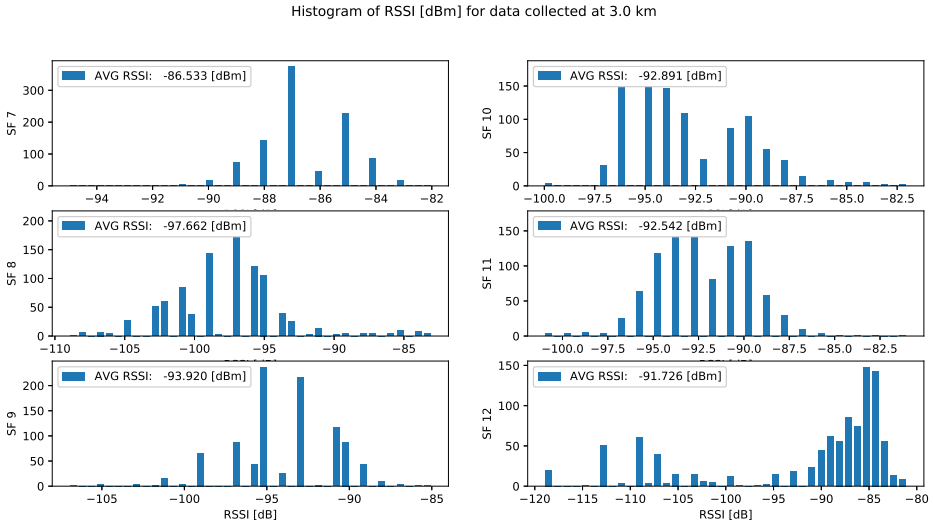


Figure A.2: Histogram of RSSI for data collected at 3.0 km.

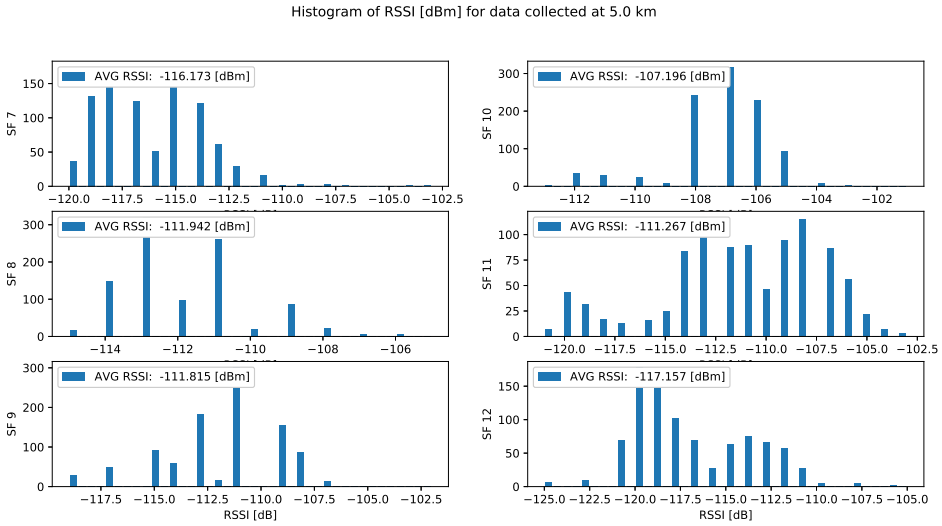


Figure A.3: Histogram of RSSI for data collected at 5.0 km.

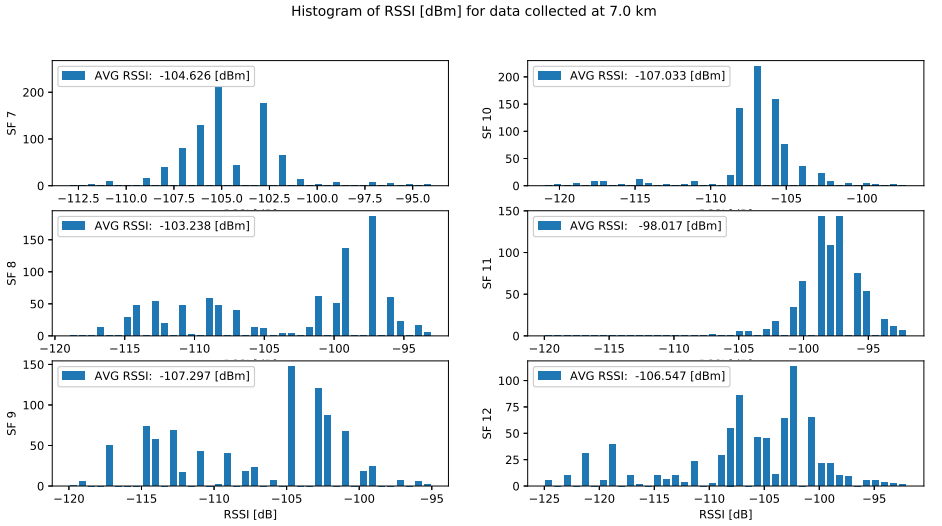


Figure A.4: Histogram of RSSI for data collected at 7.0 km.

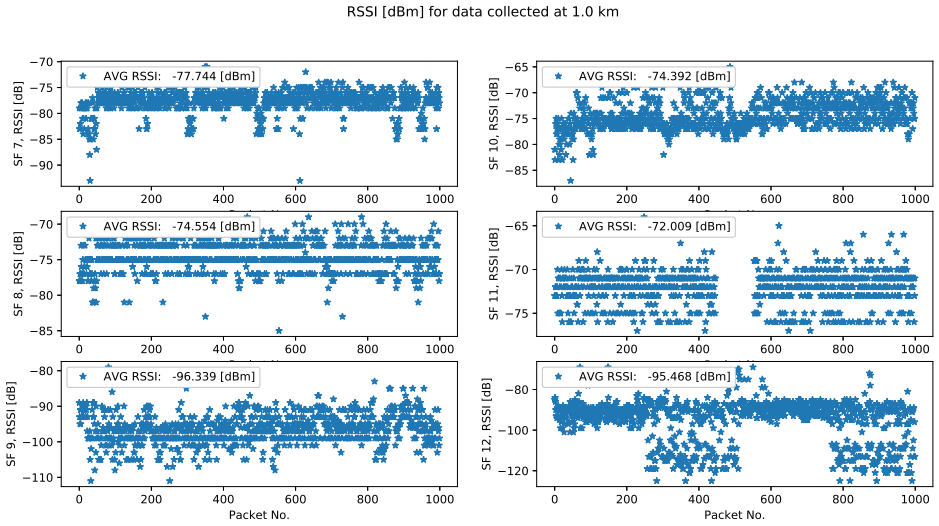


Figure A.5: RSSI for data collected at 1.0 km.

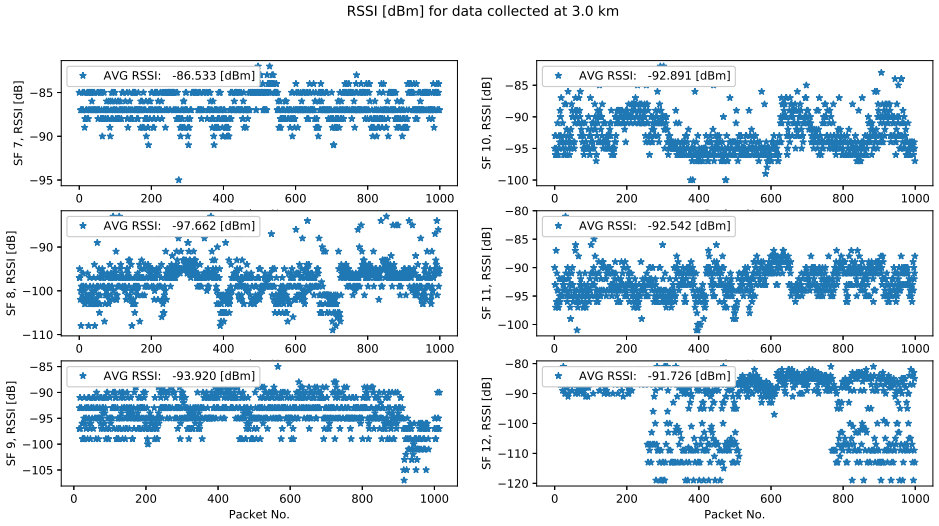


Figure A.6: RSSI for data collected at 3.0 km.

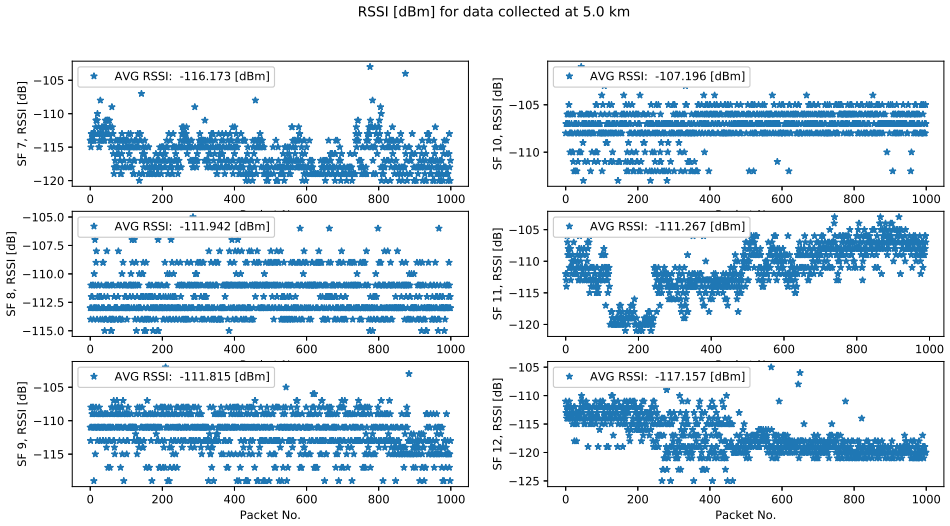


Figure A.7: RSSI for data collected at 5.0 km.

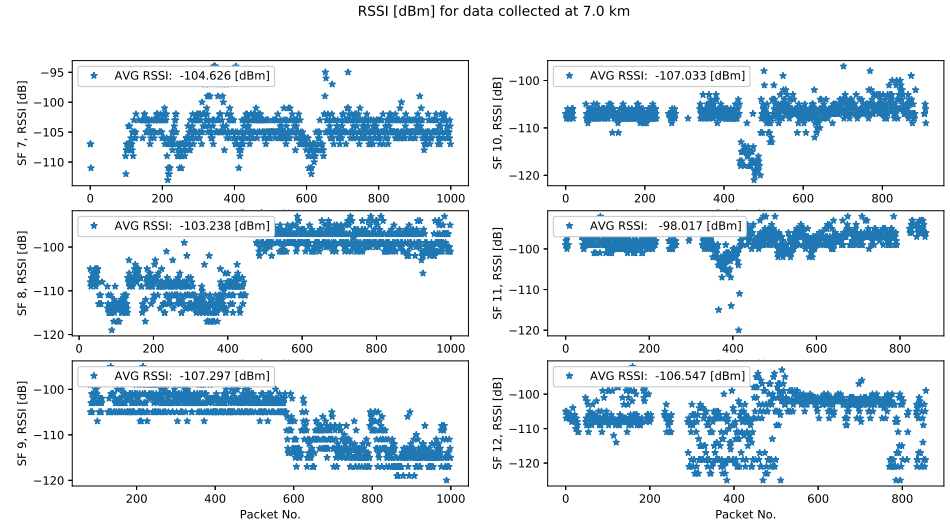


Figure A.8: RSSI for data collected at 7.0 km.

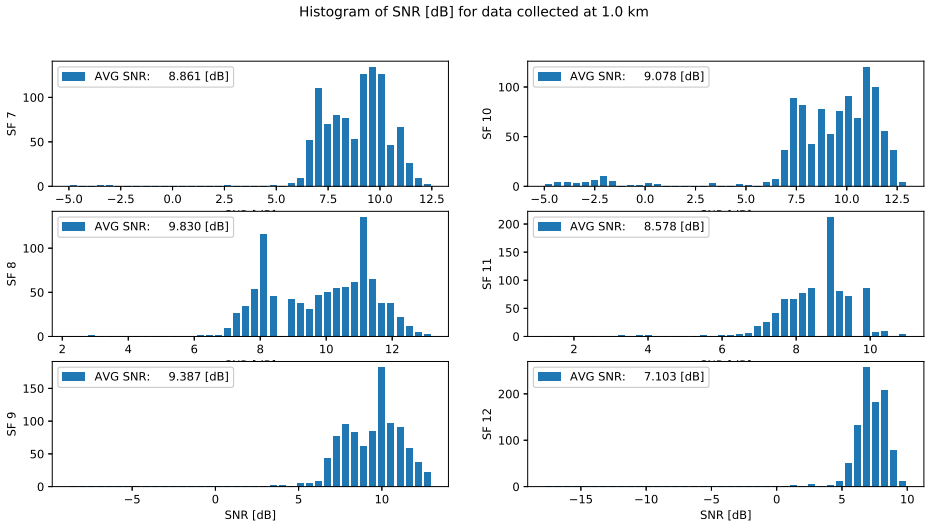


Figure A.9: Histogram of SNR for data collected at 1.0 km.

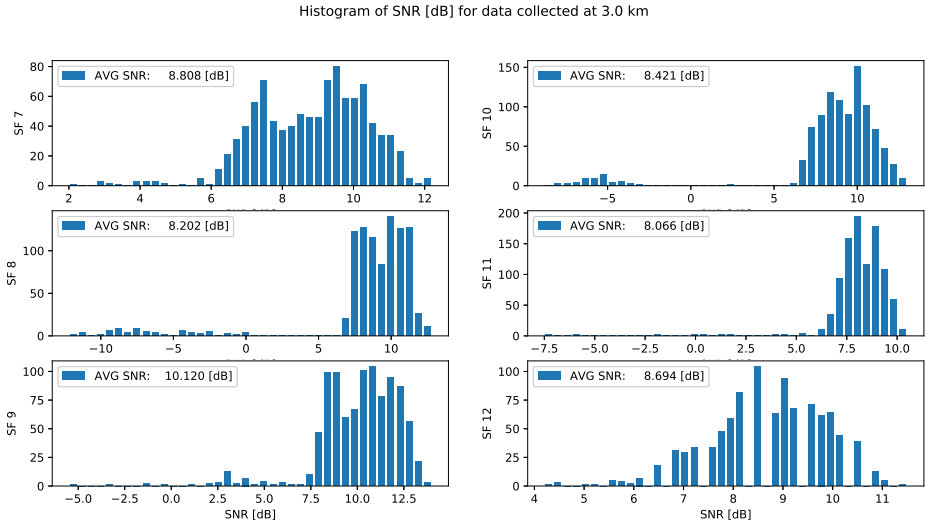


Figure A.10: Histogram of SNR for data collected at 3.0 km.

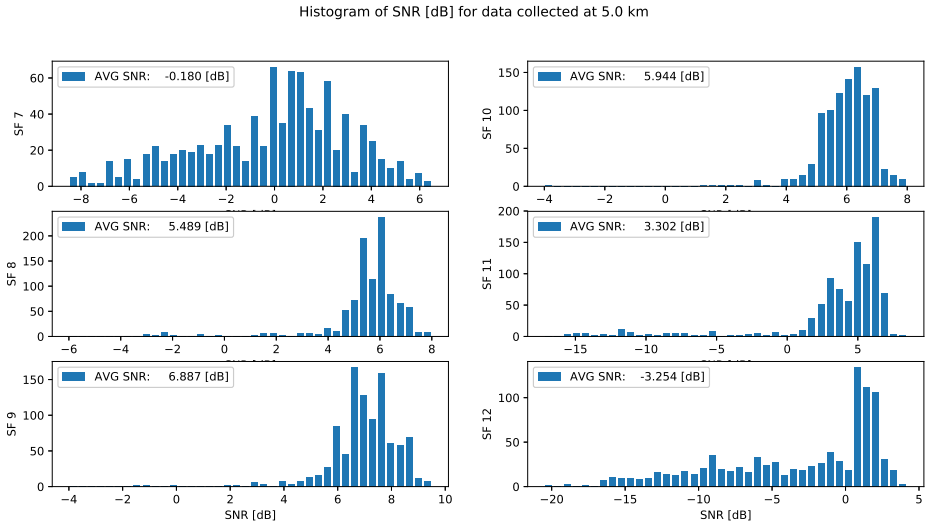


Figure A.11: Histogram of SNR for data collected at 5.0 km.

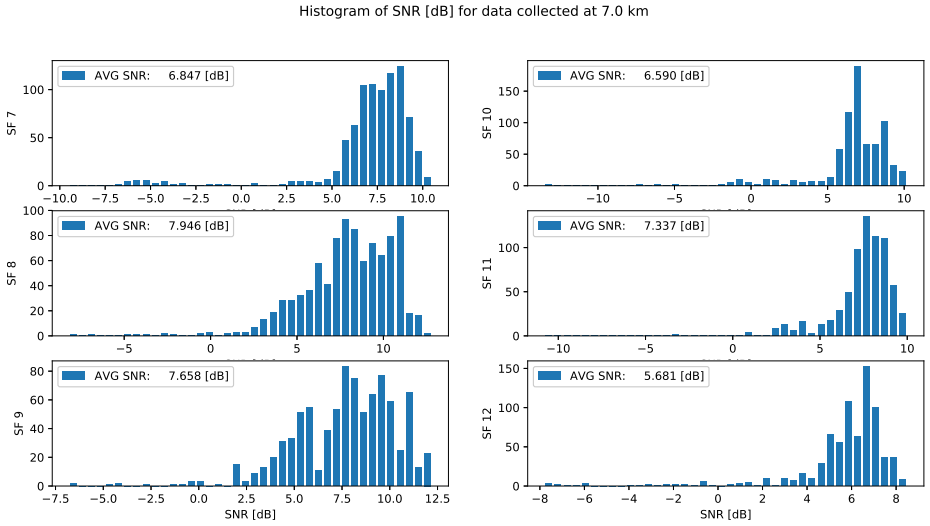


Figure A.12: Histogram of SNR for data collected at 7.0 km.

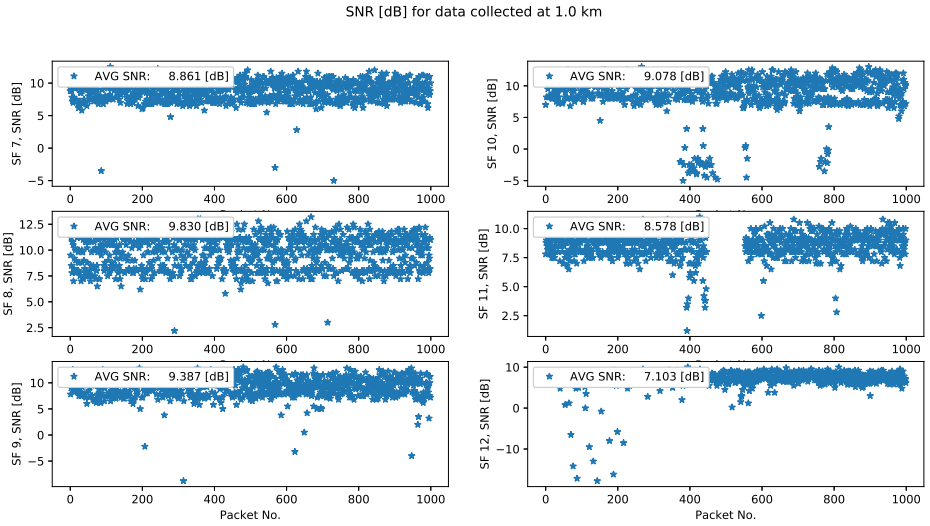


Figure A.13: SNR for data collected at 1.0 km.

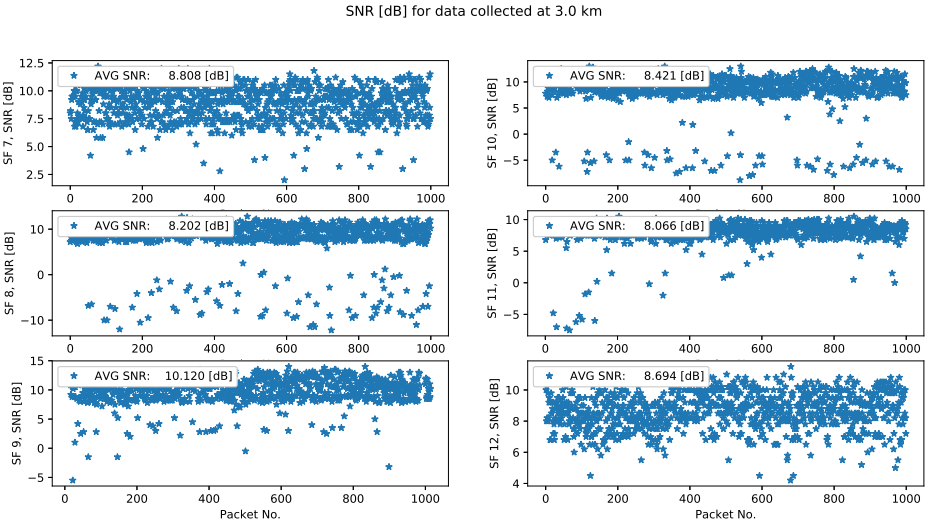


Figure A.14: SNR for data collected at 3.0 km.

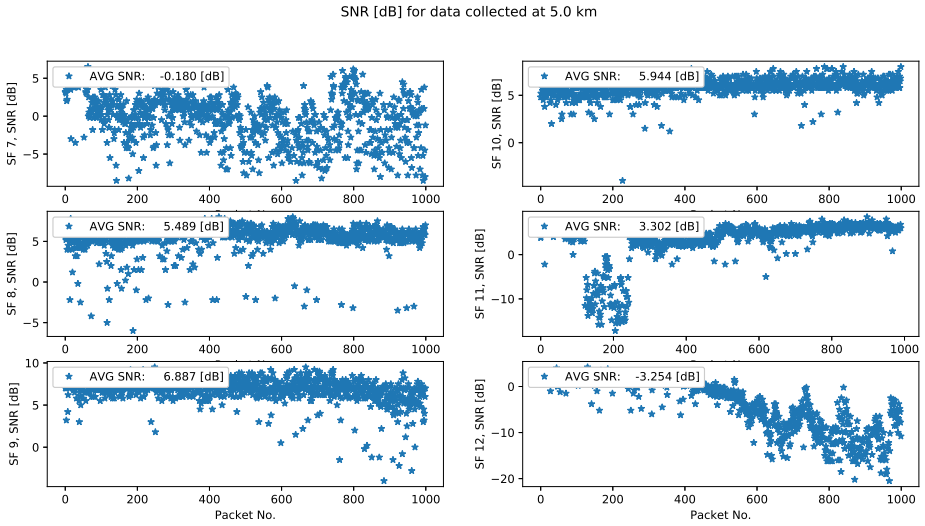


Figure A.15: SNR for data collected at 5.0 km.

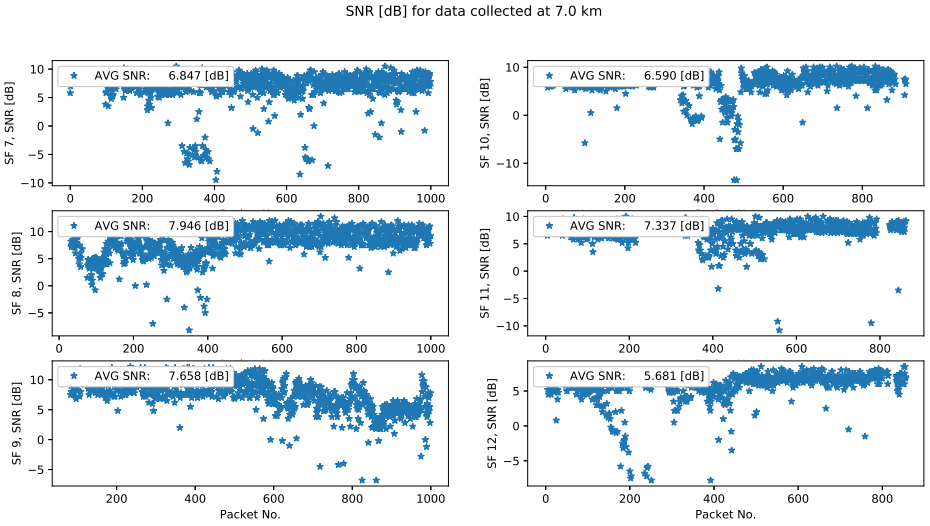


Figure A.16: SNR for data collected at 7.0 km.