Data Driven Networking

Modelling of interference probability using unsupervised learning methods

Master thesis
Asmus Bjerregaard Hansen
Group SPC-1075
SIGNAL PROCESSING
Aalborg University
4. June 2020

Aalborg University
Department of Electronic systems
Signal Processing
Fredrik Bajers vej 7
9220 Aalborg Øst
www.es.aau.dk

Title:

Data driven networking - Modelling of interference probability using unsupervised learning methods

Theme:

Signal Processing and Computing

Period of the project:

1st February - 4th June

Project group:

Group 1075

Members:

Asmus Bjerregaard Hansen

Supervisor:

Zheng-Hua Tan

Pages: 83
Appendix: A, B
Ended: 04-07-2020

Abstract:

In this thesis, a method for finding the parameters to model interference probability in a wireless channel, using the ALOHA model, is presented. For extraction of the model parameters, an algorithm which relies on a recording of interference in the wireless channel is proposed. The algorithm computes and segments the recording spectrograms to extract the parameters of individual transmission and clusters these to find the dominating interference sources in the channel. Using simulated data the parameter estimate errors are found in different congestion scenarios. For 2% congestion 92.41%, 57.57% and 15.31% of the interference transmissions is found with power levels -60dBm/Hz, -75dBm/Hz and -90dBm/Hz respectively. The fraction of mid and low powered transmissions found depends on the level of congestion in the channel. Finally, a test on real-world data quantifies the amount of interference that the algorithm is capable of extracting.

Preface

This report has been compiled by project group SPC-1075 as a Master thesis under the main theme *Signal Processing and Computing* at the Department of Electronic Systems at Aalborg University, spring 2020.

This paper is indexed in chapters chronologically numbered after the order in which they appear. Sections and subsections in chapters are numbered likewise, while sub-subsections are without index numbers. Figures, tables and equations are also indexed in numbers equivalent to the chapter and chronological order in which they appear, appendixes are lettered in alphabetical order in which they appear.

For the purpose of simulations conducted in this project, Python 3.7.4 has been used as the main simulation tool.

The Python scripts used for analysis of the spectrum recordings and a single spectrum recording are attached in a .zip file.

Asmus Bjerregaard Hansen
<ahanse18@student.aau.dk>

Contents

Introduction 1

1.1 Interference in wireless communication

When transmitting wireless messages, interference from other transmissions is always a risk. Interference can cause transmitted information to be lost and may require the information to be transmitted several times. Depending on the communication protocol, several similar transmissions can be sent to ensure that one of the transmissions is received or two-way communication with acknowledgement can be used to ensure that a transmission is successfully received and decoded. Either way, for battery driven applications this will cause higher energy consumption and lower expected life-time, given the same number of unique transmissions.

Several parts of the frequency spectrum are unlicensed where transmission does not require prior approval as long as the transmissions comply with standards set by the relevant authorities. A list of frequency spectrum allocation in Europe has been published by the ECC [1].

In recent years, Internet-of-Things(IoT) devices have provided devices with network access and, among other things, enabled automatic reading of utility meters without any human involvement. IoT applications makes extensive use of the unlicensed ISM 868-868.6MHz spectrum [2, 3] and as the number of IoT devices increases[4], the probability of interference in unlicensed bands are likely to increase as well.

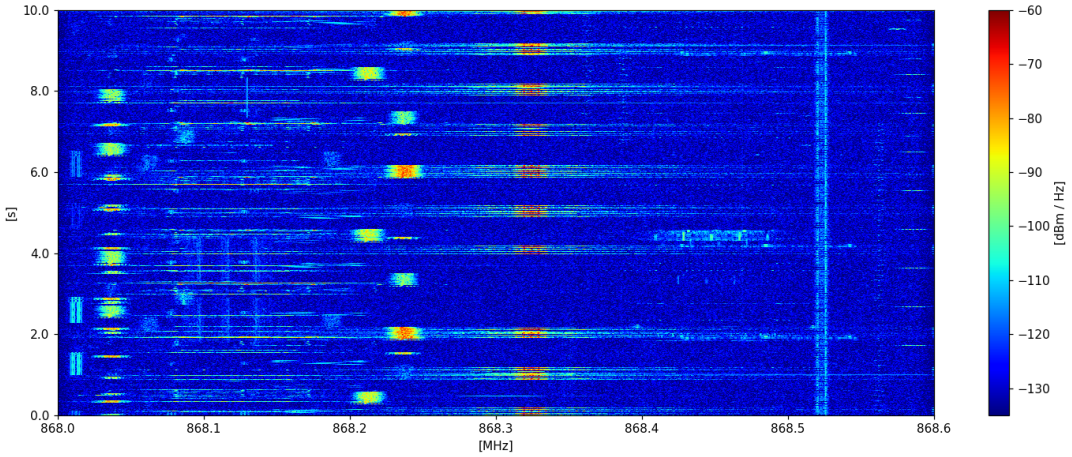

Figure 1.1: Spectrogram 868MHz to 868.6MHz.

In figure 1.1 a spectrogram of the 868.0-868.6MHz ISM band spectrum in downtown Aarhus is seen. The snapshot is a short time fourier transform(STFT) of the in-phase and quadrature signals captured

by a spectrum analyser sampling at 10MHz.

When a receiver decodes a transmission, it requires a certain signal to inference and noise ratio(SINR) and a higher level of interference will decrease the probability of successful decoding of a transmission. Some of the activity observed in the spectrum will cause interference and transmitting on frequencies with less load will increase the probability for achieving the required SINR. If given a spectrum recording, it would be beneficial to extract information about how interference sources are distributed, their bandwidth, center frequency, transmission time and duty cycle. With this information, the probability of interference for a transmission can be calculated and the channel with the least probability for interference can be chosen. The probability for interference can also be used in applications without two-way communication to determine the number of identical transmissions needed in order to obtain a sufficiently high probability of decoding success.

In this thesis, short range devices such a smart meters operating in the 868-868.6MHz ISM band are of special interest. The project is proposed by Kamstrup A/S, which is a manufacturer of smart meters for water, heating, cooling and electricity. To better understand how to efficiently transmit in unlicensed bands, an algorithm to model interference is needed. This model can be used to extract information which is used to control transmission parameters based on the observed environment. Having a transmitter adapt to its environment is termed cognitive radio [5] or adaptive protocols, which is an active field of research. The main challenge is to enable wireless transmitters to observe, adapt, reason and learn [6].

Developing algorithms to learn and reason from a given data set will be the main focus of this thesis, in order to enable a better understanding of the interference in an environment and design transmission protocols which best utilize the available spectrum. The adaption part of cognitive radio will not be considered however it is intended that the algorithms developed can eventually be used in commercial transmitters.

1.1.1 Machine learning in wireless communication

Intelligent learning algorithms can be applied to estimate the interference distributions, such as bandwidth, center frequency, transmission time and duty cycle. By finding similar transmission and clustering them into groups, a deeper understanding of the interference in a spectrum at a certain location can be obtained. Well-known machine learning algorithms can be used to both estimate the density of interference, and clustering similar transmission into groups. Since no prior information is available for interference in the spectrum, a natural choice for learning the interference distributions is using unsupervised learning algorithms. When similar transmissions are grouped, these can be extracted from the data set, and supervised classification algorithms can be used to identify modulation scheme and eventually identify which IoT technology is transmitting to further enhance the insight into spectrum interference. This however, will not be a part of the thesis.

In a similar project [7], it was found that supervised machine learning algorithms performs well when only one transmission is considered, but as soon as more than one transmissions are active at a given time(As an example, see figure 1.2), the classification accuracy decreases significantly. However, before it is possible to separate similar transmission from the rest it is necessary to learn how the interference is distributed in the spectrum.

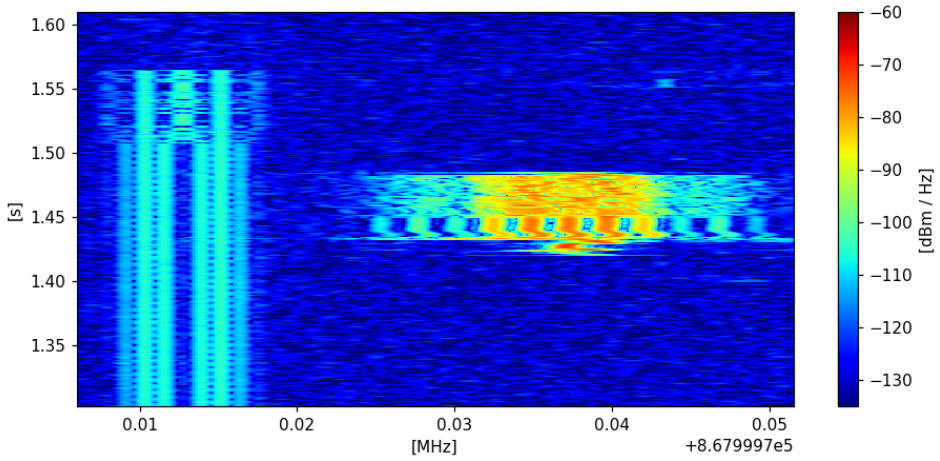

Figure 1.2: Simultanous transmissions in the spectrum

1.2 Problem statement

Based on the need to identify and model interference in a frequency spectrum, the following problem statement is formulated for this thesis.

- How can probability for interference be modelled and how can the model parameters be estimated?
- How can intelligent learning algorithms be used to find, separate and cluster similar transmissions in a spectrum recording to identify the parameters needed for modelling interference?

1.3 Delimitations

To make this thesis feasible, several assumptions and delimitations has to be made.

- It is assumed that the interference on the 868.0-868.6MHz ISM band is static such that the interference patterns does not change from day to day. However, this is a simplification since some interferers will transmit with the same pattern continuously and some may change. An example can be alarms that are triggered or devices which only transmit once every day or every week. As stated in section 1.1, more and more IoT devices are installed, which will also contribute to a change in the interference patterns. A justification for this assumption is that IoT devices are more stationary being based in buildings as alarms, meters etc. compared to mobile devices carried around by people.
- It is assumed that the time bandwidth product of transmissions is constant for a given symbol rate. Hence, the amount of information transmitted with transmission time $T = 2$s and bandwidth $BW = 1$kHz does not change if the parameters are changed to $T = 1$s and $BW = 2$kHz. The type of modulation, transmission protocol and many other factors may affect this, however this generalization is used when determining the optimal transmission parameters.

1.4 Prior work

In a study of the occupancy in the 868MHz ISM band in Erlangen and Nuremberg in Germany [8] the average and maximal power spectral density was measured to estimate the overall occupancy of the frequency bands. They concluded that overall the occupancy was 3% but that certain frequency sub-ranges had a far greater occupancy. They measured the energy for a 20kHz channel centered at 868.9MHz to estimate the inter-arrival times between transmissions and the length of the transmission to model the interference at a single channel. This supports the idea that some frequency ranges in a spectrum will be more suitable for transmission than others. In a related study, a model for interference is proposed [9]. Here, a frequency and time range termed ”playgrounds” is populated with transmission by a Poisson point process. A number of transmission classes with different bandwidth, transmission time and inter-arrival times, found as described in [8], was used to emulate various interference scenarios. These ”playgrounds” were used for simulation of interference to estimate the probability of transmission success. By finding the sources of interference in a spectrum, this method can be used to simulate real-world testing of wireless systems in a lab.

A similar approach is used in a Danish study were the interference is assumed to be distributed independently in both time and frequency [10]. Here, the probability for interference is calculated based on the interference exceeding a threshold. Inspired by [10], a German study [11] aims to find a more suitable model for the distribution of interference in the 868MHz ISM band. They compute the signal spectrogram of I/Q samples, and set an energy level threshold to detect transmission and calculate the bandwidth, transmission time, center frequency and inter-arrival times for the interference transmissions. By manually clustering the transmissions based on transmission parameters, they find the transmissions most likely to appear and estimate their bandwidth, transmission time, center frequency and inter-arrival times. Using these estimates, they calculate the probability for interference by using a method similar to the ALOHA model [12].

This thesis aims to continue were [11] finishes, finding intelligent thresholds for determining the presence or absence of an interference transmission. Furthermore, well-known machine learning algorithms will be used for clustering the interference transmissions, instead of manually clustering them, to estimate the transmission parameters of the interference transmissions most likely to appear.

Modelling interference probability

2

In this chapter, methods for computing the probability of interference are described and compared. A naive brute-force method, which can be used to compute transmission interference probability given a set of transmission parameters, is presented. However, this method does not help to model the interference in the spectrum. The probability is also modelled using the binomial distribution similar to the method used by researchers from Aalborg [10]. This method assumes that interference is distributed individually in time and frequency, which may not always be the case.

To model the spectrum interference without assuming that interference is distributed independent across time and frequency, the individual transmitters in the spectrum is assumed to use the ALOHA access protocol[12], where transmissions occur at random times. For this model, several parameters of the spectrum interference has to be estimated.

Given an I/Q data signal, a spectrogram as seen in figure 2.1 can be computed. Here, 3 simultaneous frequency modulated transmissions(2FSK) are simulated, each with a duty cycle of 0.5 and transmission time $T = 0.25s$. The transmissions are modulated at 1kHz, 10kHz and 20kHz.

Using this simulated data, the probability for interference can be computed using several methods. This simulated data set will be used as an example when discussing the different methods to estimate the interference probability. By using a simple simulated data set instead of a real data set, it is easier sanity check results.

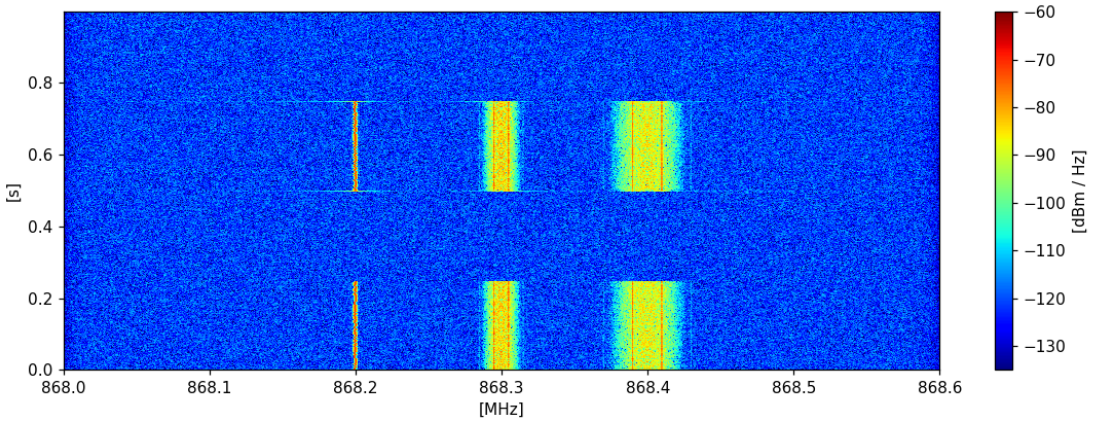

Figure 2.1: Spectrogram 868MHz to 868.6MHz with 3 simulated 2FSK signals.

The spectrogram is computed with a fixed Δf and Δt. One data point in the spectrogram covering

an area of $\Delta f \cdot \Delta t$ will be referred to as an interference unit. When the level of interference in an interference unit is above a threshold determined by the receivers required SINR[1], the interference unit is marked as being interfered. A transmission covers a certain number of interference units, depending on the bandwidth and transmission time. A transmission with $BW = 2\text{kHz}$ and $T = 2s$ will cover 4 interference units when $\Delta f = 1\text{kHz}$ and $\Delta t = 1s$. This definition of interference units is similar to the one found in [10].

In eq. (2.1) a formal definition of interference is seen. A unit is interfered when $r_m = 1$. Here, I_m is the power of interference unit m in dBm, P_R is the received power from a desired transmission in dBm and SINR is the required signal to noise plus interference ratio required for successful decoding of the transmission. In this section, an arbitrary threshold for adequate of $P_{SINR} + P_R = -105dBm$ similar to [13] is used. It is possible to combine the method described here with a distribution of received power if desired.

$$r_m = \begin{cases} 1 & I_m \geq P_R - P_{SINR} \\ 0 & else \end{cases} \tag{2.1}$$

When one interference unit in a transmission is marked interfered, the whole transmission is marked as interfered. It is unlikely that a transmission cannot be decoded when only one interference unit is marked interfered, especially if Δf and Δt are small relative to BW and T of the transmission. However, as soon as a desired transmission is overlapping an interference transmission it will be marked as interfered since it convenient for modelling. The models presented in section 2.3 and 2.4 both uses this as the criteria for interference .

[1]The required SINR varies with each receiver and finding the required SINR is not a part of this thesis.

2.1 Computing the spectrogram

In wireless communication, the spectral properties of a signal often does not match the spectral properties of the communication channel and a frequency translated version of the signal is transmitted. This is known as a band pass modulated version of the signal [14]. The base band signal is a low pass signal which contains the information transmitted using a wireless channel. An example of both is seen in figure 2.2.

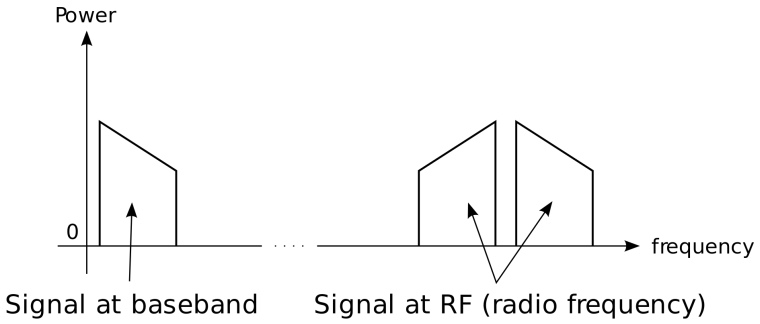

Figure 2.2: Base band and band pass signal

Given a base band signal as real-valued in-phase and imaginary quadrature components, a spectrogram can be constructed. When sampling the spectrum, a spectrum analyzer is used to shift the band pass signal to the base band representation. To represent the base band signal $x_l(t)$ as an oscillating wave with a given frequency and phase the in-phase and quadrature components are used(see eq. (2.3)). The notation can be factored to separate phase and frequency as seen in eq. (2.2).

$$x_l(t) = e^{j2\pi ft+\phi} = e^{j2\pi ft} \cdot e^{j\phi} \tag{2.2}$$

$$x_l(t) = x_i(t) + jx_q(t) \tag{2.3}$$

$$\tag{2.4}$$

When the baseband signal is quantized by the spectrum analyzer at a sampling interval T_s, it is expressed as $x_l[n] = x_l(n \cdot T_s)$

A spectrogram is a time-frequency representation of a time series, where the entire spectrum is computed at a fixed interval. The frequency spectrum is computed as a STFT such that for every N samples a new spectrum is computed.

The STFT for for the baseband signal $x_l[n]$ in the range $n = [0,N-1]$ can be computed as seen in eq. (2.5) [15], where the discrete Fourier transform is used. To compute a spectrogram, the STFT should be computed at fixed intervals over the quantized signal.

$$X[k] = \sum_{n=0}^{N-1} x_l[n]e^{-j\frac{2\pi}{N}kn} \tag{2.5}$$

The time resolution of the spectrogram depends how many samples from the quantized signal is used for each STFT. Given a signal length $K \cdot N$ samples, where $K \in \mathbb{R}$ and N is the number of samples

used for each STFT, the spectrogram will consist of K individual spectrums. The time resolution for the STFT is $\Delta T = T_s \cdot N$.

The frequency resolution is determined by eq. (2.6). Using zero-padding of the signal, the frequency resolution is increased such that more frequency bins are computed over the same range of frequencies.

$$\Delta f = \frac{N}{f_s} = N \cdot T_s \tag{2.6}$$

When computing the spectrogram, it is important to consider the characteristics of the baseband signal. If a signal contains small bursts with duration $T = 1ms$ then a time resolution of $\Delta T = 1s$ for the spectrogram will result in a poor time-frequency representation of the signal. Likewise, in a signal with several waves placed close in frequency, it may not be possible to distinguish the signals using a high Δf to compute the spectrogram. Hence, careful consideration of the signal characteristics is necessary when choosing Δf and ΔT. For every N samples used to compute the STFT, a window such as a Hanning window can be applied to lessen spectral leakage. The samples used to compute the STFT's may overlap, to make sure that the parts of the signal attenuated by the window, is present with close to 0dB gain in another window.

2.2 Brute force computation of interference probability

To find the probability for interference, given transmission bandwidth, center frequency and time, a simple brute force method can be applied. A window with fixed height(Time) and width(Frequency) segmented into interference units is slided over the computed spectrogram in steps of Δf and Δt. Each of the interference units in the transmission window is evaluated against the SINR threshold. When the window overlaps an interference transmission, it is said to be interfered. The average number of times the window is interfered at a given frequency is then evaluated across the spectrum.

The definition of interference probability is seen in eq. (2.7). If any of the m interference units is above the threshold, the transmission is interfered and the probability is set to 1.

$$P(I_m) = \begin{cases} 1 & r_m = 1 \;\; for \;\; any \;\; m \\ 0 & else \end{cases} \tag{2.7}$$

A window is slided across the spectrogram in steps of Δf and Δt which gives L seperate probabilities. $L = \frac{S_{BW} - \Delta f}{\Delta f} \cdot \frac{S_T - \Delta t}{\Delta t}$, where S_{BW} is the spectrogram bandwidth and S_T is the spectrogram time. The probability averaged over all time indexes, for every center frequency, is computed as seen in eq. (2.8) which yields the probability for intererence when transmitting at a given center frequncy. Here, I_{f_c} is the set of interference units which belongs to frequency bin f_c.

$$P_{avg}(f_c) = \frac{1}{S_T/\Delta t} \sum_{l=0}^{S_T/\Delta t} P(I_m) \quad I_m \in I_{f_c} \tag{2.8}$$

For the spectrogram generated from simulated data, as seen in figure 2.1, the computed brute force probability can be seen in figure 2.3. The transmission length is set to the lowest possible of $T = \Delta t = 1$ms to compute the probability for interference at any given time.

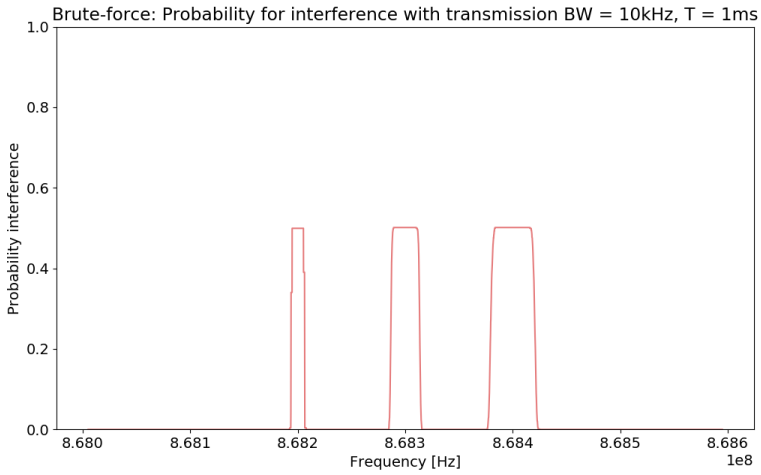

Figure 2.3: Interference probability 868MHz to 868.6MHz for a transmission with $BW = 10kHz$ and $T = 1ms$.

When transmitting in the vicinity of the three simulated transmissions in figure 2.1 it is expected that probability for interference will be 50%, since the duty cycle of the signals are 0.5. As seen in figure 2.3 this is also the value found by the brute for method.

The probability for at least one interference unit in the transmission to be interfered for all L points in the spectrogram with center frequency from $f_c = 868\text{MHz} + 10\text{kHz}/2$ to $f_c = 868.6\text{MHz} - 10\text{kHz}/2$ and transmission start time from $T_{start} = 0s$ to $T_{stop} = 1s - 0.001s$ is seen in figure 2.4.

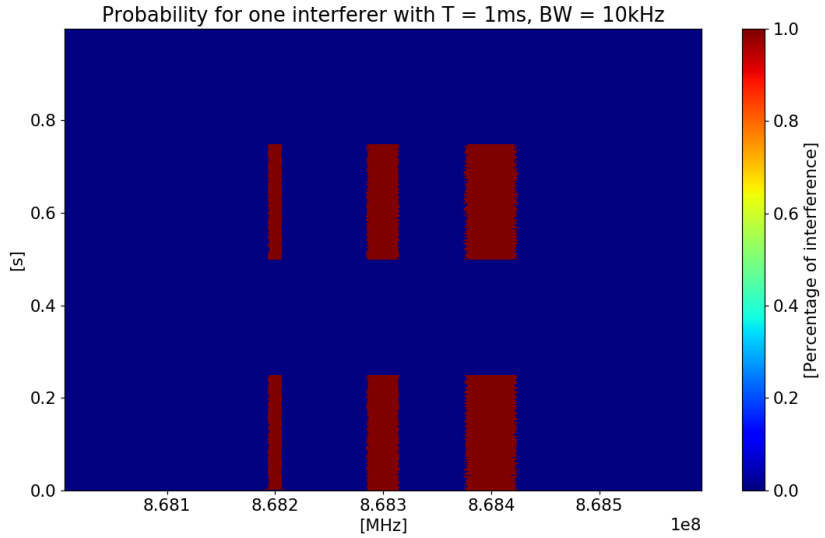

Figure 2.4: Probability for interference overlap for a transmission with $BW = 10kHz$ and $T = 1ms$.

In figure 2.5 the computed probability for a transmission with $T = 200$ms is seen. Here the probability for interference is 74.9%. As expected it is higher, since the inter-transmission time is 250ms which leaves a small margin for transmitting the 200ms long transmission.

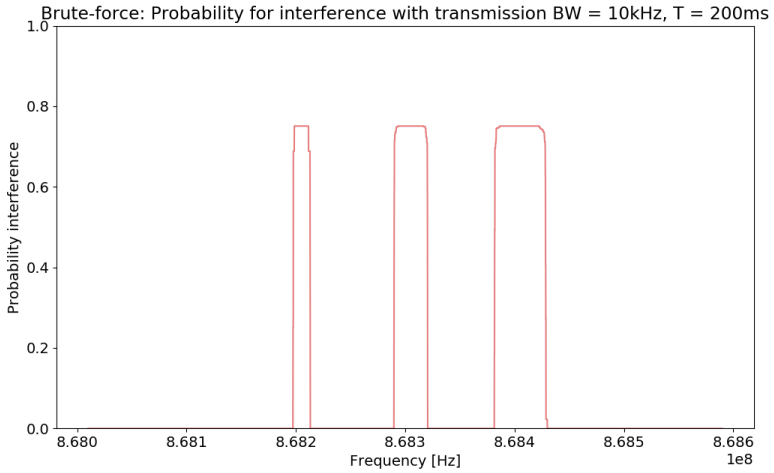

Figure 2.5: Interference probability 868MHz to 868.6MHz for a transmission with $BW = 10kHz$ and $T = 200ms$.

Given a fixed ratio $V = T \cdot BW$ of transmission parameters, the optimal transmission bandwidth can be found by running the simulation for a range of BW while keeping V constant. However, to compute the probability of interference at a single BW requires L windows to be evaluated which require evaluation of $L \cdot \frac{T}{\Delta t} \cdot \frac{BW}{\Delta f}$ interference units, for every single bandwidth. Given $\Delta t = 1ms$, $\Delta f = 100Hz$, $S_{BW} = 600kHz$, $S_T = 30s$, this amounts to more than $160 \cdot 10^9$ operations. For larger recordings, where several parameters are varied, this becomes infeasible.

2.3 Binomial computation of interference probability

If the interference is assumed to be independently distributed in time and frequency, the binomial distribution can be used to calculate the probability for interference as shown by researchers from Aalborg University[10]. The probability of one interference unit in the spectrogram to be above the interference threshold is calculated from the total spectrogram with L interference units. This is seen in eq. (2.9).

$$p_B = \frac{1}{L}\sum_{m=0}^{L-1} r_m \tag{2.9}$$

To calculate the probability that k interference units in a transmission covering N interference units are interfered, the Binomial distribution is used, see eq. (2.10).

$$P(I = k) = \binom{N}{k} \cdot p_B^k (1 - p_B)^{N-k} \tag{2.10}$$

When $k = 0$ no interference units are marked interfered, and the probability for one or more interference units to be interfered is $P(I > 0) = 1 - P(I = 0)$.

To calculate the probability that less than k interference units are interfered, the Binomial cumulative distribution is used and conversely, the probability that more than k interference units are interfered can be calculated as $P(I > k) = 1 - P(I \leq k)$.

$$P(I \leq k) = \sum_{h=0}^{k} \binom{N}{h} \cdot p_B^h (1 - p_B)^{N-h} \tag{2.11}$$

Eq. (2.9) is used to calculate the interference probability $p_B = 4.35\%$ from the spectrogram seen in figure 2.1. The probability for one or more interferers in a transmission with $BW = 10kHz$ and $T = 1ms$ can be calculated as seen in eq. (2.12), where the number of trials is $N = 100$ since $\frac{BW}{\Delta f} = \frac{10\text{kHz}}{100\text{Hz}} = 100$ and $T = \Delta t$.

$$P(I > 0) = 1 - \binom{100}{0} \cdot (0.0435)^0 (1 - 0.0435)^{100-0} = 98.83\% \tag{2.12}$$

This probability of interference is far from the value found in section 2.2, which is due to the fact that the interference is not distributed independently in time and frequency as the Binomial model assumes. This is shown by generating a spectrogram with interference units having $P = 4.35\%$ chance being above the required threshold and computing the probability for interference using the brute force method from section 2.2. The experiment yields a 98.82% chance for interference with transmission parameters $BW = 10kHz$ and $T = 1ms$ similar to the probability computed with the Binomial method. These findings highlights the need for a better model for the interference distribution.

2.4 Pure ALOHA model for interference

In a given frequency range $F = [f_c - BW/2, f_c + BW/2]$ the average number of transmissions is observed every millisecond. By assuming pure ALOHA spectrum access for all interference source, where transmissions arrive independently in time with the constant rate $\lambda = r \cdot t$ transmissions/millisecond, the probability that K transmissions occurs in t seconds can be modelled as an Poisson distribution[12], see eq. (2.13). Here r is the observed rate of events, with unit transmissions/ millisecond and t is the time interval [16].

$$P(K = k|F) = \frac{(r \cdot t)^k e^{-r \cdot t}}{k!} \tag{2.13}$$

Let T_I be the random variable modelling the time between now and the next interference transmission, if $T_I > t$ there will be no transmissions before time t which is equal to $K = 0$, and as seen in eq. (2.14) the probability that no event occur until time t follows an exponential distribution. Hence, the time between events in a Poisson distribution is exponentially distributed.

$$P(T_I > t|F) = P(K = 0|F) = \frac{(r \cdot t)^0 e^{-r \cdot t}}{0!} = e^{-r \cdot t} \tag{2.14}$$

To get the probability that an event occur before time t, the complement is used as seen in eq. (2.15).

$$P(T_I \leq t|F) = 1 - P(T_I > t|F) = 1 - e^{-r \cdot t} \tag{2.15}$$

The time interval t has to account for both the length of the desired transmission and the average length of interference transmissions in the channel. Given a spectrum with interference source $TX1$ which has transmission length T_1 and a desired transmission $TX2$ with length T_2, the time interval is $t = T_P = T_1 + T_2$ to ensure that no interference transmission is active at transmission start and that no interference transmission occur while transmitting. This is illustrated in figure 2.6

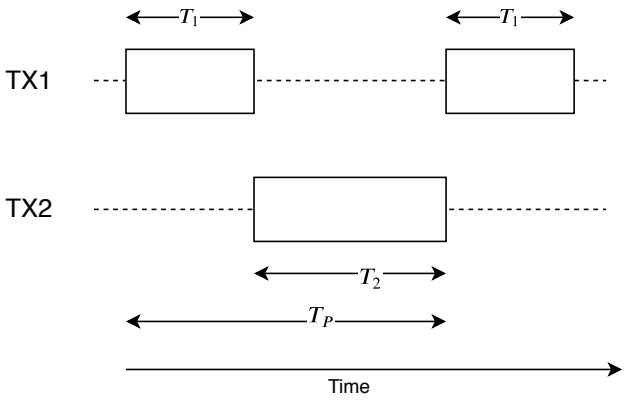

Figure 2.6: Poisson transmission time parameter.

By assuming pure ALOHA access in a channel, the probability for interference can be calculated for different transmission parameters center freqnecy(f_c), transmission time(T) and bandwidth BW.

As an example of this, consider the spectrogram in figure 2.7 where three 2FSK signals modulated at 10kHz is simulated spaced apart by 20kHz. Here transmitting with center frequency $f_c = 868.3$MHz is constrained by transmissions to each side of the center frequency. By inspecting the spectrogram, an estimate would be that the optimal transmission parameters is a bandwidth low enough to fit between the two interferers at 868.26MHz and 868.32MHz, and a transmission time short enough to avoid the middle interferer at 868.3MHz.

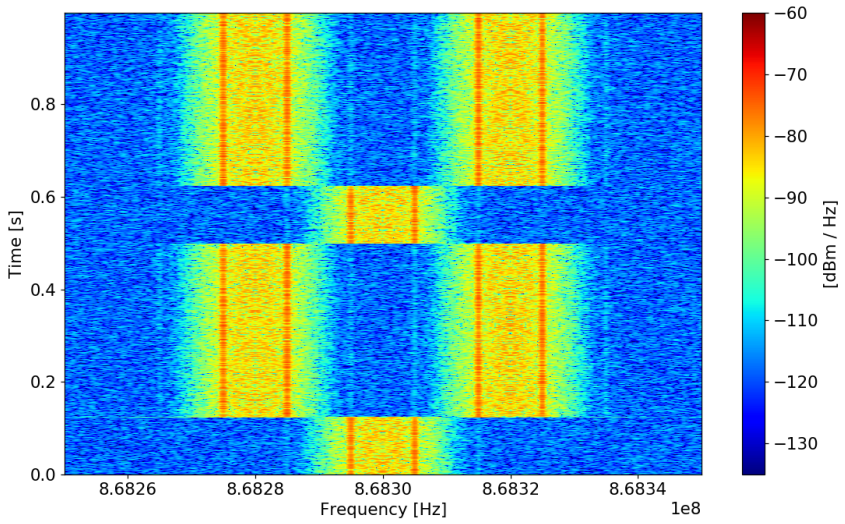

Figure 2.7: Spectrogram 868MHz to 868.6MHz with 3 simulated alternating 2FSK signals.

With a ratio between bandwidth and transmission time equal to $2500 = BW \cdot T$ the following three bandwidths are considered, $BW = 50$kHz, $BW = 10$kHz and $BW = 5$kHz.

Below the transmission rates and the required time interval T_p is calculated for each frequency range.

- Transmission 1 - $f_c = 868.3$MHz and $BW = 50kHz$: In this range 6 transmissions occur, hence $r = 6$ transmissions/second. Four transmissions with length 0.4s and two with length 0.1s yields an average length of 0.3s. This yields a time interval of $T_p = 0.3\text{s} + 0.05\text{s} = 0.35\text{s}$.
- Transmission 2 - $f_c = 868.3$MHz and $BW = 10kHz$: In this range 2 transmissions occur, hence $r = 2$ transmissions/second. Two transmissions with length 0.1s occur. This yields a time interval of $T_p = 0.3\text{s} + 0.025\text{s} = 0.55\text{s}$.
- Transmission 3 - $f_c = 868.3$MHz and $BW = 5kHz$: In this range 2 transmissions occur, hence $r = 2$ transmissions/second. Two transmissions with length 0.1s occur. This yields a time interval of $T_p = 0.3\text{s} + 0.5\text{s} = 0.8\text{s}$.

The probability for interference $P(T_I \leq t)$ is seen in table 2.1.

Interference probability	$T = 0.05$s, $BW = 50$kHz	$T = 0.25s$, $BW = 10$kHz	$T = 0.5$s, $BW = 5$kHz
$P(T_I \leq t)$	87.75%	66.71%	79.81%

Table 2.1: Interference probabilities

The results in table 2.1 show that the optimal transmission parameters are $BW = 10\text{kHz}$ and $T = 0.25s$.

Using the brute force method to evaluate the transmission parameters reveals that the optimal parameters are indeed $BW = 10\text{kHz}$ and $T = 0.25s$ as seen in table 2.2. With a time between interferers of $T_I = 0.325s$ and $\approx 10\text{kHz}$ free bandwidth between the side interferers, this is expected to be the optimum among the three options. Inspecting figure 2.7 it is clear that the two other transmissions will always overlap another transmission, hence they will always be marked as interferred.

$f_c = 868.3\text{MHz}$	$T = 0.05\text{s}$, $BW = 50\text{kHz}$	$T = 0.25s$, $BW = 10\text{kHz}$	$T = 0.5\text{s}$, $BW = 5\text{kHz}$
Brute-force	100%	53.35%	100%

Table 2.2: Interference probabilities - Brute force

To use ALOHA for modelling interference, it is important to estimate the average transmission rate and transmission length in a frequency range. This requires an algorithm to extract individual transmissions in the spectrum, group similar transmissions and estimate their bandwidth, center frequency and transmission time.

Identifying interference transmissions 3

To use the ALOHA inderference model discussed in section 2.4, the interference sources in a spectrum have to be found and their transmission parameters estimated. In order to estimate the source parameters, it is necessary to first detect interference transmissions in the spectrum. When the transmissions has been detected, the center frequency, transmission time and transmission bandwidth can be estimated. In this chapter, a signal model for the interference transmissions is presented to gain a better understanding of the signal statistics which is used to detect the transmissions. Furthermore, an algorithm for separating the interference transmission from background noise is proposed. Using this algorithm it is possible to estimate the transmission parameters seen in figure 3.1; center frequency(f_c), transmission time(T) and transmission bandwidth(BW).

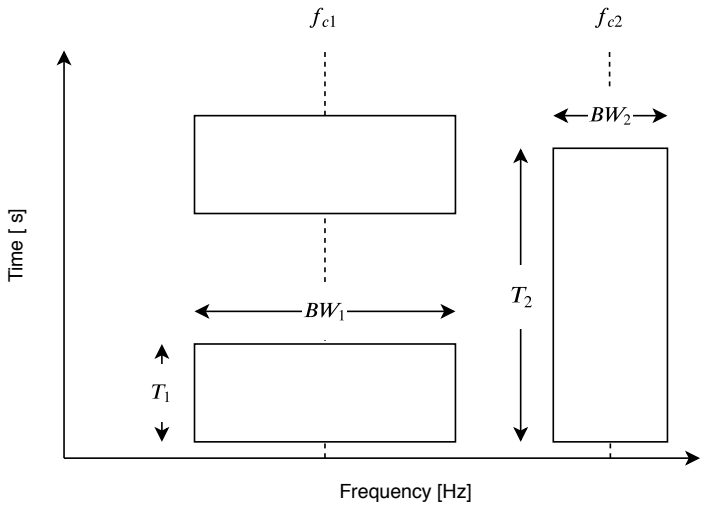

Figure 3.1: Transmission parameters.

3.1 Signal model

Consider the baseband representation of a signal with a given frequency and phase, seen in equation (3.1).

$$x_l(t) = e^{j2\pi ft+\phi} \tag{3.1}$$

This signal is transmitted through a wireless channel with additive white Gaussian noise(AWGN) and a complex channel gain α [12], no attenuation, fading or multi-path propagation is considered. The received signal will be the sum of the transmitted signal and the Gaussian noise as seen in eq. (3.2) where the Gaussian noise is limited by the bandwidth of the receiver and denoted $n_l(t)$.

$$r(t) = \alpha x_l(t) + n_l(t) = \alpha e^{j2\pi ft+\phi} + n_l(t) \tag{3.2}$$

If no interferers are present in the spectrum, the received signal will only consist of the noise component in eq. (3.2), $r(t) = n_l(t)$. The noise is modelled as a random variable N_l and is assumed to be distributed as a circularly-symmetric complex Gaussian such that $N_l \sim \mathcal{CN}(\mu, \sigma^2)$ with the real and imaginary part being independently distributed.

When an interfering transmitter transmits a signal, the received power can be calculated as seen in eq. (3.3). If no signal is transmitted the received power can be expressed as in eq. (3.4).

$$P_s(t) = |r(t)|^2 = |x_l(t) + n_l(t)|^2 \tag{3.3}$$
$$P_n(t) = |n_l(t)|^2 \tag{3.4}$$

The magnitude of the complex Gaussian N_l will be Rayleigh distributed [14], $P_n(t) \sim Rayleigh(\sigma)$. The probability density function of the Rayleigh distribution is found in eq. 3.5.

$$f_{Rayleigh}(x|\sigma) = \frac{x}{\sigma^2} \cdot e^{-x^2/2\sigma^2} \qquad x,\, \sigma > 0. \tag{3.5}$$

When a signal is transmitted(eq. (3.3)) the non-zero magnitude will cause the circularly-symmetric complex Gaussian to have a non-zero mean, hence the signal will follow a Rice distribution $P_s(t) \sim Rice(v, \sigma)$, see eq. 3.6, which, as the magnitude increases, will approximate a Gaussian distribution [12]. In eq. 3.6, $I_0(z)$ is a modified Bessel function of the first kind with order zero.

$$f_{Rice}(x|v, \sigma) = \frac{x}{\sigma^2} \cdot e^{-(x^2+v^2)/2\sigma^2} \cdot I_0\left(\frac{xv}{\sigma^2}\right) \qquad x, v, \sigma > 0. \tag{3.6}$$

Recalling the signal recording shown in figure 1.1, a histogram is computed for the first 100ms of the spectrogram to estimate the received power distribution. The estimated distribution can be seen in figure 3.2. Since the majority of the values in a spectrogram will be caused by noise, the histogram will be heavily biased towards the low values if it is calculated in the linear domain. For calculating the histogram, the power values on a logaritmic scale is used as seen in figure 3.2.

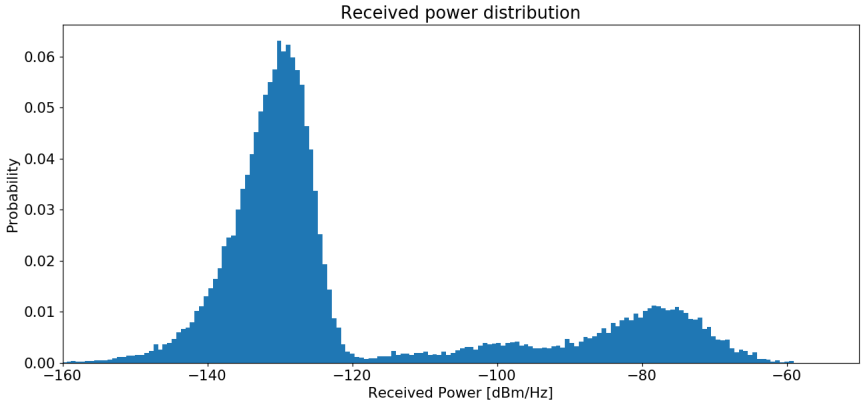

Figure 3.2: Received power distribution based on spectrum recording seen in figure 1.1.

Figure 3.2 shows a distribution which could be composed of a Rayleigh distribution(On a logarithmic scale) and several Rice distributions. It is assumed that the lowest cluster of power values belongs to the circularly-symmetric complex Gaussian noise.

3.2 Clustering of received power based on clustering algorithms

To separate the interference transmissions and noise, the received power transmissions are clustered. The signal histogram in 3.2 shows that the received power is found in varying densities, as expected when considering the signal model in section 3.1. The different interference sources will likely form their own cluster of received power levels, hence, clustering should be able to find a cluster for each interference source with an unique transmission power level.

In this section, two clustering algorithms for classifying noise and interference transmissions in spectrum snapshots are presented and tested. The tests will aim to verify that the clustering algorithms can segment signal chunks into clusters as would be expected when considering the histogram of the signal.

3.2.1 K-Means clustering

K-Means clustering is a common clustering algorithm in unsupervised machine learning. Compared to other clustering algorithms it is simple to implement and computationally inexpensive.

K-Means clustering algorithm

Given a group of samples $\{x_1,x_2..,x_N\}$ with N observations in M dimensions it is possible to separate the samples into K groups. The similarity of the samples are measured as their euclidean distance to a common reference point, the cluster center, and each sample is then assigned to one of K clusters. Given that the number of clusters K in the data set is known, the clusters should be placed such that the sum of squares from each sample to its assigned cluster center μ_k is minimized [17].

The sample x_n has a set of corresponding binary indicators which describes the samples association with a cluster k, $r_{nk} \in \{0,1\}$, where $k = 1,..,K$ are the index of each of the K clusters. If sample x_n

is assigned to cluster k, then $r_{nk} = 1$ otherwise it is 0. The objective function to be minimized is seen in eq. (3.7)

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||x_n - \mu_k||^2 \tag{3.7}$$

Eq. (3.7) should be minimized with respect to r_{nk} and μ_k which is done by iterations in a two step algorithm.

1. Choose random means μ_k to represent the cluster centers
2. Assign each sample x_n to the cluster k which minimizes $||x_n - \mu_k||^2$.
3. For every k, compute a new cluster center μ_k by using eq. (3.8).
4. If the cluster centers μ_k are moved more than some constant ϵ go to step 2, otherwise terminate.

$$\mu_k = \frac{1}{N} \sum_{n=1}^{N} r_{nk} x_n \tag{3.8}$$

The algorithm will converge to some minimum which may not be the global minimum depending on the starting position of the means [17].

K-Means clustering algorithm test

To test the K-Means algorithm for segmentation of received power levels, a simulated signal with three 2FSK modulated signals is used. This is a short version of the signal used in section 2.4. The spectrogram of the signal is seen in figure 3.3.

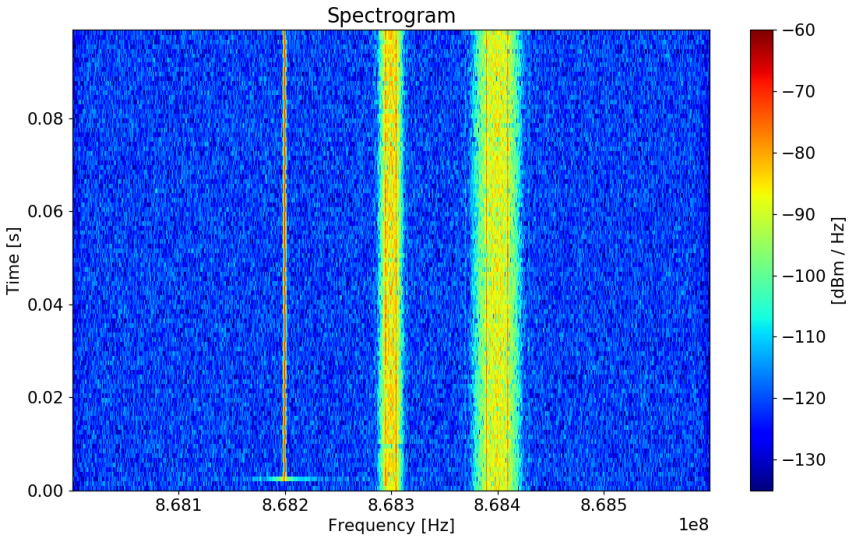

Figure 3.3: Spectrogram 868MHz to 868.6MHz with 3 simulated 2FSK signals.

The power distribution of all the interference units in figure 3.3 is seen in figure 3.4. For this test the number of clusters is $K = 2$ since the distribution seems to have two main clusters at $\approx -120dBm$ and $\approx -90dBm$.

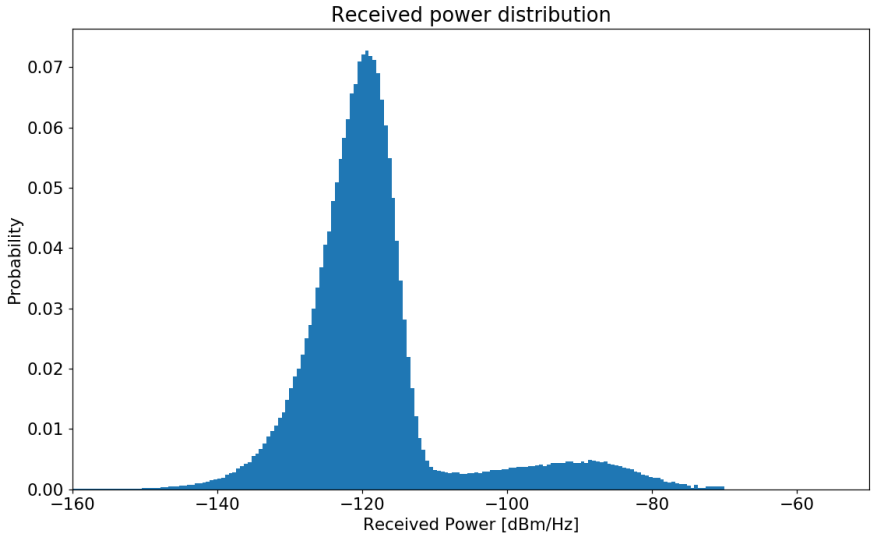

Figure 3.4: Spectrogram power distribution.

After running the K-Means clustering algorithm on the interference units, the cluster centers seen in table 3.1 are found. The first cluster represents the large amount of samples around -120[dBm/Hz] and the last cluster represents the seconds of the two tops seen in figure 3.4.

	Cluster 1	Cluster 2
Cluster center	-121.1 [dBm/ Hz]	-89.9 [dBm/ Hz]

Table 3.1: Cluster centers

By clustering the interference units according to those two clusters, the noise is clearly separated from the signal, as seen in figure 3.5.

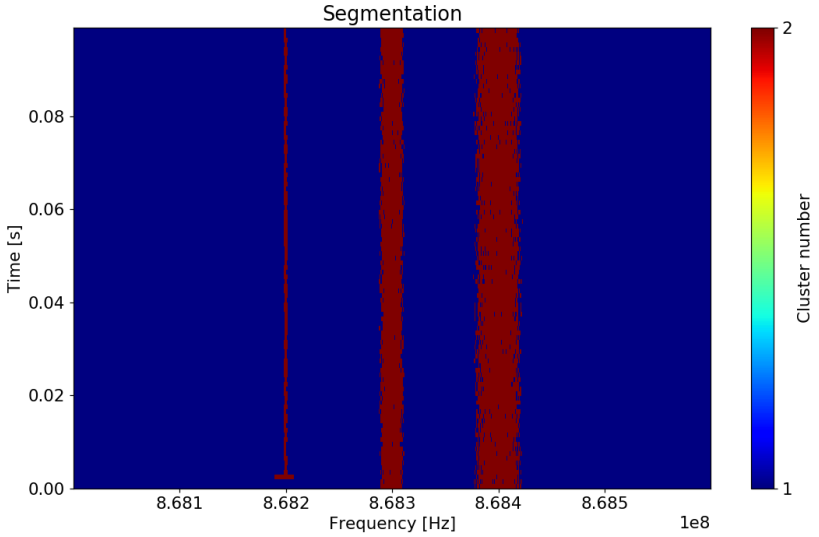

Figure 3.5: Spectrogram segmented into clusters.

A similar test is performed on the signal sampled in downtown Aarhus. In figure 3.6 the spectrogram for the first 100ms in the frequency range 868MHz to 868.04MHz is seen.

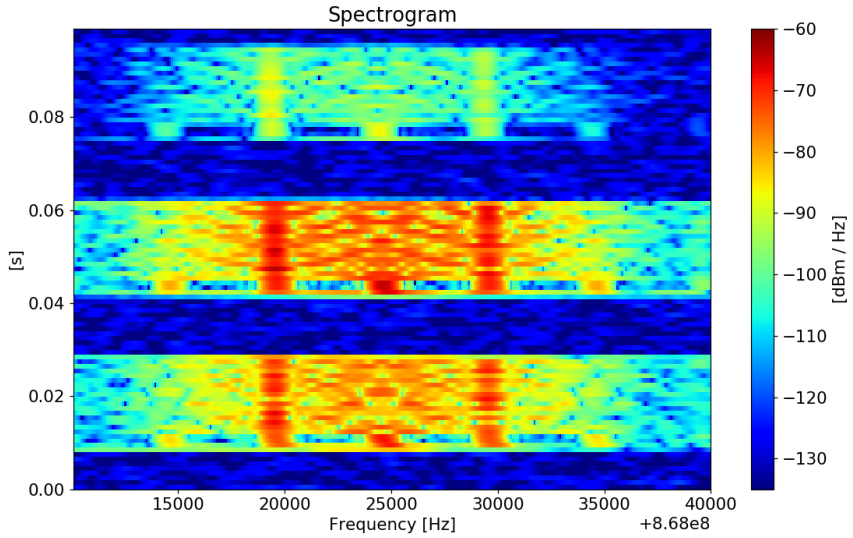

Figure 3.6: Aarhus data set spectrogram 100ms.

The distribution of the spectrogram is found in figure 3.7. Here one distinct cluster is seen at ≈ -130[dBm/Hz] and one or two is seen at ≈ -90[dBm/Hz]. For this test the number of clusters is set to $K = 3$ to see if the clustering algorithm is able to separate the noise from the different levels of received signal.

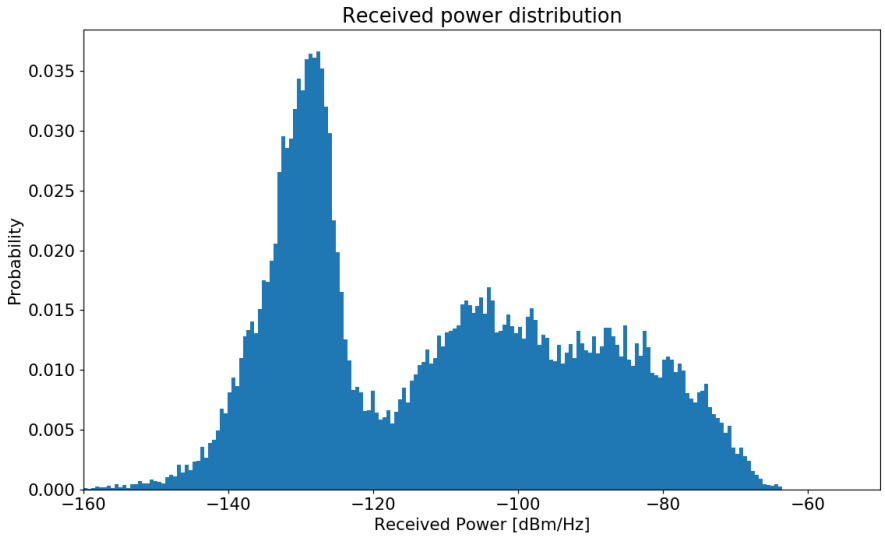

Figure 3.7: Aarhus data set spectrogram segmented into clusters.

The resulting clusters are seen in table 3.2 and, as expected, a cluster is found at ≈ -130[dBm/Hz] representing the noise and two clusters are found at ≈ -105.3[dBm/Hz] and ≈ -83.2[dBm/Hz] which represents the two other less distinct peaks.

Cluster	Cluster 1	Cluster 2	Cluster 3
Cluster center	-130.5 [dBm/ Hz]	-105.3 [dBm/ Hz]	-83.2 [dBm/ Hz]

Table 3.2: Cluster centers K-Means

In figure 3.8 the segmented interference units are seen. Inspecting figure 3.6 reveals that the two first transmissions has a higher power than the last which is also seen in the segmentation.

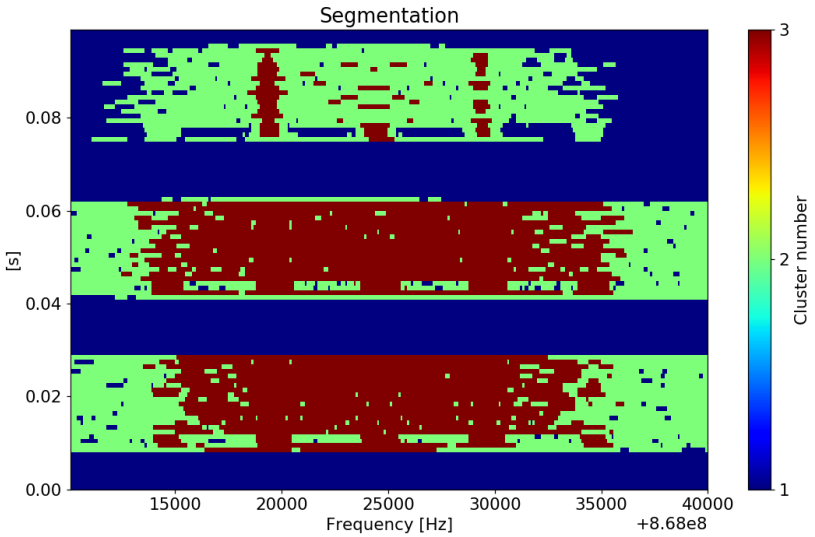

Figure 3.8: Aarhus data set spectrogram segmented into clusters.

Inspecting figure 3.7, it can be argued that the three clusters in figure 3.7 would be better described by Gaussian's than by simple means, hence a test with mixed Gaussian models i conducted.

3.2.2 Gaussian Clustering

Using a linear superposition of Gaussian's to represent the distribution may yield a better result than using K-Means. This method is called Gaussian Mixture models(GMM). The algorithm used to fit the Gaussian's to the data will be presented and a number of small tests are performed to verify that GMM is a valid clustering method for this problem.

GMM clustering algorithm

As with K-Means, a group of samples $\mathbf{x} = \{x_1,x_2..,x_N\}$ with N observation in M dimensions is given. The distribution of the data set is modelled as seen in eq. (3.9)[17].

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k,\Sigma_k) \tag{3.9}$$

A K-dimensional binary random variable Z where the k'th element z_k is one and all other is 0 is used to model the scaling factor π_k. The vector has K possible states, and the probability that it takes the value $p(z_k = 1)$ is the prior probability assigned to each Gaussian, such that $p(z_k = 1) = \pi_k$.

Each Gaussian is a conditional probability, conditioned on z_k, hence the probability that each member of $\mathbf{x}$ belongs to cluster k is seen in eq (3.10).

$$p(\mathbf{x}|z_k = 1) = \mathcal{N}(\mathbf{x}|\mu_k,\Sigma_k) \tag{3.10}$$

Multiplying $p(\mathbf{z})$ by the conditional probability yields the joint probability $p(\mathbf{x}, \mathbf{z})$. Summing over all possible values of $\mathbf{z}$ gives marginal probability $p(\mathbf{x})$ as seen in eq. (3.11)

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{z}) \cdot p(\mathbf{x}, \mathbf{z}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\mu_k,\Sigma_k) \tag{3.11}$$

To find the best representation of the given data $\mathbf{x}$ with a predefined number of clusters k, the log likelihood seen in eq. (3.12) has to be maximized.

$$ln\, p(\mathbf{x}|\pi, \mu, \mathbf{\Sigma}) = \sum_{k=1}^{K} ln\Big(\pi_k \mathcal{N}(\mathbf{x}|\mu_k,\Sigma_k\Big) \tag{3.12}$$

Maximizing the log likelihood function utilizes the Expectation-Maximization algorithm(EM). This algortihm is outlined in the following steps.

1. Initialize random means μ_k, covariances Σ_k and scaling coeffecients π_k.
2. Given the parameters π μ $\mathbf{\Sigma}$, compute the log likelihood $ln\, p(\mathbf{x}|\pi, \mu, \mathbf{\Sigma})$.
3. Assign each sample to the cluster which are the most likely to have generated it. If sample n is assigned to cluster k, the binary indicator variable r_{nk} will take on the value 1 otherwise it will be 0.
4. For every sample assigned to a cluster compute new values for μ_k, Σ_k and π_k using eq. (3.13), (3.14) and (3.15) respectively.
5. If the parameters μ_k, Σ_k and π_k change less than some constant ϵ terminate, otherwise go to step 2.

$$\mu_k = \frac{1}{N}\sum_{n=1}^{N} r_{nk}\mathbf{x_n} \tag{3.13}$$

$$\Sigma_k = \frac{1}{N}\sum_{n=1}^{N} r_{nk}(\mathbf{x_n} - \mu_k)^2 \tag{3.14}$$

$$\pi_k = \frac{1}{N}\sum_{n=1}^{N} r_{nk} \tag{3.15}$$

GMM clustering algorithm test

The GMM clustering algortihm is tested on the spectrogram generating the distribution seen in figure 3.7 and produces the clusters seen in table 3.3 which have means nearly identical to the clusters produced by the K-Means algorithm. Hence, the segmentation of the spectrogram will be almost identical to the one seen in figure 3.8. In this example the added complexity of using the GMM algorithm is not justified.

Cluster	Cluster 1	Cluster 2	Cluster 3
Cluster center	-130.6 [dBm/Hz]	-105.5 [dBm/Hz]	-83.4 [dBm/Hz]
Cluster variance	165.66 [dBm^2/Hz^2]	166.62 [dBm^2/Hz^2]	165.39 [dBm^2/Hz^2]

Table 3.3: Cluster centers GMM

Using a larger part of the spectrogram obtained from the Aarhus data set as seen in figure 3.9 another test is conducted to display the benefits of using GMM instad of K-Means.

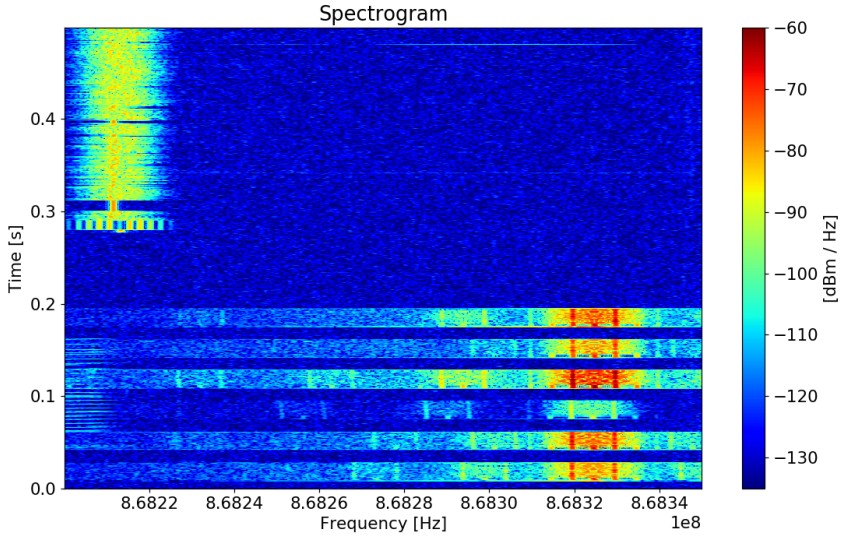

Figure 3.9: Aarhus data set 0.5s spectrogram.

The distribution of received power from figure 3.9 is seen in figure 3.10. Here the peaks are less distinct.

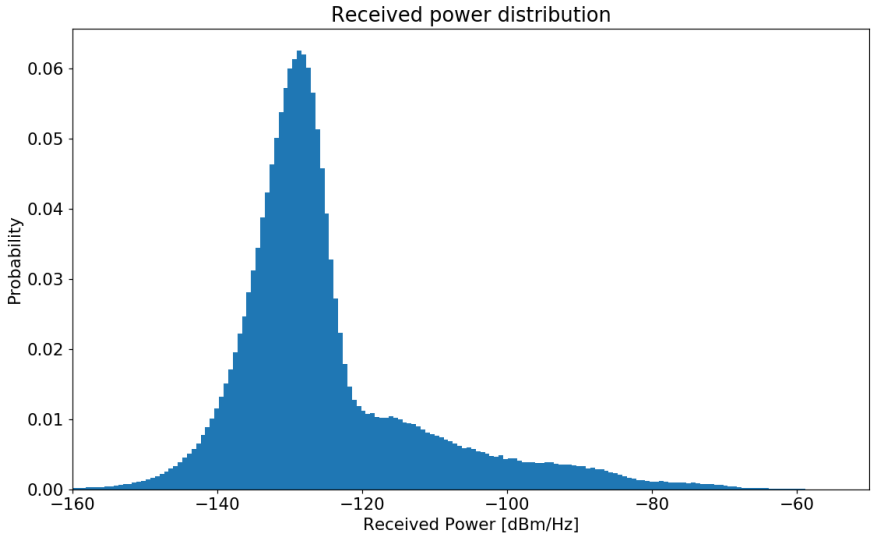

Figure 3.10: Aarhus data set 0.5s spectrogram segmented into clusters.

Both GMM and K-Means is tested here to see if GMM has any advantage, the number of clusters is still $k = 3$. The cluster parameters for both K-Means and GMM is seen in table 3.4.

Cluster	Cluster 1	Cluster 2	Cluster 3
Cluster center K-Means	-132.2 [dBm/Hz]	-119.2 [dBm/Hz]	-94.5 [dBm/Hz]
Cluster center GMM	-130.4 [dBm/Hz]	-112.8 [dBm/Hz]	-93.1 [dBm/Hz]
Cluster variance GMM	184.66 [dBm^2/Hz^2]	182.65 [dBm^2/Hz^2]	184.90 [dBm^2/Hz^2]

Table 3.4: Cluster centers K-Means and GMM

For K-Means, the segmentation of the spectrogram is seen in figure 3.11.

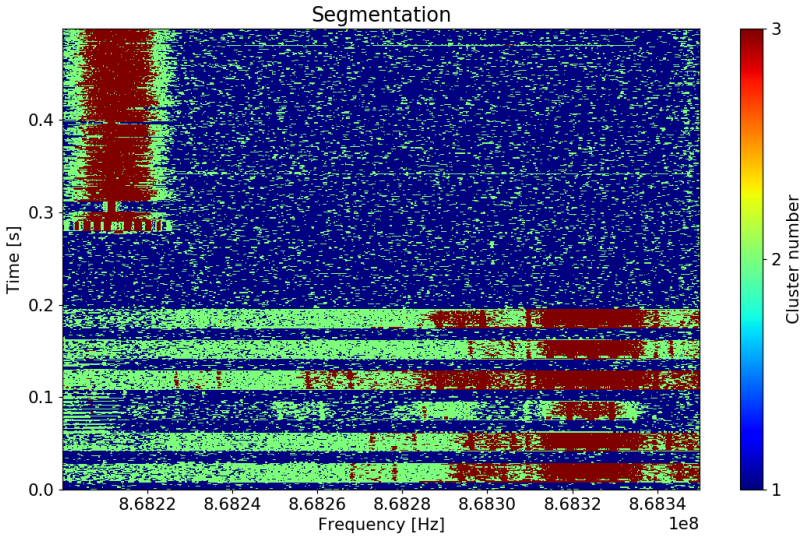

Figure 3.11: Aarhus data set 0.5s spectrogram segmented into clusters by K-Means.

The GMM segmentation is seen in figure 3.12.

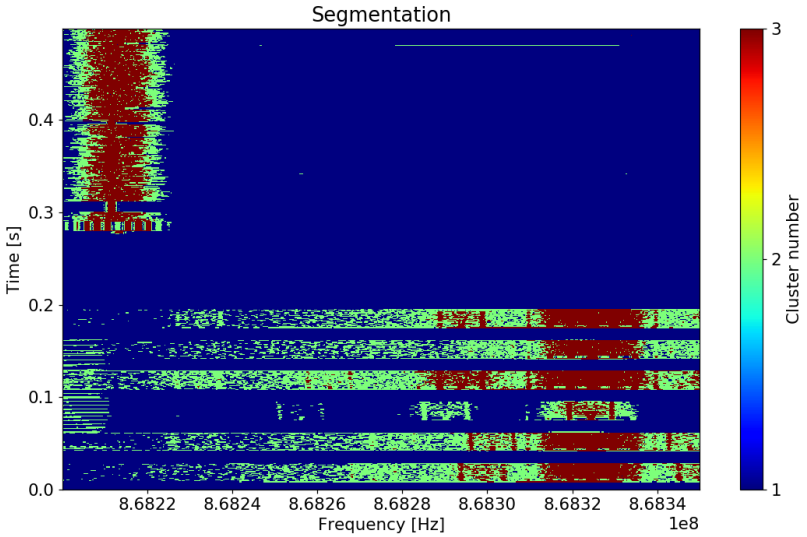

Figure 3.12: Aarhus data set 0.5s spectrogram segmented into clusters by GMM.

By inspecting figure 3.11 and figure 3.12 it becomes clear that the GMM method yields a segmentation with less variation. K-Means has difficulty separating background noise from the different signal levels which GMM seems to handle better.

The two segmentation methods will be tested on a data set with known distribution in section 3.5 to see which is better for segmenting power levels.

3.3 Silhouette score

So far, the number of clusters is found by inspecting the distribution and guessing how many clusters the distribution is composed of. However, the process of choosing an appropriate number of clusters can be automated by using a measurement of how well the clusters matches the samples in a data set. A common method is to test several values of k, compute a metric of fit and then choose the value k that yields the best fit.

One metric for evaluating how well a clustering algorithm fits a data set is the "silhouette score"[18]. It assigns a score in the range $[-1, 1]$ of how well a sample matches its own cluster compared to other clusters. Negative or low values indicates the data set is not well represented by the k clusters and that the number of clusters may be too high or too low.

In eq. (3.16) the mean distance a between a sample $i \in C_k$ and all the other samples assigned to cluster k is calculated. N_k is the number of samples in cluster C_k. A low value indicates that the sample well assigned to the cluster.

$$a_i = \frac{1}{N_k - 1} \sum_{j \in C_k, i \neq j} ||x_i - x_j||^2 \tag{3.16}$$

The dissimilarity of a sample to some cluster is defined as seen in eq. (3.17). It is defined as the mean of the distance to all samples in the neighbouring cluster k which i is not a member of. A high value indicates that the sample is far away from the neighbouring cluster, hence its assignment to its current cluster is appropriate.

$$b_i = \min_{k \neq i} \frac{1}{N_k} \sum_{j \in C_k} ||x_i - x_j||^2 \tag{3.17}$$

The silhouette score is defined as eq. (3.18).

$$s_i = \frac{b_i - a_i}{\max\{a_i, b_i\}} \tag{3.18}$$

A more transparent way of expressing eq. (3.18) is seen in (3.19). When $a_i < b_i$ the mean distance of the sample i to each other sample in the cluster is low, and the distance to the nearest neighbouring

cluster is high and the score goes to 1. Conversely, when $a_i > b_i$ the score goes to -1.

$$s_i = \begin{cases} 1 - a_1/b_i, & a_i < b_i \\ 0 & a_1 = b_i, \\ a_1/b_i - 1, & a_i > b_i \end{cases} \tag{3.19}$$

The silhouette score can be used to determine the optimal value of cluster by using the algorithm below.

1. Chose a range of $k = 1,..,K$ to evaluate.
2. Use a clustering algorithm to fit k clusters to the data.
3. Evaluate the silhouette score s_i for each value of k.
4. Use $k = \underset{k}{\operatorname{argmin}}\, s_i$ to represent the data.

To show that the silhouette scores can be used to find an optimum value k when fitting clusters to the received power distribution obtained from the spectrogram seen in figure 3.9, the silhouette scores are evaluated for the range $k = [2,7]$. In this case the GMM algorithm is used but it might as well have been K-Means or some other clustering algorithm. In figure 3.13 a plot of the silhouette scores is seen. An optimum is found at $k = 3$ where the score is $s_i \approx 0.58$. This is the same number used in the segmentation seen in figure 3.12, which shows that $k = 3$ is a sensible choice for segmenting the power levels.

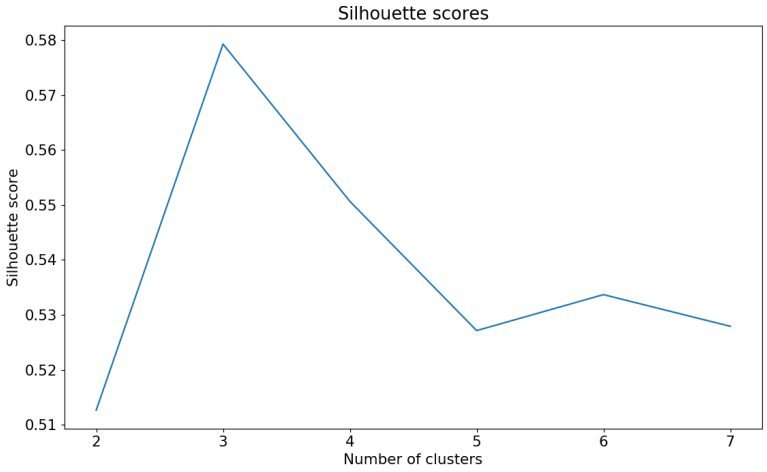

Figure 3.13: Silhouette scores for the Aarhus data set 0.5s

In the following sections and chapters, the number of clusters k will be determined by the algorithm outlined above to ensure optimal number of custers are used given a certain clustering algorithm.

3.4 Detection of transmissions

In section 3.2, a method to segment the power levels in a spectrogram is presented. The transmissions still needs to be identified and their center frequency, bandwidth and transmission time need to be estimated. From a spectrogram segmented into different power levels, this is a two step process. First, the segmented power levels are smoothed, and second an algorithm is used to find the parameters of each separate transmission in a power level cluster.

3.4.1 K-Nearest neighbours post-processing of clustering

The segmentation of power levels in figure 3.15 and 3.17 contains a lot of noise, which can cause problems when estimating the parameters of an interference transmission. To eliminate some of that noise, a k-NN post-processing algorithm is used. Other methods for smoothing the results exists, however the k-NN algorithm is chosen due to low computational complexity and ease of implementation.

Typically the k-NN algorithm is used for regression or classification[17] as shown in the following example. Consider the example in figure 3.14. Here, a sample is the center of a volume containing $k = 5$ classified samples. In this volume, the probability for a sample belonging to class i is $p(C_i|\mathbf{x}) = \frac{N_i}{N}$ where N_i is the number of samples belonging to class i in the volume and N is the total number of samples in the volume. Hence, the probability of mis-classification is minimized by assigning sample $\mathbf{x_n}$ to the same class as the k nearest neighbours, which are the samples contained by the volume.

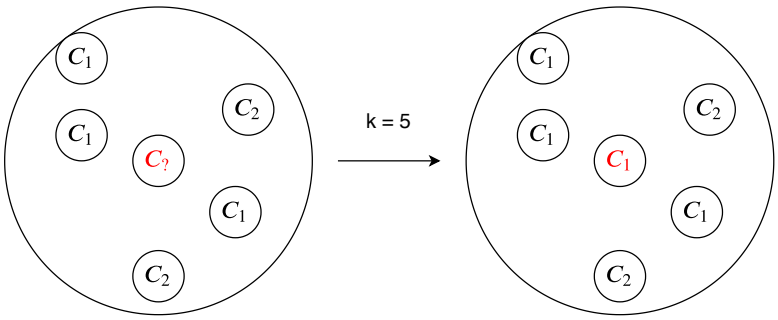

Figure 3.14: K-Nearest neighbour algorithm example.

For each interference unit in figure 3.15 and 3.17 it is assumed that no information is available about which cluster it belongs to. The probability that an interference unit belongs to one of the clusters is evaluated with $k = 24$. The interference unit is then classified as class C_i based on the posterior probability $p(C_i|\mathbf{x}) = \frac{N_i}{N}$. As seen in the spectrogram in figure 3.9, the interference power levels groups into interference transmissions which indicates that an interference unit is likely to belong to the same group as it's neighbours.

In figure 3.16 and figure 3.18 the output of the k-NN classification is seen. Comparing to the spectrogram in figure 3.9, a more coherent segmentation is obtained where neighbouring interference units belongs to the same class.

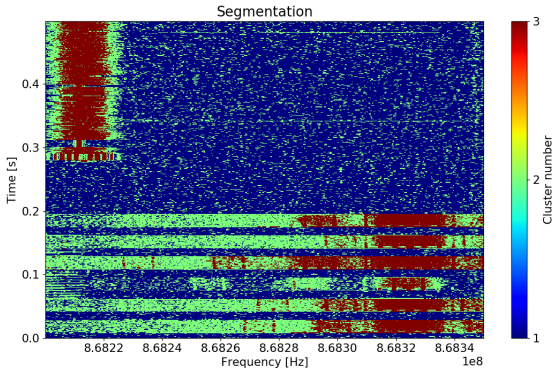

Figure 3.15: K-Means segmentation

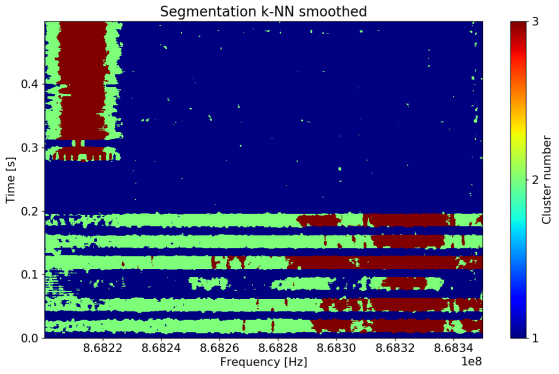

Figure 3.16: K-Means segmentation smoothed

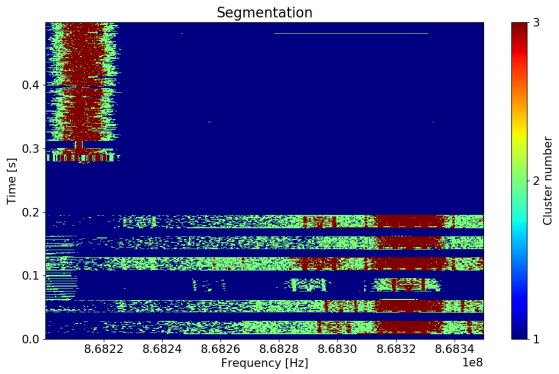

Figure 3.17: GMM segmentation

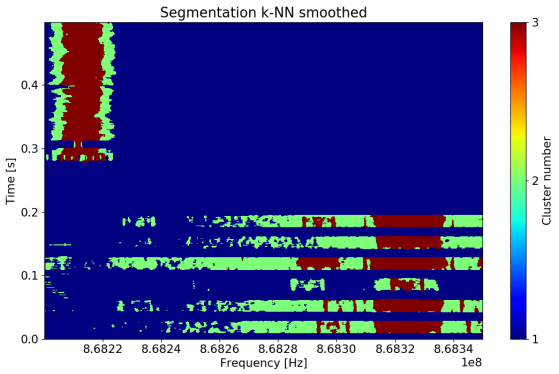

Figure 3.18: GMM segmentation smoothed

3.4.2 Estimation of transmission parameters

To estimate the transmission parameters of each interference transmission, one or more of the power levels can be used. In figure 3.19, the cluster with the largest power of figure 3.12 is plotted. To illustrate how the transmission parameters is extracted, only the maximum power cluster is used.

Figure 3.19 consists of several groups of neighbouring interference units which are neighbours belonging to the same cluster. Each of these groups should be located and their transmission time(height), bandwidth(width) and center frequency(center coordinate) estimated.

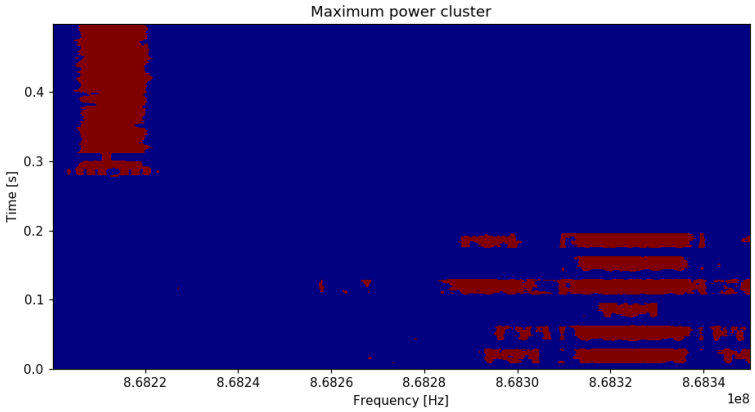

Figure 3.19: Maximum power cluster.

An algorithm to find the borders of each grouping in the cluster is used to estimate the parameters. The algorithm is outlined below.

1. Initialize maximum $x_{max} = 0$, $y_{max} = 0$, $x_{min} = \infty$ and $y_{min} = \infty$
2. Find first non-zero entry in cluster.
3. Save x-y coordinates as start coordinates.
4. Initialize direction vector as $[x = 0, y = 1]$.
5. Go to neighbour interference units, test possible neighbours in ascending order. Use first valid option.
   a) Rotate direction by 90° and take one step ahead.
   b) Take one step ahead.
   c) Rotate direction by −90° and take one step ahead.
   d) Rotate direction by 180° and take one step ahead.
6. Update x_{max}, y_{max}, x_{min} and y_{min}
7. If start position is reached, terminate. Otherwise go to next non-zero entry in cluster not part of $[x_{min}, x_{max}]$ and $[y_{min}, y_{max}]$.

This algorithm aims to find and follow the border of the transmission in a counter-clockwise manner. The coordinate x_{max}, y_{max}, x_{min} and y_{min} is used to find the transmission time(eq. (3.20)), bandwidth(eq. (3.21)) and center frequency(eq. (3.22)).

$$T = (y_{max} - y_{min}) \cdot \Delta t \tag{3.20}$$
$$BW = (x_{max} - x_{min}) \cdot \Delta f \tag{3.21}$$
$$f_c = BW/2 + x_{min} \cdot \Delta f \tag{3.22}$$

To illustrate how the algorithm finds the borders of a transmission, consider figure 3.20. Here the transmission area is updated every time a new maximum or minimum coordinate is found.

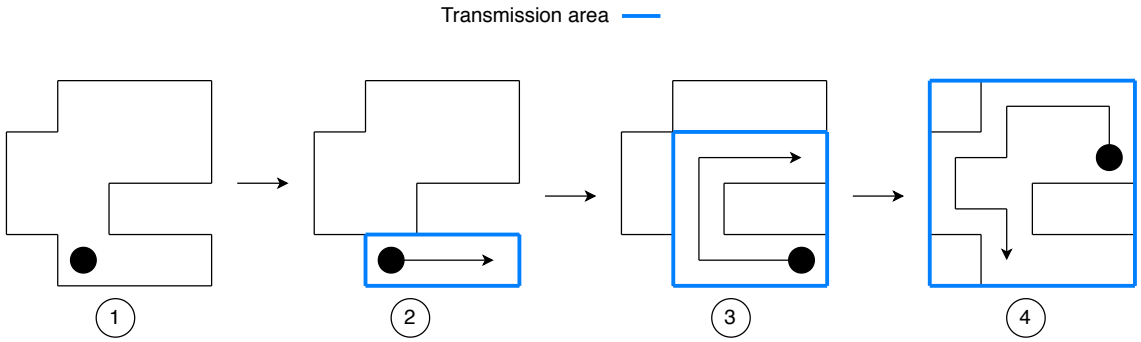

Figure 3.20: Transmission bounds algorithm.

The algorithm outlined above is run on the interference units in figure 3.19 and the result is seen in figure 3.21. It is seen how the transmission centered at $f_c \approx 868.2$MHz if found as well as the six transmissions centered at $f_c \approx 868.32$MHz. The transmissions at $f_c \approx 868.32$MHz contain harmonics which are classified as signals on their own, although not as consistent as the main part of the transmission. Smaller areas at $f_c \approx 868.26$MHz are labelled as transmissions, however a lower bound on the bandwidth-time product can be considered to prevent classification of signals with low bandwidth and transmission time. It is seen that for non-overlapping transmission this methods succeeds in finding the interference transmission borders.

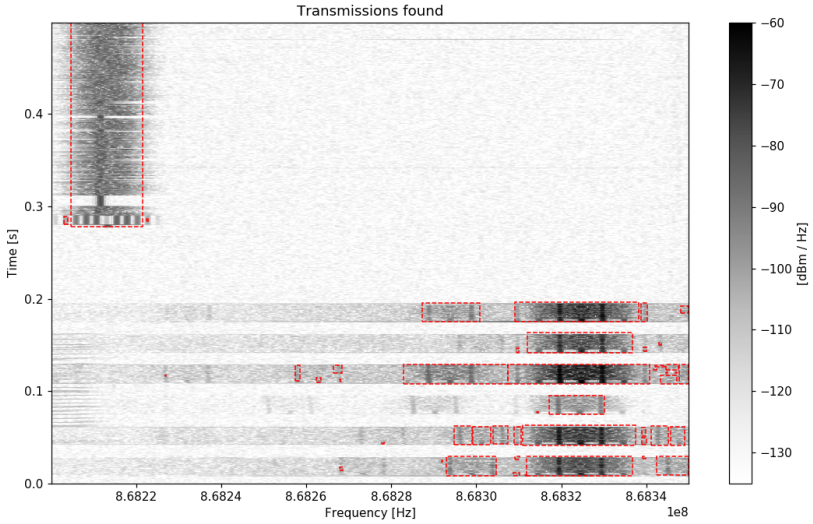

Figure 3.21: Transmission borders(Red dotted line).

3.5 Evaluation of clustering in the transmission extraction algorithm

In this section, the ability of the algorithm to find the transmissions in a spectrogram is evaluated for K-Means and GMM clustering. Based on the findings here, either K-Means or GMM is chosen as the main clustering method for the received power levels. Two different transmissions types are evaluated which can be found in table 3.5. The first type is a random example with a relatively narrow bandwidth and a long transmission time, whereas the second type has a transmission time close to the spectrogram resolution and a relatively wide bandwidth. The purpose of using these signals is to see how the algorithm handles transmissions loading the spectrum differently, i.e. in time or in frequency.

Transmission type	BW [kHz]	T [ms]
1	10	200
2	50	2

Table 3.5: Transmission types

In each test, a test signal is constructed by addition of several individual transmission spaced evenly apart. The signals is offset by a random number of samples, and with different duty cycle, such that the transmissions will drift in time relative to each other. Additive Gaussian noise with a power level of -130dBm/Hz is added to each separate signal. The sample rate of the signal is $f_s = 10$MHz.

The purpose of the test is to quantify how many transmissions are found at different power levels. The transmission parameters f_c, BW and T are not subject to the test.

3.5.1 Transmission type 1

For the first types of transmissions, a 500s test signal composed of 4 different transmissions is constructed; the transmission parameters can be found in table 3.6. For each test signal a spectrogram is computed, which is segmented into power levels and transmissions are located.

#	f_c [MHz]	Offset from f_c [kHz]	Power level [dBm]	BW [kHz]	T [ms]
1	868.3	-150	-60	10	200
2	868.3	-50	-75	10	200
3	868.3	50	-90	10	200
4	868.3	150	-105	10	200

Table 3.6: Transmission type 1 parameters.

A spectrogram of the signal is computed with the parameters seen in table 3.7. The spectrogram can be found in figure 3.22.

Δ T [ms]	Δ f [kHz]	Spectrogram overlap [%]	Decimation factor
2	100	50	5

Table 3.7: Spectrogram parameters

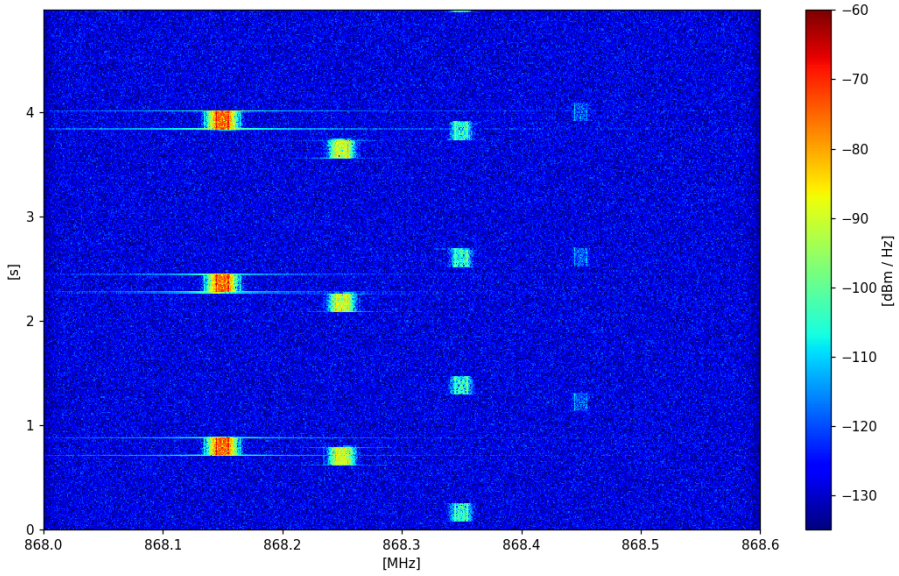

Figure 3.22: Transmission type 1 signal spectrogram.

To evaluate how both K-Means segmentation and GMM segmentation performs, both methods will be tested on the signal as described from section 3.1 to 3.4.

Figure 3.23 shows the transmissions located by using K-Means to segment the power levels.

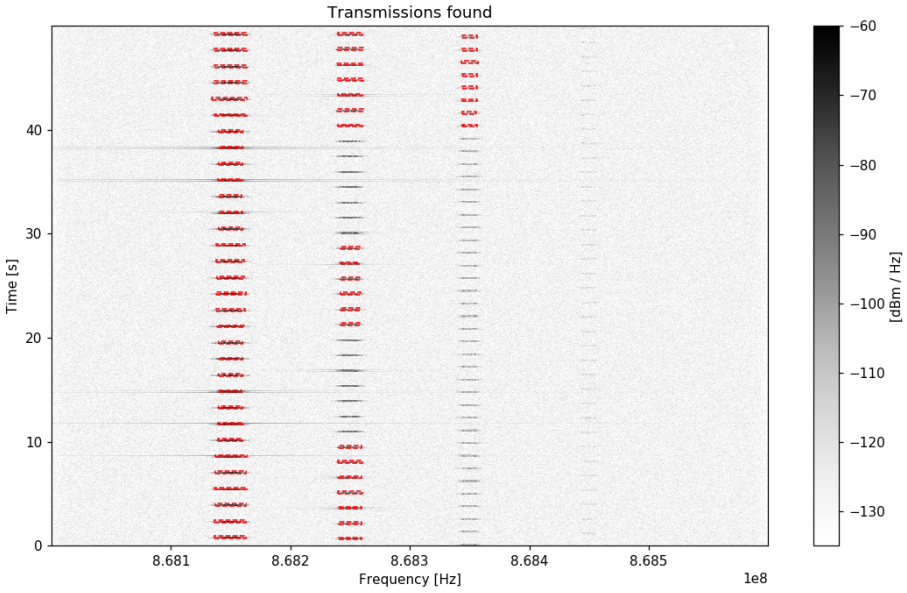

Figure 3.23: Transmission type 1 - Interference transmission found K-Means.

Figure 3.24 shows the transmission located by using GMM to segment the power levels.

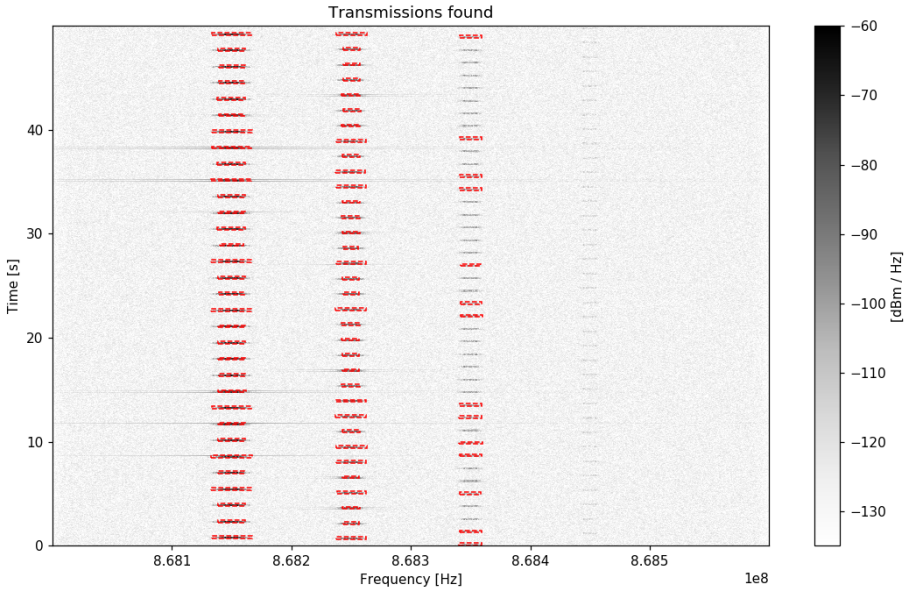

Figure 3.24: Transmission type 1 - Interference transmission found GMM.

To evaluate the two segmentation types, the percentage of transmissions found for each type is calculated and the result is seen in table 3.11. The results show that segmentation using GMM is better for locating low power interference transmissions than K-Means segmentation.

Transmission #	1	2	3	4
Percentage found K-Means [%]	100	59.2	19.2	0
Percentage found GMM [%]	100	100	34.2	0

Table 3.8: Segmentation result K-Means

3.5.2 Transmission type 2

For the second types of transmission, a test signal with duration 20s composed of 4 different transmissions is constructed, the transmission parameters can be found in table 3.9. For each test signal, a spectrogram is computed, which is segmented into power levels and transmissions are located.

#	f_c [MHz]	Offset from f_c [kHz]	Power level [dBm]	BW [kHz]	T [ms]
1	868.3	-200	-60	50	2
2	868.3	-50	-75	50	2
3	868.3	80	-90	50	2
4	868.3	200	-105	50	2

Table 3.9: Transmission type 2 parameters.

The signal spectrogram is computed and seen in 3.22. The transmission paramerers of the signal is computed with the parameters seen in table 3.10.

Δ T [ms]	Δ f [kHz]	Spectrogram overlap [%]	Decimation factor
2	100	50	5

Table 3.10: Spectrogram parameters

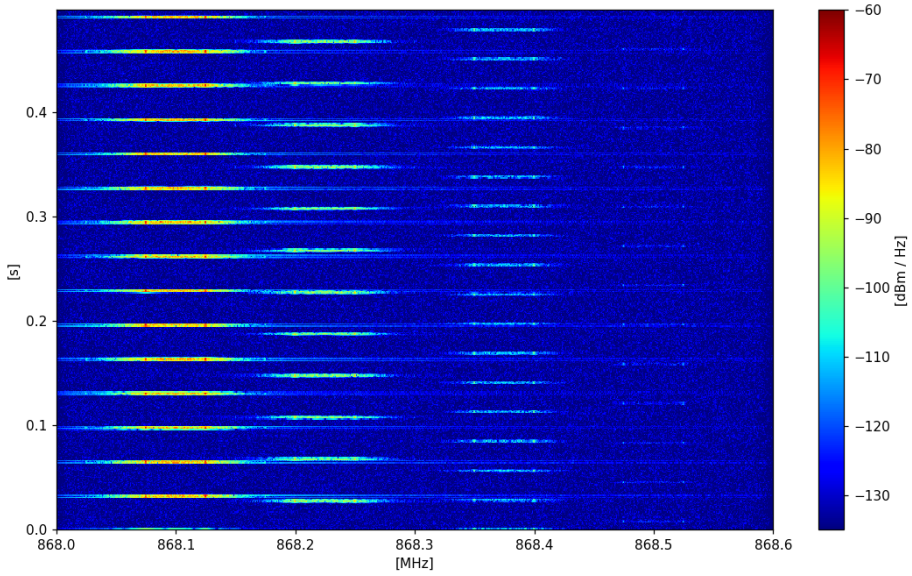

Figure 3.25: Transmission type 2 signal spectrogram.

To evaluate how both K-Means segmentation and GMM segmentation performs, both methods will be tested on the signal as described from section 3.1 to 3.4.

Figure 3.26 shows the signal located by using K-Means to segment the power levels

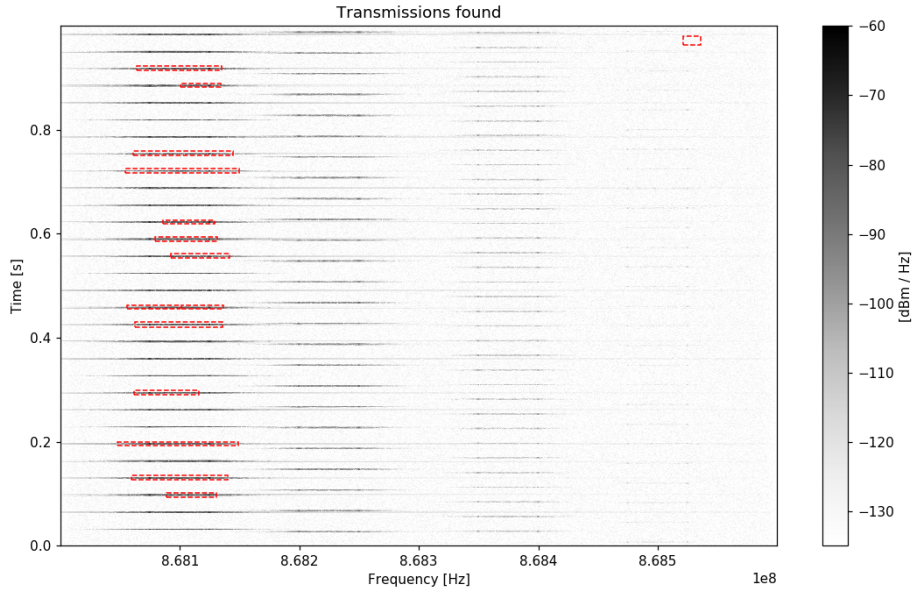

Figure 3.26: Transmission type 2 - Interference transmission found K-Means.

Figure 3.27 shows the signal located by using GMM to segment the power levels.

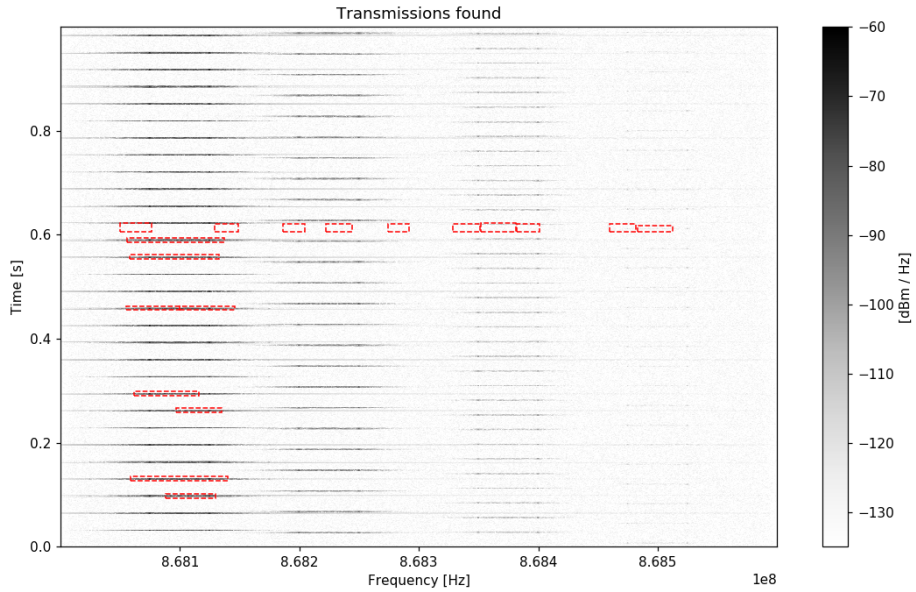

Figure 3.27: Transmission type 2 - Interference transmission found GMM.

Transmission #	1	2	3	4
Percentage found K-Means [%]	41.6	0	0	0
Percentage found GMM [%]	23.6	0	0	0

Table 3.11: Segmentation result K-Means

For this test, segmentation using GMM performs worse in terms of finding transmissions. However both GMM and K-Means finds fewer than half of the $-60dB$ transmission and none of the others. More stray transmissions are located as seen in figure 3.27, these are not included in the results. A low percentage of transmissions is located compared to the 10kHz signals, which is likely to be a consequence of the interference transmission times to be close to the spectrogram time resolution, such that a transmission will only be represented by a few pixels on the time axis. To accommodate short transmissions, it may be necessary to change the time resolution, either by decreasing the frequency resolution or using a lower decimation rate. As a consequence, only transmissions significantly longer than or equal to the time resolution($T \geq 5 \cdot \Delta t$) will be considered in this thesis, in order to maintain the same time and frequency resolution for the spectrograms.

When the transmission length is well above the spectrogram time resolution, the GMM method finds a higher fraction of the transmissions located in the spectrogram, therefore it is used in the final algorithm.

Clustering interference transmissions 4

In section 3.4.2, an algorithm for extracting transmission parameters is presented. Each transmission will have an estimated center frequency, transmission time and transmission bandwidth. To determine which interference transmissions are the most common, clusters in the estimated transmission parameters are found. Based on the cluster centers, a set of characteristic interference transmissions for the spectrum is extracted for modelling of interference in the spectrum. Each cluster should ideally represent an independent transmission source.

Since the number of independent sources may not be known, the number of clusters have to be determined. As seen in section 3.4.2, the estimated transmission parameters may be noisy and the algorithm may identify random noise or harmonics as independent transmission. This implies that some of the data points for clustering may be outliers and the clustering algorithm should be able to handle these while determining the number of clusters. In this chapter, two algorithms are presented and tested; Gaussian Mixture Models as seen in section 3.2 and the "Density-based algorithm for discovering clusters in large spatial databases with noise"(DBSCAN) [19] algorithm.

4.1 GMM Clustering

In section 3.2, the GMM clustering algorithm is presented, as well as a method for finding the appropriate number of clusters. This method is used to cluster the received transmissions. This includes using silhouette scores to determine the number of clusters.

4.2 DBSCAN Clustering

The DBSCAN algorithm is capable of finding clusters with different densities, sizes and shapes in large data sets while handling outliers. The algorithm does this by separating areas with high density from areas with low density. To quantify area density, the algorithm relies on two parameters ϵ and V. For each data point p, a sphere with radius ϵ and center p will contain a number of data points. The number of data points determines if the point p is a "core point", a "border point" or noise.

Each data point has a set of neighbouring points. All the points within the distance ϵ from the center point is said to be neighbours of point p and the set of neighbours is defined as seen in eq. 4.1.

$$N(p) = \{q \in D \mid ||q - p||^2 \leq \epsilon\} \tag{4.1}$$

- Core point: A core point is a point with at least V individual data points within distance ϵ, $|N(p)| \geq V$.

- Border point: A point p which lies in the neighbourhood of a core point and has less than V individual data points in its neighbourhood, $|N(p)| < V$.
- Noise: A point which is not a core or a border point.

In figure 4.1 the three types of points are illustrated, here $V = 3$. Point A is a core point, point B is a border point and point C is noise.

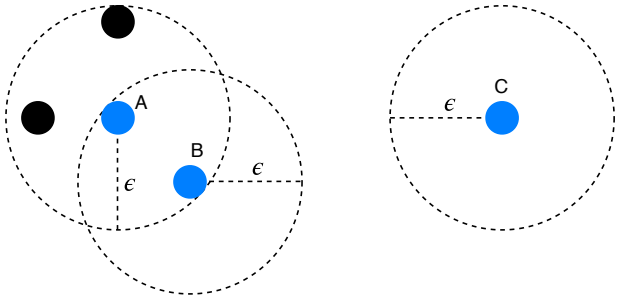

Figure 4.1: DBSCAN Points classification.

Three important definitions in DBSCAN rely on the definitions of points shown above. Directly density reachable, density reachable and density connected.

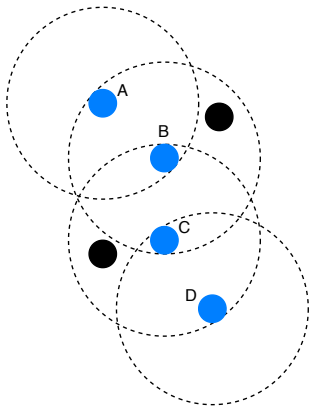

Figure 4.2: DBSCAN Connection classification.

- Directly density reachable: If point B is a core point and point A is in the neighbourhood of B, then A is directly density reachable from B.
- Point A is density reachable from point B if a chain of points $b_1, ..b_n$ with $b_1 = B$ and $b_n = A$, where the $b_i + 1$ is directly density reachable from b_i for all i, connects the point A and B.
- Density connected: Point A and B are density connected of they belong to the same cluster, but do not share a common core point.

In figure 4.2, the point B is directly density connected to point A, point D is density reachable from point B and point A and D is density connected.

The algorithm executes as follows:

1. The neighbourhood set for an arbitrary point p which is not yet visited is found.

2. If $|N(p)| \geq V$ for point p the data point is classified as belonging to cluster k, otherwise the data point is classified as noise. This point can later be found to be density reachable from another point and made part of the corresponding cluster.
3. If the point p is a core point, all the points density reachable from point p is also part of cluster k. This may cause clusters to be merged.
4. If all points are visited, terminate. Otherwise, go to step 1.

The downside of using DBSCAN is that finding a meaningful ϵ parameter requires domain knowledge, hence it may be difficult to find. However, for this application the input features will be normalized to the range [0,1] and the ϵ parameter can be computed by deciding a percentage threshold for how much the parameters are allowed to deviate with respect to the maximum center frequency, transmission time or transmission bandwidth.

4.3 Interference clustering test

To show that the clustering methods GMM and DBSCAN can cluster transmissions and extract information about the dominating interferers in the spectrum, a test of both algorithms is performed. For testing the clustering algorithms, two different test signals has been constructed. The first test signal consists of four individual transmissions and the parameters for the individual transmissions can be seen in table 4.1. Here all the transmission has the same power level, and the purpose of using this signal, is to show that the algorithm is capable of determining the number of interferers, as well as the correct center frequency, transmission time and transmission bandwidth.

#	f_c [MHz]	Offset from f_c [kHz]	Power level [dBm]	BW [kHz]	T [ms]
1	868.3	-200	-60	5	40
2	868.3	-75	-60	10	20
3	868.3	75	-60	15	13.33
4	868.3	200	-60	20	10

Table 4.1: Transmission type 1 parameters.

In figure 4.3 a spectrogram of the first 2 seconds of the test signal is seen.

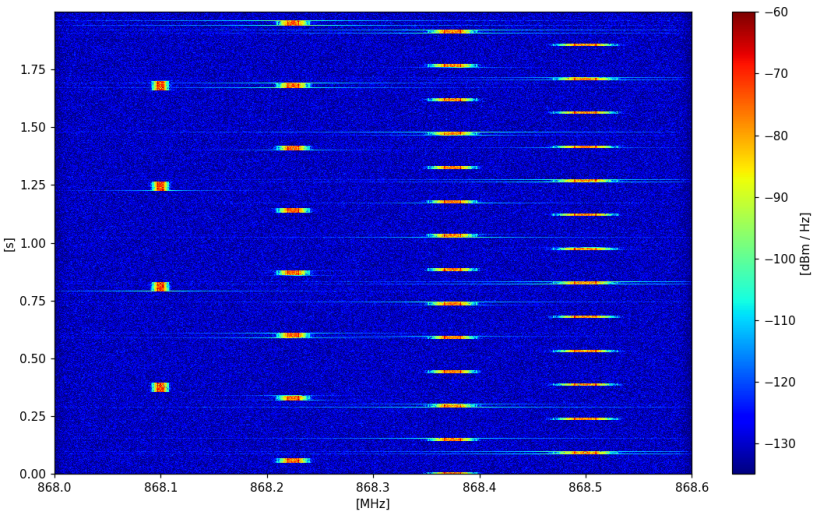

Figure 4.3: Clustering test signal 1 spectrogram.

The second test signal consists of six individual signals. The purpose here is to verify that for different power levels, the algorithm is still capable of determining the correct transmission parameters. Hence, three different power levels are used as seen in table 4.2.

#	f_c [MHz]	Offset from f_c [kHz]	Power level [dBm]	BW [kHz]	T [ms]
1	868.3	-250	-60	10	20
2	868.3	-150	-60	20	10
3	868.3	-50	-75	10	20
4	868.3	50	-75	20	10
5	868.3	150	-90	10	20
6	868.3	200	-90	20	10

Table 4.2: Transmission type 2 parameters.

In figure 4.3 a spectrogram of the first 2 seconds of the test signal is seen.

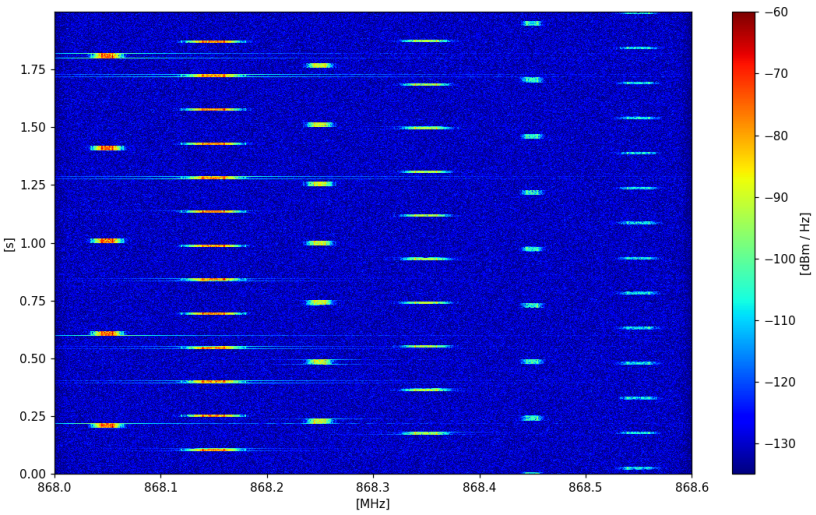

Figure 4.4: Clustering test signal 2 spectrogram.

In section 3.2, the GMM segmentation of the received power proved to be the more effective than K-Means, when the transmission time was several times longer than the time resolution of the spectrogram. Since both these test signals has a transmission time $T \geq 5 \cdot \Delta t$, GMM segmentation is used in this test.

The results in section 3.5 show that the estimated bandwidth of the signals varies much more than the transmission time and the center frequency. For simplicity, the bandwidth will not be used as a parameter for clustering the transmission parameters in this test. In the rest of the thesis, this is not the case. The cluster centers will be marked by an opaque sphere with the same colour as the samples in the cluster.

4.3.1 GMM Clustering test

Silhouette tests are performed for 2 to 20 clusters, and the number of clusters with the largest score is used to cluster the transmission parameters.

GMM clustering - Results signal 1

In table 4.3, results for GMM clustering of the transmission parameters obtained from test signal 1 is seen. Four clusters are found as expected, since four individual signals is used to generate test signal 1. The average error in percentage is 0.0042% for frequency offset, 135.41% for the bandwidth and 32.88% for transmission time.

Cluster #	Offset from f_c [MHz]	T [ms]	BW [kHz]	Percentage of samples [%]
1	199.93	15.42	44.53	41.27
2	−200.05	45.04	12.55	10.07
3	−75.03	24.97	23.82	19.38
4	74.97	18.65	34.48	29.28

Table 4.3: GMM Cluster centers for test signal 1.

In figure 4.5 the result of the clustering is seen.

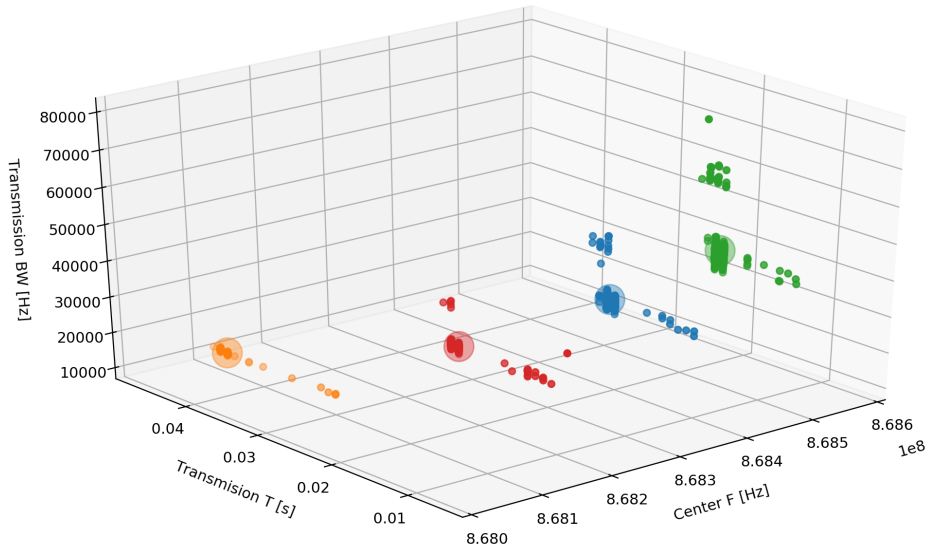

Figure 4.5: GMM Clustering of transmission parameters for test signal 1.

GMM clustering - Results signal 2

In table 4.4 the results for GMM clustering of test signal 2 is seen. Six clusters are found as expected, since six individual signals is used to generate test signal 2. The average error in percentage is 0.167% for frequency offset, 84.93% for the bandwidth and 38.82% for transmission time.

Cluster #	Offset from f_c [MHz]	T [ms]	BW [kHz]	Percentage of samples [%]
1	−250.05	25.47	23.41	15.31
2	49.97	14.94	26.33	29.35
3	−152.30	15.44	43.37	36.24
4	−50.05	25.01	16.39	17.05
5	243.51	15.10	33.51	1.41
6	150.00	25.14	19.55	0.64

Table 4.4: GMM Cluster centers for test signal 2.

In figure 4.6 the result of the clustering is seen.

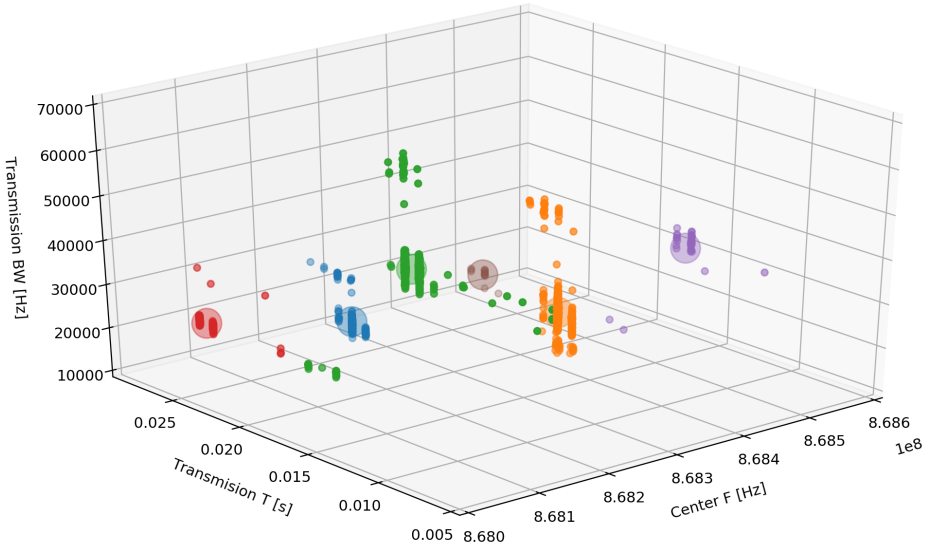

Figure 4.6: GMM Clustering of transmission parameters for test signal 2.

4.3.2 DBSCAN Clustering test

DBSCAN needs two parameters, the minimum number of samples in a cluster, V, and the core distance, ϵ, as described in section 4.2. It is difficult to argue for a specific value of V, but the value $V = 10$ yields good results for this specific signal and application. It may be necessary to change the parameter to better match other scenarios. For the parameter ϵ, it is decided that samples with center frequencies within 6kHz belongs to the same cluster, which gives $\epsilon = 0.01$ with a total bandwidth of 600kHz.

DBSCAN clustering - Results signal 1

In table 4.5 the results for DBSCAN clustering of test signal 2 is seen. Five clusters are found which is one more than expected. However, cluster 5 contains less than 1% of all samples, with parameters similar to cluster 1. The average error in percentage is 0.0042% for frequency offset, 135.41% for the bandwidth and 34.39% for transmission time.

Cluster #	Offset from f_c [MHz]	T [ms]	BW [kHz]	Percentage of samples [%]
1	−75.03	25.51	23.80	18.34
2	199.93	15.53	44.55	40.57
3	74.97	18.80	34.50	28.75
4	−200.05	45.47	12.55	9.72
5	−75.03	15.65	23.62	0.99

Table 4.5: DBSCAN Cluster centers for test signal 2.

In figure 4.7 the result of the clustering is seen. The DBSCAN algorithm classifies noise separately, which is seen as the blue samples.

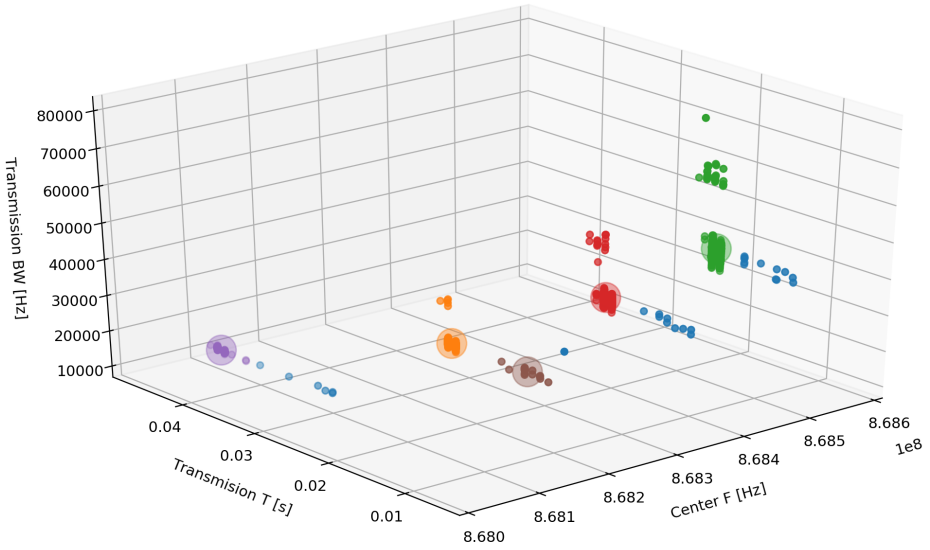

Figure 4.7: DBSCAN Clustering of transmission parameters for test signal 1.

DBSCAN clustering - Test signal 2

In table 4.6 the results for DBCAN clustering of test signal 2 is seen. Seven clusters are found where six is expected, since six individual signals is used to generate test signal 2. However, cluster 7 seems to be a duplicate of cluster 4 which are merged to compute the average error percentage. The average error in percentage is 0.012% for frequency offset, 85.43% for the bandwidth and 39.16% for transmission time.

Cluster #	Offset from f_c [MHz]	T [ms]	BW [kHz]	Percentage of samples [%]
1	249.96	15.25	34.56	1.28
2	−150.04	15.49	43.85	34.96
3	49.97	14.94	26.33	29.35
4	−250.05	25.53	23.38	15.13
5	−50.05	25.01	16.39	17.05
6	150.00	25.14	19.55	0.64
7	−250.03	16.83	22.95	0.82

Table 4.6: DBSCAN Cluster centers for test signal 2.

In figure 4.8 the result of the clustering is seen. The DBSCAN algorithm classifies noise separately, which is seen as the blue samples.

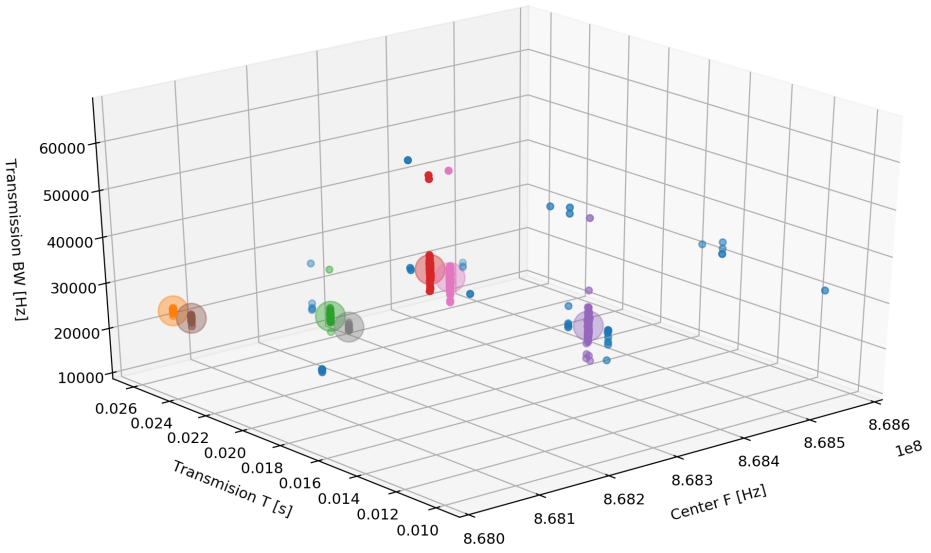

Figure 4.8: DBSCAN Clustering of transmission parameters for test signal 2.

4.3.3 Clustering results - GMM vs. DBSCAN

Both GMM and DBSCAN finds good estimates for the center frequency, a decent estimate of the transmission time and overestimates the transmission bandwidth with more than a factor of two for test signal 1 and almost a factor of two for test signal 2. The cluster centers found by the two algorithms are more or less identical however the DBSCAN algorithm tends to overestimate the number of individual transmitters. For both test signals, the DBSCAN algorithm finds one more cluster than there are actual transmitters.

In section 3.5, it is found that the interference transmissions with low transmission power is harder to detect in presence of interference transmission with higher power. In test signal 2, the low power interference transmissions are found much less frequently than other transmissions as seen in table 4.6, and 4.4. One approach to mitigate this problem, would be to separate the spectrogram into several frequency bands and segment the frequency bands separately.

However, using a more complex signal, as described in section 5.1, where the data points are not that well separated, the silhouette score cannot determine the correct number of data points. In figure 4.9, it is seen how clusters are formed by the DBSCAN algorithm, relative to figure 4.10, where the GMM algorithm is used. In figure 4.10, clusters are formed across large ranges of bandwidth, center frequency and transmission time indicating that the number of clusters is too low. To mitigate this, other methods than the silhouette score could be used to find the optimal number of clusters if the GMM method is preferred.

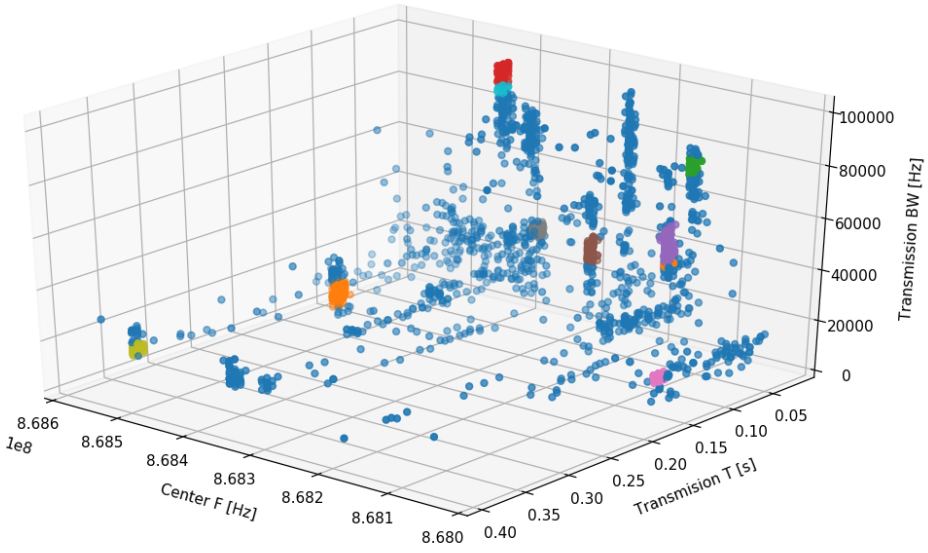

Figure 4.9: DBSCAN Clustering of transmission parameters for test signal.

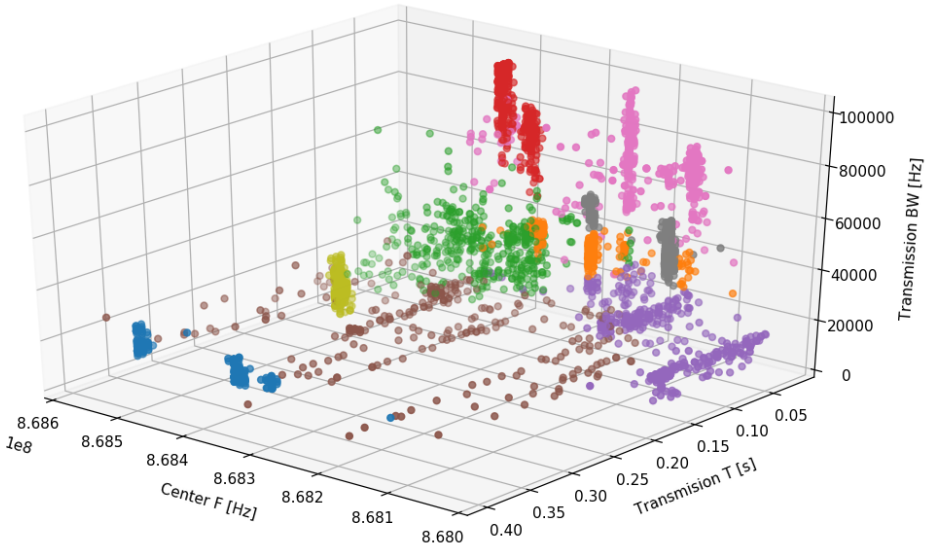

Figure 4.10: GMM Clustering of transmission parameters for test signal.

Algorithm test 5

In this chapter the algorithm developed in chapter 3 and 4 to estimate the parameters in the ALOHA interference model is evaluated. In the model, both transmission rate and average interference transmission length in a frequency range is needed. The error of those two estimates depends on the algorithms ability to find a large fraction of the transmissions in a spectrum as well as estimating the transmission parameters; center frequency, transmission time and bandwidth. A labelled dataset is simulated with known interference sources to test the fraction of transmissions found and the estimate error at different spectrum congestion scenarios. Furthermore, a real-world unlabelled dataset provided by Kamstrup is analyzed. For the real-world dataset the interference sources are not known, however the clusters found are compared with the most common interferers in the signal spectrogram as a sanity check of the algorithm. The amount of interference found in the signal by the algorithm is compared to the actual amount where a high fraction will indicate good estimates for the transmission rate and average transmission length.

5.1 Test signal generation

To generate realistic test signals, the histogram of received power levels from the Aarhus dataset is considered, see figure 5.1. In a real-life interference scenario, transmitters far away from the receiver will cause low power interference transmission to be seen at the receiver and transmitters closer to the receiver will contribute with high power transmission at the receiver. The tail seen in the distribution in figure 5.1 is the combination of these transmissions. The large amount of values at ≈ -130dBm/Hz is complex Gaussian noise as discussed in section 3.1.

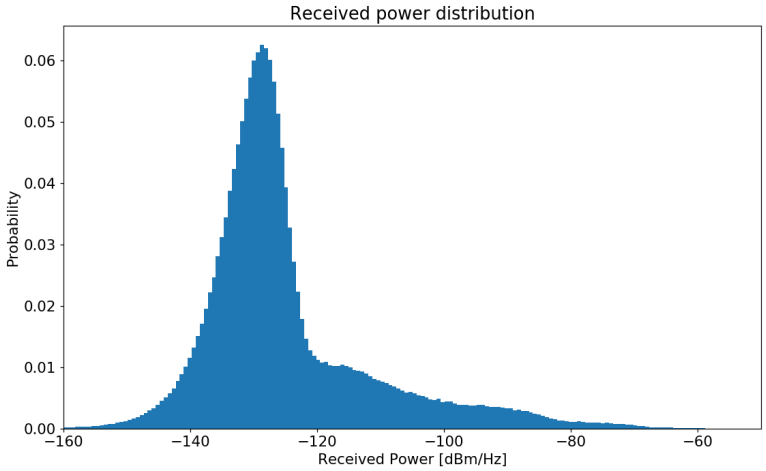

Figure 5.1: Aarhus data set 0.5s spectrogram.

To approximate the low power interferers, a range of interference signals is constructed where the power levels are drawn from the distribution of power levels in the Aarhus dataset.

By normalizing the histogram, it is possible to compute a probability distribution from where power levels can be drawn. The transmission parameters f_c, T and BW are choosen with uniform distribution from the ranges seen below.

- Center frequency, f_c: 868MHz to 868.6MHz
- Transmission time, T: 0.034s to 340s
- Transmission bandwidth, BW: 2kHz to 50kHz

To keep the computational complexity at a reasonable level, the transmission time and transmission bandwidth is kept in these ranges. If a large range of parameters are chosen, the resolution required to detect transmissions in each end of the ranges will become too computationally expensive.

Each interference signal will transmit at a a predefined duty cycle. Furthermore, white Gaussian noise is added with power level -130dBm/Hz(Approximate mode of the noise distribution) to simulate thermal noise.

An interference signal with 40 low power sources generated as described above is constructed, in figure 5.2 the spectrogram of the signal is seen.

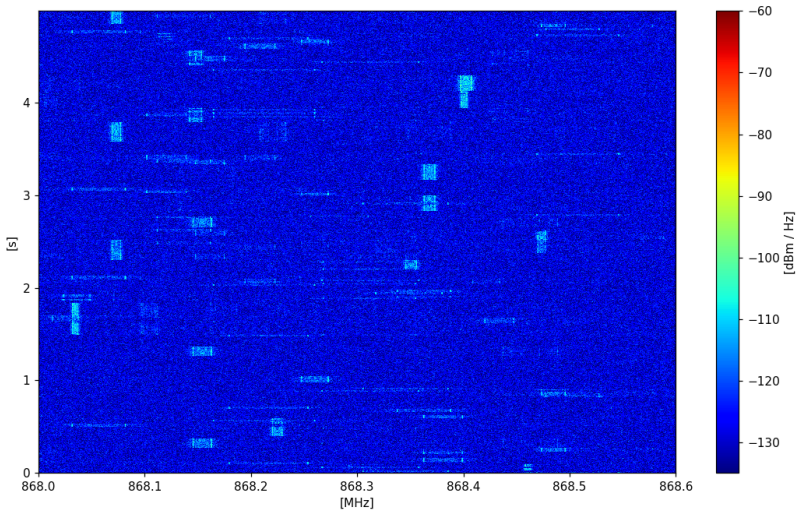

Figure 5.2: Interference noise spectrogram example.

Since all signals drawn from the power levels distribution are likely to be low power, a number of high power signals are added to the interference signal. These interference sources models the transmitters placed closer to the transceiver.

Transmission #	T [ms]	BW [kHz]	Power level range [dBm]	Modulation type
1	340	5	[-90, -75, -60]	2GFSK
2	170	10	[-90, -75, -60]	2GFSK
3	68	25	[-90, -75, -60]	2GFSK
4	34	50	[-90, -75, -60]	2GFSK

Table 5.1: Interference transmission types

In figure 5.3, an example of 40 low power interference sources combined with three of each transmission type shown in table 5.1 is seen. Here the four transmission types are transmitted with power levels -60dBm/Hz, -75dBm/Hz and -90dBm/Hz.

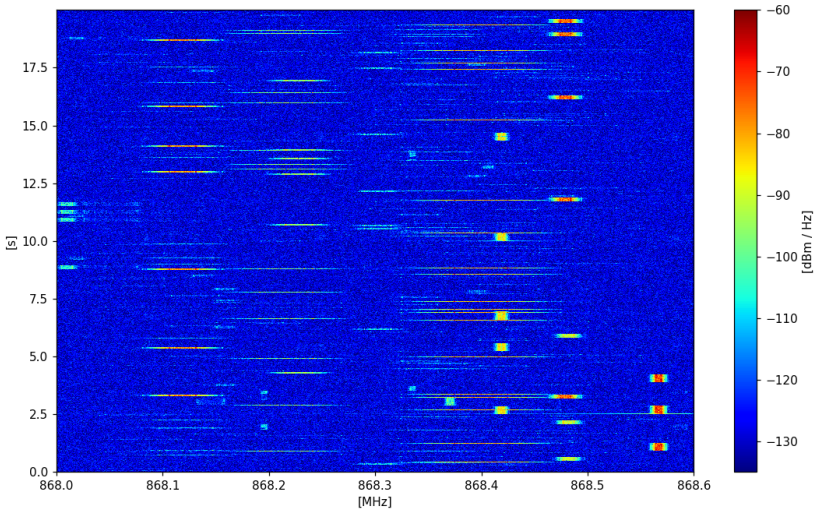

Figure 5.3: Interference noise spectrogram example.

Two types of test signals are constructed which are used to verify that the algorithm is capable of reliable extraction and clustering of transmission parameters. The test signals has a length of 15 minutes.

Parameter estimatation test signal

To simulate a real spectrum with interference, 3 separate signals are constructed. Each with 40 low power noise sources and 12 interference sources, 3 of every transmission type seen in 5.1 with separate power levels. The transmissions are assigned a random center frequency, uniformly chosen in the range 868MHz $-$ 868.6MHz. The time between the transmissions are randomly drawn, such that the total transmission time matches a given duty cycle over the total signal length.

Each separate signal transmits with a duty cycle to simulate different levels of congestion in the spectrum. Both the low power and the high power sources has the same duty cycle of 2%, 4% or 6%. Since the allowed duty cycle for many unlicensed spectrum bands is in the range $0.1 - 1\%$ [20], these levels of congestion should ideally simulate worst case conditions.

Transmission extraction test signal

For the second type of test signals, only one signal is constructed with 12 interference sources, 3 of every transmission type seen in 5.1 with separate power levels. The center frequencies are chosen such that the interference transmissions are spread out evenly in the 868MHz − 868.6MHz spectrum with a minimum of overlap. Time between transmissions and duty cycle are identical to the previous test signal. Again, the signal is constructed at three different congestion levels, 2%, 4% and 6%.

5.2 Algorithm parameters

From the algorithm development in chapter 3 and 4, a set of parameters for the algorithm is found. This set is used on the test signals to test the performance of the algorithm without any changes. The test signals in the development tests are generated differently from the test signals used in this chapter. They have different transmission time and bandwidth ranges, fixed time between transmissions and only complex Gaussian noise with no low power interference sources. The algorithm is tested on both real and simulated test signals, which are generated differently from the development test signals used to find the algorithm parameters. This is important, since tuning the algorithm parameters on the test data will cause bad generalization to other datasets and may yield parameters which only work for that specific dataset. Furthermore, by finding the parameters from a dataset generated differently from the tests sets, it is expected that the results will be comparable to any results found by analysing previously unseen data. The algorithm parameters can be found in table 5.2.

For these tests, DBSCAN will be used, since the test in section 4.3 showed that the silhouette tests for the GMM method has trouble finding a suitable number of clusters in data where the clusters are less distinct. An upper bound for the bandwidth and transmission time is used in parameter extraction. The clustering algorithm normalizes the data before clustering, and outliers with large values will cause the remaining data points to be close in value. This causes large variation in the performance of the DBSCAN algorithm since the ϵ parameter depends on reliable scaling. By introducing these upper bound, as well as using them as max when normalizing the parameters, the effect of epsilon can be calculated for all three parameters. With $\epsilon = 0.01$, the parameters for the clustering will be $\Delta BW = 1\text{kHz}$, $\Delta T = 5\text{ms}$[1] and $\Delta f_c = 1\text{kHz}$.

Spectrogram		
Δt	Δf	Overlap
2 ms	100 Hz	50%
Power level segmentation		
Clustering method	Clustering interval	Max clusters
GMM	500 ms	10
Parameter extraction		
Max bandwidth	Max transmission time	kNN Smooting
100 kHz	500 ms	K = 9
Transmission parameters clustering		
Method	ϵ	Min samples/cluster
DBSCAN	0.01	10

Table 5.2: Algortihm parameters

[1] Five interference units on a spectrogram with $\Delta t = 2\text{ms}$ and 50% overlap

5.3 Algorithm test - Transmission extraction test

In section 2.4 the ALOHA model for interference is introduced. The model parameters are the average interference transmission rate r and the transmission time length T_P. The parameter T_P is the length of the desired transmission added to the average length of the interference transmissions in a frequency range. If a large fraction of the transmissions in a spectrum is detected, the estimated transmission rate will be close to the true value. The average length of the interference transmissions also depends on the fraction of transmissions which are detected. As an example, if two transmissions are present in a spectrum and only a large fraction of the first type is detected, the average transmission time will be biased heavily towards this transmissions average length.

Hence, a prerequisite for obtaining good estimates of the model parameters is to detect a large fraction of the transmissions in a spectrum. It is hard to quantify the error in average transmission length, since this depends heavily on the frequency range used for a desired transmission and the individual interference sources in the spectrum. On the other hand, the error when estimating the transmission rate is easier to quantify, since a success rate of 90% in detecting transmissions will result in the estimated transmission rate to be 90% of the true value[2].

Since the fraction of detected transmissions is critical to model the interference, a test to detect the the number of false negatives(Missed transmissions) is conducted for the test signals specified in table 5.1. The transmission extraction test signal described in section 5.1 is used.

Transmission extraction - 2% duty cycle

In figure 5.4 a spectrogram of the transmission extraction test signal with 2% duty cycle is seen.

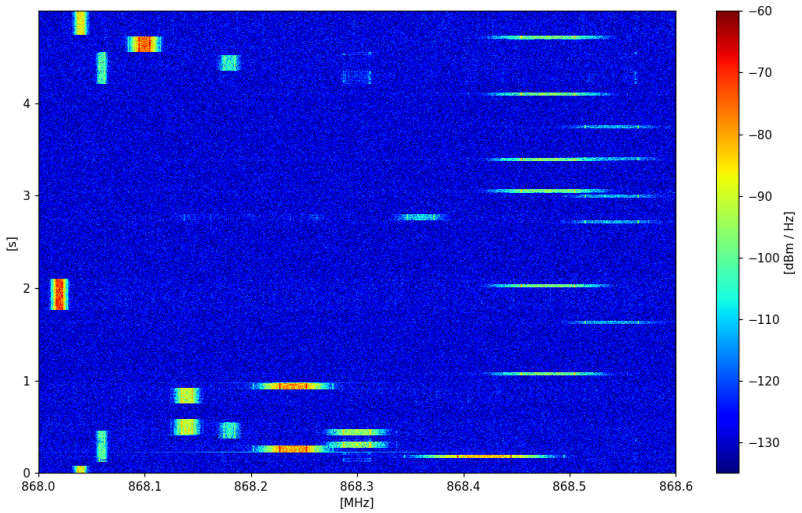

Figure 5.4: Interference first 5 for 2% duty cycle.

[2]Since transmission rate has units Transmissions/second

Transmission extraction - 2% duty cycle results

Table 5.3 shows the percentage of transmission found for each interference source in the test signal.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]	% found
1	868.020	340	5.000	-60	87.50
2	868.040	340	5.000	-75	74.07
3	868.060	340	5.000	-90	19.23
4	868.100	170	10.000	-60	96.30
5	868.140	170	10.000	-75	67.74
6	868.180	170	10.000	-90	29.41
7	868.240	68	25.000	-60	97.92
8	868.300	68	25.000	-75	61.54
9	868.360	68	25.000	-90	10.42
10	868.420	34	50.000	-60	87.93
11	868.480	34	50.000	-75	26.92
12	868.540	34	50.000	-90	2.17

Table 5.3: Interference transmission clustering results with 2% duty cycle

All the interference sources are found, but it is clear that the high powered sources are easier for the algorithm to detect. The 12'th transmission is barely found at 2.17% of the transmissions detected.

Transmission extraction - 4% duty cycle

In figure 5.5 a spectrogram of the transmission extraction test signal with 4% duty cycle is seen.

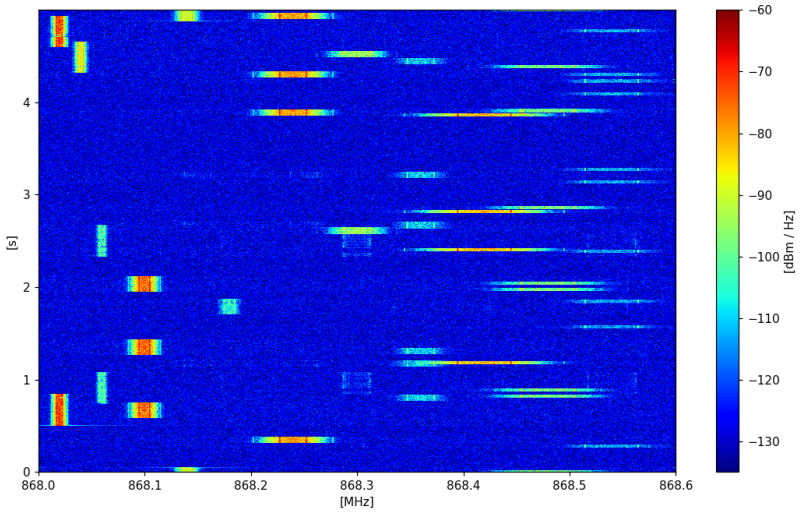

Figure 5.5: Interference first 5 for 4% duty cycle.

Transmission extraction - 4% duty cycle results

Table 5.4 shows the percentage of transmission found for each interference source in the test signal.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]	% found
1	868.020	340	5.000	-60	92.86
2	868.040	340	5.000	-75	71.43
3	868.060	340	5.000	-90	0.00
4	868.100	170	10.000	-60	97.87
5	868.140	170	10.000	-75	57.89
6	868.180	170	10.000	-90	10.42
7	868.240	68	25.000	-60	97.01
8	868.300	68	25.000	-75	45.31
9	868.360	68	25.000	-90	6.33
10	868.420	34	50.000	-60	78.86
11	868.480	34	50.000	-75	26.21
12	868.540	34	50.000	-90	0.00

Table 5.4: Interference transmission clustering results with 4% duty cycle

Here, only one of the −90dBm sources are found, and a comparable fraction of the −60dBm power sources if found. The fraction of −75dBm power sources found have declined a bit.

Transmission extraction - 6% duty cycle

In figure 5.6 a spectrogram of the transmission extraction test signal with 6% duty cycle is seen.

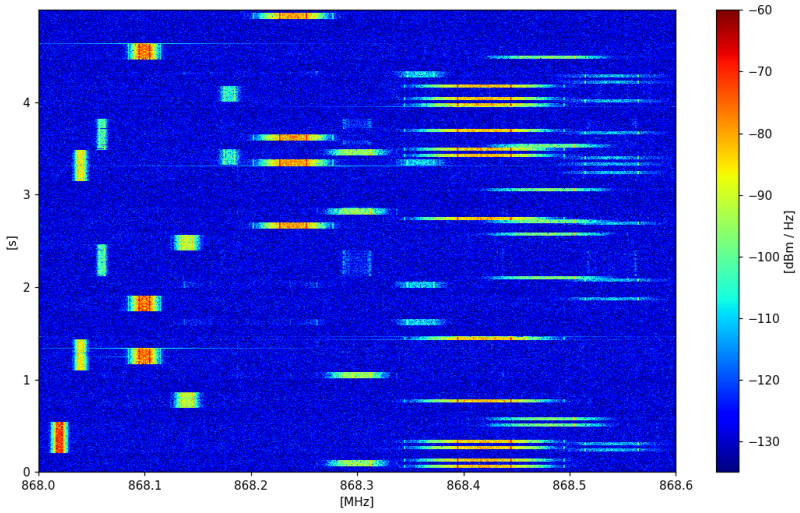

Figure 5.6: Interference first 5 for 6% duty cycle.

Transmission extraction - 6% duty cycle results

Table 5.5 shows the percentage of transmission found for each interference source in the test signal.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]	% found
1	868.020	340	5.000	-60	93.55
2	868.040	340	5.000	-75	89.19
3	868.060	340	5.000	-90	0.00
4	868.100	170	10.000	-60	97.83
5	868.140	170	10.000	-75	54.55
6	868.180	170	10.000	-90	0.00
7	868.240	68	25.000	-60	93.81
8	868.300	68	25.000	-75	25.45
9	868.360	68	25.000	-90	0.00
10	868.420	34	50.000	-60	86.53
11	868.480	34	50.000	-75	15.08
12	868.540	34	50.000	-90	0.00

Table 5.5: Interference transmission clustering results with 4% duty cycle

For this test none of the -90dBm power sources found and the fraction of -74dBm power sources found has decreased.

5.3.1 Transmission extraction test discussion

As expected, the high power transmission has a larger probability for being found than the low power transmissions. From the tables 5.3, 5.4 and 5.5 the results in figure 5.7 are compiled.

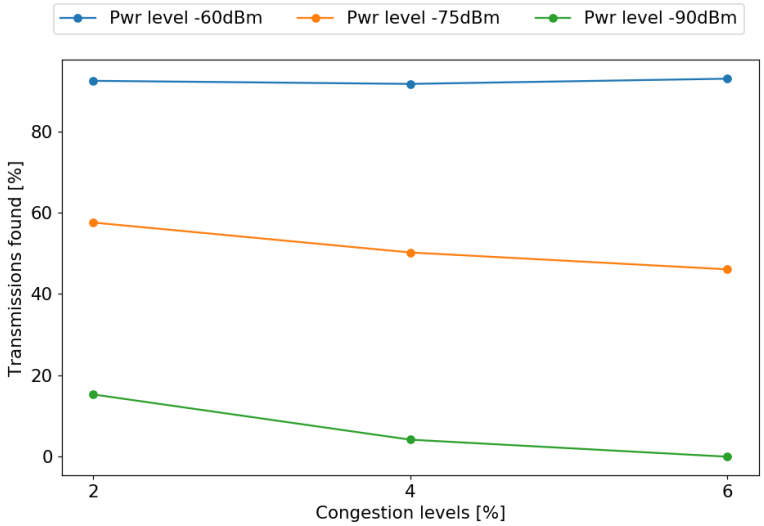

Figure 5.7: Average percentage of transmissions found

The values in figure 5.7 indicates how close the ALOHA parameter estimates are to the true value given transmission power and spectrum congestion levels. In table 5.6 the values from figure 5.7 is found.

Power level/Congestion	2%	4%	6%
−60dBm/Hz	92.41	91.65	92.93
−75dBm/Hz	57.57	50.21	46.07
−90dBm/Hz	15.31	4.19	0

Table 5.6: Transmissions found

Here, the average percentage for each power level at each duty cycle scenario is seen. These shows that the high powered transmissions are more likely to be found at the high duty cycle scenario and the probability for finding $-75dBm$ and $-90dBm$ transmission declines as the duty cycle increases.

The reason for this, is found in the way power levels are segmented. When the spectrograms are segmented, they are split into chunks in time. If a high powered transmission exists in the same time frame as a low powered transmission, the low powered transmission will not be assigned to the cluster with the highest power and in turn not be considered by the parameter extraction algorithm.

From figure 5.7 the average error in transmission rate can be found for the three power levels at 2%, 4% and 6% duty cycle. A reliable estimate of the $-60dBm$ source can be expected however as soon as the power level decreases, the estimate becomes less reliable. To understand how the parameter estimation error translates to deviation in real-world interference probability it is necessary to perform field tests where the transmission success rate is measured under different interference scenarios. This is outside the scope of this thesis, however in section 5.5 the percentage of interference located in a real-world spectrum is evaluated to better understand the possible parameter estimation error.

In figure 5.8 a signal segmented into power layers is seen. At ≈ 1.75s and ≈ 868.1MHz a high powered transmission causes the two transmission at ≈ 868.475MHz to be clustered into the second highest cluster.

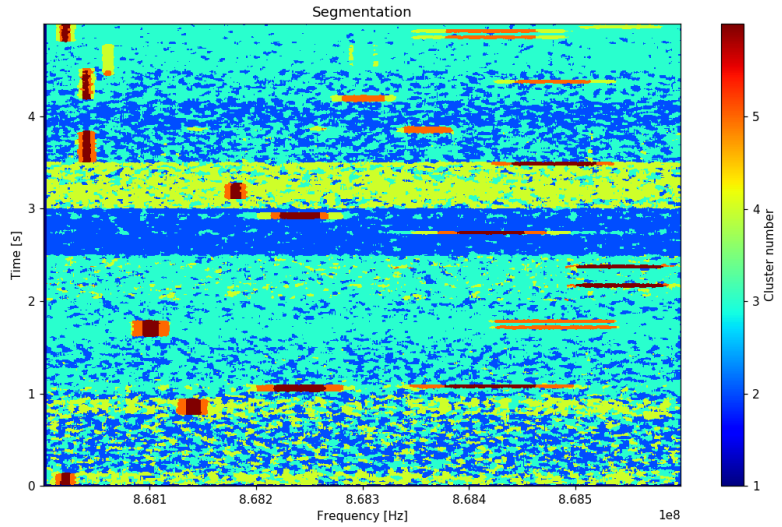

Figure 5.8: Power level segmentation

Since the two transmissions at ≈ 868.475MHz are clustered into the second highest cluster, they are not considered by the parameter extraction algorithm, which is seen in figure 5.11 where they are not marked. Examples like this can be found several times in these figures.

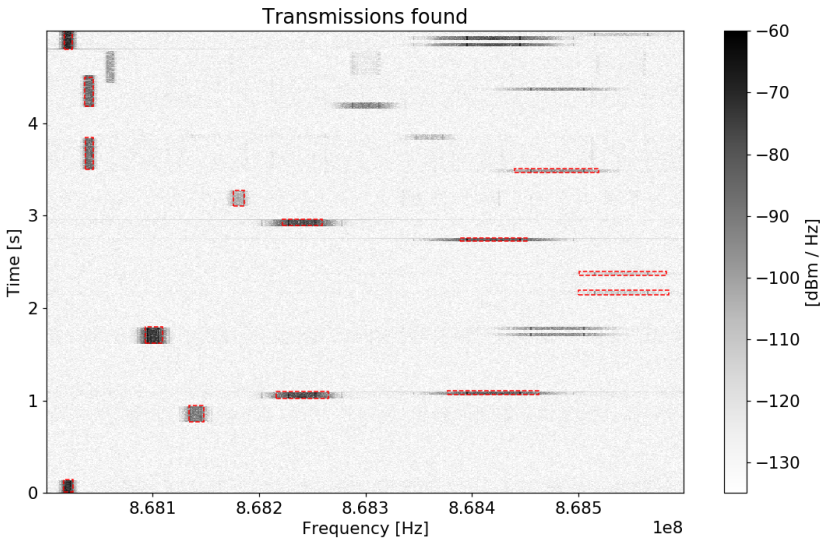

Figure 5.9: Transmission found

In a more congested spectrum it becomes increasingly unlikely that transmissions with low power is not considered in the same time frame as a higher powered transmission, which is seen in table 5.3, 5.4 and 5.5. To mitigate this, the spectrum can be divided into several sub-spectra. One strategy for

dividing the spectrum while minimizing the probability of dividing transmissions into several parts is to separate the spectrum along a frequency with a low probability interference transmission to appear.

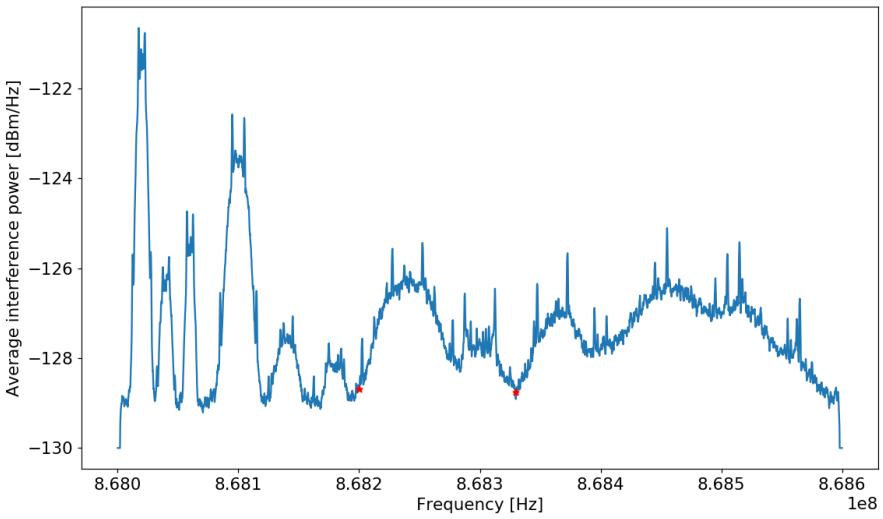

Figure 5.10: Average interference power along all frequencies

In figure 5.10, the average interference power level along a given frequency is seen. Here two frequencies(Marked red) are chosen to divide the spectrogram. These are both close to being a local minima, chosen conveniently at $f = 868.2$MHz and $f = 868.33$MHz

The divided spectrogram can be seen figure 5.11.

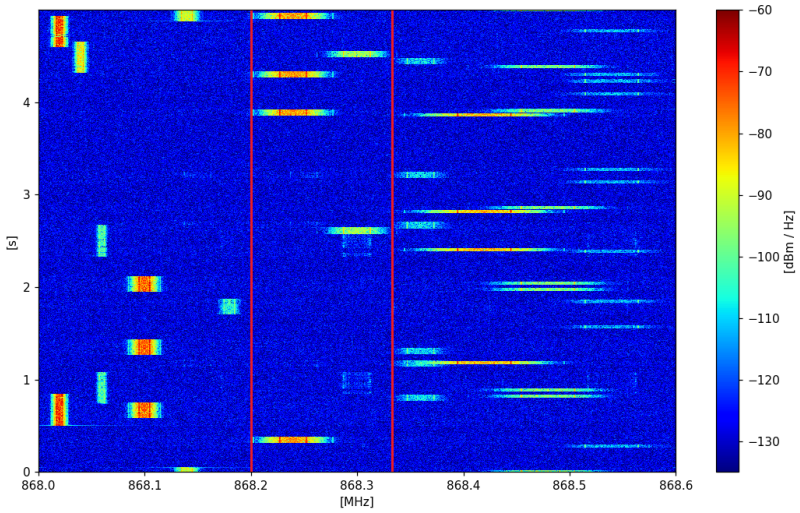

Figure 5.11: Spectrogram segmented

If the spectrogram is divided into three parts and the algorithm is run individually on these, the low powered transmission are more likely to be found, as seen in table 5.7.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]	% found
1	868.020	340	5.000	-60	92.86
2	868.040	340	5.000	-75	82.14
3	868.060	340	5.000	-90	20.69
4	868.100	170	10.000	-60	97.87
5	868.140	170	10.000	-75	78.95
6	868.180	170	10.000	-90	22.92
7	868.240	68	25.000	-60	97.01
8	868.300	68	25.000	-75	92.19
9	868.360	68	25.000	-90	26.58
10	868.420	34	50.000	-60	63.41
11	868.480	34	50.000	-75	57.24
12	868.540	34	50.000	-90	17.27

Table 5.7: Interference transmission clustering results with 4% duty cycle on several sub spectrums

In the 4% duty cycle test without spectrogram segmentation, three out of four of the $-90dBm$ transmissions are found but when the spectrogram is segmented, all of the $-90dBm$ interference sources are found. For nearly all sources, more transmissions are found except for transmission 10, where the percentage of found transmission declines.

Hence, it is shown that the fraction of low powered transmissions being found by the algorithm, relates the the amount of high powered transmission in the spectrum.

5.4 Algorithm test - Parameter clustering test

When estimating the average interference transmission length used in the ALOHA interference model, discussed in section 2.4, it is important to make a reliable estimate of the transmission length of each individual transmission source. Also, it is important to know which frequency range a transmission source occupies, which can be calculated from the center frequency f_c and the transmission bandwidth BW. To quantify the error of these estimates, the interference sources in the 4 test signals described in section 5.1 is is found by clustering as discussed in 3.2 and 3.4.2 and the error of the estimated parameters is found.

The transmission parameters for each transmission found is stored in a vector. The vectors consisting of f_c, BW and T, for each of the duty cycle levels is clustered separately using the DBSCAN clustering algorithm.

Transmission parameters clustering - Duty cycle 2%

In figure 5.12 a spectrogram of the first 20s of the 2% duty cycle test signal is seen.

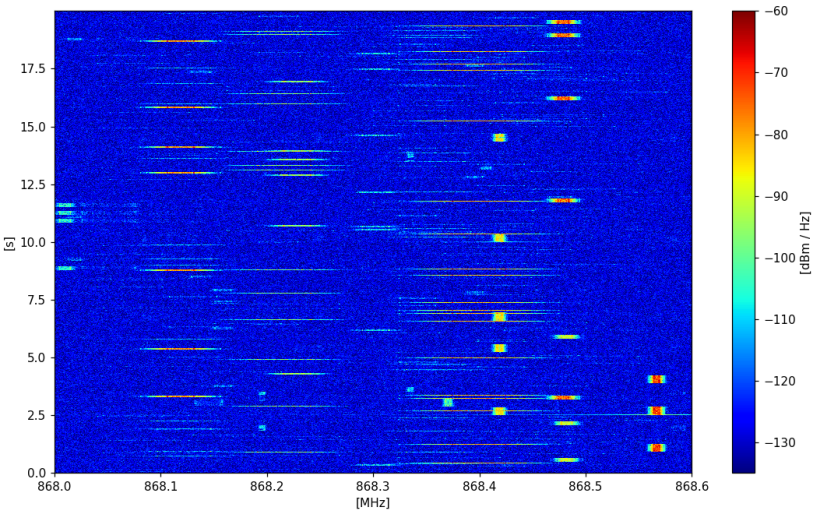

Figure 5.12: Spectrogram of test signal with 2% duty cycle.

In figure 5.13 the extracted transmission parameters are seen. The blue data points are classified as noise and the clusters found are each marked by a color. Some colors repeat, even though they represent separate clusters. This is intentional, to use a small set of easily distinctive colors, instead of using more nuances which could look alike.

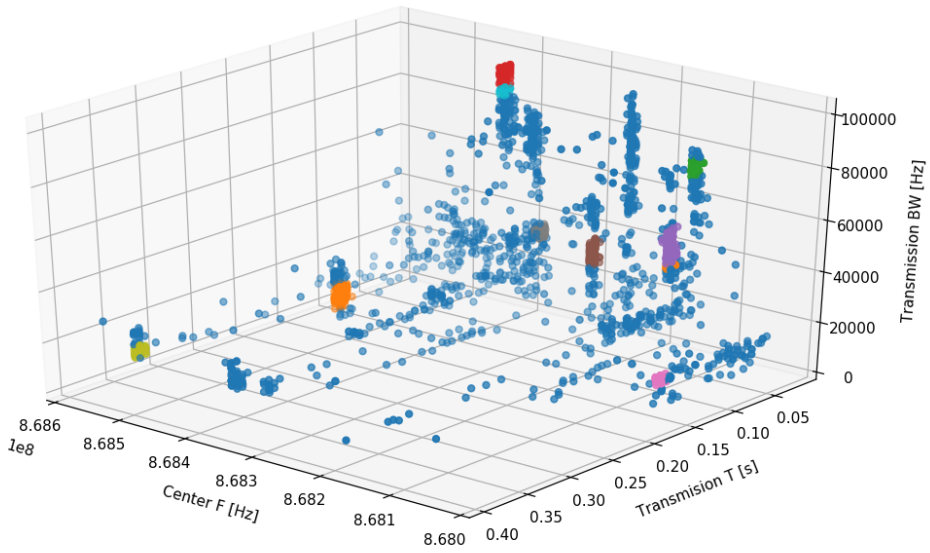

Figure 5.13: DBSCAN Clustering results, 2% duty cycle.

The clusters seen in figure 5.13 is compared to the actual transmitted interferers in table 5.8. It is seen that for each transmitted cluster, one or more similar clusters are found except for a single transmission source with power -90dBm. Some transmitted signals, such as transmission 4 yields several clusters. This happens when the estimated bandwidth of the transmissions has a large variance. The errors in the estimated parameters are seen to be lower than what is found in section 4.3. The average absolute error in percentage for the transmitted clusters found is 0.0047% for f_c, 6.99% for T and 71.92% for BW. Compared to the estimates in section 4.3 the BW estimate is further from true value while the center frequency is still close to the true value. The transmission time estimate has improved significantly compared to the estimates in section 4.3.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]
Transmission 1	868.010	170.0	10.00	-90.0
Found cluster:	868.012	176.0	17.69	-
Transmission 2	868.119	68.0	25.00	-60.0
Found cluster:	868.119	74.1	49.10	-
Transmission 3	868.122	34.0	50.00	-90.0
Found cluster:	868.123	44.2	75.80	-
Transmission 4	868.217	34.0	50.00	-75.0
Found cluster:	868.217	41.8	84.41	-
Found cluster:	868.217	41.0	79.89	-
Found cluster:	868.217	41.3	72.62	-
Transmission 5	868.229	68.0	25.00	-75.0
Found cluster:	868.229	78.1	60.00	-
Found cluster:	868.229	74.2	40.99	-
Transmission 6	868.303	68.0	25.00	-90.0
Found cluster:	868.303	77.8	44.58	-
Transmission 7	868.371	340.0	5.00	-90.0
Transmission 8	868.400	34.0	50.00	-60.0
Found cluster:	868.400	42.0	93.72	-
Found cluster:	868.400	41.4	84.75	-
Transmission 9	868.419	340.0	5.00	-75.0
Found cluster:	868.420	343.0	10.33	-
Transmission 10	868.479	170.0	10.00	-60.0
Found cluster:	868.481	174.1	19.56	-
Transmission 11	868.482	170.0	10.00	-75.0
Found cluster:	868.482	177.8	27.86	-
Transmission 12	868.567	340.0	5.00	-60.0
Found cluster:	868.568	342.8	10.99	-

Table 5.8: Interference transmission clustering results with 2% duty cycle

Transmission parameters clustering - Duty cycle 4%

In figure 5.14, a spectrogram of the first 20s of the 4% duty cycle test signal is seen.

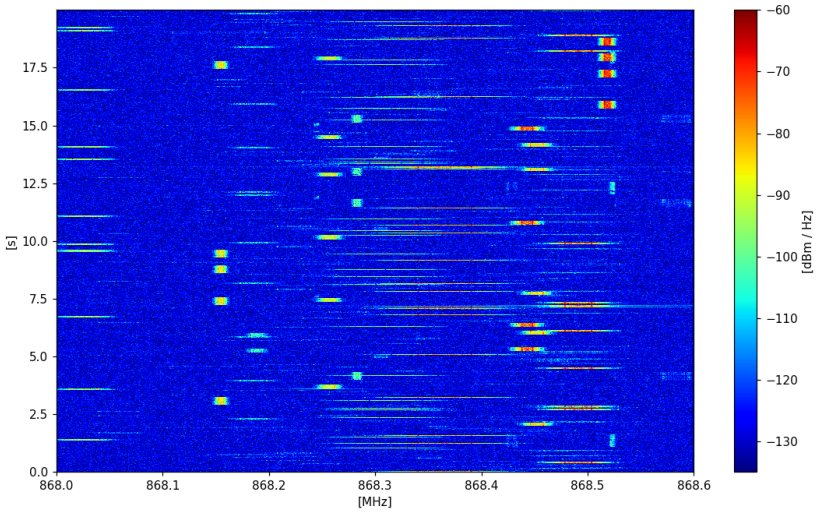

Figure 5.14: Spectrogram of test signal with 4% duty cycle.

In figure 5.15 the extracted transmission parameters are seen. Again, the blue data points are classified as noise and the clusters found are each marked by a color.

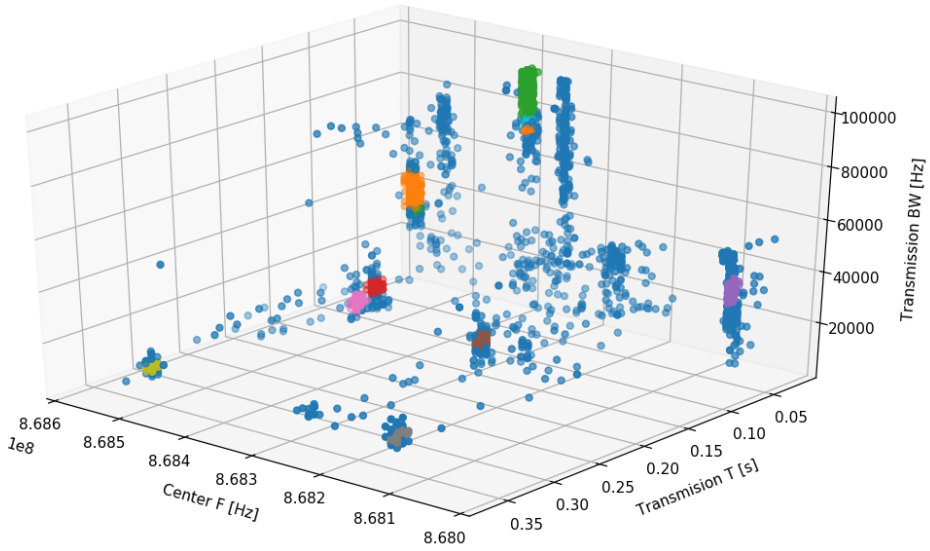

Figure 5.15: DBSCAN Clustering results, 4% duty cycle.

The clusters seen in figure 5.15 are compared to the actual transmitted interferers in table 5.9.

It is seen that no clusters for sources transmitted with transmission power −90dBm are found. For the transmissions with power level −75dBm and −60dBm, one or more similar clusters are found. As with the 2% test some transmission yields several clusters which still is due to the large variance of the bandwidth estimate. The errors in the estimated parameters are seen to be lower than what is found in section 4.3 and the previous test. The average absolute error in percentage for the parameters estimated by the clusters are 0.0079% for f_c, 8.56% for T and 81.14% for BW.

The center frequency is still accurate, but the transmission time estimate is closer to the true value compared to the 2% test and the estimates in 3.5. The reason for this, is that the power levels for detected transmissions are on average much higher, since less low power transmission are detected. When the power levels are higher, more of the sidelobes will be belong to the cluster which is used for transmission extraction.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]
Transmission 1	868.026	68.0	25.00	-75.0
Found cluster:	868.027	73.4	38.73	-
Found cluster:	868.029	77.6	52.57	-
Transmission 2	868.155	340.0	5.00	-75.0
Found cluster:	868.156	340.7	9.33	-
Transmission 3	868.186	68.0	25.00	-90.0
Transmission 4	868.189	170.0	10.00	-90.0
Transmission 5	868.257	170.0	10.00	-75.0
Found cluster:	868.258	173.1	16.90	-
Transmission 6	868.283	340.0	5.00	-90.0
Transmission 7	868.310	34.0	50.00	-75.0
Found cluster:	868.310	42.3	75.85	-
Transmission 8	868.366	34.0	50.00	-60.0
Found cluster:	868.366	41.1	91.51	-
Found cluster:	868.366	40.9	76.55	-
Found cluster:	868.366	41.4	80.00	-
Transmission 9	868.444	170.0	10.00	-60.0
Found cluster:	868.444	172.8	19.72	-
Transmission 10	868.484	34.0	50.00	-90.0
Transmission 11	868.492	68.0	25.00	-60.0
Found cluster:	868.492	74.4	49.11	-
Found cluster:	868.492	69.9	40.64	-
Transmission 12	868.519	340.0	5.00	-60.0
Found cluster:	868.519	341.4	10.92	-

Table 5.9: Interference transmission clustering results with 4% duty cycle

Transmission parameters clustering - duty cycle 6%

In figure 5.16 a spectrogram of the first 20s of the 6% duty cycle test signal is seen.

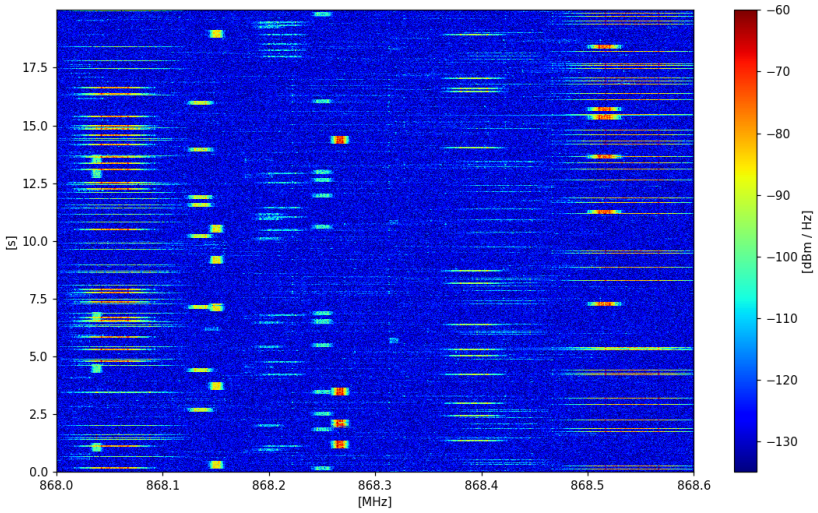

Figure 5.16: Spectrogram of test signal with 6% duty cycle.

In figure 5.17 the extracted transmission parameters are seen. Again, the blue data points are classified as noise and the clusters found are each marked by a color which repeats.

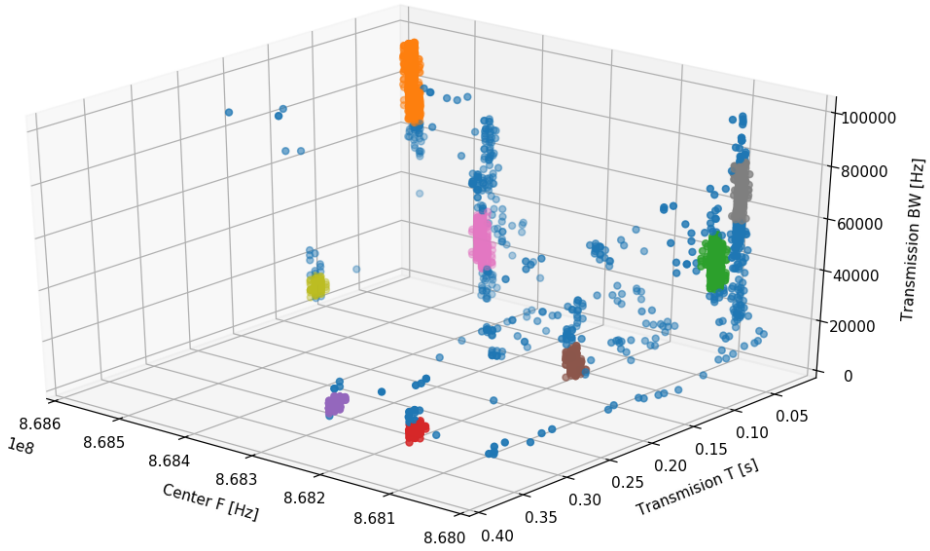

Figure 5.17: DBSCAN Clustering results, 6% duty cycle.

The clusters seen in figure 5.17 is compared to the actual transmitted interferers in table 5.10.

Similar to the previous test, no transmitted interference with transmission power −90dBm is found. For all interferers with power level −60dBm and −75dBm, one or more similar clusters are found. As with the previous tests some transmission yields several clusters which is due to the large variance of the bandwidth estimate. The errors in the estimated parameters are seen to be lower than what is found in section 3.5. The average absolute error in percentage for the transmitted parameters and the clusters found is 0.0052% for f_c, 6.03% for T and 73.57% for BW.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]
Transmission 1	868.038	340.0	5.00	-90.0
Transmission 2	868.055	68.0	25.00	-60.0
Found cluster:	868.056	72.9	47.88	-
Transmission 3	868.063	34.0	50.00	-75.0
Found cluster:	868.064	40.3	70.56	-
Transmission 4	868.136	170.0	10.00	-75.0
Found cluster:	868.136	171.6	14.92	-
Transmission 5	868.151	340.0	5.00	-75.0
Found cluster:	868.151	340.6	8.79	-
Transmission 6	868.213	68.0	25.00	-90.0
Transmission 7	868.251	170.0	10.00	-90.0
Transmission 8	868.267	340.0	5.00	-60.0
Found cluster:	868.268	340.4	10.97	-
Transmission 9	868.393	68.0	25.00	-75.0
Found cluster:	868.394	72.7	34.74	-
Transmission 10	868.421	34.0	50.00	-90.0
Transmission 11	868.516	170.0	10.00	-60.0
Found cluster:	868.517	172.7	19.91	-
Transmission 12	868.542	34.0	50.00	-60.0
Found cluster:	868.543	39.1	86.72	-

Table 5.10: Interference transmission clustering results with 6% duty cycle

5.4.1 Transmission parameter test discussion

The results in table 5.8, 5.9 and 5.10 show that the algorithm is capable of finding clusters that matches the actual interference transmissions found in the spectrogram. It is also shown that the congestion of the spectrum affects the algorithms capability to detect low powered transmission.

In table 5.11 the average parameter estimate error at each congestion level is seen.

Parameter/Congestion	2%	4%	6%
f_c	0.0047	0.0079	0.0052
T	6.99	8.56	6.03
BW	71.92	81.14	73.57

Table 5.11: Transmission parameter estimate error

It is expected that the algorithm overestimates the bandwidth of the interference transmission, since it is hard to distinguish between the actual transmission and the transmission sidelobes. Also, the

sidelobes of high powered transmissions may cause interference of desired transmissions and should be considered when the transmission rate in a frequency range is computed. The ALOHA model relies on accurate estimation of transmission time, and with an average error of $\approx 7\%$ the estimates are close to the true value. Hence, the largest error in the parameter estimates must be caused by the algorithm missing transmissions and bias the estimates towards transmission parameters from transmissions which are more likely to be detected.

5.5 Algorithm test - Aarhus dataset

A final test to evaluate the real-world performance of the algorithm is performed. The purpose of this test is to determine the percentage of interference the algorithm is capable of detecting. The accuracy of the transmission parameter estimates have been tested in section 5.4 and the transmission parameters are not available in this unlabelled dataset. Hence, only the percentage of found interference is evaluated. When modelling the interference spectrum as discussed in section 2.4, the accuracy of the model depends on the accuracy of the parameters r(Transmission rate) and T_p(Average interference transmission length). If a large fraction of the interference in a spectrum is found, the estimates of these parameters will be close to their true value.

Kamstrup has provided a set of preexisting measurements, the Aarhus dataset, a 1 hour recording of the 868MHz − 868 6MHz band in the center of Aarhus. For recording the spectrum a Signal Hound BB60C spectrum analyzer [21] is used to sample the 868MHz to 868.6MHz frequency range with center frequency 868.3MHz. The received signal is mixed down to 0.6MHz and sampled into I/Q samples with a rate of $f_s = 10$MHz.

A spectrogram of the first 2 seconds of the Aarhus dataset signal i seen in figure 5.18.

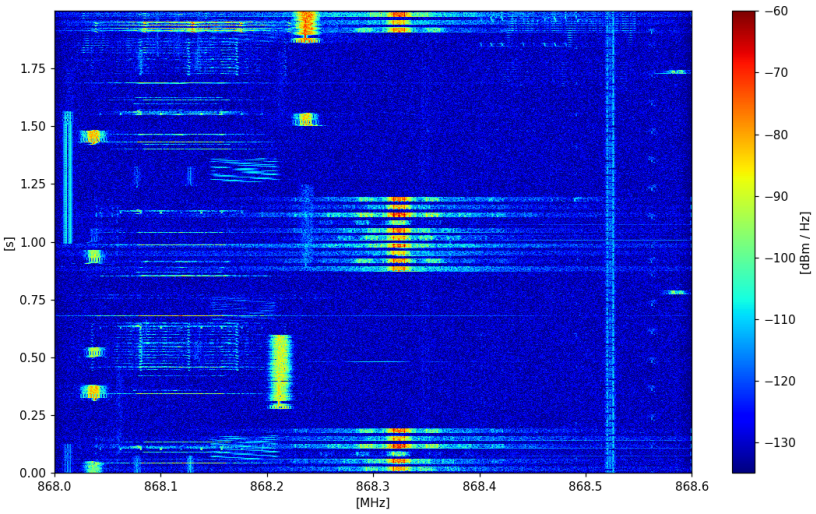

Figure 5.18: Aarhus dataset signal spectrogram, first 2 seconds.

As with the other test signals, power levels segmentation is performed. Power level segmentation of the first 2 seconds is seen in figure 5.19.

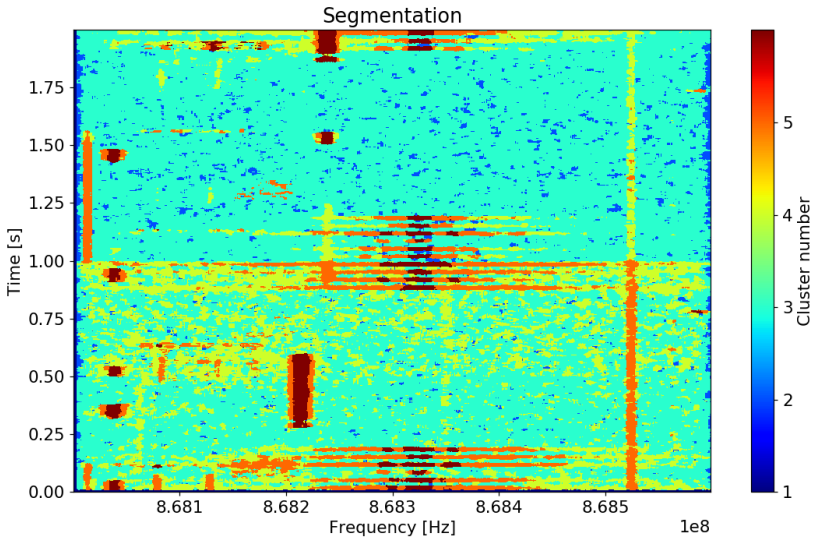

Figure 5.19: Aarhus dataset signal power level segmentation, first 2 seconds.

After the power level segmentation, transmissions are located and the transmission parameters are extracted. In figure 5.20 the transmissions located in the first 2 seconds can be seen.

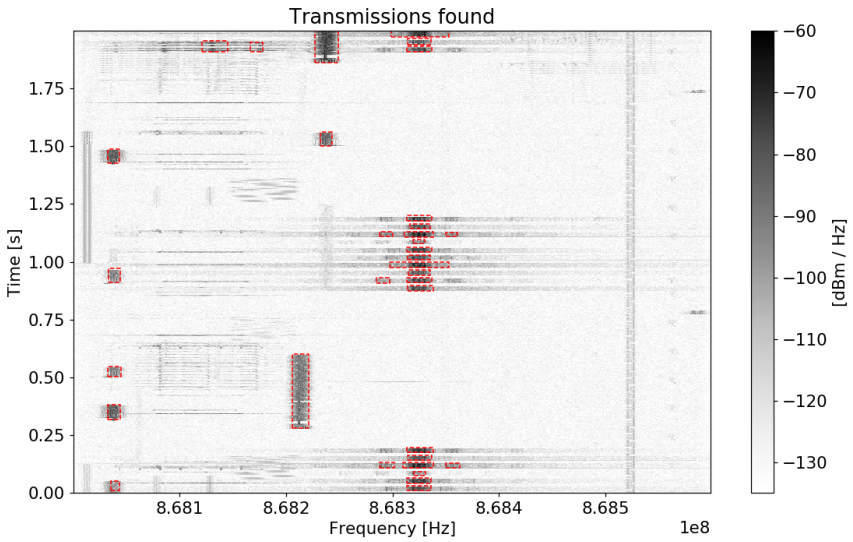

Figure 5.20: Aarhus dataset signal located transmissions, first 2 seconds.

Finally, the transmission parameters are clustered to find clusters of repeating transmissions. This is seen in figure 5.21. Again a set of easily distinguishable colors which repeats is chosen.

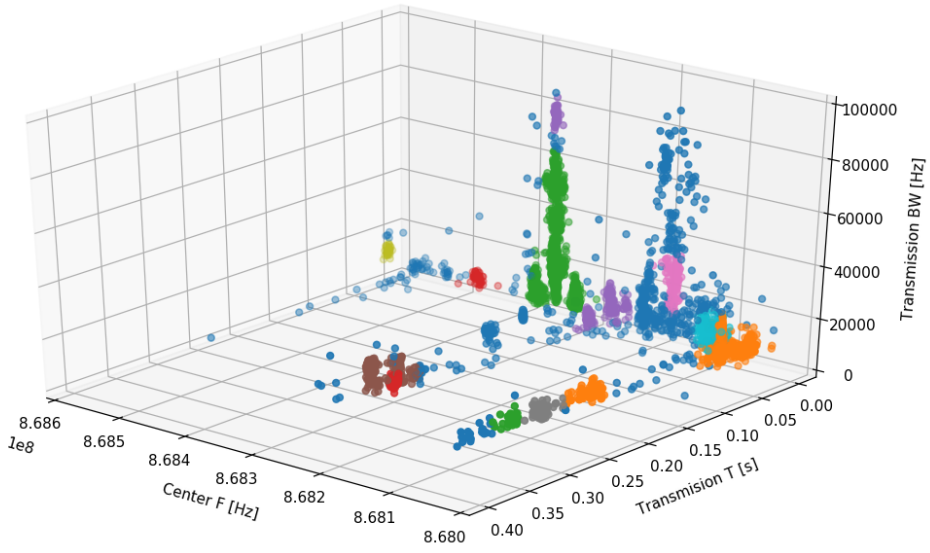

Figure 5.21: Aarhus dataset signal transmission parameters clusters.

The clusters seen in figure 5.21 is found in table 5.12 with the estimated parameters and a silhouette score.

Cluster #	f_c [MHz]	T [ms]	BW [kHz]	Silhouette score
1	868.038	57.32	14.05	0.590
2	868.038	330.57	18.30	0.396
3	868.038	292.21	17.27	0.761
4	868.038	233.56	17.13	0.550
5	868.087	29.12	13.73	−0.024
6	868.131	40.40	31.01	0.485
7	868.171	32.08	16.91	0.636
8	868.214	322.39	19.46	0.915
9	868.235	41.65	13.84	0.547
10	868.238	313.13	20.42	0.392
11	868.325	24.26	81.30	0.681
12	868.325	21.30	23.21	0.697
13	868.438	24.70	12.09	0.877
14	868.587	16.16	12.75	0.954

Table 5.12: Interference transmission clustering results

Some of the clusters in table 5.12 are easily recognized from the spectrogram in figure 5.22.

Cluster 1 and 2 found at 868.038MHz can be seen as two different transmissions(One long and one short), in figure 5.22 at $\approx$ 868.038MHz. Cluster 3 and 4 are similar to cluster 2 which could indicate that at least three different transmission times are used for that particular transmission type.

Cluster 8 at 868.214MHz is found as the long transmissions at ≈ 868.2MHz in the spectrogram. Two more clusters, 9 and 10 at ≈ 868.236MHz, is seen as a short transmission followed by a long transmission. Finally, cluster 12 captures the repeated short transmissions in the spectrogram seen at ≈ 868.33MHz.

It is difficut to verify the validity of the remaining clusters of table 5.12. Some may be transmission occurring at a later time in the signal and some may be results of noise. The test in section 5.4 shows that for simulated data, the algorithm is capable of finding the actual interference sources by clustering the interference transmission parameters. However, to verify if this is also the case for real-world spectrum recordings field tests with a known interference spectrum will be necessary.

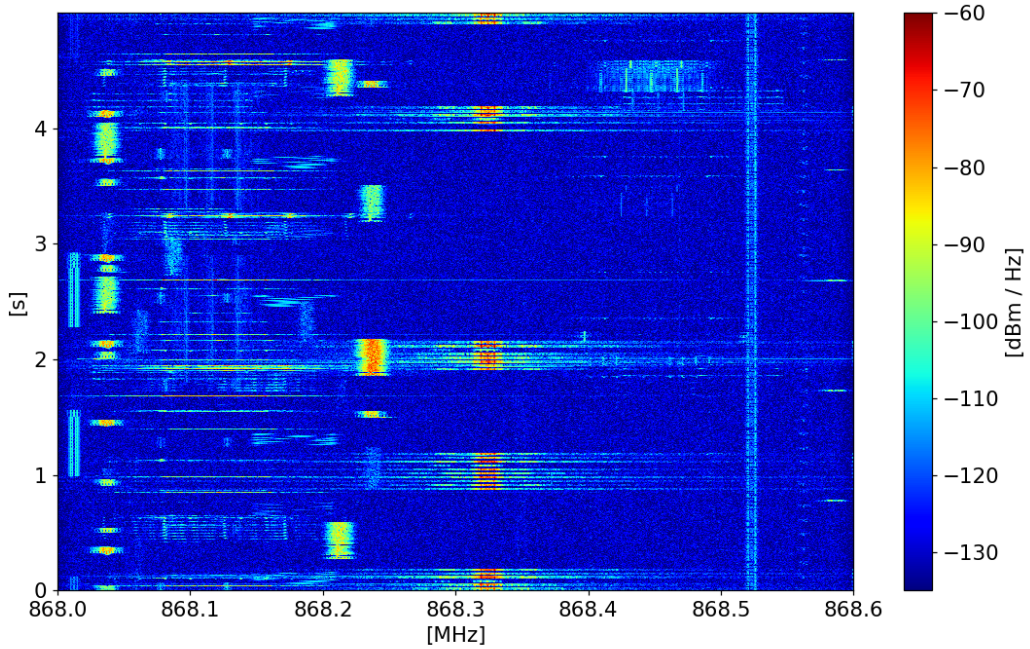

Figure 5.22: Aarhus dataset signal spectrogram, first 5 seconds.

Since the accuracy of the transmission parameters relies on the fraction of transmissions found, the percentage of found interference is calculated. Since the dataset is unlabelled and the actual number of transmissions is unknown, a range of thresholds $[-120, -60]$ is used to quantify the percentage of transmissions found. The percentage is calculated as the number of interference units above the threshold found by the algortihm divided by the number of interference units above the threshold present in the spectrogram. In figure 5.23 the percentage of interference found for each threshold value is seen. Also, the number of interference units in the spectrogram above the threshold is seen. This is included to explain the sharp decline in found interference after $\approx -65 dBm/Hz$ since almost no interference units exists above this threshold the estimate goes towards zero.

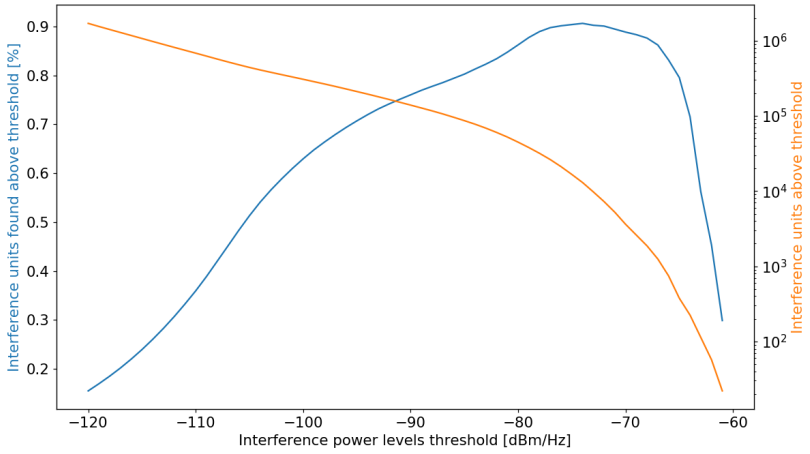

Figure 5.23: Aarhus dataset signal - Percentage of interference found by clustering.

In table 5.13, a range of results from figure 5.23 is seen. The results cannot be directly compared to the results in section 5.3, however it is seen that already at $-100dBm/Hz$ more than half of the interference transmissions above that threshold are found. Considering only interference above -90dBm/Hz, the transmission rate estimate for these transmissions are likely to be $\approx 76\%$ of the true value. However, as discussed in section 5.3 the difference in real-world transmission success compared to the ALOHA model with estimated parameters needs to be evaluated by fields tests.

Power threshold [dBMm/Hz]	Interference units found [%]	# interference units in spectrogram
-120	15.48	1702614
-110	35.93	683958
-100	62.95	307257
-90	76.04	139297
-80	86.25	44765
-70	88.88	3620

Table 5.13: Percentage of interference units found

Looking at figure 5.24 where the transmissions for the first 25 seconds of the signal is marked, it is clear that a large fraction of the transmissions is indeed found as indicated by the results in table 5.13. Some of the low power transmissions, for example at ≈ 868.53MHz, are not located as seen in figure 5.23. This is expected when considering the results in section 3.5 which shows that low powered transmissions existing in the same time frame as high power transmissions are likely to be missed by the algorithm.

Appendix A shows $5x25$s of the Aarhus dataset, for a longer time period. Here a large fraction of the interference transmissions are marked as found. Appendix B shows a more detailed version of the Aarhus dataset to illustrate how well the markings matches the actual transmissions. This is a prerequisite for accurate estimates of the center frequency, transmission time and transmission bandwidth.

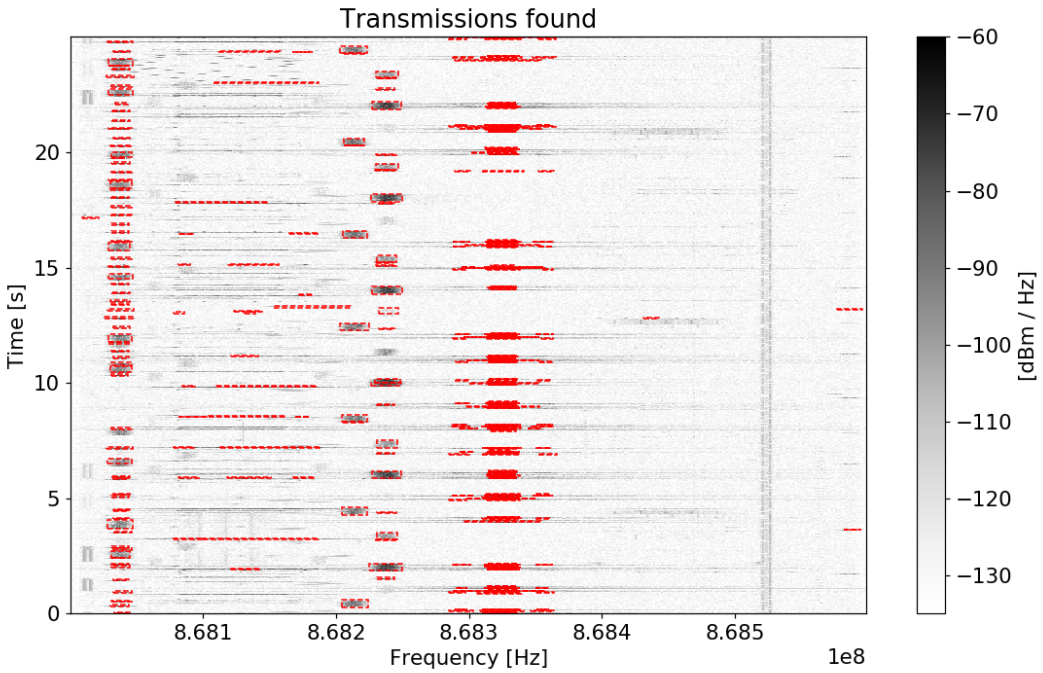

Figure 5.24: Aarhus dataset signal transmissions found.

5.6 Comparision with prior work

To evaluate the results obtained in the previous tests, an approach similar to the one used by German researchers [8] is tested. In this paper, a threshold for activity in the spectrogram is defined, and everything above the threshold is classified as interference. A method similar to the approach discussed in section 3.4.2 is used to find the outline of individual transmissions on the spectrogram and the individual transmission parameters are clustered to identify the main interference sources.

The test signal used is the 4% duty cycle signal from section 5.3. Here 12 interference transmissions with either -60dBm, -75dBm or -90dBm populates the spectrum.

The paper does not mention how to find the threshold used to detect activity in the spectrum. This threshold is not trivial to find since a threshold too low will result in random noise being identified as interference transmissions and a threshold too high will result in interference transmissions not being found. The spectrogram segmentation algorithm in section 3.2 avoids this by finding the clusters in the received interference spectrum.

For now, a threshold of -115dBm/Hz is used since the power level of the interference sources are known, and this threshold can completely separate the noise from the interference sources. This is to show the best case performance of the algorithm. However, for this algorithm to be used on real-world data, it is necessary to find a method for estimating the detection threshold.

Furthermore, the paper does not mention a clustering algorithm, hence it is assumed that the clustering is done manually. If this is the case, the algorithm is hard to scale, and for spectra with high interference activity, it is a cumbersome process. Looking at figure 5.25 the local density of the samples can be hard to determine with only visual inspection since some of the samples are either close to each other or directly on top of each other. Since no clustering method is mentioned and due to the possible errors in manually clustering the samples, the DBSCAN clustering algorithm is used in this test. The parameters for the DBSCAN algorithm is the same as used in the other tests. By using this algorithm, this test will only show the effect of using a predetermined static threshold against an adaptive approach as used in section 3.2. Without exact information on how the clusters in the paper are obtained it is difficult to make a better comparison of the algorithms.

In figure 5.25 the clusters obtained by using a fixed -115dBm/Hz threshold, marking the interference transmissions and clustering them is seen.

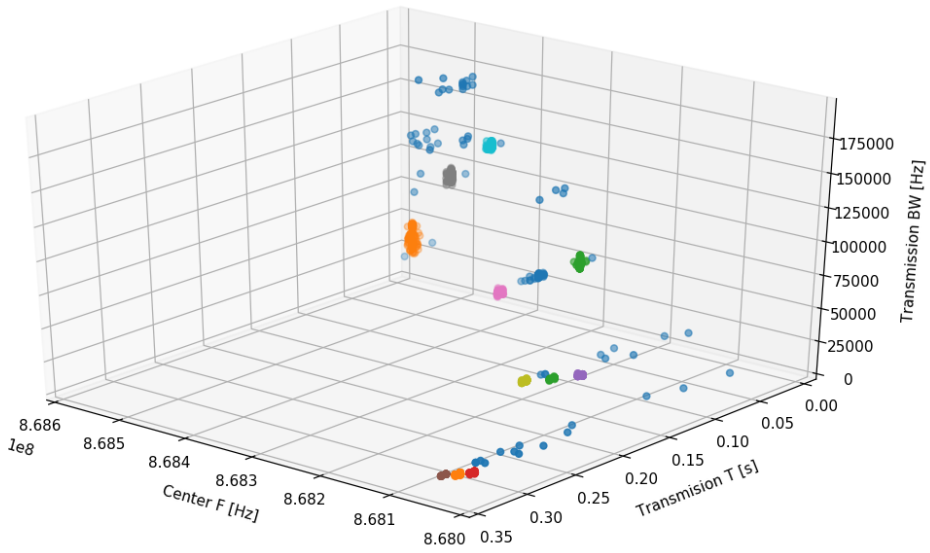

Figure 5.25: Interference transmissions clustered.

In table 5.14 the clusters from figure 5.25 with estimated parameters, f_c, T and BW for each cluster as well as transmission power and fraction of transmissions found is seen. The fraction of transmissions which is found exceeds what is found in section 5.3, which is expected when the threshold is set lower than the transmission power of the interference transmissions. The average error is $< 1\%$ for f_c and 11% for transmission time. This is similar to the results seen in section 5.4. The results in table 5.14 shows that the bandwidth estimate depends on the transmission power of the interference transmissions and consequently will also depend on the interference activity detection threshold. The choice of threshold will not only determine how many of the interference transmissions are found, but it will also determine the bandwidth estimate error. Furthermore, the bandwidth estimates will be highly location dependent. As an example, if the spectrum is recorded in two different places, any path loss due to terrain, buildings etc. will alter the received transmission power and the estimated bandwidths will be different for the two locations. Hence, the estimated parameters for the interference model will only be valid close to where the recordings are made and will not provide a general interference model for the area.

Transmission #	f_c [MHz]	T [ms]	BW [kHz]	Power level [dBm]	% found
Transmission 1	868.020	340.0	5.00	-60.0	-
Found cluster:	868.020	332.5	16.65	-	0.93
Transmission 2	868.040	340.0	5.00	-75.0	-
Found cluster:	868.040	331.6	12.52	-	0.89
Transmission 3	868.060	340.0	5.00	-90.0	-
Found cluster:	868.061	331.5	9.00	-	0.79
Transmission 4	868.100	170.0	10.00	-60.0	-
Found cluster:	868.100	161.6	30.98	-	0.98
Transmission 5	868.140	170.0	10.00	-75.0	-
Found cluster:	868.140	161.6	23.00	-	0.97
Transmission 6	868.180	170.0	10.00	-90.0	-
Found cluster:	868.180	161.0	15.68	-	0.92
Transmission 7	868.240	68.0	25.00	-60.0	-
Found cluster:	868.240	60.1	71.39	-	0.91
Transmission 8	868.300	68.0	25.00	-75.0	-
Found cluster:	868.300	60.1	53.97	-	0.91
Transmission 9	868.360	68.0	25.00	-90.0	-
Found cluster:	868.360	58.9	33.03	-	0.81
Transmission 10	868.420	34.0	50.00	-60.0	-
Found cluster:	868.421	25.7	129.71	-	0.78
Transmission 11	868.480	34.0	50.00	-75.0	-
Found cluster:	868.480	25.5	99.93	-	0.80
Transmission 12	868.540	34.0	50.00	-90.0	-
Found cluster:	868.541	24.9	45.12	-	0.70

Table 5.14: Interference transmission clustering results with 4% duty cycle

Discussion and Conclusion 6

6.1 Discussion

When the resolution of interference units in the spectrogram, Δt and Δf, is close to the transmission parameters T and BW, the Binomial distribution seen in section 2.3 can be used to model interference probability. When these conditions are satisfied, the model assumes that the transmissions occur independently in time and frequency, which for a completely unknown spectrum with several independent sources can be an appropriate assumption. However, when Δt and Δf are lower than the transmission parameters and the transmissions cover several interference units, the assumption does no longer hold. Neighbouring interference units will have a high probability for being close in value, which is not accounted for in the Binomial model. Hence, the model is not inherently bad, but the spectrogram resolution Δt and Δf has to be chosen carefully.

The pure ALOHA model used in this thesis assumes that all interference transmissions are transmitted by a finite number of sources, which transmit with a fixed center frequency, transmission time and bandwidth. It assumes that all interference sources transmits at random times with a given average transmission rate in $transmissions/second$. This may not always be true, since some of the transmissions can be two-way communication between two or more units. For example, a gateway in a network can acknowledge the transmissions sent by the individual nodes, or request data from the nodes[1]. If this happens, the nodes will communicate on request of the initiator, hence they are no longer independent of each other. Several such scenarios can be constructed where the interference sources in a spectrum no longer transmits independently. However, it is assumed that most units using the spectrum will not be aware of other units and, operate independently. As stated in [12], pure ALOHA spectrum access is a common model for such a communication channel which is the reason why this model is chosen.

When segmenting the received power levels for finding the transmissions, a straightforward approach would be to use a set threshold as shown by German researchers [8]. The reason for not choosing this approach, is that low powered interference transmissions can have the same maximum power levels as the sidelobes of a high powered transmission. This will cause the bandwidth estimate of the high powered transmissions to become much larger than a comparable low powered transmission. To mitigate this the threshold can be set to only consider the high powered transmissions which will greatly affect the the estimated parameters for the ALOHA model. However, if a spectrum recording is made locally to evaluate the interference in a wider area not only locally where the recording is made, the high powered transmissions can be low powered transmission(Due to path loss) somewhere else. This is shown in section 5.6 where a static threshold is used to detect interference transmissions. Here a greater fraction of the interference transmissions are found, compared to the tests in section 5.3, but the method require that a static threshold can be determined which the paper does not provide.

[1] This could be any scenario where two units communicate over a wireless channel.

To obtain a model for interference in a area, regardless of location(Which translates to power level) it is important to only consider the main part of the transmission when evaluating the bandwidth and avoid as much as the sidelobes as possible.

The estimated parameters of the ALOHA model will result in an interference probability for a frequency range and transmission time both specified by a desired transmission. In section 5.3, the fraction of transmissions found by the algorithm is evaluated. As discussed in that section, finding 90% of the transmissions in a given frequency range will result in an estimated transmission rate which is 90% of the true value. For a real spectrum, it is impossible not to miss some of the interference transmissions and the estimates of both transmission rate and average transmission time for the interference transmission will always have some errors. The error in the estimates will be highest for the low powered transmissions which implies that the model will be most accurate close to the receiver.

The transmission rate and average transmission time used in the ALOHA model varies with the parameters of a desired transmission. When the bandwidth or center frequency changes, both parameters will change since another set of interference transmissions will occupy the frequency range used to transmit the desired transmission. This results in a large number of permutations for even a small set of transmission parameters. In the tests, either the fraction of located transmissions is calculated or the error in estimating transmission parameters is found. If a large fraction of transmissions is found and the individual transmission parameters are estimated close to their true value, the parameters in the ALOHA model can be estimated with a small error. The model parameters has not been calculated directly for any of the tests, but in section 5.3, any desired transmission covering only a frequency range with one interference source will have a transmission rate estimate error corresponding to the fraction of transmissions found. The error in average transmision time, can be found from the parameter estimate error in section 5.4. Instead of calculating the ALOHA parameter estimate errors for a small number of arbitrary transmissions, the underlying parameter estimate errors have been highlighted in the results in the test chapter.

In a known interference scenario ,the ALOHA parameter estimates can be calculated and compared to the true ALOHA parameters. If the calculated interference probability is compared to the real-world packet error rate for a system transmitting in a spectrum with the known interference, the implications of the parameter estimate error can be evaluated. To fully understand this relation between the estimate errors and the deviation in interference probability, from model to the real world, a test campaign with known interference spectrum and a transmitter/receiver should be conducted. Due to limited laboratory access at the time of writing this thesis, this has not been performed.

6.2 Conclusion

This thesis aims to answer the following problem statement:

- *How can probability for interference be modelled and how can the model parameters be estimated?*
- *How can intelligent learning algorithms be used to find, separate and cluster similar transmissions in a spectrum recording to identify the parameters needed for modelling interference?*

Chapter 2 aims to answer the first part of the problem statement. Here the pure ALOHA spectrum access model is used to calculate the probability for interference in a wireless channel. It is shown how to find the model parameters, the transmission rate and transmission time in a frequency range specified by the parameters from the desired transmission. Furthermore, an example of choosing the optimal transmission parameters by using the model is presented and the result is verified by use of a brute-force interference probability algorithm. The second part of the problem statement is treated in chapters 3, and 4. Afterwards, the performance of the algorithm is evaluated in chapter 5.

By assuming noise in the wireless channel as AWGN [2][12], a signal model for the received signal with and without interference present is constructed. Spectrograms are calculated from recordings of interference in the 868.0MHz − 868.6MHz frequency range and the received power at each time and frequency index with resolution Δt and Δf is computed. From the signal model it is clear that the Gaussian noise in the spectrogram will be Rayleigh distributed and interference sources in the the channel, with additive Gaussian noise, will be Rice distributed. To segment the received power in the spectrogram into clusters of either noise or one of several possible interference sources, two methods, K-Means and Gaussian mixture models, have been presented, implemented and tested.

When inspecting the histograms computed from the spectrograms, the findings matches the expected distribution from the signal model. A significant part of the interference levels is Rayleigh distributed and several Ricians with different means, due to interference transmission with different power, appear in the histogram. The Rice distribution with mean $v \neq 0$ will have a shape similar to a Gaussian, and the Gaussian mixture model also proves to be the best in separating the noise from interference transmissions given the spectrogram resolution is sufficient. To use the ALOHA model, good estimates of parameters are needed. To obtain these, the algorithm should be able to locate a large fraction of the transmissions in a spectrogram. In table 6.1 the fraction of transmissions located with different power levels and spectrum congestion settings is seen. The fraction of found transmissions is lower for power levels −75dBm/Hz and −90dBm/Hz compared to the reference test in section 5.6 where a method proposed by German researchers [8] is tested.

Power level/Congestion	2%	4%	6%
−60dBm/Hz	92.41	91.65	92.93
−75dBm/Hz	57.57	50.21	46.07
−90dBm/Hz	15.31	4.19	0

Table 6.1: Transmissions found

To extract the transmission parameters from a segmented spectorgram, an algorithm to find neighbouring clusters of non-zero entries in the spectrogram is developed. With this it is possible to find the center frequency, transmission time and bandwidth parameters of an interference transmission.

[2]Additive white Gaussian noise

To find similar transmissions, likely emitted by the same source, the samples containing the extracted transmission parameters are clustered. The DBSCAN clustering algorithm proved to be superior compared to the Gaussian mixture models when more noise samples existed in the samples. It is desired to minimize the parameter estimate error, which is seen in table 6.2. The closer these estimates are to the true value, the closer the ALOHA parameter estimates are to their true value.

The results in table 6.2 shows that the algorithm proposed in this thesis provides interference parameter estimates which is better than the estimates obtained in the reference test in section 5.6.

Parameter/Congestion	2%	4%	6%
f_c	0.0047	0.0079	0.0052
T	6.99	8.56	6.03
BW	71.92	81.14	73.57

Table 6.2: Transmission parameter estimate error

With the results in table 6.1 and 6.2, it can be concluded that for high powered transmissions the algorithm is capable of making reliable estimates for the transmission rate. For the low/mid power transmission, the estimate depends highly on how congested the spectrum is. It is seen that when the spectrum becomes less congested, the fraction of transmissions located rises. Furthermore, it can be concluded that for the interference sources found, both center frequency and transmission time can be estimated close to their true value where the bandwidth is overestimated as expected.

Compared to the method proposed in [8], the algorithm proposed here finds better bandwidth estimates while the fraction of interference transmissions found is worse. If a reliable method for finding the interference activity threshold is available, it will be possible to combine the algorithm proposed in this thesis with the algorithm proposed in [8] to estimate the transmission parameters and the transmission rate respectively. This will combine the strengths of both algorithms while reducing the weaknesses.

The real world test in section 5.5 shows that a large fraction of the interference sources in a spectrum can in fact be located by the algorithm, as seen in table 5.13 and in appendix A and B where the transmissions are marked in the spectrogram. These results indicates that a large part of the dominating interference sources are in fact located by the algorithm. Furthermore, the clustering of transmission parameters locates what appears to be the most dominating interference sources. This enhances the belief that the algorithm is capable of making reliable estimates of the ALOHA model parameters which can be used to predict the interference probability for a desired transmission.

6.3 Future work

In this thesis, only the fraction of interference transmissions which is found in a spectrum and the estimate error of f_c, T and BW are evaluated.

To validate the algorithm, it is necessary to conduct real-world tests with a known interference spectrum to evaluate the algorithms ability to find the individual interference sources and estimate their parameters.

A number of IoT devices should be set-up to communicate on the 868.0MHz − 868.6MHz frequency range with fixed center frequencies, transmission times and bandwidths. The duty cycle of each transmitter should be noted, such that the true transmission rate and average transmission time of the interferers is known. A recording of the spectrum should be made with a spectrum analyzer similar to the one used for the recordings presented in this spectrum. By using the proposed algorithm to analyze the recording and estimating the source parameters, the ALOHA model parameters can be estimated and the probability for interference calculated.

During this test, two or more transmitters/receivers should communicate back and forth and the packet error rate should be logged, to calculate the probability for interference. By comparing the actual interference probability with the one estimated by the ALOHA algorithm it is possible to determine if the algorithm is suitable for extracting the model parameters and estimating the interference probability.

Bibliography

[1] ECC (Electronics Communication Comittee). THE EUROPEAN TABLE OF FREQUENCY ALLOCATIONS AND UTILISATIONS IN THE FREQUENCY RANGE 9 kHz to 3000 GHz. Technical report, ECC, 2009.

[2] SIGFOX SA. Sigfox connected objects: Radio specifications. Technical report, Accessed: 2020-02-01.

[3] LoRa Alliance. LoRaWAN™ 1.1 Specification. Technical report, Accessed: 2020-02-01.

[4] Gartner. Gartner says 5.8 billion enterprise and automotive iot endpoints will be in use in 2020, Accessed: 2020-02-20.

[5] J. Mitola and G. Q. Maguire. Cognitive radio: making software radios more personal. *IEEE Personal Communications*, 6(4):13–18, Aug 1999.

[6] L. Gavrilovska, V. Atanasovski, I. Macaluso, and L. A. DaSilva. Learning and reasoning in cognitive radio networks. *IEEE Communications Surveys Tutorials*, 15(4):1761–1777, Fourth 2013.

[7] Mads Helge Jespersen Alexander Korsvang Hagelskjær, Benjamin Højmose Grevenkop-Castenskiold. Radio signal technology classification using machine learning techniques. 2018.

[8] Gerd Kilian-Joerg Robert Albert Heuberger Hendrik Lieske, Frederik Beer. Characterisation of channel usage in ism/srd bands. 2014.

[9] J. Robert, S. Rauh, H. Lieske, and A. Heuberger. Ieee 802.15 low power wide area network (lpwan) phy interference model. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2018.

[10] Benny Vejlgaard, Mads Lauridsen, Huan Cong Nguyen, István Kovács, Preben Elgaard Mogensen, and Mads Sørensen. Interference impact on coverage and capacity for low power wide area iot networks. In *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, I E E E Wireless Communications and Networking Conference. Proceedings, United States, 2017. IEEE.

[11] S. Rauh, J. Robert, M. Schadhauser, and A. Heuberger. Lpwan occupancy model parameter identification for license exempt sub-ghz frequency bands. In *2018 11th German Microwave Conference (GeMiC)*, pages 111–114, March 2018.

[12] Andreas Molisch. *Wireless Communications*. Wiley-IEEE Press, 2005.

[13] Mads Lauridsen, Benny Vejlgaard, István Kovács, Huan Cong Nguyen, and Preben Elgaard Mogensen. Interference measurements in the european 868 mhz ism band with focus on lora and sigfox. In *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, I E E E Wireless Communications and Networking Conference. Proceedings, United States, 3 2017. IEEE.

[14] John G. Proakis and Masoud Salehi. *Communication Systems Engineering*. Prentice-Hall, second edition, August 2001.

[15] Alan V. Oppenheim, Ronald W. Schafer, and John R. Buck. *Discrete-Time Signal Processing*. Prentice-hall Englewood Cliffs, second edition, 1999.

[16] A. Croft and T. Croft. *Engineering Mathematics: A Foundation for Electronic, Electrical, Communications and Systems Engineers*. Pearson, 2017.

[17] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[18] Peter Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, 20(1):53–65, November 1987.

[19] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[20] ECC (Electronics Communication Comittee). Erc recommendation 70-03 -relating to the use of short range devices (srd), Accessed: 2020-02-01.

[21] Signalhound. Bb60c production data sheet, Accessed: 2020-04-30.

Aarhus data set - Found transmissions results A

To show how many of the transmissions are found by the algorithm, the first $5x25$s of the Aarhus dataset are shown as spectrograms with their corresponding overlay which marks transmissions found. The figures are matched page for page for better overview. For this reason, the rest of this page is left blank and the results are shown from the next page.

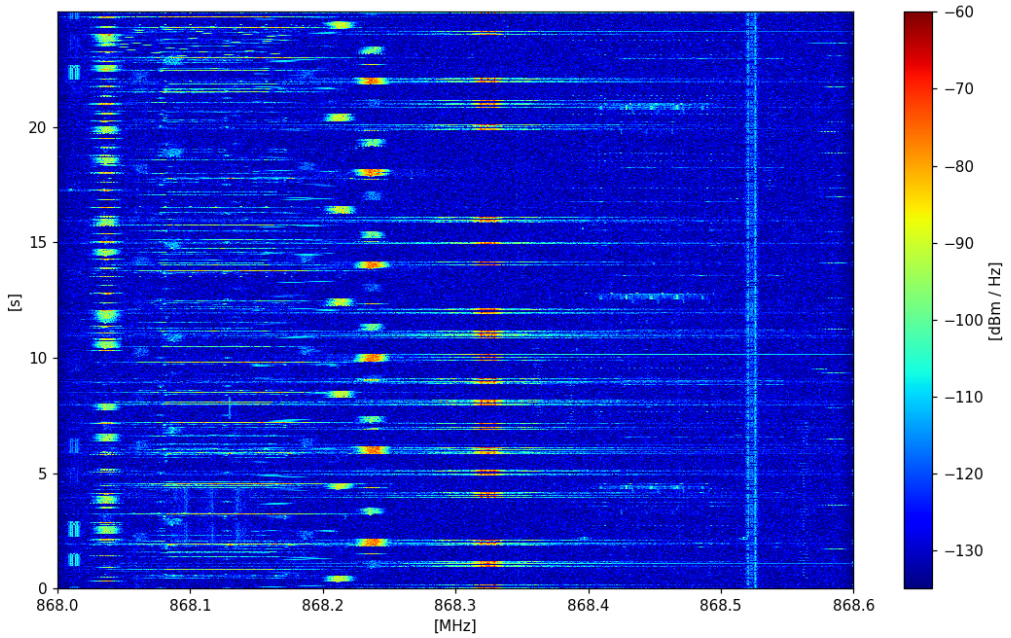

Figure A.1: Aarhus dataset signal spectrogram 1.

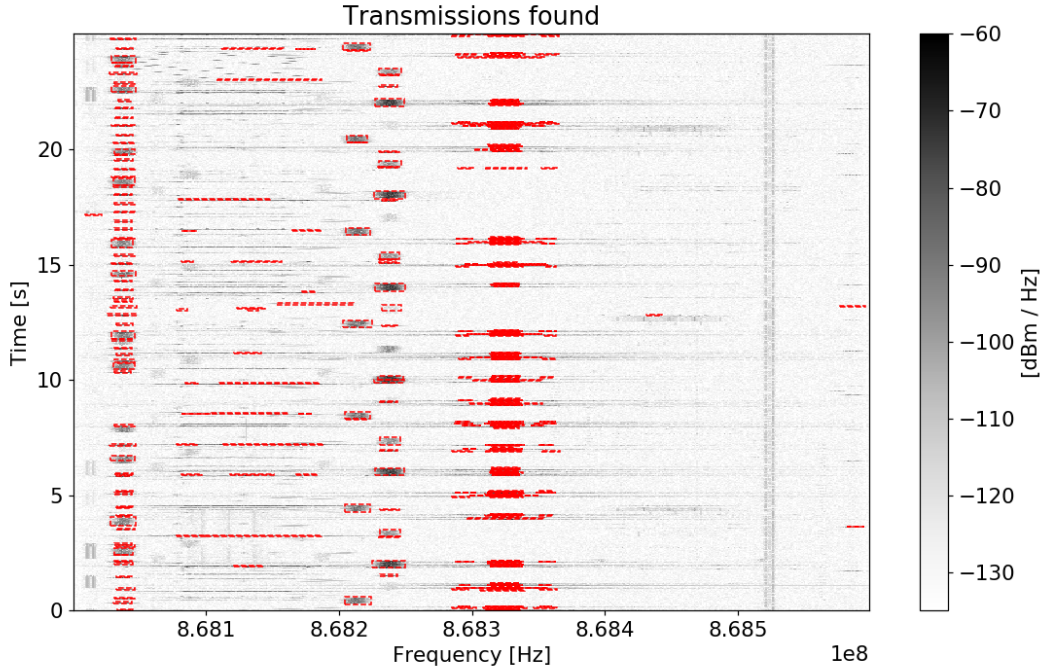

Figure A.2: Aarhus dataset signal transmissions found 1.

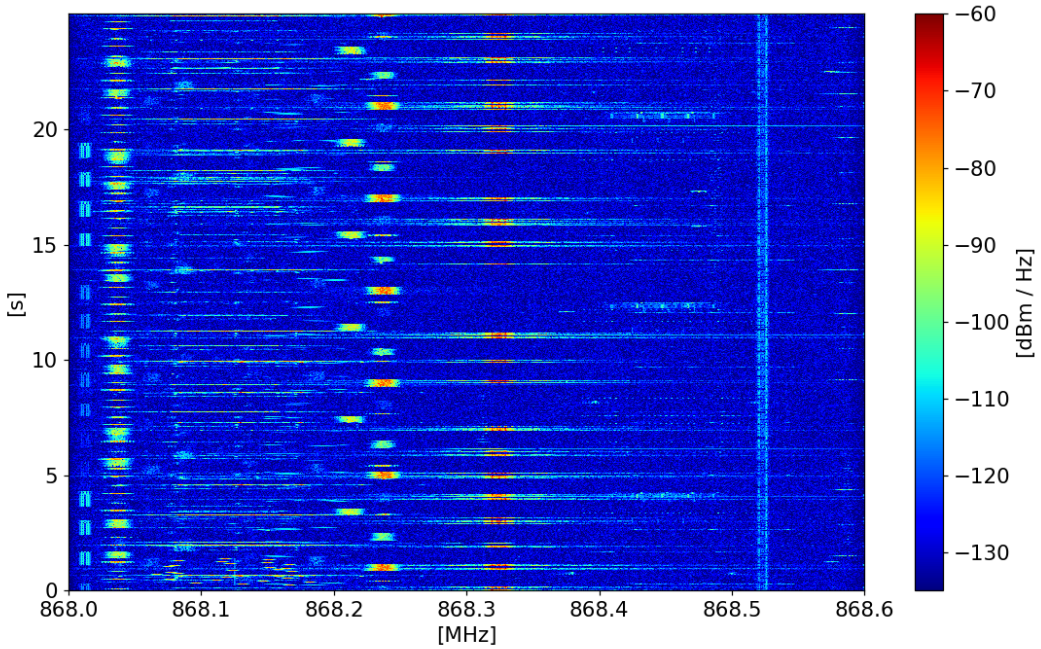

Figure A.3: Aarhus dataset signal spectrogram 2.

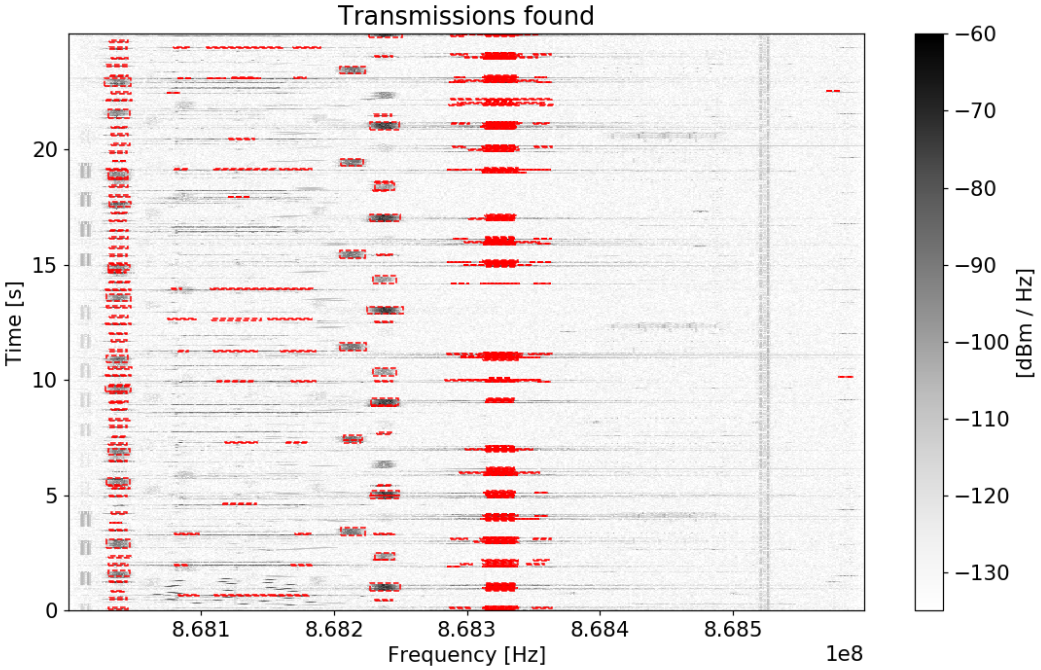

Figure A.4: Aarhus dataset signal transmissions found 2.

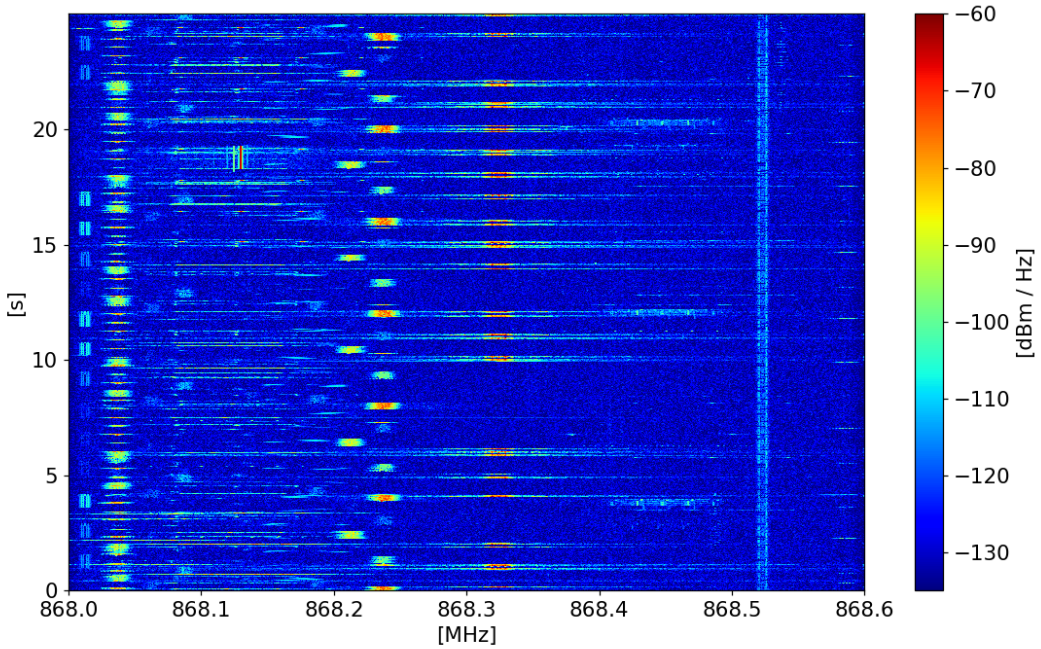

Figure A.5: Aarhus dataset signal spectrogram 3.

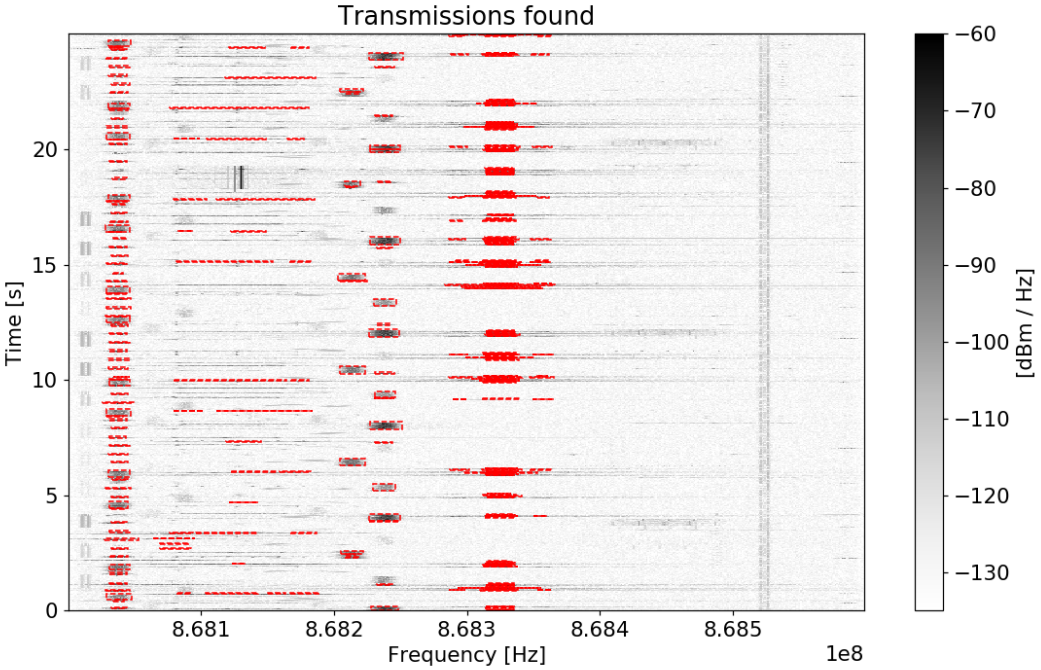

Figure A.6: Aarhus dataset signal transmissions found 3.

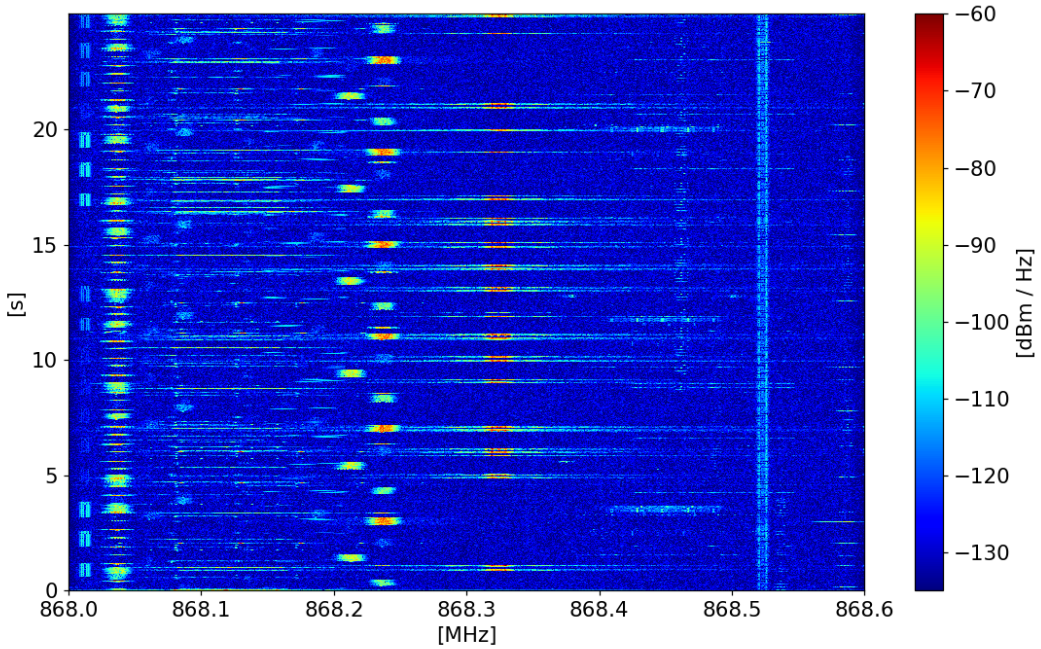

Figure A.7: Aarhus dataset signal spectrogram 4.

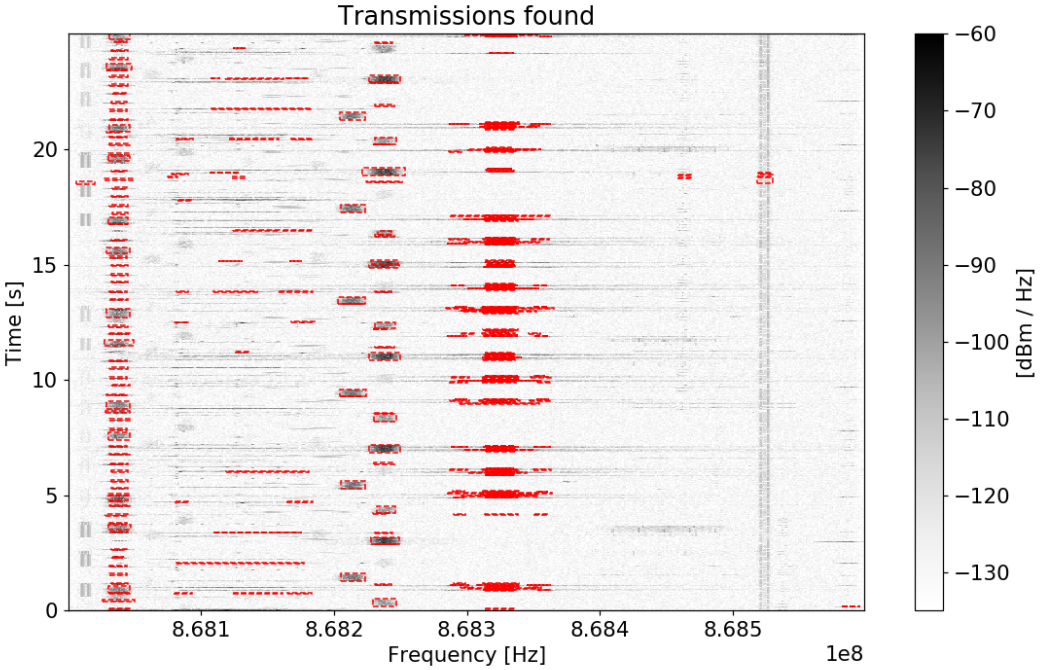

Figure A.8: Aarhus dataset signal transmissions found 4.

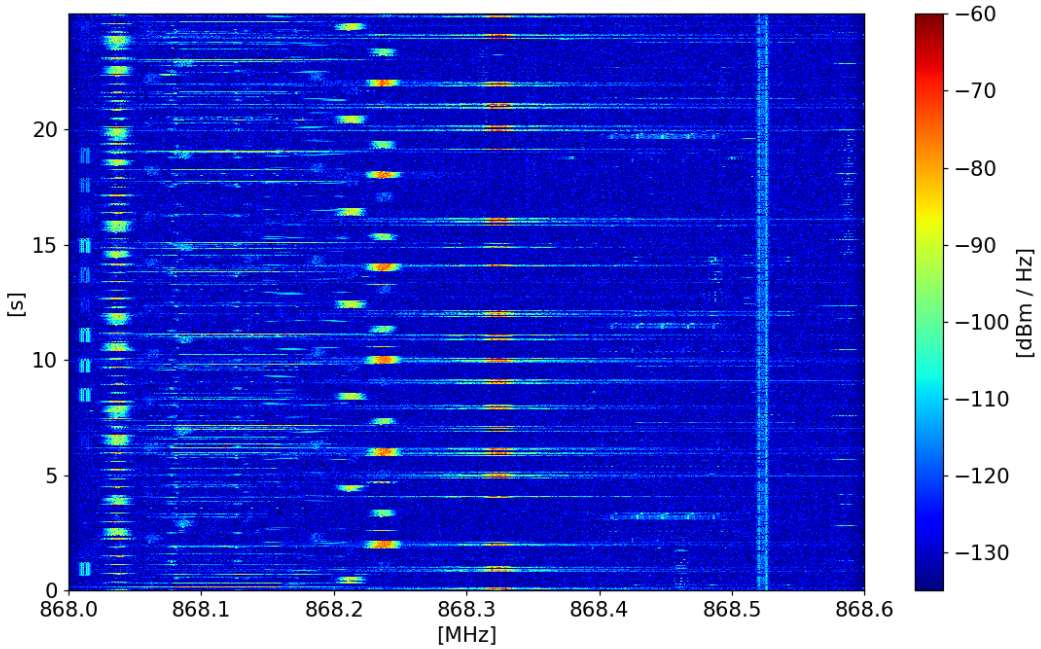

Figure A.9: Aarhus dataset signal spectrogram 5.

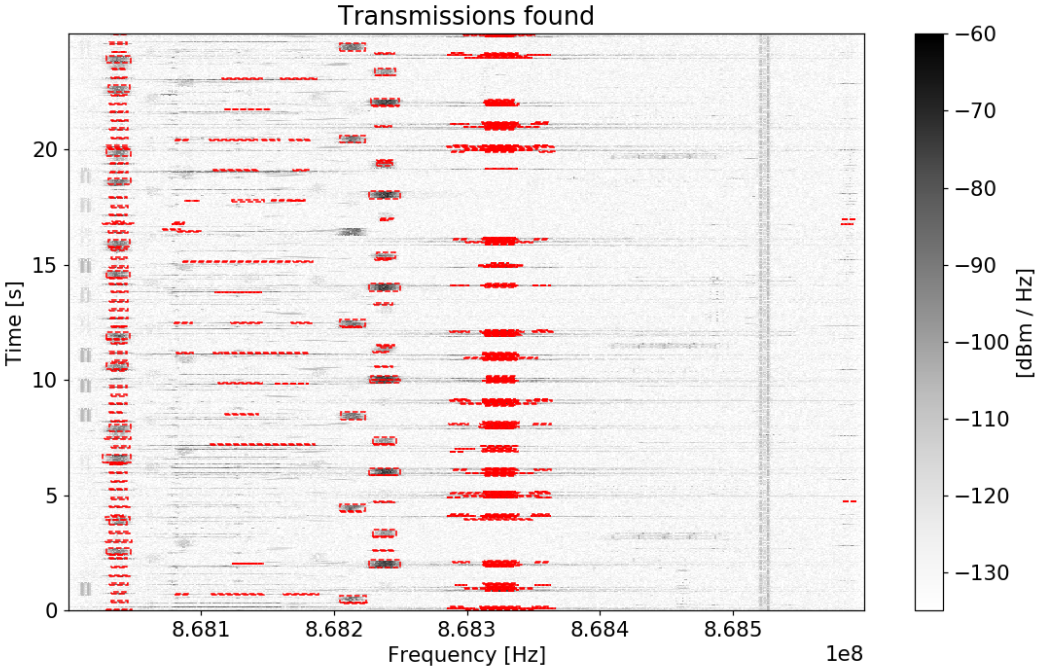

Figure A.10: Aarhus dataset signal transmissions found 5.

Aarhus data set - Found transmissions results zoom B

To show how many of the transmissions are found by the algorithm, the first $5x2.5$s of the Aarhus dataset are shown as spectrograms with their corresponding overlay which marks transmissions found. This is the first 2.5s of each spectrogram in A. The figures are matched page for page for better overview. For this reason, the rest of this page is left blank and the results are shown from the next page.

It is seen how the short transmissions in the range $868.05\text{MHz} - 868.2\text{MHz}$ are not well marked. This is due to the time resolution of the spectrogram as discussed in section 3.5. To find these transmissions, it is necessary to use a much higher time resolution. To maintain the same frequency resolution, this would require a higher sample rate of the spectrum analyzer as well as increasing the computational complexity of the algorithm.

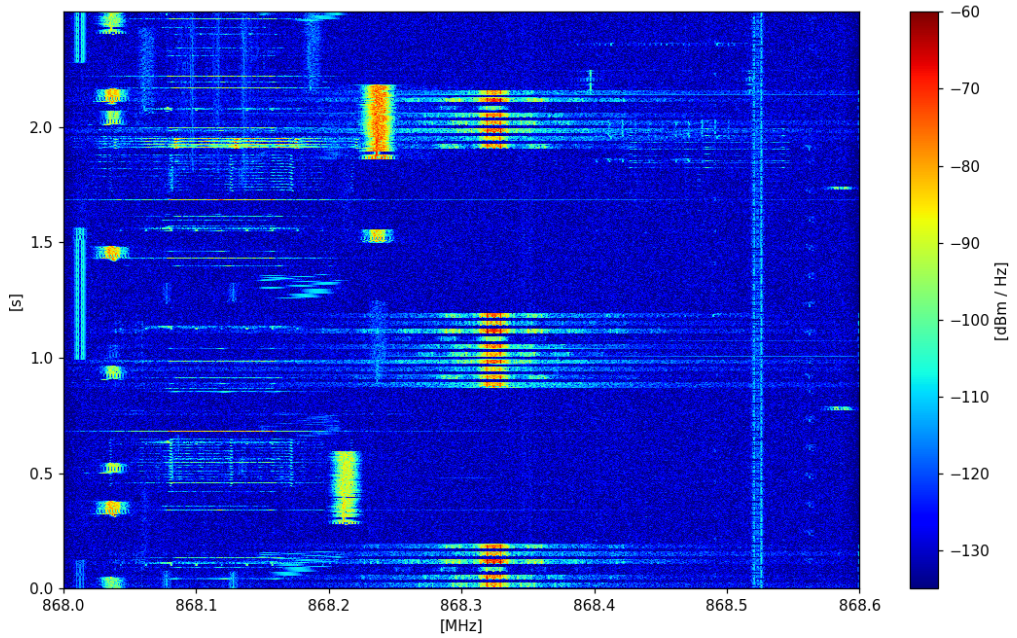

Figure B.1: Aarhus dataset signal spectrogram 1.

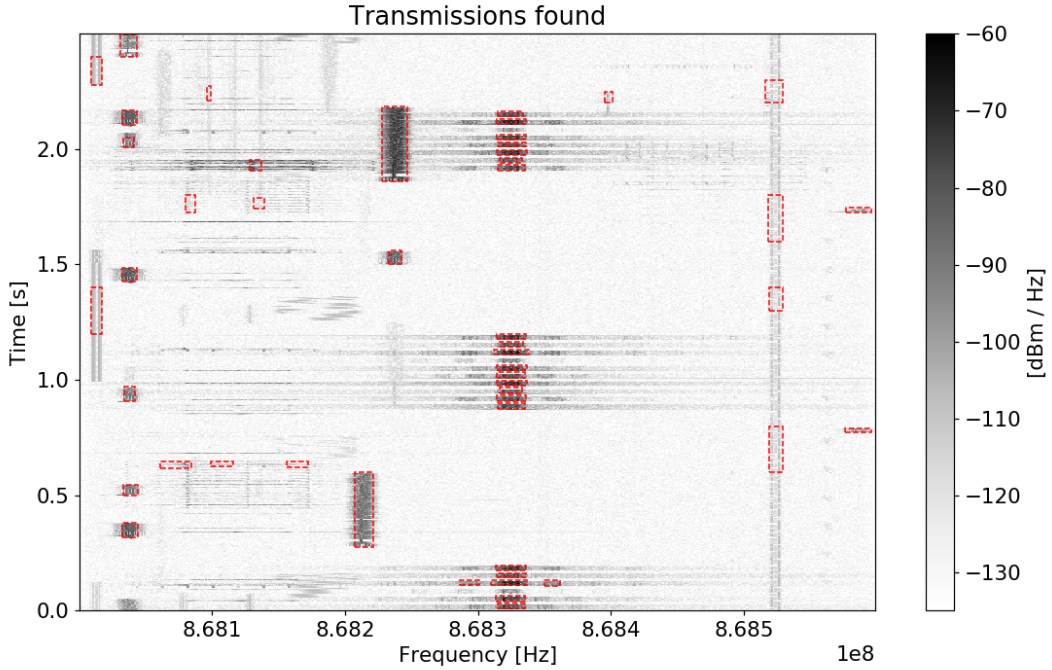

Figure B.2: Aarhus dataset signal transmissions found 1.

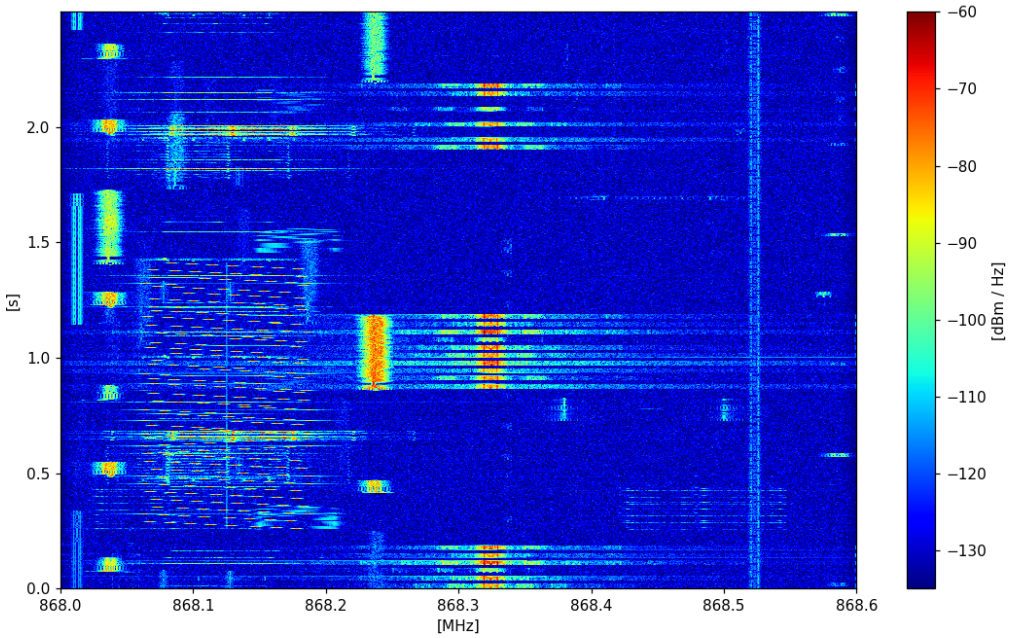

Figure B.3: Aarhus dataset signal spectrogram 2.

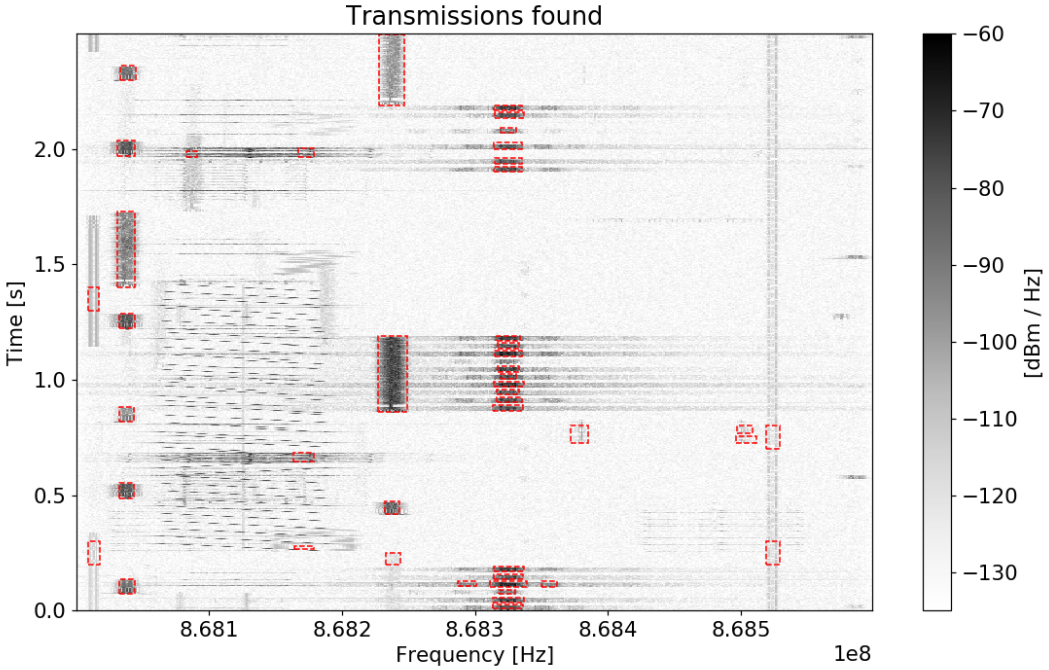

Figure B.4: Aarhus dataset signal transmissions found 2.

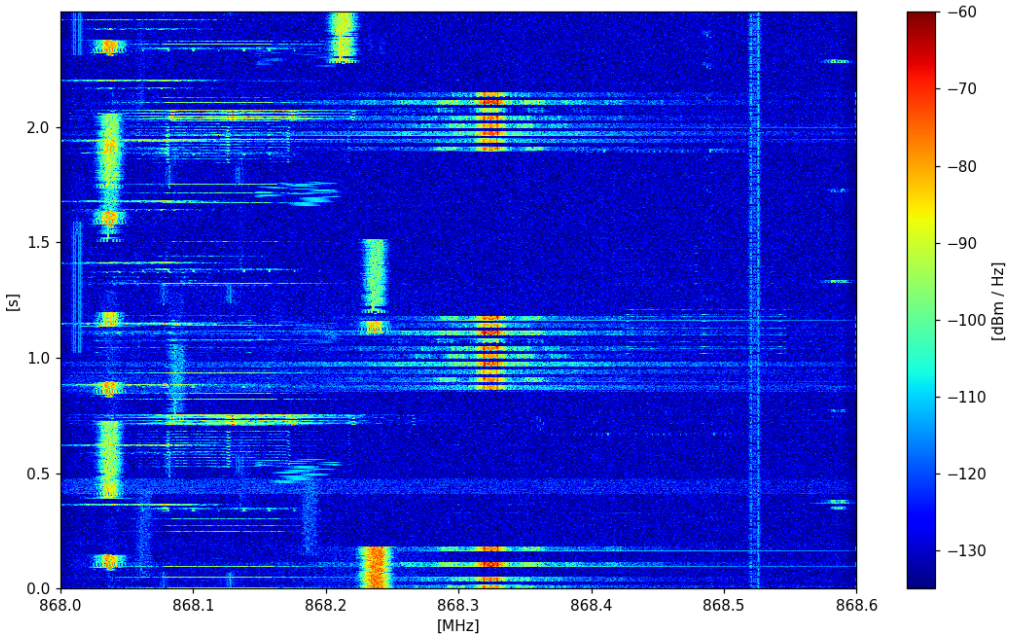

Figure B.5: Aarhus dataset signal spectrogram 3.

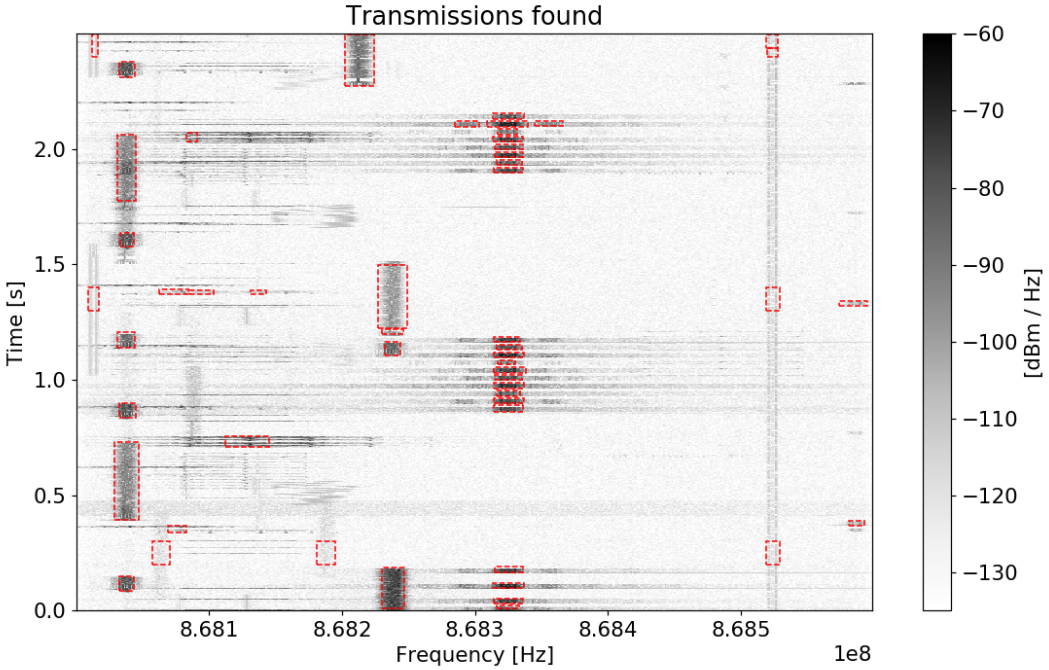

Figure B.6: Aarhus dataset signal transmissions found 3.

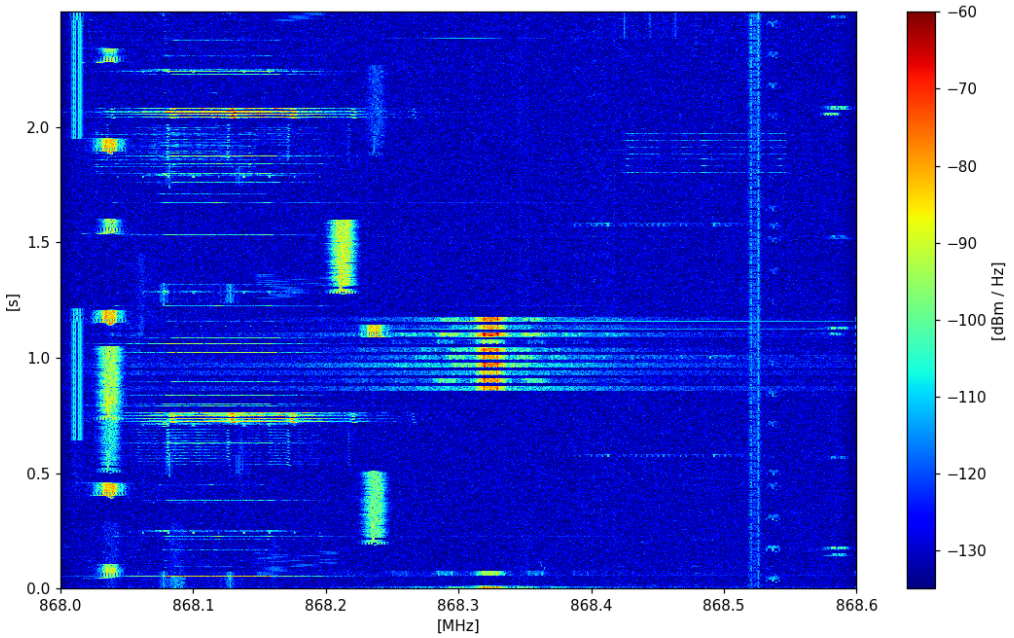

Figure B.7: Aarhus dataset signal spectrogram 4.

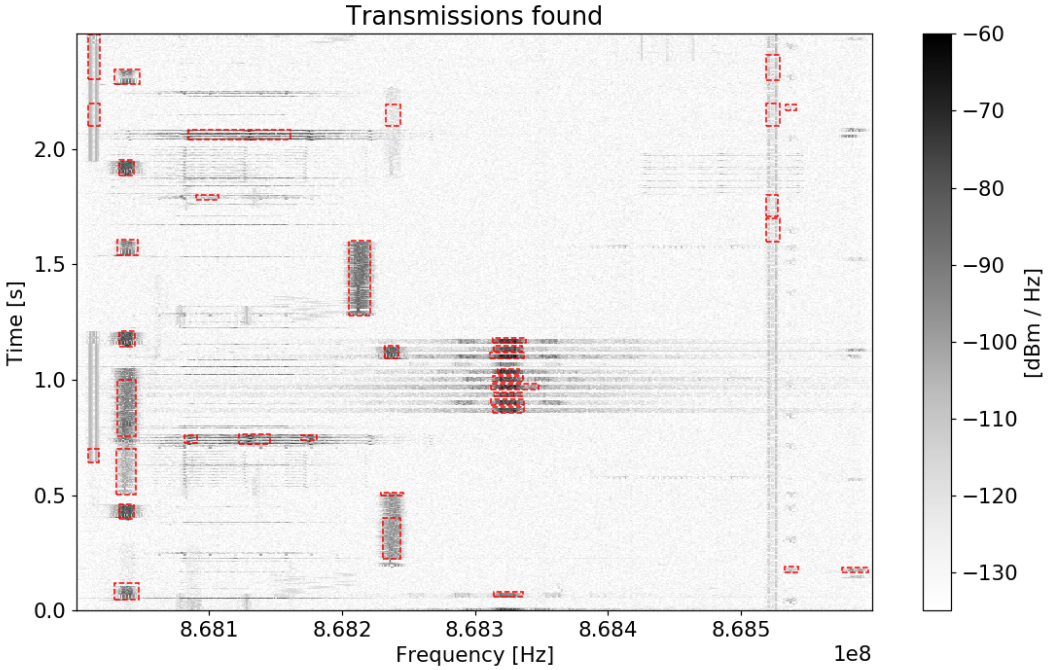

Figure B.8: Aarhus dataset signal transmissions found 4.

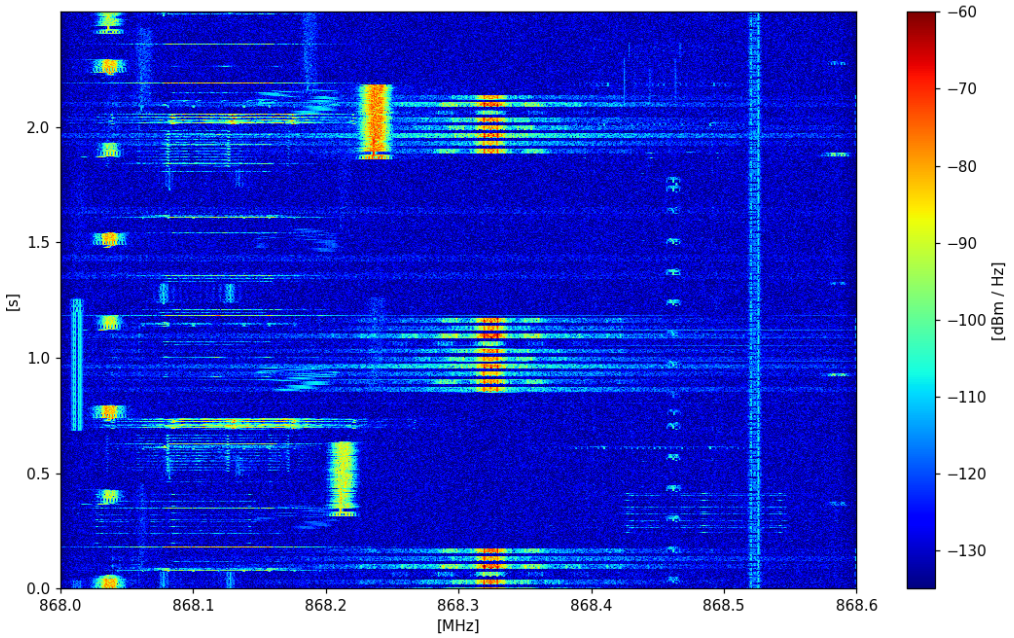

Figure B.9: Aarhus dataset signal spectrogram 5.

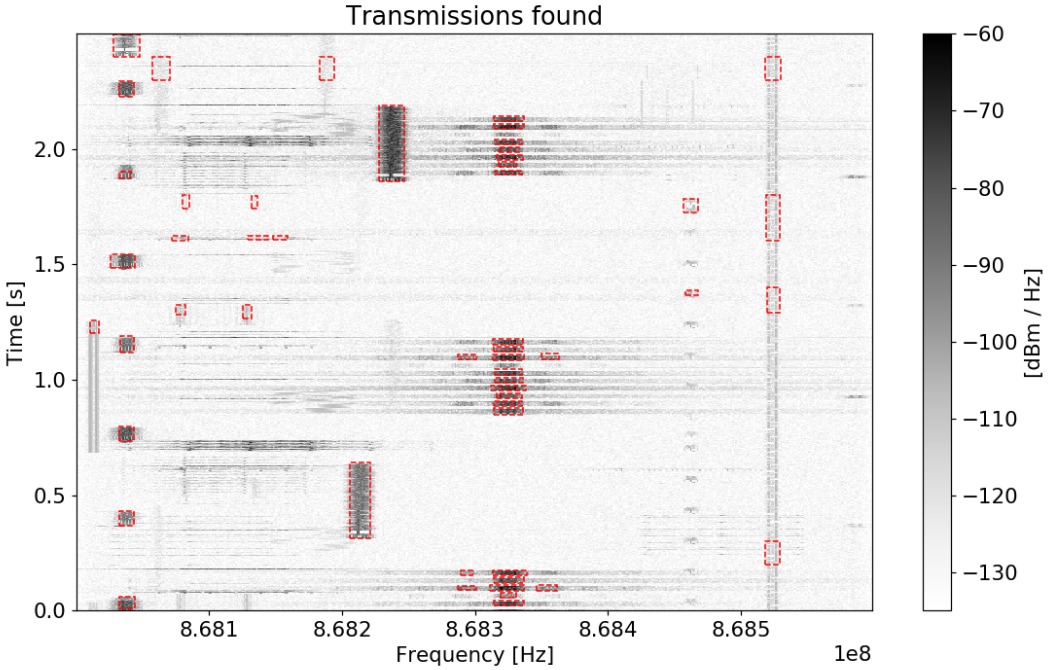

Figure B.10: Aarhus dataset signal transmissions found 5.