Investigating the cost of ML-based fingerprinting localisation for integrated sensing and communication in cell-free massive MIMO systems

Masters Thesis

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7B
DK-9220 Aalborg Ø

Department of Electronic Systems
Fredrik Bajers Vej 7B
9220 Aalborg Ø
www.es.aau.dk

Title

Investigating the cost of ML-based fingerprinting localisation for integrated sensing and communication in cell-free massive MIMO systems

Project type

Masters Thesis

Project period

Spring 2023

Project group

Group 1022

Participants

Mathias Thorsager
Sune Krøyer

Supervisors

Fabio Saggese
Shashi Raj Pandey
Israel Mayorga

Abstract

This report investigates the relation between the accuracy of localisation methods and their cost on the underlying communications network. The considered localisation methods are all based on the concept of integrated sensing and communications, where the location estimates are made using data extracted from existing communication.

The report details the development of three machine learning based methods. Two methods which use machine learning as an intermediary step in angle of arrival based fingerprinting, and one method which uses machine learning to perform location estimations directly. The first two aim to improve on the localisation accuracy of a considered fingerprinting-based baseline method, while the third aims to reduce the required setup cost of gathering large amounts of fingerprints.

It was found that despite the reduced cost of the third method, it was not able to perform in the considered system. For the two fingerprinting methods, both outperform the baseline while imposing little to no additional strain on the communications network.

Number of pages: 62
Date of completion: June 1, 2023

Nomenclature

Abbreviations	
AN	Anchor Node
ANN	Artificial Neural Network
AoA	Angle of Arrival
BS	Base Station
CNN	Convolutional Neural Network
DCNN	Deep Convolutional Neural Network
DFT	Discrete Fourier Transform
DNN	Deen Neural Network
ISAC	Integreated Sensing And Communication
LoS	Line of Sight
MAE	Mean Absolute Error
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MLP	MultiLayer Perceptron
mMIMO	massive Multiple Input Multiple Output
MSE	Mean Squared Error
NLoS	Non Line of Sight
NR	New Radio
PDoA	Phase Difference of Arrival
OFDM	Orthogonal Frequency Division Multiplexing
RSS	Received Signal Strength
ToA	Time of Arrival
TDoA	Time Difference of Arrival
UMi	Urban Microcell
UT	User Terminal
WMSE	Weighted Mean Squared Error

Preface

References follow the Harvard standard, where author name and year of creation are used. Further reference information can be found in the bibliography provided in the end. The bibliography is sorted alphabetically by author. However, if the author is unknown, the publisher is used instead. If the reference is a website, the latest visiting date is also included. An reference example is: [author's last name, year of creation]. Figures and tables are sorted according to the chapters. This means that the first figure in chapter 2 is numbered 2.1, and the next figure in the chapter is numbered 2.2. Every figure found externally will have a source reference. Figures without a reference are created by the group itself.

Numbers are written using periods as decimal separators and spaces as thousand separators.

The authors of this project are:

Mathias Thorsager	Sune Krøyer
<mthors18@student.aau.dk>	<skraye18@student.aau.dk>

Contents

1 Introduction

In 6G, one of the goals is to utilize the THz frequency band for communication which has the potential to enable significantly higher data rates than those achievable in current 5G NR (New Radio) systems. However, signals at higher frequencies are subject to higher attenuation which limits the effective communication range. To counteract this, techniques such as beamforming can be utilised to focus the transmit power and extend the communication range [One6G, 2022]. In order to achieve beamforming which has a meaningful improvement on communication, communication resources must be spent on aligning and maintaining the beam between the tx (transmitter) and rx (receiver). One way of reducing this communication overhead is to utilise estimated positions of the communicating devices [Talvitie et al., 2020].

Beyond the uses for beamforming in 6G, the position estimation and tracking of UTs (User Terminals) is already a focus in 5G and beyond under the term of sensing. Several current applications in 5G require accurate localisation, such as the use of autonomous drones and robots for search and rescue operations, road traffic management, and more [Kabiri et al., 2022]. The required accuracy of the localisations may vary for the different applications, ranging from a horizontal accuracy of 10 m down to 30 cm [3GPP, 2018].

While some methods of sensing such as GPS and RADAR are commonly used, they often require the use of additional resources on the UTs, BSs (Base Stations) and the communication between them. Additionally, under open sky GPS is typically accurate to within a 4.9 meter radius [PNT, 2022], which may not continue to support the localisation accuracy requirements of future 5G and beyond applications. However, recent developments within both sensing and communication show a promising alternative of combining the two areas in what is known as ISAC (Integrated Sensing And Communication) [Liu et al., 2022].

Radio sensing as well as radio communication are moving towards higher frequency bands and larger antenna arrays. This means that the channel characteristics, signal processing, and hardware are already quite similar for both sensing and communication. This means that sensing and communication could be done using the same wireless infrastructure. Furthermore, another development for beyond 5G is the shift from monolithic cell-based systems to cell-free mMIMO (massive Multiple Input Multiple Output) systems [Zhang et al., 2020]. The improvements of 5G in terms of data rates and traffic volumes are predominantly achieved by the UTs near the cell centers. This is due to the inter-cell interference and handover issues which are innate problems of the cellular structure. To mitigate these problems the beyond 5G networks are shifting to cell-free paradigms [Zhang et al., 2020]. In conjunction with mMIMO technologies, this allows for new opportunities for integrated sensing especially for environment- or localisation sensing. Utilising multiple MIMO connections with multiple distributed antenna arrays increases the points of reference which should allow for improved environment/localisation sensing.

Based on the similar channel characteristics, signal processing and hardware that create the basis for ISAC, the option of using the existing communication for localisation could be a possible promising alternative. Using the existing communication while taking advantage of the benefits of obtaining data from multiple MIMO BSs should further increase the viability and accuracy of this approach. As such, the focus of this report will be on investigating the development of a localisation method using existing communication in cell-free MIMO systems.

This leads to the initial problem statement:

How can localisation information be extracted from existing communication, and how is this information used to estimate the locations of user terminals?

Problem Analysis 2

This chapter aims to answer the initial problem statement presented in the introduction of the report. The chapter starts by introducing the components of a wireless signal and explains how location information can be extracted from it. Based on this information, the chapter next covers the current state of the art for estimating the location of UTs using geometry-based and fingerprinting-based localisation methods. Following this, a brief overview of the prevalent machine learning structures used in localisation methods. Lastly, the chapter present the overall scope of the report, which is to investigate the cost of localisation methods.

There are many ways to estimate the location of a UT based on uplink or downlink transmissions. However, when considering the localisation process of a UT as an integral part of communication as with ISAC, the information used to estimate the location must be extracted or decoded from the existing communication. While it is possible to have the UT estimate its own location through separate means and then transmit the location to the BSs, this involves extra work which can be avoided. Instead, it is possible to extract information regarding the UT's location which is naturally present in the uplink signals received by the BSs. After the location information is extracted, it can be used to estimate the location of the user. In 4G and 5G, UTs are mainly in range of single BSs and as such, the localisation estimates must be based on the information which can be extracted from the transmissions between the two. However, for 6G and beyond, the concept of cell-free massive MIMO will allow multiple BSs to receive the uplink transmissions. This enables the use of localisation methods which take advantage of the multiple reference points to increase the accuracy of the location estimations.

2.1 Extracting Location Information

In order to explain how location information is extracted from transmissions, the basic elements of signals must be explained. Considering communication over a flat frequency channel, a simple representation of a received signal y is that of a single transmit- and receive-antenna where only a signal traveling over a LoS (Line of sight) path is received. This signal consists of a complex symbol y which is defined as:

$$y = hx + w, \tag{2.1.1}$$

where x is the transmitted symbol, h is the channel coefficient, and w is the noise.

The received signal is received as a waveform, which is an electromagnetic wave, over some time T_s. In order for this signal to carry data it is modulated by altering the phase (phase shift keying), amplitude (amplitude shift keying) or frequency (frequency shift keying) in specific ways which represent a set of complex symbols. While the exact methods for modulating the signal are not important for localisation, the fact that the transmitted symbol x is specified by these factors is important.

The received signal y is not a perfect representation of the transmitted symbol. It is changed by the channel coefficient and noise which affect the phase, amplitude, and frequency. The noise is white Gaussian which means it cannot be estimated and removed from the received signal. However, the channel coefficients are determined by the environment and relative position of the transmitter and receiver. While the coefficients in most environments are not static, the rate at which they change is often slow enough that they can be estimated and removed from the received signal. In practice this is done by transmitting periodic pilot symbols, where the modulated data is known by both the transmitter and receiver – for practical purposes the modulated data is often the Kronecker delta. By knowing what x is, the channel coefficient can be estimated by demodulating x from y. For the following transmissions, the estimated channel coefficient can be demodulated from y to give a more clear representation of x.

From this description of the received signal, the only apparent cause of noise is the w term in equation (2.1.1). However, this is only the case if the signal is received through a single LoS path. In practice, this would not be the case. When a signal is transmitted it encounters obstacles and walls which reflect, diffract, and scatter the signal. This causes the receiver to receive multiple versions of the signal which is called multipath propagation [Popovski, 2020, p. 284].

A convenient consequence of the existing use of pilot symbols to estimate channel coefficients, is that the channel coefficients carry information which is useful in localisation. As mentioned, the channel coefficients are based on the relative positions of the transmitter and receiver. In particular, the phase of the channel coefficients is related to the distance between the transmitting antenna and the receiving antenna. Furthermore, the amplitude of the channel coefficients is also based on the distance. The further a signal propagates, the more power it loses which means that the amplitude of the channel coefficients becomes lower.

The channel coefficients are, however, not the only aspect of the received signal which carries information regarding the position of the transmitter or receiver. Just as the amplitude of the channel coefficient is related to the distance between them, so is the time it took for the signal to propagate. As a result, there are three aspects of a transmission whose use in location estimations will be investigated in the following subsections: the phase, the power, and the time of arrival of the recieved signal.

2.1.1 Phase of the Signal

In order to use the phase of the channel coefficients as localisation information, a more complex equation for the received signal is needed. For this, the signal is still transmitted from a single antenna, but is now received over an antenna array consisting of N antenna elements. Such a setup is called MISO (Multiple Input Single Output) but is generally referred to as a version of MIMO despite the lack of multiple transmitting antennas. This means that the received signal is now defined as:

$$y_n = h_n x + w_n, \quad n \in [1, N] \tag{2.1.2}$$

where n is the specific receiver antenna.

Since the signal originates from a single transmission from a single antenna, the symbol x is the same at every receiver antenna. This means that when a pilot symbol is transmitted, different channel coefficients are estimated for each antenna (the noise is still a factor so it is never a perfect reconstruction of the channel coefficients). One of the main differences in the channel coefficients across the receiver antennas is the phase of the received signal. If the antennas on the receiver are spaced slightly apart – less than or exactly one half the wavelength of the transmission – the phase will be different on each antenna (assuming that the signal does not arrive orthogonal to the antenna array). This means that the difference in phase across the receiver antennas can be translated to a relative difference in distance.

From trigonometry, the difference in distance the signal travels to each antenna element is determined by the relative AoA (Angle of Arrival) to the orientation of the antenna array. Likewise, the phase difference at each antenna element is correlated to the AoA of the signal. This means that by analysing the channel coefficients, it is possible to extract the angle which the signal arrived at.

As mentioned in section 2.1, multipath propagation can cause extra noise for the received signal. This is one of the main issues for AoA estimations based on the phase difference as the different paths can hit the receiver from many different angles. If each individual signal from the different paths could be distinguished at the receiver, then there would not be a problem. However, the signal is not an instantaneous value which is detected by the antenna. Instead, it is a waveform which takes some amount of physical time to be received. This means that the other paths are likely to arrive during the reception of the initial LoS signal and interfere with the original signal. The multipath effect causes the receiver to detect a signal which has components of all the different paths which each has a different AoA. The challenge is therefore to accurately extract the AoA of the LoS path only, as this is what is relevant.

An example of dealing with NLoS (Non Line of Sight) paths interfering with the LoS AoA detection is used in the paper by Shen et al. [2021]. Here, the method used for estimating the LoS AoA is based on calculating the amplitude of the different PDoA (Phase Difference of Arrival) frequencies encoded by the different paths. It is assumed that the NLoS paths lose a significant amount of power compared to the LoS path which can be detected in the amplitude of the PDoA frequencies. The AoA is found through a beam alignment approach which works by taking a modified DFT (Discrete Fourier Transform) of the estimated channel coefficients of the transmitter. From this modified DFT, the AoA is estimated based on the element in the resulting row vector with the largest power, which corresponds to the amplitude of the frequency. Other note worthy AoA estimation methods based on the received signal are the MUSIC (MUltiple SIgnal Classification) and ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) methods as detailed in the papers by Wang et al. [2019]; Lin et al. [2018].

A further aspect which can complicate the process of extracting the AoA is when interference occurs from multiple transmitters. One of the main benefits of using MIMO systems is that they can transmit to and receive from multiple devices at the same time. However, this is only possible if the signals from or to the transmitters do not cause interference with each other. While this can be solved through time division based MAC protocols, these do not allow for the simultaneous transmissions. Instead, Frequency division can be used, in particular OFDM (Orthogonal Frequency Division Multiplexing) can be used to send multiple signals which are known as subcarriers over orthogonal frequencies [Popovski, 2020, p. 247]. As long as the pilot signals for each transmitter in the uplink are orthogonal, then all channel coefficients can estimated without interference [Popovski, 2020, p. 330].

2.1.2 Signal Power

When a signal propagates through the air, the power of the signal attenuates as a function of the distance, known as path loss. This means that if the transmitted power is known, it is possible to estimate how much power was lost during the transmission. However, a general problem is that the power lost during transmission is not only caused by the distance which the signal travelled. Other factors such as shadowing and blockages will also affect the RSS (Received Signal Strength) of the transmitted signal. The issue lies in the fact that these variables are generally unknown and cannot be accurately accounted for when estimating the distance from the loss in power. Usually, a generic path loss model, such as the log normal path loss model, is used where the unknown factors are estimated as a Gaussian random variable. This model is defined as:

$$\beta = \alpha_0 + 10\alpha \cdot \log_{10}\left(\frac{d}{d_0}\right) + \xi[dB], \tag{2.1.3}$$

where α_0 is the path loss in dB at the reference distance d_0 calculated based on the Friis free-space path loss model, α is the path loss exponent, d is the distance between the transmitter and receiver, and ξ is a Gaussian random variable describing the shadowing of the channel. The Friis free-space path loss model calculates the path loss of a purely LoS transmission:

$$\alpha_0 = \left(\frac{4\pi df}{c}\right)^2, \tag{2.1.4}$$

where d is the distance between the transmitter and receiver (it is assumed that the distance is large enough such that the antennas are in the far field of each other), f is the frequency of the transmitted signal in Hz, and c is the speed of light in m/s.

Opposite to the AoA estimation, it is not required for there to be multiple antennas on the receiver in order to get distance estimates using the RSS. However, the requirements of knowing the exact transmit power and path loss exponent for the environment is not always possible. Depending on the devices used for transmission, it is not necessarily possible to guarantee that they always transmit with the correct power. Furthermore, environments are not static which means that the path loss exponent will not remain the same at all times. This means that while it is possible to fit a path loss exponent to an environment, over time this value will change which will cause the distance estimates to become inaccurate.

To deal with these problems, the paper by Prasad and Bhargava [2019] presents a localisation method which does not require perfect prior knowledge of the path loss exponent or transmit power. Due to the assumed instability in the transmit power, the proposed method works with the differential RSS with respect to a chosen reference BS. Furthermore, it is assumed that environmental changes will cause the path loss exponent to change following a uniform distribution in a predefined interval. Based on these assumptions, the authors have defined a linear least squares solution which can map estimates for the ratio of distances w.r.t. the reference BS to an estimated location of the UT.

2.1.3 Time of arrival

There are two ways which the time of arrival of a signal can be used to extract information regarding the location of the transmitter. One is to translate the ToA (Time of Arrival) to a distance based on the propagation time of the signal and the other is to calculate an AoA based on the TDoA (Time Difference of Arrival) at the different antennas on the receiving antenna array.

The ToA of a signal can be converted into a distance if the exact propagation time of the signal is known. This requires the exact transmission time to be known along with the ToA. There are two main problems with this. First, to know the exact transmission time requires strict synchronisation between the transmitter and receiver, which can be difficult to achieve in practise. Instead, using RTToA (Round Trip Time of Arrival) or using the TDoA of multiple receivers will remove the synchronisation requirement. The second issue is that the timestamps taken at the transmitter and receiver may be affected by the network jitter at the layers above the physical layer. This can, however, be mitigated by using methods from synchronisation schemes. One example is the TPSN (Timing-sync Protocol For Sensor Networks) Ganeriwal et al. [2003] where the timestamps are taken at the MAC-layer on the transmitter and receiver.

The second way of utilising the time of arrival is based on a multi antenna receiver where each antenna records a different time of arrival. This removes the problem of synchronisation entirely as there are no longer multiple devices which are receiving the signal. Instead, the difference in time of arrival between each antenna can be translated into an angle of arrival based on simple trigonometry.

2.2 Location Estimation

In the above, three methods of extracting location information regarding the transmitter is described. However, the real problem is how to use this information to get an estimated location. When there is a single receiver of the signal, it is especially difficult. A single distance or AoA estimation is not enough to get a 2D location of the transmitter. It is, however, still possible to estimate the location of a transmitter based on a single receiver. One way is to use more than one type of information. By combining an AoA and distance estimation it is possible to estimate a 2D location by describing the position as a polar coordinate.

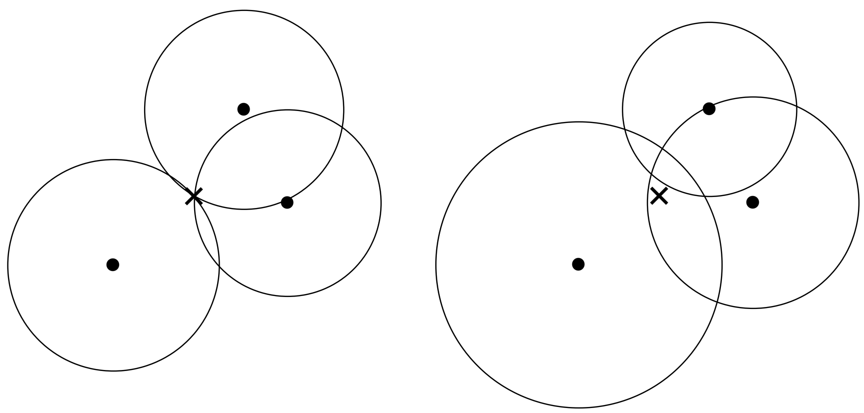
Another way is to have more than one antenna array at the receiver which are spaced some distance apart. This will allow a single receiver to get two or more AoA or distance estimates – one for each antenna array. For 4G cell towers, this may not be a problem as the physical size of the tower is large enough to allow a significant spacing between the antenna arrays. However, for smaller 5G base stations, it may not be possible to get sufficient spacing between them. The closer the antenna arrays are to each other, the more alike the angle or distance estimates will be, which makes it harder to use them for location estimations. It is therefore generally desired to have the antenna arrays placed at locations far apart. One way of achieving this is through the concept of distributed MIMO, which is found in cell-free mMIMO, where several BSs with single antenna arrays are placed within the range of transmitters. Instead of having a single BS receive a signal, there are now multiple BSs which each receive the signal and who can cooperate to more accurately estimate the location of the transmitter.

2.2.1 State of the Art

In the current state of the art for localisation, there are mainly two categories of methods in use: 1) geometry-based methods and 2) fingerprinting-based methods [Alamu et al., 2021].

1) Geometry-Based Localisation

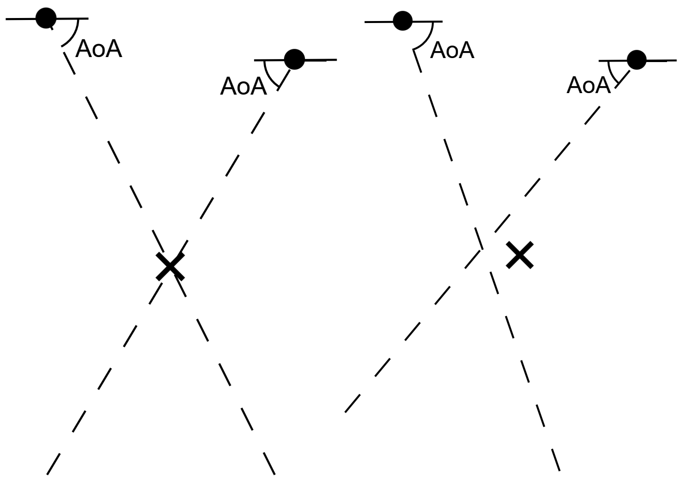
The two main ways of estimating the user location based on their geometric information, is the multilateration and multiangulation methods. Multilateration works by estimating the position of a transmitter based on estimated distance between the transmitter and a number of receivers. As illustrated in figure 2.1, each receiver draws a circle around them where the radius is the estimated distance – estimated from the RSS or ToA. The location of the transmitter is then based on the point which has the least summed distance to the peripheries of the circles. In the case that there is no noise and the distance estimates are all perfect, the circles will intersect in exactly one point (there will be two points if only two distances are used), and the point with the least summed distance is this intersection.



(a) No Noise ***(b)*** Noisy

Figure 2.1. Multilateration for distance estimations with and without noise. The dots are the locations where the distance estimations where made, the radius of the circles are the distances which where estimated, and the X is the location of the transmitter.

Multiangulation works by finding the intersection of lines which exit the receivers at the AoA of the received signal, as illustrated in figure 2.2. If there are only two angles, then the estimated location will always be where the two lines intersect. However, if there are more than two angles, the noise may affect the angle estimates in a way where there is no point where all lines intersect. In this case, the location estimate is based on a similar process as for the multilateration – finding the point which is closest to all lines.



(a) No Noise ***(b)*** Noisy

Figure 2.2. Multiangulation for AoA estimations with and without noise. The dots are the locations where the AoA estimations where made, the dashed line is the estimated path of the signal, and the X is the location of the transmitter.

In the context of mMIMO, the AoA based localisation methods have proven to be the best performers of the single type geometry-based methods. Alamu et al. [2021] attributes this to the fact that more antenna elements allows for finer angular resolution whereas more antennas do not help in alleviating the problems of fast fading or the tight time synchronisation requirement which plagues the RSS and ToA methods respectively. However, as the authors show when comparing the accuracy of geometry-based localisation techniques, the best localisation accuracy is found when a combination of AoA and RSS or ToA is used. An example of this is presented in the paper by Garcia et al. [2017] where the authors use the ToA to limit the amount of NLoS paths which are considered in the AoA estimation. For these hybrid localisation techniques, the accuracy goes from a general accuracy of meter-level for the individual techniques, to centimeter- and even millimeter-level for the hybrid techniques.

2) Fingerprinting-Based Localisation

The fingerprinting-based methods differentiate from the geometry-based methods by including an offline step to the localisation process. Instead of using the RSS, AOA, or ToA values directly to infer the location of the transmitter, the fingerprinting-based methods record these values as fingerprints for a number of known locations in an offline phase. The fingerprints are then used in the online phase where the location estimation takes place.

The traditional fingerprinting approach is akin to the KNN (K-Nearest Neighbors) ML (Machine Learning) method. For this, the offline phase consists of mapping known locations of an area to fingerprints consisting of measurements of either RSS, CSI (Channel State Information), or AoA. In the online phase a UTs location is estimated by comparing the fingerprint of the UT to the fingerprints of the known locations. This is typically done using an MSE approach and the location is estimated as the known location with the best fit or some function of the best locations, hence the comparison to KNN.

More recent types of ML models can also be considered versions of fingerprinting. For example, an ANN (Artificial Neural Network) can be used in a similar way to give direct location estimates based on an offline training phase where transmissions, or information extracted from transmissions, are given as inputs and the known locations are the targets.

When considering the impact of the different types of information used in the fingerprinting methods, Alamu et al. [2021] shows that the CSI based methods generally perform better than those using AoA and RSS. Furthermore, this is mainly shown to be the case when the location estimates are done using DNNs (Deep Neural Networks) and DCNNs (Deep Convolutional Neural Networks) where the accuracy are increased from meter-level to centimeter-level.

2.3 Machine Learning

As mentioned in the state of the art section on fingerprinting-based localisation, ML is widely used for localisation. Furthermore, two of the most promising structures are the DNN and DCNN models. As such, in order to get a better understanding of these, a brief introduction to the theory behind them is presented. Both the DNN and DCNN are based on the general structure of an MLP (MultiLayer Perceptron), however, the DCNN utilises one or more convolutional layers.

MLPs are feed forward networks where every layer consists of nodes that apply an activation function on the sum of weighted outputs of every node of the previous layer. By applying weights to the outputs of the previous layer, the output (or activation) of a node becomes dependent on parameters which can be changed, which allows for training of the model. The weights of an MLP model are adjusted using a process called backpropagation. This process calculates an error between the output of the model and a specified target and propagates the error backwards through the network, adjusting relevant weights appropriately [Bishop, 2006]. In addition to weights at every layer, a bias term is also introduced. However, the bias can be absorbed in the weights by adding an additional bias node at every layer with no connection to the previous layers, which leads to the following equation for the activation, z, of a node, k [Bishop, 2006]:

$$z_k = g\left(\sum_j w_{kj} z_j\right), \tag{2.3.1}$$

where k is the current node, j iterates through the nodes in the previous layer, g is the activation function, and w is the weight assigned between k and j.

A graph depicting the general structure of MLPs is shown in figure 2.3.

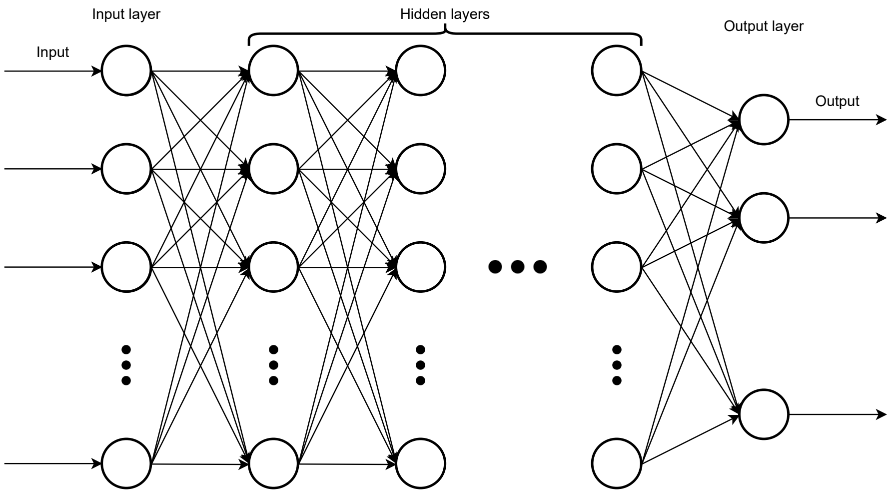


Figure 2.3. Generalised structure of an MLP showing the nodes of the input layer, hidden layer(s), and output layer, as well as the connections between them.

When training an ML model, be it a DNN, DCNN, or any other structure, the model is provided with a set of training data with corresponding target values. The goal is for the model to learn the statistical correlation between the training data and the provided targets such that it can correctly predict targets given some input. In this regard, the performance of a model is not evaluated on its ability to predict the targets for the training data, but rather on its ability to correctly predict the targets of unseen data. If a model excels at this, it is said to be great at generalising.

Relating this to the process of estimating the locations of UTs, the input of an MLP can be a vector of the estimated channel coefficients at the BSs which received a transmission from the UT. Through the hidden layers, information regarding the position of the UT is extracted and an estimated position is returned as the output – for instance in the form of an x- and y-coordinate. As mentioned in section 2.1, coefficients from individual BSs contain information regarding AoA and distance. When multiple BSs are considered at once, the set of channel coefficients can be considered as spatial data. For this type of data, a category of neural networks called CNNs (Convolutional Neural Networks) is often used [Bishop, 2006].

2.3.1 Convolutional Neural Networks

CNNs are neural networks commonly used for processing spatial data, such as images. CNNs are networks that apply a convolution operation, instead of the usual matrix multiplication, in one or more layers. The notable layers of CNNs are: 1) convolutional layers and 2) pooling layers.

1) Convolutional layer

A convolution is a process where an input tensor is transformed using a smaller tensor called a kernel. The kernel slides over the input tensor, multiplying its values with the overlapping input values. This process repeats until all possible kernel positions have been covered, essentially performing a matrix multiplication.

The expression of a convolution operation is:

$$(\mathbf{I} * \mathbf{K})(i, j) = \sum_m \sum_n \mathbf{I}(m, n)\mathbf{K}(i - m, j - n) \tag{2.3.2}$$

Where:

I	Input tensor
K	Kernel tensor

For convolutional operations, the kernel is flipped. However, the operation is usually implemented using the cross-correlation, which avoids the need for flipping the kernel while providing a similar result. As such, the convolutional operation is usually implemented as follows:

$$(\mathbf{I} * \mathbf{K})(i, j) = \sum_m \sum_n \mathbf{I}(i + m, j + n)\mathbf{K}(m, n) \tag{2.3.3}$$

2) Pooling layer

Another common type of layer used in CNNs is a pooling layer. Pooling is an operation that reduces the size of the input, thereby decreasing the number of parameters and computations in the network. An example of a commonly used pooling function is Max pooling which returns the maximum value within a region. Pooling makes the CNN model approximately invariant to small translations of the input [Goodfellow et al., 2016, p. 342]. This means that a CNN model using max pooling may be able to generalise better.

When convolutional or pooling layers are used in for example DCNNs, the output of the final convolutional or pooling layer is flattened and used as an input to the following series of MLP layers, also known as dense layers.

2.4 Problem scope

As was shown in section 2.2.1, there are several ways of achieving localisation estimations using the transmissions of already existing communication. It was further shown that there was at times a quite significant difference in the accuracy of the different methods, where some methods could achieve a meter-level precision and others even exceeded the centimeter-level precision. However, the accuracy of a method is not the only indicator of its performance. While some methods may be able to achieve a significantly higher accuracy than others this may be irrelevant if the use case only calls for a certain degree of localisation accuracy. In this case, more important aspects to consider may be the cost of the methods when they operate at the required accuracy. This cost can cover many things depending on the specific use case, however, general aspects of the cost include the impact of the methods on the communication and the amount of manual work required to achieve and maintain the localisation accuracy.

For the impact on the communication, the first aspect to consider is the impact of the localisation method on the communication between the UTs and the BSs. Due to possible inaccuracies on individual location estimations, most localisation methods require multiple transmissions for each localisation estimation. However, the exact number of transmissions required to achieve the desired accuracy can vary from method to method. Furthermore, when considering that the UTs are moving, it is important to consider the frequency with which the localisations are done. It may be that if a method can achieve a significantly higher accuracy with a few extra transmissions, the frequency of localisations will be decreased enough that the overall cost is lowered.

The second aspect of the communication to consider is the impact on the backhaul. Due to the fact that the localisation methods considered in this project utilise distributed MIMO, it is required that data is shared between the BSs. For example, for a fingerprinting-based method, the localisation is done via a comparison of the fingerprints from all BSs. This means that the contents of the fingerprints must be shared and gathered at a central location where the comparison will take place. Other methods may require more of less data to be shared over the backhaul depending on how much of the work is done at the individual BSs or the central location.

Besides the impact on the communication, the fingerprinting-based methods require a set of fingerprints which are gathered based on transmissions from known locations. This is a significant amount of manual work which must be done initially to build up the fingerprint database, and then maintained as the environment changes. Furthermore, if there are ML models involved, then these need to be trained and kept up to date as well.

This leads to the following problem statement:

How does the accuracy of ISAC-based localisation methods relate to their cost on the underlying communications network?

For the scope of this project, fingerprinting-based localisation methods are the focus of this investigation. Traditional fingerprinting-based localisation methods where the received signal is processed at the BSs before any data is transmitted over the backhaul have the potential of limiting the backhaul load. However, they are quite expensive in regards to the manual work required to use. As such, two additional investigations are carried out regarding the accuracy and cost of fingerprinting-based localisation methods:

1) How can the accuracy of traditional fingerprinting-based localisation methods be improved?

2) Can machine learning be used to lessen the amount of required manual work in fingerprinting?

System Model 3

This chapter introduces the system model which is used as the basis for the testing performed in the report. The system model is first presented in terms of the network topology it is based on. The topology covers the spatial relation between the BSs and the UTs, as well as the overall parameters for the BSs. Lastly, two channel models are presented, which are used in the design of the localisation methods and final results chapter.

3.1 Network Topology

The general structure of the network topology consists of a number of BSs placed within a close enough proximity to each other such that a UT will be in range of multiple BSs regardless of its position. The distances considered also means that the BSs are always considered to be in the far field of any transmitting UT. Furthermore, each BS is connected through a backhaul to a centralized server. In the context of location estimation, the BSs will transmit data over the backhaul which the centralized server will use to estimate the location of the UT. An illustration of this topology can be seen in figure 3.1.

Two different network topologies are used in this project. Both topologies share the same overall structure with the main difference being the density of the BSs. Taking inspiration from Shen et al. [2021], in first topology the BSs are placed within a close proximity to each other – closer than what would be necessary in a real use case. The second topology takes into account the use of a UMi (Urban Microcell) channel model 3GPP [2022] and is designed based on the specifications of 5G base stations. According to the paper by MacCartney et al. [2013], a typical UMi BS can achieve a coverage area with a radius of 200 meters. The specific size of the considered grids and density of the BSs is covered in more detail in section 4.2.

As illustrated in the network topology figure, the base stations are equipped with antenna arrays. Based on the usage of the UMi channel model, the BSs are designed in accordance with existing 5G BSs. For example, Flex ltd. Flex [2020] and Nokia Nokia [2023] make 5G BSs which consists of 32 to 128 antenna elements, for which half are vertically polarized, and the rest are horizontally polarized. The antennas are placed with a spacing of half the wavelength of the frequency of the considered transmissions, which is set at 28 GHz.

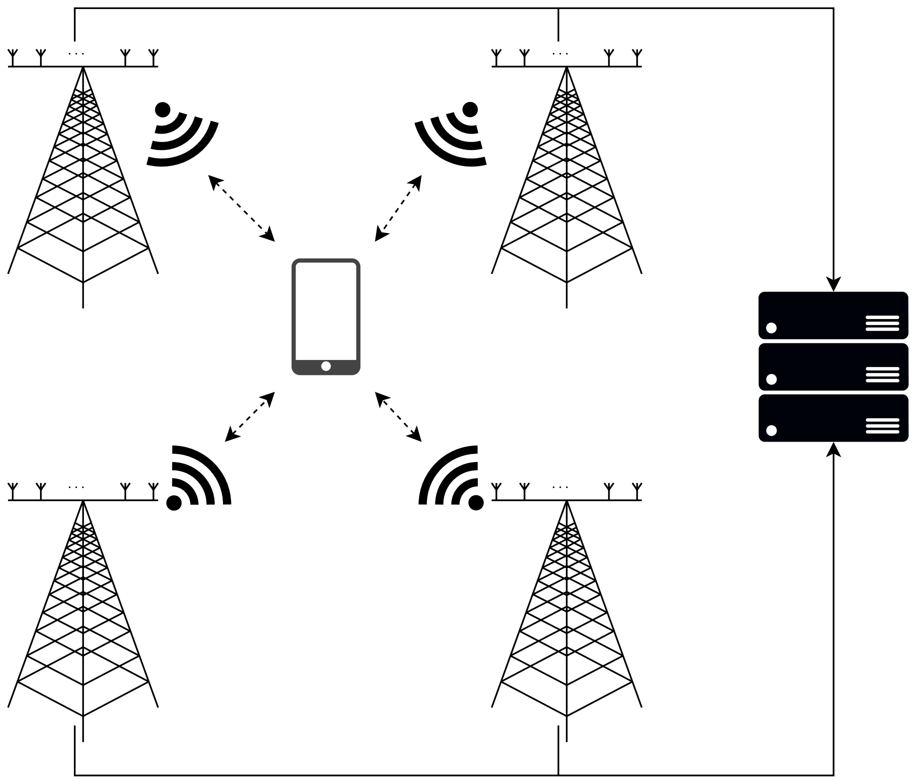


Figure 3.1. Illustration of an example network topology where 4 MIMO base stations are evenly spaced apart. All four BSs are within range of the user terminal and can wirelessly transmit to and receive from it, depicted by the dashed lines. The solid lines going from all BSs is the backhaul which is a wired connection to a centralised server.

3.2 Channel Models

Two channel models are considered for this project: a realistic channel model for urban environments and a LoS only channel model. The main channel model which the final results are based on, is the UMi channel model which models transmissions in urban areas where canyon like features are present in the street layouts 3GPP [2022]. Furthermore, the microcell aspects means that the model is specified for BSs with relatively small coverage areas. This lends itself nicely to the concept of distributed MIMO where base stations are placed at higher densities and cover smaller overlapping areas. The second channel model is a less complex model which only considers LoS paths. While this is not considered representative of any real use cases, it is still considered useful for the project as its relative simplicity aids the validation process of the localisation methods. If a localisation method is tested on the UMi model with unfulfilling results, it is not immediately clear if it is due to a flaw in the method or due to the high complexity of the model. By testing the methods on the LoS model first, the core idea of the method can be validated before testing it on the more challenging channel.

3.2.1 LoS Channel Model

The LoS channel model is inspired by the channel model used in the paper by Shen et al. [2021]. The paper presents a channel model with both a LoS and NLoS element, however, to acheive the desired simplicity as presented above, only the LoS aspect is used in this project. The model defines the channel coefficients of a channel between a transmitter k and receiver r as:

$$h_{kr}^{\text{LOS}} = \sqrt{\beta_{kr}} e^{j\mu_{kr}} a(\theta_{kr}), \tag{3.2.1}$$

where β is the LoS path loss, μ is the phase of the signal at the antenna which is closest to the transmitter, and $a(\theta)$ is the steering vector.
The LoS path loss, β, is defined as:

$$\beta_{kr} = \alpha_1 + 10\alpha_2 \log_{10}(d_{kr}) + \xi[\text{dB}], \tag{3.2.2}$$

with α_1 and α_2 being variables particular to the specific environment, and ξ being a normal random variable representing shadow fading. Considering the path loss model in equation (2.1.3), this path loss model is roughly based on the same structure though with a few slight alterations.
The phase at the closest receiving antenna, μ, is defined as:

$$\mu = \frac{d}{\lambda}, \tag{3.2.3}$$

where d is the distance between the transmitting and receiving antenna, and λ is the wavelength of the transmission.
The steering vector, $a(\theta)$, is defined for each receiving antenna as:

$$[a(\theta)]_n = e^{\frac{j2\pi(n-1)d_a}{\lambda}\cos(\theta)}, \quad n \in [1, N] \tag{3.2.4}$$

where θ is the AoA in radians, n is the antenna number starting at the closest antenna to the receiver, N is the total number of antennas at the receiver, and $\frac{d_a}{\lambda}$ is the ratio of the antenna spacing to the wavelength of the received signal.

3.2.2 UMi Channel Model

The UMi model is specified by the 3GPP standard in specification 38.901 [3GPP, 2022].

The channel coefficients are defined as a Ricean K-factor scaled addition of a LoS term and NLoS term:

$$h_{u,s}^{\text{LOS}}(\tau, t) = \sqrt{\frac{1}{K_R + 1}} h^{\text{NLOS}}(\tau, t) + \sqrt{\frac{K_R}{K_R + 1}} h_{u,s,1}^{\text{LOS}}(t)\delta(\tau - \tau_1), \tag{3.2.5}$$

where u is the receiver antenna, s is the transmitter antenna, τ is the path delay, t is the point in time where the channel coefficients are taken, δ is the Dirac-delta function, and K_R is the Ricean K-factor. The K-factor describes the relative power of the LoS path with respect to the NLoS paths. The K-factor follows a log-normal distribution which means that there will always be some amount of power over NLoS paths. As explained in 2.1, this will affect the location methods by introducing an element of noise compared to the purely LoS channel.

The LoS channel coefficients are defined using the following equation:

$$h_{u,s,1}^{\text{LOS}}(t) = \begin{bmatrix} F_{rx,u,\theta}(\theta_{\text{LOS,ZOA}}, \Theta_{\text{LOS,AOA}}) \\ F_{rx,u,\Theta}(\theta_{\text{LOS,ZOA}}, \Theta_{\text{LOS,AOA}}) \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} F_{tx,u,\theta}(\theta_{\text{LOS,ZOA}}, \Theta_{\text{LOS,AOA}}) \\ F_{tx,u,\Theta}(\theta_{\text{LOS,ZOA}}, \Theta_{\text{LOS,AOA}}) \end{bmatrix}$$
$$\cdot \exp\left(-j2\pi \frac{d_{3\text{D}}}{\lambda_0}\right) \exp\left(j2\pi \frac{\hat{r}_{\text{rx,LOS}}^T \bar{d}_{\text{rx},u}}{\lambda_0}\right) \exp\left(j2\pi \frac{\hat{r}_{\text{tx,LOS}}^T \bar{d}_{\text{tx},s}}{\lambda_0}\right) \exp\left(j2\pi \frac{\hat{r}_{\text{rx,LOS}}^T \bar{v}}{\lambda_0} t\right) \tag{3.2.6}$$

Compared to the LoS channel coefficients from the LoS channel model, the UMi models many more aspects of the communication. However, for the purpose of extracting localisation information, the models are not that dissimilar. The first exponential of the function $\exp\left(-2j\pi\frac{d_{3\text{D}}}{\lambda_0}\right)$ is a direct encoding of the phase at the specific receiving antenna u. $d_{3\text{D}}$ is the distance between the transmitting antenna s and the receiving antenna u and λ_0 is the base band frequency of the transmitted signal. This means that while the phase difference at each antenna may be more obfuscated in the UMi channel model than the LoS channel model, the information is still present which should make it possible to extract it in the same way.

However, the NLoS coefficients further complicate the extraction of the LoS AoA by encoding similar phase difference information for the NLoS paths into the LoS channel coefficients. The reason it will still be possible to extract the LoS path AoA from the channel coefficients, is due in part to the K-factor and that the power of each ray in each cluster is scaled by the number of rays per cluster. These two factors combined mean that the power of the LoS path at the receiver should be significantly higher than that of the individual NLoS paths, which should allow for the detection of the LoS path AoA.

After the channel coefficients are generated, the path loss is applied. The LoS path loss for the UMi model is calculated through one of two functions dependent on a threshold distance, called the breakpoint distance, between the UT and BS. The breakpoint distance, d'_{BP}, is calculated based on the frequency of the transmission and the heights of the UT and BS:

$$d'_{\text{BP}} = 4v_{\text{BS}}v_{\text{UT}}\frac{f_c}{c}, \tag{3.2.7}$$

where v_{BS} and v_{UT} are the heights of the BS and UT, f_c is the center frequency of the transmission, and c is the velocity of the transmission in free space which is $c = 3 \cdot 10^8 m/s$.

Based on this breakpoint distance, the LoS path loss for the UMi is:

$$\text{PL}_{\text{UMi-LOS}} = \begin{cases} \text{PL}_1 & 10m \leq d_{2\text{D}} \leq d'_{\text{BP}} \\ \text{PL}_2 & d'_{\text{BP}} \leq d_{2\text{D}} \leq 5km \end{cases} \tag{3.2.8}$$

$$\text{PL}_1 = 32.4 + 21\log_{10}(d_{3\text{D}}) + 20\log_{10}(f_c) + \xi[\text{dB}] \tag{3.2.9}$$

$$\text{PL}_2 = 32.4 + 40\log_{10}(d_{3\text{D}}) + 20\log_{10}(f_c) - 9.5\log_{10}((d'_{\text{BP}})^2 + (v_{\text{BS}} - v_{\text{UT}})^2) + \xi[\text{dB}], \tag{3.2.10}$$

where ξ[dB] is the normally distributed shadow fading with a standard deviation of 4.

The NLoS path loss is defined as the maximum of the LoS path loss $\mathrm{PL}_{\text{UMi-LoS}}$ and the NLoS path loss $\mathrm{PL}'_{\text{UMi-NLoS}}$ for all distances between 10 meters and 5 kilometers:

$$\mathrm{PL}_{\text{UMi-NLoS}} = \max(\mathrm{PL}_{\text{UMi-LoS}}, \mathrm{PL}'_{\text{UMi-NLoS}}) \tag{3.2.11}$$

$$\mathrm{PL}'_{\text{UMi-NLoS}} = 35.3\log_{10}(d_{3\mathrm{D}}) + 22.4 + 21.3\log_{10}(f_c) - 0.3(v_{\mathrm{UT}} - 1.5) + \xi[\mathrm{dB}], \tag{3.2.12}$$

where the shadow fading has a standard deviation of 7.82.

Design of Localisation Methods 4

This chapter covers the description of the baseline localisation method used for this project as well as the design of three ML-based localisation methods – two versions of a fingerprinting based method which uses two different ML structures and a channel coefficient based ML localisation method. The design of the proposed fingerprinting methods is presented as an iterative process where the improvements are based on intermediary validation results. Furthermore, the design of the first two proposed methods is concluded by illustrating the performance gain of the individual improvements when compared using the LoS channel. The third proposed localisation method follows a different localisation process from the fingerprinting based methods. The third method employs ML to estimate the position of UTs directly from the estimated channel coefficients.

4.1 Overview of Presented Methods

Before the design of the localisation methods is presented, the relation between the baseline and proposed methods is introduced. Along with the baseline, three proposed methods are designed and presented in this chapter. All of the presented methods are based on ISAC localisation where the existing communication between UTs and BSs are used to facilitate the localisation and tracking of UTs. The methods are further divided into two main approaches: A fingerprinting approach which uses estimated AoA values as fingerprints, and an ML based direct localisation approach where a DNN model predicts the coordinates of the UTs directly.

For the fingerprinting based methods, a baseline localisation method is presented which uses a DFT method for estimating the AoA values. Furthermore, two ML based method are proposed as improvements on the baseline fingerprinting method. The two proposed methods use two different ML structures for estimating the AoA values: one uses a CNN while the other uses a DNN. The relation between the presented methods is summarised and shown in figure 4.1.

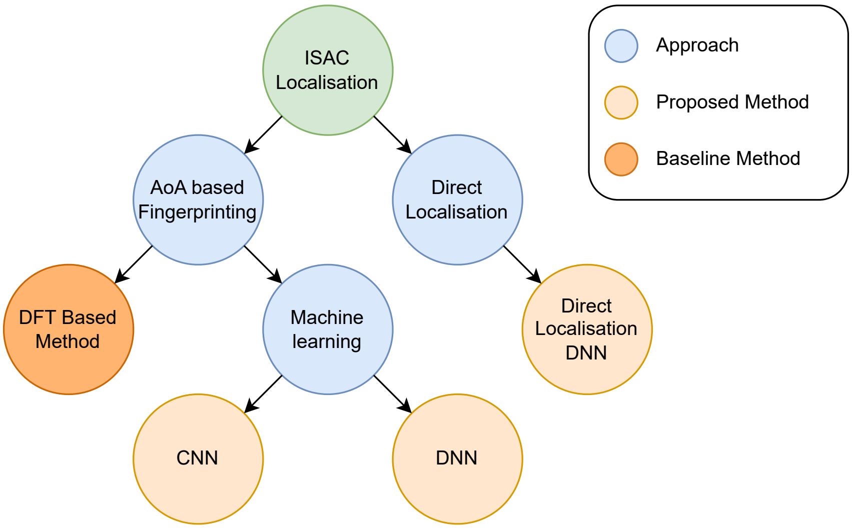


Figure 4.1. Overview of the relation between the baseline localisation method [Shen et al., 2021] and proposed localisation methods.

4.2 Data Generation for Method Validation

In order to validate the methods which are designed in this chapter, a way of generating data containing the channel coefficients from the channel models, presented in section 3, is needed. For this, there are two ways to produce the data: Gather the data from real transmissions produced by a test setup based on a realistic usage scenario, or simulate the transmissions. While the real data may be more attractive as they can be a closer representation of the data which would be received during real use, the main problem with this approach is the amount of data needed. It is considered infeasible to gather enough data to validate the localisation methods and for this reason, the use of simulated data transmissions is chosen for this project.

To streamline the process of generating data from the channel models, the Python library *Sionna* [Hoydis et al., 2022], which is a 5G and 6G physical layer simulator developed by NVIDIA, is used. Sionna is built on top of Tensorflow which allows for direct integration of the channel models with ML models.

While the UMi channel model is natively supported in Sionna, the LoS channel model is not. Instead, the channel coefficients for the LoS channel are generated manually according to equation (3.2.1) in subsection 3.2.1 on page 16. With these channel coefficients, a custom Sionna channel model is made which draws randomly from the pre-generated channel coefficients. Both channel models are implemented in an OFDM channel which manages the frequency shifts to subcarrier frequencies as well as the addition of white Gaussian noise.

In order to generate the channel coefficients for the validation tests, the exact topology which they are based on must be specified. In section 3.1, it was introduced that two grid sizes are considered for the validation tests. The small grid is 40×40 meters with a BS density of 9 per 40 m^2 and the large grid is 300×300 meters with a BS density of 9 per 300 m^2. For both grid sizes, the BSs are evenly spread across the grid with BSs placed at $x = 0, 20, 40$ and $y = 0, 20, 40$ for the small grid and at $x = 0, 150, 300$ and $y = 0, 150, 300$

for the large grid. Each BS is equipped with 32 antennas, where for the LoS channel model all 32 antennas are vertically polarized and for the UMi channel model, 16 antennas are vertically polarised and 16 are horizontally polarised.

4.3 Baseline AoA Based Fingerprinting

The baseline method is an AoA based fingerprinting method proposed in Shen et al. [2021], which is the same paper that the LoS channel model is based on. As described in section 2.2.1 fingerprinting consists of two steps: 1) an offline fingerprinting training step and 2) a fingerprint comparison step. In step one, fingerprints are collected for ANs (Anchor Nodes) which are UTs placed at reference positions. In step two, a fingerprint is collected for a specific UT whose position is to be estimated. The fingerprint collected in step two is then compared with the fingerprints collected in step one in order to estimate the position. Since the baseline is an AoA based fingerprinting method, the fingerprints consists of the AoA values estimated at all receiving BSs. This means that for each of the two steps, a method of estimating the AoA values is needed. For the baseline method a DFT based AoA estimator is used.

4.3.1 DFT Based AoA Estimations

The AoA estimator described in the baseline paper [Shen et al., 2021] uses a process of taking the matrix multiplication of estimated channel coefficients, h_{ar}, with an $N \times N$ normalised DFT matrix F, where N is the number of antennas on the BS, and an $N \times N$ rotation matrix $O(\phi)$. The rational behind this process is to find the frequency of the phase difference between the antennas. The transmitted signal is a (sine) wave with some frequency f. The phase difference across the antennas at the receiver is determined by the AoA and the distance between each antenna. This can be seen in equation (3.2.4), where the cosine of the AoA determines how quickly the phase changes from antenna to antenna on the receiver. The initial phase is determine by the distance between the transmitting antenna and the antenna on the receiver which is closest to the transmitter. This means that when the received channel coefficients for all antennas are considered, the values form a sine wave whose frequency is determined by the AoA and antenna spacing. The AoA estimator is defined based on a transmitting AN, a and receiving BS r:

$$\tilde{h}_{ar} = FO(\phi_{ar})h_{ar} \tag{4.3.1}$$

with F defined as:

$$F_{p,q} = N^{-\frac{1}{2}} e^{-j\frac{2\pi}{N}pq}, \quad \forall p, q \in \{0, \dots, N-1\} \tag{4.3.2}$$

and $O(\phi)$ defined as:

$$O(\phi_{ar}) = diag(e^{jn\phi_{ar}}), \quad \forall n \in \{0, \dots, N-1\}, \tag{4.3.3}$$

where ϕ is an optimisation parameter with a range from $\phi \in [-\frac{\pi}{N}, \frac{\pi}{N})$, and $diag$ is an $N \times N$ diagonal matrix.

Multiplying the channel coefficients with the DFT matrix, is equivalent to taking a DFT of the channel coefficients which returns the frequency of the phase difference. In particular, the frequency is found based on the maximum absolute value of $\tilde{h}_{ar}$. In this process, however, it is not the actual frequency that is of importance. Instead, it is the index, i, of the maximum absolute value which is needed to calculate the AoA. However, simply checking the maximum value for this DFT multiplication does not guarantee that the correct AoA is found. This is why the rotation matrix involving the optimisation value ϕ is used as well. Changing the value of ϕ, will cause the values of $\tilde{h}_{ar}$ to differ slightly. The goal is then to find the value of ϕ which results in the maximum absolute value of $\tilde{h}_{ar}$ and use that along with the index i to calculate the AoA. Based on these two values, ϕ and i, the AoA is estimated using one of two equations. The exact equation is determined by the value of i, where equation (4.3.4) is used for values of $i \in [1, \frac{N}{2}+1)$ and equation (4.3.5) is used for values of $i \in [\frac{N}{2}+1, N]$:

$$\cos(\theta_{ar}) = \frac{\lambda}{d_a}\left(\frac{i-1}{N} - \frac{\phi_{ar}}{2\pi}\right), \quad i \in [1, \frac{N}{2}+1) \tag{4.3.4}$$

where λ is the wavelength of the transmission, and d_a is the antenna spacing.

$$\cos(\theta_{ar}) = \frac{\lambda}{d_a}\left(\frac{i-1}{N} - \frac{\phi_{ar}}{2\pi} - 1\right), \quad i \in [\frac{N}{2}+1, N] \tag{4.3.5}$$

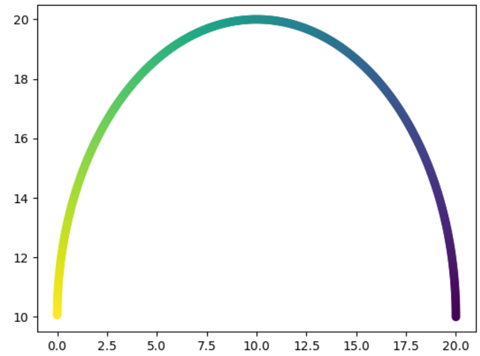
In order to combat the randomness which can occur for the angle estimations in multipath conditions or when the SNR is low, each AoA value in the fingerprints are based on multiple angle estimates using separate estimated channel coefficients. Specifically, the angle estimates used in the fingerprints are based on a maximum likelihood estimation approach which is used to calculate the statistical mean of the samples of θ_{ar}.

Given S samples of θ_{ar} and i, the vectors ψ_s and x_s, $s \in 1, \dots S$ contain the samples of θ_{ar} and i respectively. The probability that a sample of x_s is equal to a specific index value i is denoted by p_i and the corresponding set is $\{I_i\} = \{s | x_s = i, \quad \forall s \in \{1, \dots S\}\}, \quad \mu_i = |\{I_i\}|$. Based on this, statistical mean of θ_{ar} is calculated as:

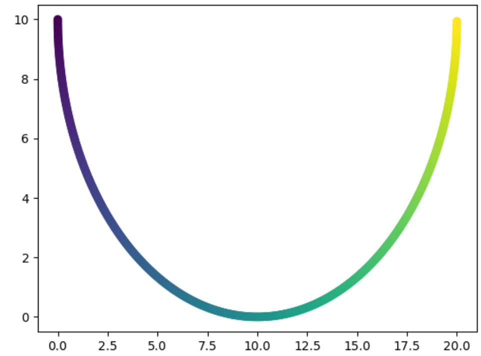
$$w_{ar} = \sum_i \left(\hat{p}_i \sum_{s \in \{I_i\}} \frac{\psi_s}{\mu_i}\right), \tag{4.3.6}$$

where $\hat{p}_i$ is calculated as $\frac{\mu_i}{S}$.

Since the antenna arrays used with the LoS channel model and that of the UMi model have different antenna polarisations, the number of total antennas used in equation (4.3.2) to (4.3.5) are different even though the total amount of antennas are the same. For the LoS channel model, all 32 antennas use the same polarisation as the transmitter. To visualise the performance of the AoA estimator with the LoS channel, a brief test is conducted. The estimator is tasked with estimating the AoA of transmissions from positions following a circle with a BS at its center. The positions of the transmitters are shown in figure 4.2. Each estimation is done based on estimated channel coefficients for a single pilot transmission. The circle is cut into two halves along the x-axis as the AoA estimator in (4.3.4) and (4.3.5) can only estimate the cosine of the angle and is incapable of distinguishing between positive and negative sine. This means that if the entire circle is considered at once, the AoA estimations for both halves would be predicted with a positive sine. The points along the circle are coloured in a gradient from yellow to blue through green. This is done so the estimated angles can be evaluated on their accuracy. If the estimated points are perfectly accurate, then the gradient of the estimated points should follow that seen in figure 4.2. Any deviation from this continuous gradient will be considered as incorrect estimates. The angle estimates using this estimator for the LoS channel are shown in figure 4.3.

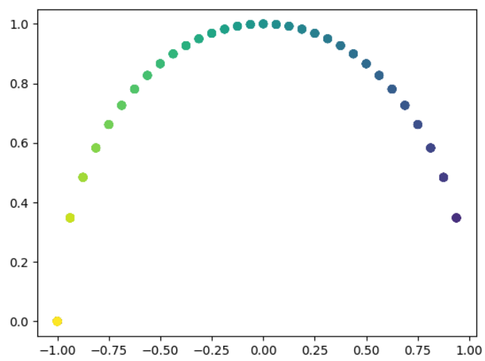


(a) First half

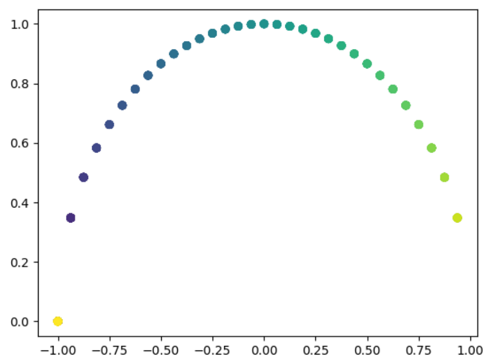


(b) Second half

Figure 4.2. Circle of UT positions circling a BS at position 10,10 with a radius of 10 meters. Each half circle contains 500 equally spaced UTs.



(a) First half



(b) Second half

Figure 4.3. Estimated AoAs based on the equations in (4.3.4) and (4.3.5) using 32 vertically polarised antennas with the LoS channel model. The estimated AoAs are projected onto the unit circle. The transmissions are made from positions following a circle around the BS placed at the center with a radius of 10 meters.

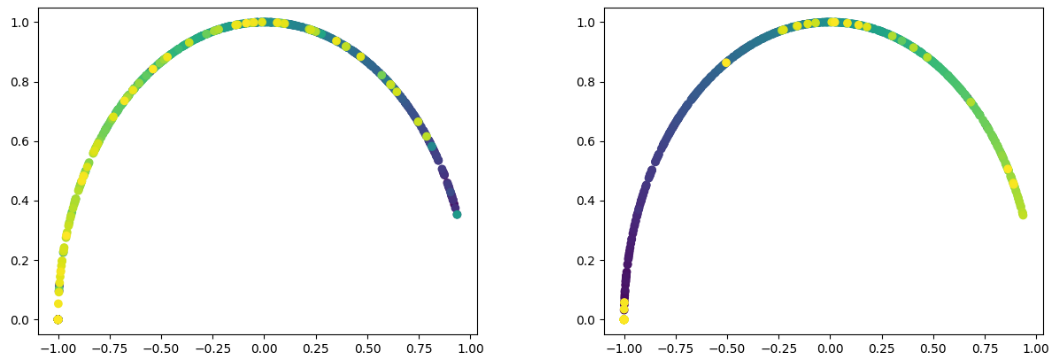
There are three main things which are noteworthy when inspecting the angle estimates. First, the estimates fall into exactly 32 groups in each of the circle halves. This lines up with the total number of antennas used in the angle estimation. This means that despite the use of the phase rotation optimisation value ϕ, the resolution of the angles are not improved. It is expected that this optimisation value mainly serves a purpose in the case of multipath transmissions where it can help find the path with the strongest power. This is backed up by the following test using the UMi channel model in figure 4.4, where the estimates are more spread out despite using fewer antennas.

The second thing to note is that there is a blind spot where the estimator is not able to estimate angles correctly. This blind spot is from transmissions which are parallel or close to parallel with the antenna array. This does, however, make sense as the cosine of these AoAs is 0 which results in no phase difference over the antennas. From the color of the estimates in the second half, it seems that transmissions from both sides of the antenna array are estimated to the same angle of 180.

Lastly, despite the low resolution and presence of the blind spot, the estimates look to be fairly consistent. The half circles show a discretised version of the gradient seen with the original points in figure 4.2.

In order to test the angle estimations for the UMi channel, the points need to be shifted by $\frac{\pi}{2}$ due to the orientation of the antenna array. With this change, the points in the first half are placed behind the BS, and the points in the second half are placed in front. For the LoS model, receiving a signal from the front or back had no impact on the angle estimations as the channel model did not distinguish signals coming from different angles in sine. However, as seen in figure 4.4, there is a significant difference in the angle estimates from the front and back.

Besides the difference in the front and back, it can be seen that the angle estimation is able to take advantage of the optimisation parameter to get a more continuous gradient of the angle estimates. However, it is not able to estimate all angles equally well. There is a significant blind spot similar to that for the LoS channel, however, the estimated angles are more spread for the angles in this blind spot.



(a) First Half – Back *(b)* Second Half – Front

Figure 4.4. Estimated AoAs based on the equations in (4.3.4) and (4.3.5) using 16 vertically and horizontally polarised antennas with the UMi channel model. The estimated AoAs are projected onto the unit circle from the perspective of the BS. The transmissions are made from positions following a circle around the BS placed at the center with a radius of 10 meters.

4.3.2 Fingerprint Training

The first step of the localisation method is the offline fingerprint training where fingerprints are gathered for known locations. This consists of transmitting from a set of locations and estimating the AoA at each BS which receives the transmissions. These devices which are placed at the various locations are called ANs. While there are multiple ways of using the fingerprints to estimate the locations of UTs in the online phase, intuitively, the more densely the ANs are placed, the higher the accuracy of the location estimates can be.

4.3.3 Location estimation

The location estimation process for the baseline consists of three main steps. The first step is to find the relevant ANs to compare the UT's fingerprint with. This is done based on the coverage range of the BSs. Only the ANs which are in range of the BSs which received the transmissions from the UT is included in the location estimation. In the grid sizes considered for the validation tests, all ANs are considered to be in range of all BSs. As such, all ANs are used for each location estimation.

The second step consists of comparing the fingerprints of the relevant ANs with that of the UT. This is done using the a WMSE (Weighted Mean Squared Error) function which is weighted by a confidence factor for the angle estimates θ_{ar} in the fingerprints. The confidence factor of θ_{ar} is calculated using information entropy in order to characterize the variance of θ_{ar}:

$$\eta_{ar} = exp(-E_{ar}), \quad r \in [1, \dots, R], \tag{4.3.7}$$

where R is the amount of BSs in each fingerprint, and the entropy, E_{ar}, is calculated as:

$$E_{ar} = \sum_i -\hat{p}_i \log_2 \hat{p}_i, \tag{4.3.8}$$

In order to use the confidence factors η_{ar} as weights in the WMSE, they are normalised as η'_{ar} such that $\sum_{r=1}^{R} \eta'_{ar} = 1$. Based on these weights, the WMSE of the fingerprints between UT k and all ANs is calculated as:

$$X_k = \sum_{r=1}^{R} \left(\eta'_{ar}(w_{ar} - \hat{\theta}_{kr})^2 \right), \quad \forall a \in \mathcal{A}, \tag{4.3.9}$$

where $\hat{\theta}_{kr}$ is the fingerprint of the UT k, w_{ar} is the fingerprint of AN a, and $\mathcal{A}$ is the set containing all ANs.

The third and final step consists of using the best ANs to estimate the location of the UT. Instead of using only one AN for this, a subset of ANs, $\mathcal{A}_m$, with the lowest MSEs is used. The exact amount of ANs in $\mathcal{A}_m$ is not fixed. Instead, it can be changed to include more or less ANs in each estimation. By including more ANs, the probability that the mean location of the ANs is close to the UT should increase. However, including too many ANs also increases the risk that more outlier AN locations are included in $\mathcal{A}_m$. It was found that the number of ANs which most often gave the best location accuracy was 4. Using these ANs the estimated coordinates, k_x and k_y, of UT k is found by taking the average position of the ANs contained in A_m:

$$k_x = \frac{1}{A_m} \sum_{a \in \mathcal{A}_m} \zeta_{x,a}, \tag{4.3.10}$$

where $\zeta_{x,a}$ is the x-coordinate of AN a.

$$k_y = \frac{1}{A_m} \sum_{a \in \mathcal{A}_m} \zeta_{y,a} \tag{4.3.11}$$

where $\zeta_{y,a}$ is the y-coordinate of AN a.

4.3.4 Accuracy of Location Estimations

Combining the DFT-based AoA estimations with the fingerprinting-based localisation method, a small validation test of the estimation accuracy is carried out. This is done to investigate if there are certain characteristics to the inaccuracies of the baseline method. These inaccuracies are then used as the basis for some of the improvements presented for the proposed method in section 4.4. For these tests, the baseline localisation method is tested in the LoS and UMi channel models for both the large and the small grid sizes. For each test, a total of 1 000 random positions are estimates, and the angle estimates recorded in the fingerprints are made using estimations of channel coefficients for 20 different pilot transmissions from the same locations. For this, the amount of ANs which are used is specified. For the small grid size, ANs are placed 1 meter apart, totalling 1672 ANs. For the large grid, the ANs are placed 4 meters apart, totalling 5772 ANs. Due to the increased distance between the ANs for the larger grid, the small grid performance is likewise presented with ANs spaced 4 meters apart.

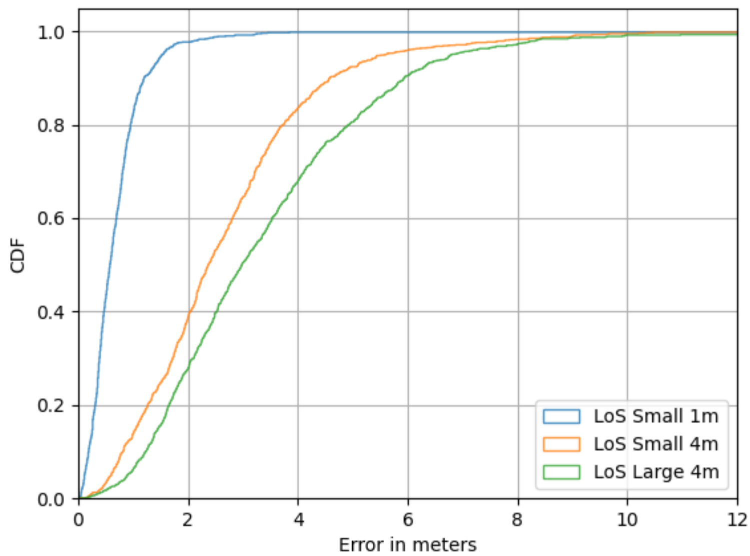
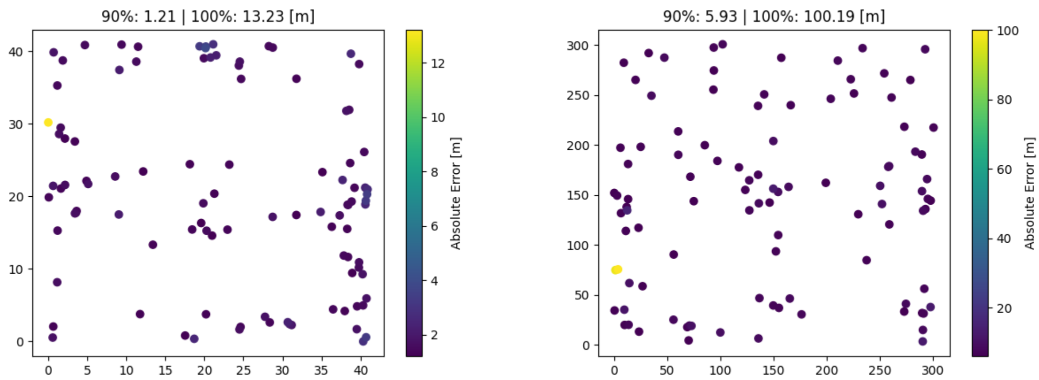


Figure 4.5. CDF of the estimation errors made using the baseline location estimation method for the LoS channel model. The plots show the difference in accuracy for the small grid size with ANs placed 1 and 4 meters apart and the large grid with ANs placed 4 meters apart.

As seen in figure 4.5, there is a difference in the accuracy of the localisation method for the small and large grid, despite there being the same distance between the ANs. However, looking at figure 4.6, it can be seen that the errors occur in similar locations, where the UTs are vertically or horizontally aligned with the BSs. As explained earlier, these angles result in a phase difference of 0 across the antennas. Furthermore, when a signal is received at 90 or 270 degrees, there is no difference in distance to the antennas (assuming far field propagation) which means the phase is again the same across all antennas. This means that when a signal is received at either of the 4 directions, the phase information cannot distinguish them. This was also shown in the angle estimates from the circle test where it could be seen that the angle estimates where clumped up in these cases leaving blind spots.

If an AoA estimation of a specific angle is incorrect, intuitively this is bad for the fingerprint comparison. However, if the error occurs in such a way that the outcome of all estimations of a specific true angle is the same, then it will pose no problems during the fingerprint comparison step. In spite of this, when the estimations of multiple true angles all result in the same incorrect angle, the fingerprint comparison is compromised. It is assumed that this is the problem which is causing the large errors in the positions shown in figure 4.6.



(a) Small grid *(b)* Large grid

Figure 4.6. Visualisation of the true locations of the largest localisation errors using the baseline estimation method on the LoS channel model. The distance between the ANs is 1 meter for the small grid and 4 meters for the large grid. The color of the points is related to the error of the estimation following the gradient bar to the right of each plot.

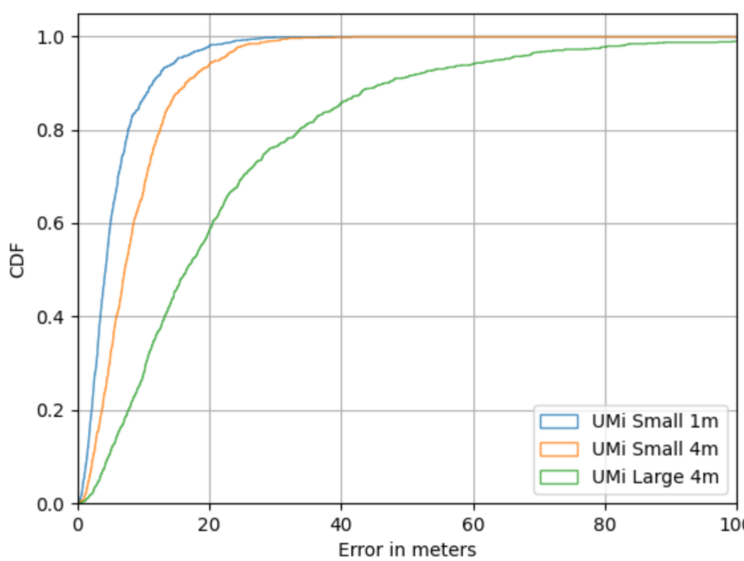
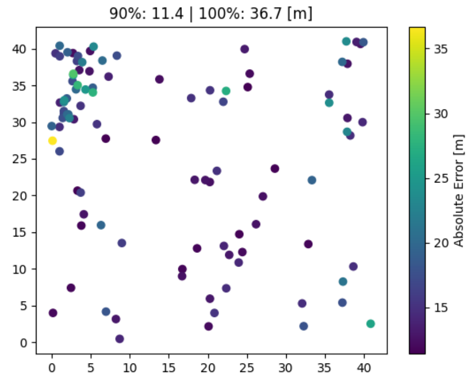


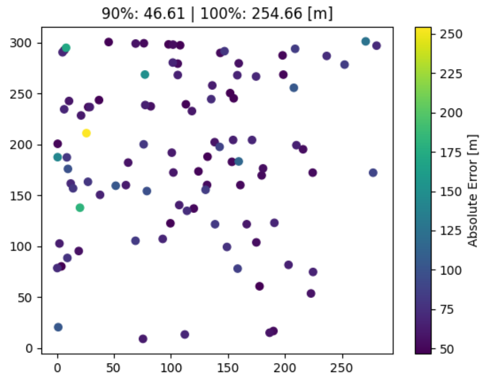
Figure 4.7. CDF of the estimation errors made using the base line location estimation method for the UMi channel model. The plot shows the difference in accuracy for the small grid size with ANs placed 1 and 4 meters apart and the large grid with ANs placed 4 meters apart.

Considering the results for the UMi channel, shown in figure 4.7, the same relation between the accuracy is seen for the large grid and small grid with ANs 4 meters apart. However, for the UMi model, the disparity is even more pronounced. Furthermore, the accuracy has drastically decreased compared to the LoS channel model. This is, however, expected as the UMi model includes multipaths, some of which are NLoS paths. Comparing the 90th percentile error for the LoS and UMi channel in the small grid, the accuracy drops from 1.21 to 11.4 meters.

Looking at the positions where the biggest errors occur in figure 4.8, it seems that for the baseline method, the biggest problem in the UMi channel is not when the transmissions are in line with the BSs. While this was the case for the LoS channel, it is assumed that the increased noise caused by the multipaths in the UMi channel is a bigger problem. Furthermore, the plots suggest that there may be an issue regarding the orientation of the antenna arrays on the BSs. Mainly in the larger grid there is a tendency for the larger errors to occur towards the left side of the grid. While it is not a strong tendency, it lines up with what was observed in the angle estimations shown for the circle test, shown in figure 4.4.



(a) Small grid 1m



(b) Large grid 4m

Figure 4.8. Visualisation of the true locations of the largest localisation errors using the baseline estimation method on the UMi channel model. The distance between the ANs is 1 meter for the small grid and 4 meters for the large grid. The color of the points is related to the error of the estimation following the gradient bar to the right of the plot.

4.4 Proposed AoA Based Fingerprinting

Through the validation testing of the baseline localisation method, certain aspects of the method were identified as shortcomings with the possibility of improving upon them. This section will cover these improvements and compare them briefly with the baseline method.

4.4.1 ML-Based AoA Estimation

Instead of using the DFT-based method for estimating the AoAs, this project investigates the use of DNN and CNN ML models for AoA estimation. The prediction of AoAs using ML models are considered for the two channel models separately. This is due to the complexity difference of the two channels models where it is not expected that the same ML structure will perform the best for both.

DNN for LoS channel model

For the LoS channel model, a DNN model is first tested. The DNN model takes estimated channel coefficients for a single pilot as input and predicts the arccos of the cosine of the AoA, which results in an estimation range of 0 to π. This is done since the AoA is only encoded into the channel coefficients as the cosine of the AoA in the LoS channel model.

In order to determine the exact structure of the DNN, a relatively simple hyperparameter tuning process is carried out, where the validation loss is considered as the performance metric. The models are trained on the same data set of channel coefficients recorded from 100 000 randomly generated UTs within the 40x40 grid and secondly within the 300x300 grid. Since there are 9 BSs which each receive the transmissions from the UTs over 32 antennas, the entire data set consists of 900 000 target angles which are represented through estimated channel coefficients from the 100 000 positions. In order to get the validation loss, the training set is split [9:1], leaving 810 000 training targets and 90 000 validation targets. While the DNN might be able to work with the complex numbers directly, the channel coefficients are split into their real and imaginary parts making the input dimension of the DNN 64 nodes. Furthermore, the data is normalised based on the individual sets of channel coefficients. While this means that the path loss information is lost, this is not important for the angle estimations.

For the hyperparameter tuning in both grids, the loss is calculated using the MAE (Mean Absolute Error) in radians and the model uses the Adam optimiser with a learning rate of 0.001. The parameters which are investigated in the tuning process are: Number of hidden layers, size of hidden layers, and activation function on hidden layers.

From the hyperparameter tuning it was found that while there is a difference in loss between the DNN models for the two grid sizes, both models benefit from the same parameters changes. Mainly it was found that the number of hidden layers and size of hidden layers did not have a significant impact on the loss. For the hidden layers, several structures of varying sizes where tested with no tangible difference in the loss. Instead, the main factor was the activation function of the hidden layers, where the ReLu function was found to be the best performing activation function.

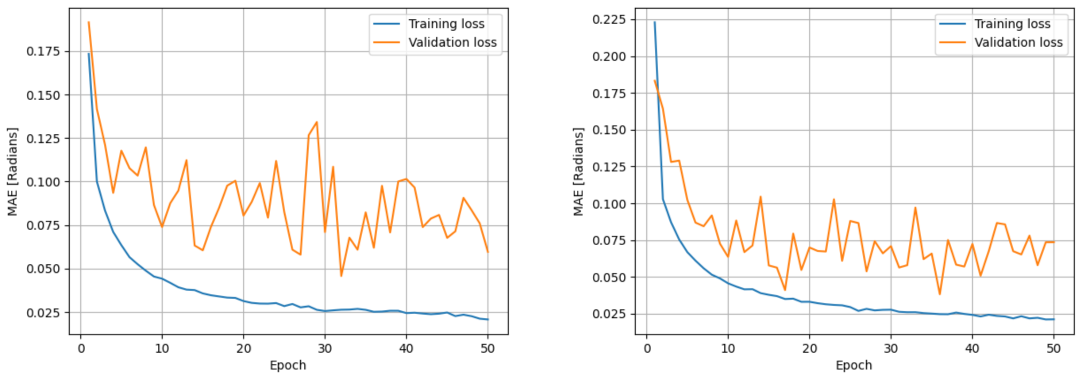
Based on the hyperparameter tuning, the final structure of the DNN is shown in table 4.1.

Layer Index	Layer Type	Size	Activation
1	Input layer	64	N/A
2	Normalisation layer	N/A	N/A
3	Dense Layer	128	Relu
4	Dense Layer	64	Relu
5	Dense Layer	32	Relu
6	Output Layer	1 + bias	Linear

Table 4.1. The structure of the DNN angle estimation neural network for the LoS channel model.

With these parameters, the two models where trained over 50 epochs, saving a unique model at each epoch. At the end of the training, the validation and training loss is investigated to determine what epoch produced a model with a desirable validation loss. This is based on when the improvement in validation loss stagnates as well as the relation between the validation loss and training loss. Despite the fact that the validation loss might continue to improve slightly after the stagnation point, it is often the case that the training loss keeps improving at a significantly faster rate. This means that the model is starting to overfit, which is undesirable. As a result, the epochs are chosen where the validation loss is low, and the training loss has not yet decreased too far from the validation loss.

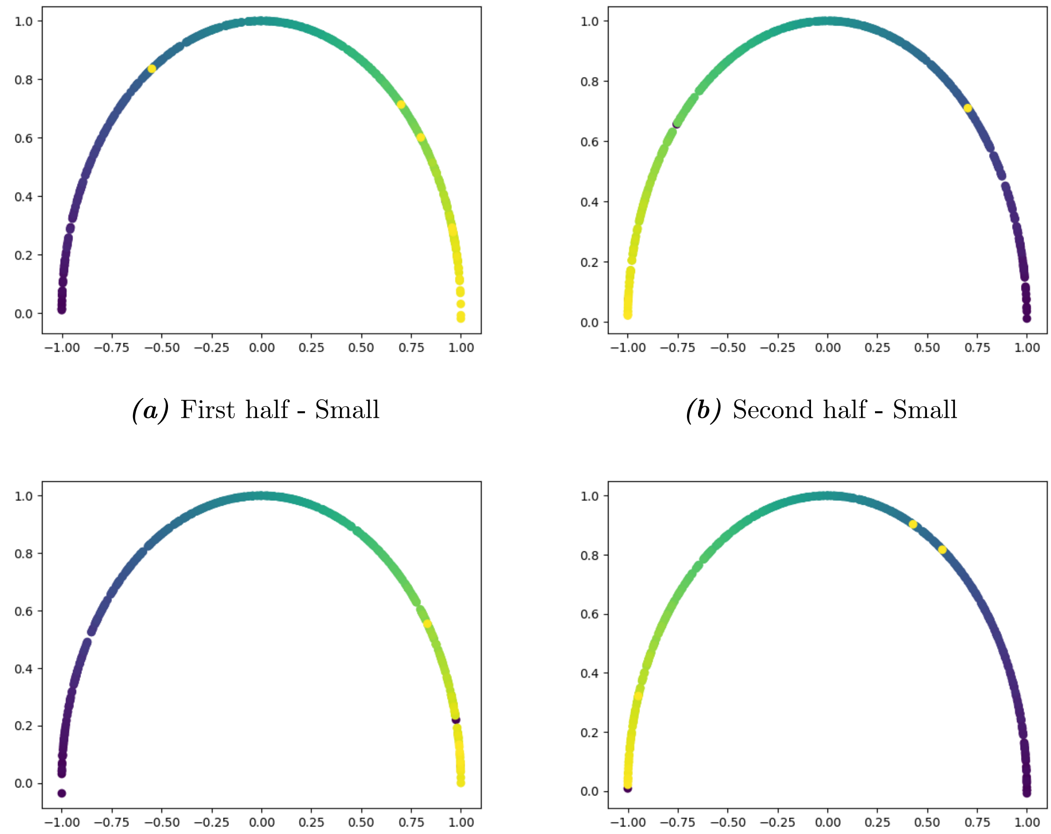
The validation loss and training loss of the DNN model is shown in figure 4.9. While the validation loss for both models is relatively unstable, both models seem to converge between 10 and 20 epochs. As such, the small grid uses the model generated at 15 epochs with a validation loss of 0.061 and the large grid uses the model at 17 epochs with a loss of 0.041.



(a) Small grid ***(b)*** Large Grid

Figure 4.9. Training and validation loss for the DNN model trained on the LoS channel model for the small and large grid size.

Comparing the angle estimations made using these models in figure 4.10 with those from the DFT-based estimation in figure 4.3, the DNN manages a more continuous gradient similar to that for the baseline in the UMi channel in figure 4.4. The estimates show the same problem of estimating incorrectly at 0 and 180 degrees. While this is expected due to the lacking information present in the data at these angles, the estimations of these angles are now more spread. The DNN does not simply estimate the same angle for both 180 and 0 degrees.



(a) First half - Small

(b) Second half - Small

(c) First half - Large

(d) Second half - Large

Figure 4.10. Estimated AoAs based on the DNN model tuned for the small and large grid size using 32 vertically polarised antennas with the LoS channel model. The estimated AoAs are projected onto the unit circle. The transmissions are made from positions following a circle around the BS placed at the center with a radius of 10 meters for the small grid size and 100 for the large grid. The points with a negative sine are a consequence of the DNN model predicting a negative radian for the angle.

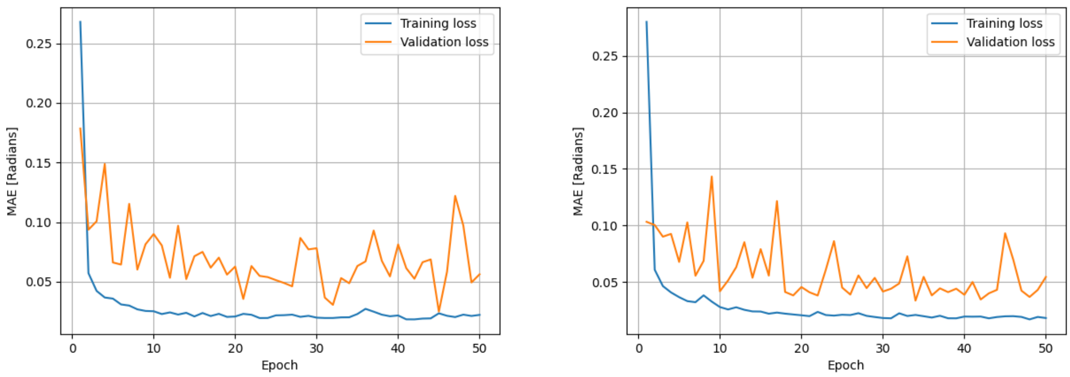
CNN for LoS Channel Model

While the above shows that a DNN can achieve relatively accurate angle estimations, it might not be the best model structure for this specific task. In the paper by Naseri et al. [2022] the authors investigate the use of a 1D CNN for AoA estimations based on channel coefficients. The CNN makes intuitive sense given its use of a kernel which can perform operations similar to that of the DFT in the baseline. However, since it is an ML model, it may be more flexible and able to generalise better than the DFT based estimations. As a result, a similar structure to that presented in the paper is tested for the AoA estimations done in this project. The differences to the structure presented in the paper is the size of the input layer, the omission of dropout layers, and the omission of pooling layers. The adapted structure of the CNN is shown in table 4.2.

Layer Index	Layer Type	Size	Kernel Size	Activation
1	Input Layer	64x1	N/A	N/A
2	Normalisation Layer	N/A	N/A	N/A
3	Conv1d	64	3	Relu
4	Conv1d	64	3	Relu
5	Normalisation Layer	N/A	N/A	N/A
6	Conv1d	64	3	Relu
7	Conv1d	256	3	Relu
8	Normalisation Layer	N/A	N/A	N/A
10	Flatten	N/A	N/A	N/A
11	Dense Layer	1024	N/A	Relu
12	Dense Layer	768	N/A	Relu
13	Dense Layer	512	N/A	Relu
14	Dense Layer	256	N/A	Relu
15	Output Layer	1 + bias	N/A	Linear

Table 4.2. The structure of the CNN angle estimation neural network for the LoS channel model. The size parameter refers to the output size when addressing the convolutional layers.

A plot of the loss for the training of the two CNN models is shown in figure 4.11. Similar to the DNN models, the validation loss for the CNN models are relatively unstable, however, they seem to converge after 10-20 epochs. Despite starting with a lower loss than the DNN models, both CNN models converge at roughly the same validation loss. The small grid uses the model at 21 epochs with a validation loss of 0.036 and the large grid uses the model at 10 epoch with a loss of 0.042.

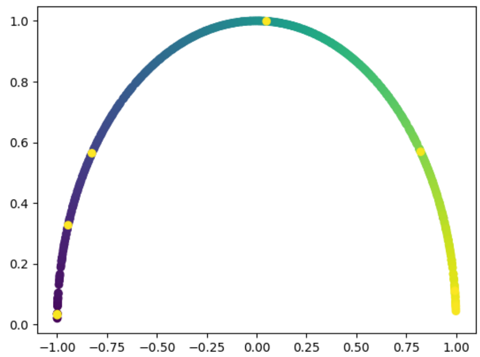


(a) Small grid ***(b)*** Large Grid

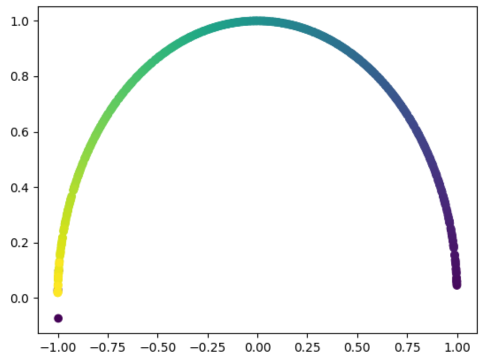
Figure 4.11. Training and validation loss for the CNN model trained on the LoS channel model for the small and large grid size.

Both the validation loss and the angle estimates shown in figure 4.12 show a similar estimation accuracy to the DNN model. However, when considering that both ML structures achieve nearly perfect performance in their circle tests, the similarity of their results is not surprising. Both structures show a nearly continuous gradient with few out of place estimations. Furthermore, while both structures do suffer from the same blind spots, the severity of the blind spots is significantly less than those for the baseline AoA estimator.

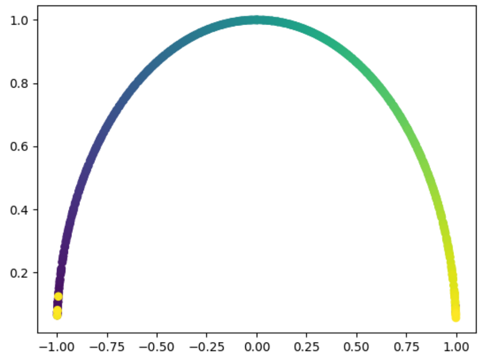
With this said, it is assumed that this performance is due to the relative simplicity of the LoS model. As such, both ML structures are tested and re-evaluated in the UMi channel.



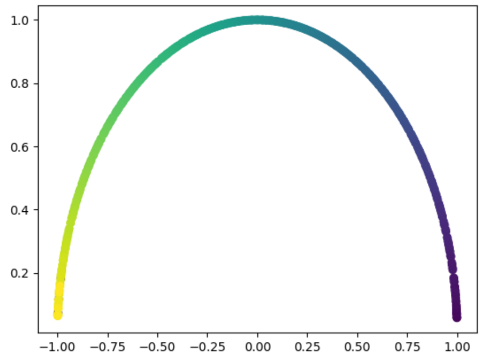
(a) First half - Small



(b) Second half - Small



(c) First half - Large



(d) Second half - Large

Figure 4.12. Estimated AoAs based on the CNN model tuned for the small and large grid size using 32 vertically polarised antennas with the LoS channel model. The estimated AoAs are projected onto the unit circle. The transmissions are made from positions following a circle around the BS placed at the center with a radius of 10 meters for the small grid size and 100 for the large grid. The points with a negative sine are a consequence of the CNN model predicting a negative radian for the angle.

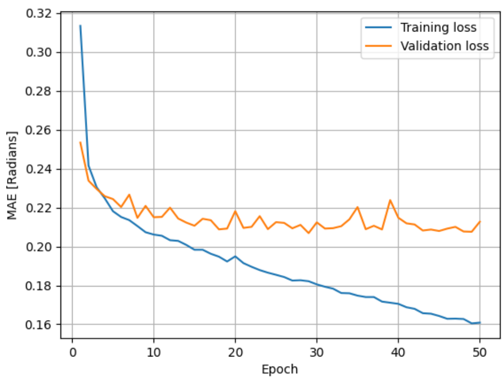
DNN for UMi channel model

Due to the increased complexity of the UMi channel model, it is not assumed that the same model structures, which work for the LoS channel model, will work for the UMi model. As a result, a separate hyperparameter tuning process for the DNN and the CNN model structures is carried out. Despite the fact that the grid size did not have a significant impact on the training of the ML models for the LoS channel model, both grid sizes are still tuned independently. It makes sense that the results are similar for the LoS channel as the only difference between the grids is the path loss values which when normalised is used is removed anyways. For the UMi, however, the model includes NLoS paths which are assumed to impact the channel coefficients differently at small and large distances.

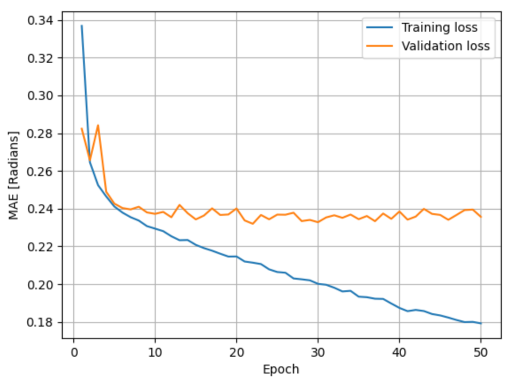
Before the results of the tuning is presented, the necessary changes to the training data are briefly discussed. For the LoS channel, the estimated channel coefficients where simply divided into their real and imaginary parts and normalised. For the UMi channel model this does, however, not work. Regardless of ML model structure, the models were incapable of learning using this data. They would always converge to predicting the same angle regardless of input, resulting in a loss of 0.79 – exactly the middle of the range of 0 to π radians. As described in subsection 3.2.2, the channel coefficients for the UMi channel model are affected by more factors which obfuscate the AoA information. As such, it is expected that this inability to learn is caused by the relatively higher complexity of the UMi channel which makes it too difficult to extract the correct information directly from the channel coefficients. As a result, the input is changed using a feature extraction step where the phase at each antenna is calculated. By calculating the phase before the input is passed to the ML model, the information which is assumed to be the most important is isolated. The phase is still affected by the noise at the receiver and the multipaths, but the ML models will have something more concrete to work with when estimating the AoA.

The hyperparameter tuning is performed on the same parameters as for LoS channel model. To reiterate, these are the number of hidden layers, size of hidden layers, and activation function on hidden layers. Despite the change in input data, parameters which affected the loss where similar to those for the LoS channel. The ReLu activation function still performs the best, with the Sigmoid function failing to learn anything at all. However, for both the small and large grid size, the number and size of the hidden layers had a bigger impact on the loss compared to the LoS channel. The size of the model is therefore increased to include 6 hidden layers, each of which has an increased number of nodes compared to that for the LoS channel model.

Figure 4.13 shows that the DNN model converges just before 10 epochs for the small grid and at around 5 epochs for the large grid. As such, the model for the small grid is stopped at epoch 8 hitting a validation loss of 0.215 and the model for the large grid is stopped at epoch 6 hitting a validation loss of 0.240.



(a) Small grid



(b) Large grid

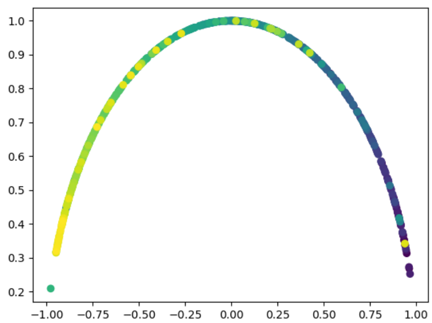
Figure 4.13. Training and validation loss for the DNN model trained on the UMi channel model for the small and large grid size.

The final structure for the DNN models used for the UMI channel model is shown in table 4.3.

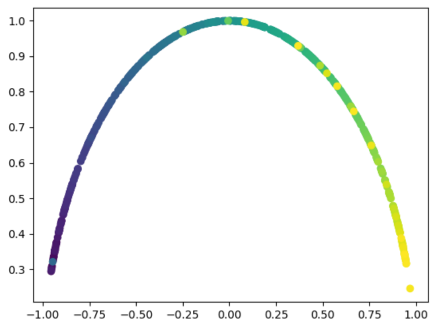
Layer Index	Layer Type	Size	Activation
1	Input layer	32	N/A
2	Normalisation layer	N/A	N/A
3	Dense Layer	1280	Relu
4	Dense Layer	1024	Relu
5	Dense Layer	512	Relu
6	Dense Layer	256	Relu
7	Dense Layer	128	Relu
8	Dense Layer	64	Relu
9	Output Layer	1 + bias	Linear

Table 4.3. The structure of the DNN angle estimation neural network for the UMi channel model.

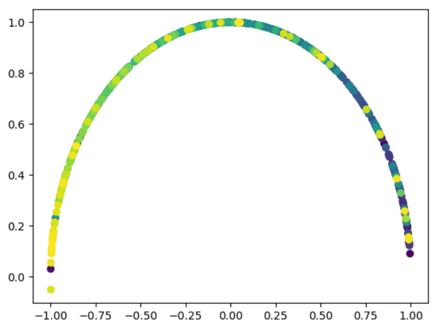
Looking at the angle estimations made by the DNN model in figure 4.14, it can be seen that the DNN model still suffers from inaccurate estimations for signals received at the back of the antenna array. Besides the difference on front to back, the figure also shows that the estimations are more difficult at the same areas as for the simple channel model – when the cosine is close to 1 and -1. However, unlike the estimations shown for the DNN in the LoS model, the estimations are generally more noisy – even for the signals received from the front. The gradient is not as continuous and it is no longer exclusively the true angles from the aforementioned blind spots which fail.



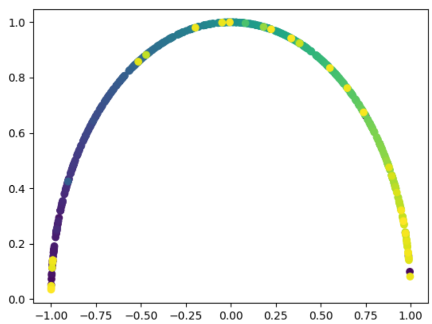
(a) First half - Small



(b) Second half - Small



(c) First half - Large

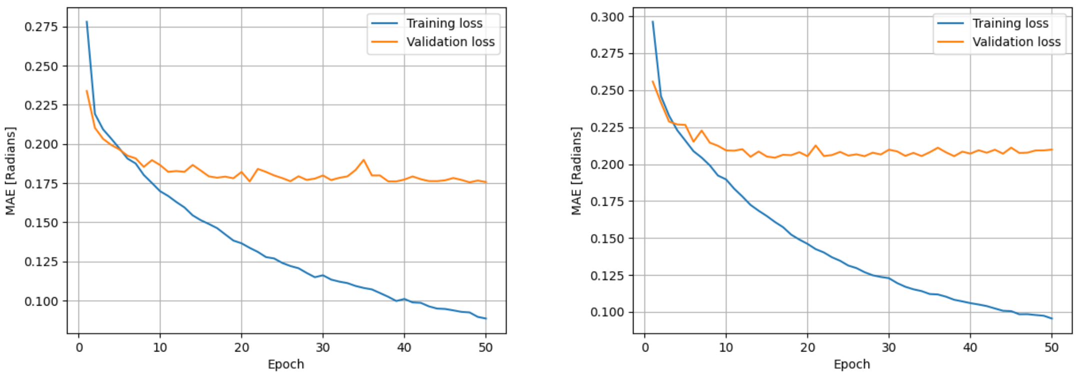


(d) Second half - Large

Figure 4.14. Estimated AoAs based on the DNN model tuned for the small and large grid size using 16 vertically and 16 horizontally polarised antennas with the UMi channel model. The estimated AoAs are projected onto the unit circle. The transmissions are made from positions following a circle around the BS placed at the center.

CNN for UMi channel model

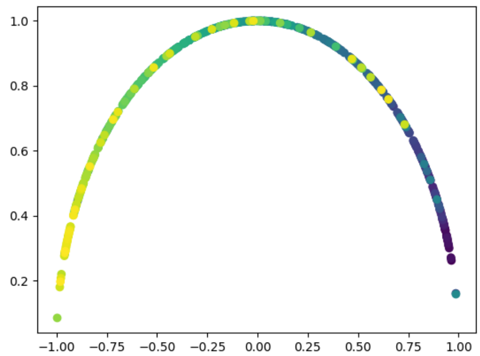
The CNN model is based on the same structure as presented for the LoS channel in table 4.2, however, with the key difference of the reduced input size to accommodate the phase values instead of real and imaginary parts of the channel coefficients. The training and validation loss of the models using these parameters are shown in figure 4.15. The figure shows that the models for both grids start to converge after 10 epochs and as such, the small grid uses the model at 11 epochs with a validation loss of 0.182 and the large grid uses the model at 14 epochs with a loss of 0.205.



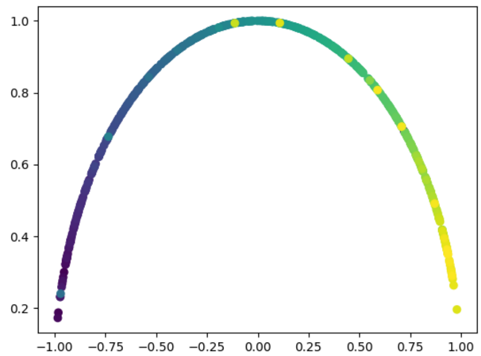
(a) Small grid *(b)* Large Grid

Figure 4.15. Training and validation loss for the CNN model trained on the UMi channel model for the small and large grid size.

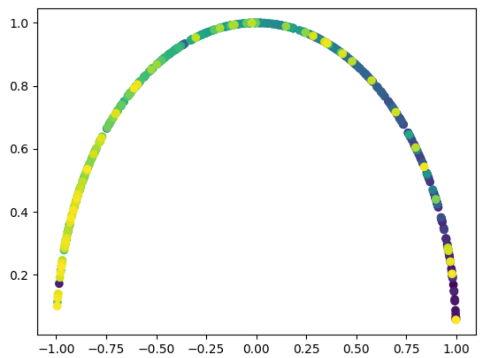
For the LoS channel model, the difference between the DNN and CNN models where negligible, both for the validation loss and the angle estimations. However, for the UMi channel model, it seems that the CNN is more robust, both in terms of the validation loss, where it shows a roughly 15% improvement, and in the angle estimates, as shown in figure 4.16. Though it is a slight difference, the angle estimates made by the CNN seem to have less errors. However, it does show the same difficulty for the signals received at the back of the antenna array, and some slight issues for angles at 0 and 180 degrees.



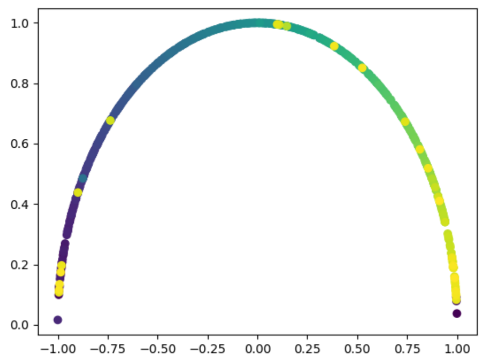
(a) First half - Small



(b) Second half - Small



(c) First half - Large



(d) Second half - Large

Figure 4.16. Estimated AoAs based on the CNN model tuned for the small and large grid size using 16 vertically and 16 horizontally polarised antennas with the UMi channel model. The estimated AoAs are projected onto the unit circle. The transmissions are made from positions following a circle around the BS placed at the center with a radius of 10 meters for the small grid size and 100 for the large grid.

4.4.2 Fingerprint Training

The fingerprint training of the proposed method follows the same overall approach as the baseline. However, since the maximum likelihood based method from equation 4.3.6 is designed to use the vector x_s consisting of index values i of $|\tilde{h}_{ar}|$, the method cannot be applied directly to the ML-based angle estimations, as no index values are produced in these estimations. Instead, the size of the range, which is determined by the difference between the maximum and minimum value of the angle estimates contained in the vector ψ_s, is used to determine how the angles for the fingerprints are calculated. It has been observed that the angle estimates using different OFDM pilot symbols can be characterised by two cases based on the size of the range of the estimated angle values in ψ_s.

For the first case, the size of the range of the estimated angle values is below 1 degree and the estimates are generally close to the true AoA. In the second case, the size of the range is high (more than 1 degree) but most of the estimates are still close to the true AoA. In this second case, there are outlier estimates which would negatively impact an average of the angles. Instead, when the size of the range is high, a histogram is taken of the estimates with a bin size of 1 degree. The bin with the majority of estimates is chosen as the angle which is saved in the fingerprint. For the estimates with a size of the range below 1 degree, the mean angle is saved in the fingerprint. Furthermore, the variance value of the estimates in ψ_s is additionally saved in the fingerprint to be used in the location estimation.

4.4.3 Location estimation

Similar to the fingerprint training described in the above, the location estimation follows the same overall process as for the baseline method. The main differences are in the two last steps: the fingerprint comparison and location calculation. As was seen in figure 4.6, which showed the positions where the estimation errors where the highest, the locations which are in line with the BSs can cause problems for the location estimations. This is caused by some of the problematic angle estimates in the fingerprints where the estimated angles are either 0, 90, or 180 degrees. However, regardless of the position of the UT, there will be several BSs where the angle is not one of those three.

For the baseline method, these problematic angles are dealt with by virtue of the confidence factor which weights the MSE calculated with the angles. However, another approach is to remove the problematic angles from the fingerprints, such that only the non problematic angles are used in the comparison. This approach of removing problematic angles from the fingerprints is regarded as a pruning step. From the angle estimates in the circle tests shown in figures 4.14 and 4.16, it was shown that the estimated angles of close to 0 and 180 degrees where randomly spread across the circle. For both the DNN and CNN models, these angles, and angles close to them, result in angle estimates which have a higher variance. Furthermore, the circle tests which are based on the UMi channel model show that estimations of signals received from the back of the antenna array likewise has a higher inaccuracy and higher variance. This means that using the variance to remove certain angles from the fingerprints will also help to alleviate the problem of these angle estimates.

By looking at the variance of each angle estimate in a UTs fingerprint and excluding those with the highest variance, the probability of using the angle estimates from the problematic areas is decreased. This should result in a higher location estimation accuracy. If certain angles are pruned from the fingerprint of a UT, the corresponding angles must be removed from the fingerprints of the ANs when comparing them. In practice, the fingerprints are pruned by sorting the fingerprints in terms of variance and removing all but the 4 angles with the lowest variance. The number of 4 angles is chosen based on the maximum number of problematic angles which can occur. The maximum number of problematic angles which can occur, is when a UT is placed in close proximity to one of the BSs. For these locations, 5 angles can be problematic leaving 4 non-problematic angles.

While it may seem that using the approach presented in the fingerprint training does not have an effect when the estimates with a high variance is removed, it will not always be the case. There is no guarantee that there will be any angle estimates in the fingerprint where the variance is low enough for the mean to be used. However, the location estimation will not work if too many of the estimates are removed from the fingerprint. As a result, it will still be beneficial to try and improve the accuracy of the estimates included in the fingerprints when the variance is high.

Instead of using a WMSE function to compare the fingerprints as done in the baseline, a normal MSE function is used with the pruned fingerprints. The MSE function is defined as:

$$Z_{ka} = \frac{1}{R'} \sum_{r=1}^{R'} (w'_{ar} - \hat{\theta}'_{kr})^2, \quad \forall a \in \mathcal{A}, \tag{4.4.1}$$

where w'_{ar} is the pruned fingerprint for AN $a \in \mathcal{A}$, $\hat{\theta}'_{kr}$ is the pruned fingerprint of the UT k, and R' is the amount of BSs in the pruned fingerprint.

Based on the resulting vector of MSE values, the ANs with the lowest MSEs are selected, denoted by set $\mathcal{A}_m$. For the location calculation itself, the coordinates of the selected ANs are weighted by the inverse of the MSE. As such, the estimated coordinates, k_x and k_y of UT k is defined as:

$$k_x = \sum_{a \in \mathcal{A}_m} Z_{ka}^{-1} \zeta_{x,a}, \tag{4.4.2}$$

where Z_{ka}^{-1} is the inverse of the MSE defined in (4.4.4), and $\zeta_{x,a}$ is the x-coordinate of AN a.

$$k_y = \sum_{a \in \mathcal{A}_m} X_a^{-1} \zeta_{y,a}, \tag{4.4.3}$$

where $\zeta_{y,a}$ is the y-coordinate of AN a.

The inverse MSE is calculated by summing the MSEs for the ANs in $\mathcal{A}_m$, subtracting each MSE from the sum, and dividing it by the sum of the subtractions:

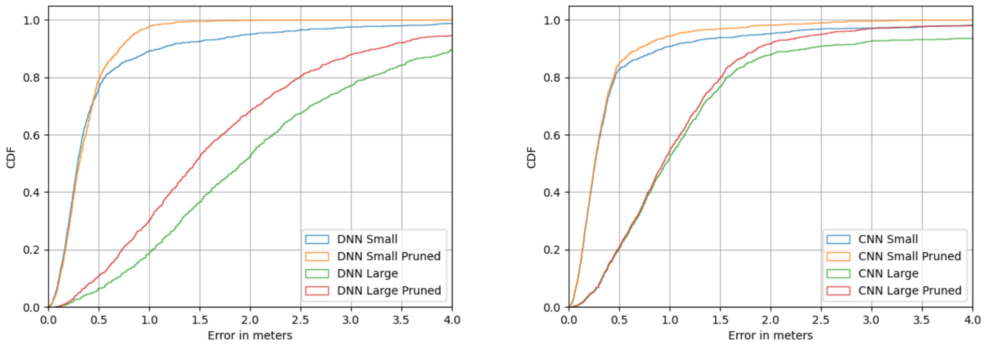
$$Z_a^{-1} = \frac{\sum_{a' \in \mathcal{A}_m} (Z_{a'}) - Z_a}{\sum_{a'' \in \mathcal{A}_m} (\sum_{a' \in \mathcal{A}_m} (Z_{a'}) - Z_{a''})} \tag{4.4.4}$$

4.4.4 Accuracy of Location Estimations

In this subsection the accuracy of the proposed localisation method is briefly investigated by introducing each improvement one at a time. For this, the ML-based AoA estimations are first used with and without pruning. For each, the inverse MSE weighted location calculation is omitted. Following this, the inverse MSE weighted location calculation is included. Lastly, the baseline method is compared against the proposed fingerprinting method. As done in the validation tests performed for the baseline, the angle estimates are based on channel coefficients from 20 OFDM pilot symbols.

LoS Channel

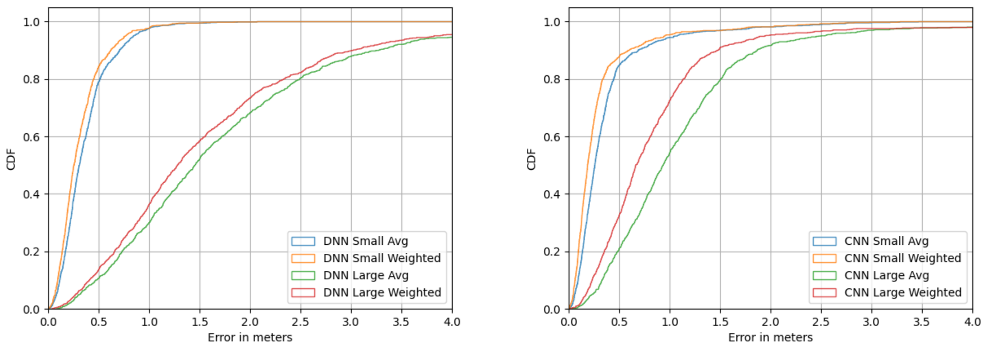
Starting with the LoS channel, figure 4.17 shows the improvement introduced by pruning the fingerprints. This shows that the localisation accuracy for both the DNN and CNN is increased for the small and large grid. Furthermore, it can be seen that for all but the DNN in the large grid, the increased accuracy is gained at the extreme outliers. For 80% of the estimates there is little to no difference, but for the remaining estimates fewer reach the extreme outlier errors.



(a) DNN *(b)* CNN

Figure 4.17. CDFs of the resulting localisation errors made with and without the pruning of the fingerprints for the LoS channel model for both the small and large grids. Results in the small grid have ANs spaced 1 meter apart, whereas for the large grid the ANs are spaced 4 meters apart.

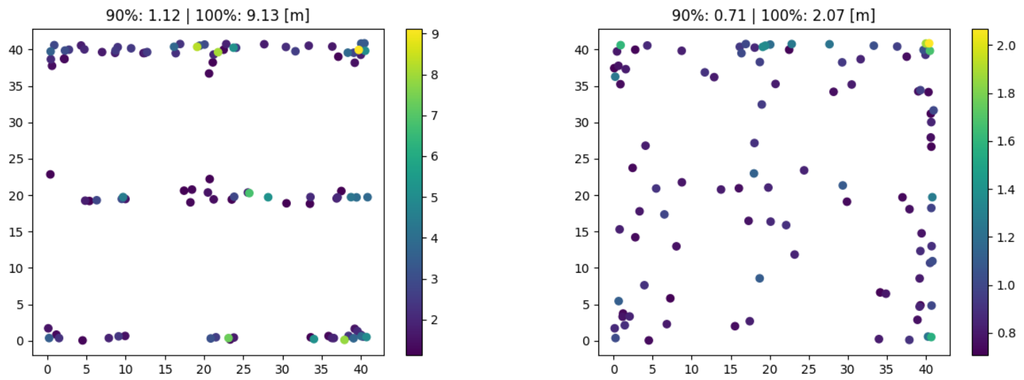
The results of using the inverse MSE to weight the AN positions in the final location calculation are shown in figure 4.18. The figure shows that the use of the inverse MSE does increase the accuracy of the location estimations. However, the accuracy is increased in a different way compared to the inclusion of pruning. The inverse MSE does not improve the accuracy for the already problematic positions. Instead, it increases the accuracy of the positions which already had relatively decent location estimations.



(a) DNN *(b)* CNN

Figure 4.18. CDFs of the resulting localisation errors made using the DNN (a) and CNN (b) for the angle estimation step for the LoS channel model with and without pruning. Results in the small grid have ANs spaced 1 meter apart, whereas for the large grid the ANs are spaced 4 meters apart.

Both the inclusion of pruning and weighted location calculation improve the localisation accuracy. However, it is not yet clear if the improvements address the areas which where specified during the design. Looking at figure 4.19, it can be seen that using omitting pruning results in a clear characteristic for the locations of the worst errors. All 10% of the largest estimation errors, which are between 1.12 and 9.13 meters, are where the signals arrive at 0 or 180 degrees. When the pruning of fingerprints is added, the maximum error is significantly decreased and the locations of the errors change. While it does not seem to be perfect – the errors are still aligned with the BSs to some degree – it is not as severe as without pruning. Furthermore, when the weighted location calculation is included, the 90% error decreases even further to 0.62 meters. It does, however, not change what locations the worst errors occur at.



(a) DNN No Pruning *(b)* DNN With Pruning

Figure 4.19. Comparison of the true locations of the largest localisation errors using the DNN angle estimation method with and without pruning on the LoS channel model. The distance between the ANs is 1 meter with both subplots using the small grid size. The color of the points is related to the error of the estimation following the gradient bar to the right of each plot.

Lastly, comparing the proposed fingerprinting method with the baseline, figure 4.20 shows that the proposed method does in fact perform better than the baseline for the validation case in the LoS model. However, it must be noted that the baseline does have a disadvantage in the LoS channel by the reduced resolution of the possible angle estimates it can make, as shown in figure 4.3. Furthermore, the lacking variance in the angle estimations in the LoS channel means that the maximum likelihood based angle calculation used by the baseline does not offer much value. However, as shown in figure 4.4, the baseline angle estimator did not have the same problem with the reduced resolution in the UMi channel model. It is therefore possible that the baseline will see an improved relative performance in the UMi model.

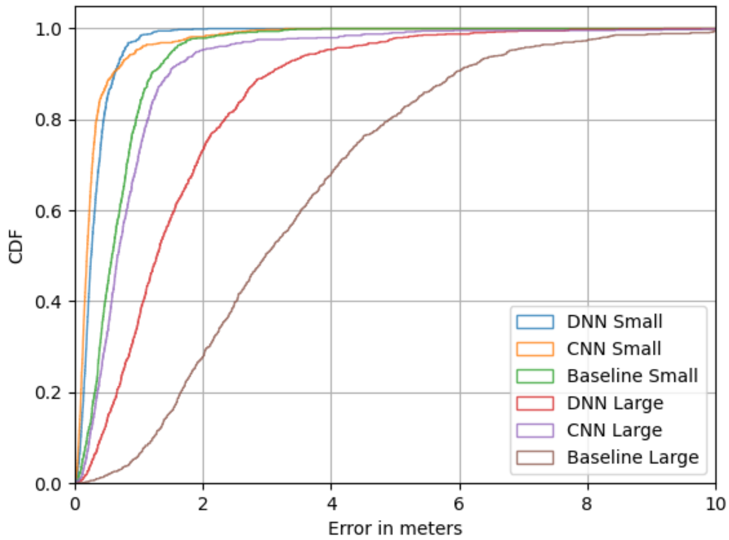
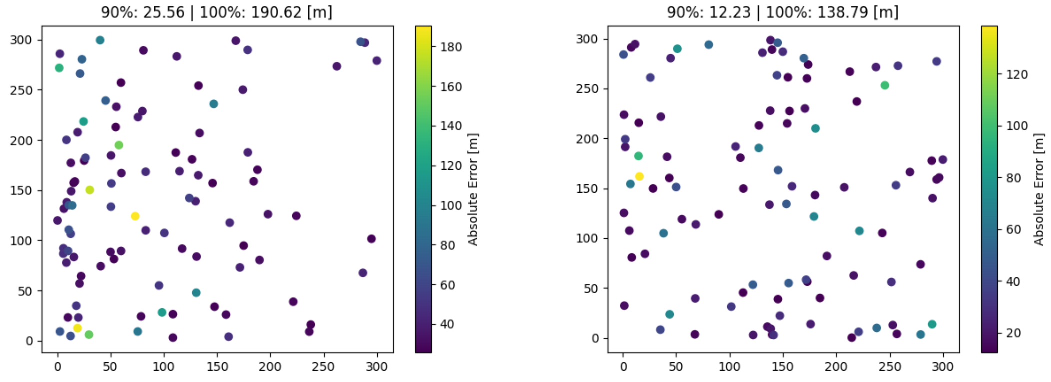


Figure 4.20. CDFs of the resulting localisation errors made using the implemented angle estimation methods for the LoS channel model for both the small and large grid. Results in the small grid have ANs spaced 1 meter apart, whereas for the large grid the ANs are spaced 4 meters apart.

UMi Channel

For the UMi channel, the accuracy gain of the individual improvements are not re-investigated. Instead, an oversight of the BS locations is presented and investigated. Following this, a final overview of the relative performance of the baseline and the proposed fingerprinting method is presented.

As shown in the results of the circle tests using the UMi channel, there is a significant difference in receiving the signals from the front of the antenna array and the back. It was further alluded to in the discussion of figure 4.8 that there is bias of the errors to tend towards the left side of the grid. As seen in figure 4.21a, the estimations based on the DNN show that the localisation method does struggle significantly more on the left side. The reason for this is that there are no BSs placed outside of the 300×300 grid. This means that the UTs on left side do not have enough BSs which can receive their signal from the front and not an angle close to 0 or 180 degrees. However, by extending the number of BSs by 7 – one more row and column at $x = -150$ and $y = -150$ – the accuracy increases significantly with no more tendency towards the left side, as shown in figure 4.21b.



(a) DNN 9 BSs *(b)* DNN 16 BSs

Figure 4.21. Comparison of the true locations of the largest localisation errors using the DNN angle estimation method in the LoS channel model. The estimations are based on 9 BSs in a) and 16 BS in b). The distance between the ANs is 4 meters with both subplots using the large grid size. The color of the points is related to the error of the estimation following the gradient bar to the right of the plot.

The results of the comparison between the proposed method and the baseline using the extended grid is shown in figure 4.22. The figure shows that both versions of the proposed method still outperforms the baseline, both in the small grid and the large grid. In fact, the proposed method using the CNN model in the large grid even outperforms the baseline in the small grid. The magnitude of the difference in performance between the proposed method and the baseline, suggests that the main benefit of the proposed method comes from the ML-based angle estimates. The improvements shown from using pruning and weighted location calculation did not seem significant enough to warrant such a disparity between the methods.

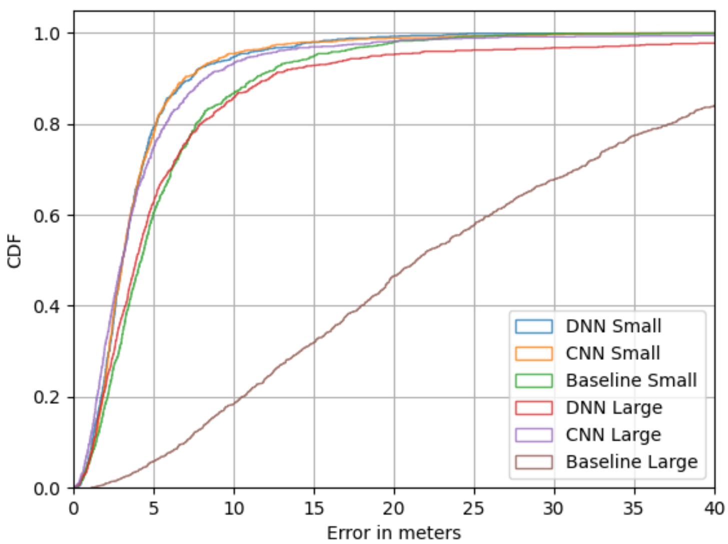


Figure 4.22. CDFs of the resulting location estimation errors made using the implemented angle estimation methods for the UMi channel model. Results in the small grid have ANs spaced 1 meter apart, whereas for the large grid the ANs are spaced 4 meters apart.

4.5 Direct Localisation

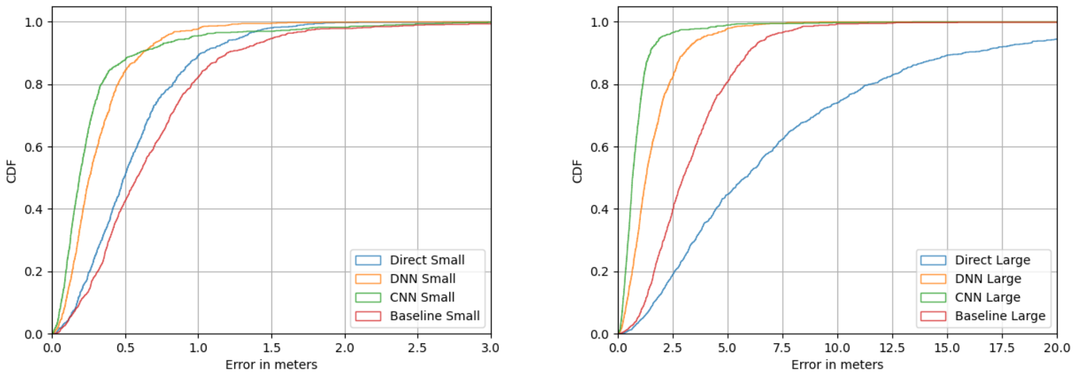
The fingerprinting-based methods described in the sections above all require a considerable amount of work in collecting the fingerprints of the ANs. As an alternative to this, an ML based approach is presented which skips this step entirely. Instead of using the ML model as an intermediary step in estimating the positions of UTs, this proposed method predicts the locations directly from the inputs – hence the name of direct localisation.

The proposed direct localisation method uses a DNN structure, which takes the channel coefficients from a single pilot to all BSs as input. Similar to the proposed DNN for the AoA estimations in the LoS channel, the channel coefficients are further split into their real and imaginary parts. As the output, the proposed model uses a two point regression output layer – one for the x- and y-coordinate respectively. The exact structure of the direct localisation DNN is shown in table 4.4.

Layer Index	**Layer Type**	**Size**	**Activation**
1	Input Layer	Small grid: 576 ($9 \times 32 \times 2$) Large grid: 1024 ($16 \times 32 \times 2$)	N/A
2	Normalisation Layer	N/A	N/A
3	Dense Layer	256×6	Relu
4	Dense Layer	256×5	Relu
5	Dense Layer	256×4	Relu
6	Dense Layer	256×3	Relu
7	Dense Layer	256×2	Relu
8	Dense Layer	256	Relu
9	Output Layer	2	Linear

Table 4.4. The structure of the direct localisation DNN for the UMi channel mode for both the small and large grid.

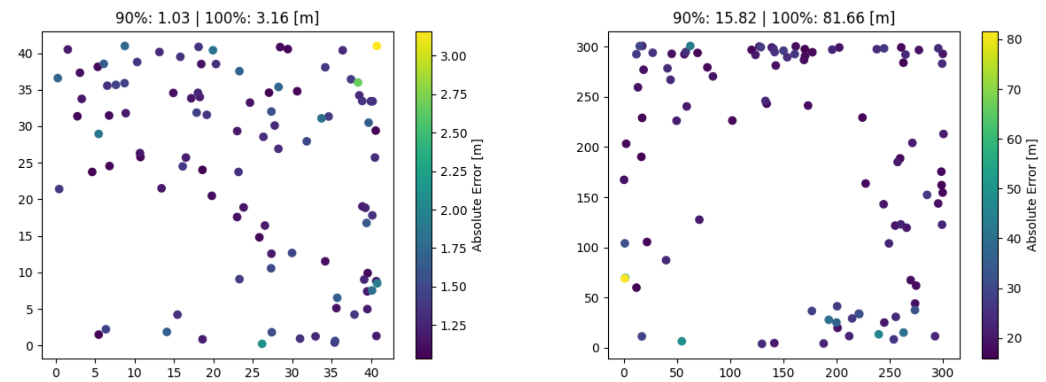
Training the direct localisation DNN using the LoS channel model with the small and large grid sizes required two different input sizes. For the small grid, there are 9 BSs while there are 16 for the expanded large grid. Using the hyperparameters from the AoA DNN as a starting point for the hyperparameter tuning, it was found that the validation loss was relatively invariant to changes in the DNN structure. This was the case for the small grid as well as the large grid and as such, the hyperparameters are kept the same.



(a) Small grid *(b)* Large Grid

Figure 4.23. CDFs of the resulting localisation errors made using direct localisation DNN compared against the baseline and proposed fingerprinting methods using the LoS channel model. Results for the non-direct localisation methods in the small grid have ANs spaced 1 meter apart, whereas for the large grid the ANs are spaced 4 meters apart.

Looking at the CDFs in figure 4.23 it is clear that the proposed fingerprinting-based methods outperform the direct localisation DNN. While the direct localisation method does outperform the baseline in the small grid, the direct localisation does not scale as well to the larger grid. Furthermore, looking at figure 4.24 the distribution of the least accurate estimation targets can be seen to show different results. In the small grid the most points are somewhat equally distributed with more of them lying in the top half of the grid. However, for the large grid the points seem to be located near the edges of the grid. This is likely due to the larger distances present in the larger grid increasing the effect of the pathloss such that the relative power of the signal to the noise is much lower.



(a) Small grid *(b)* Large grid

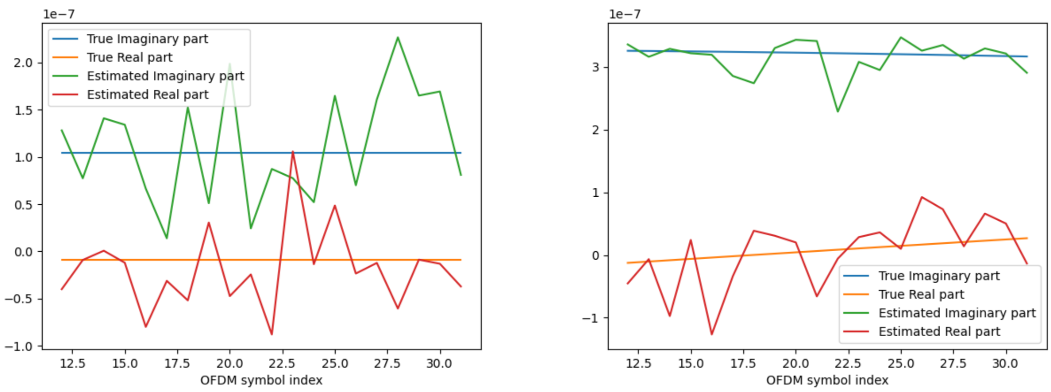
Figure 4.24. Visualisation of the true locations of the largest localisation errors using the direct location estimation method on the LoS channel model. The estimations are based on 9 BSs in the small grid and 16 BSs in the large grid. The distance between the ANs is 1 meter for the small grid and 4 meters for the large grid. The color of the points is related to the error of the estimation following the gradient bar to the right of the plot.

Attempting to train a direct localisation model in the UMi channel proved to be more troublesome. While a model could be trained in the small grid, the resulting accuracy was significantly worse than the proposed fingerprinting methods as well as the baseline method. Furthermore, the model was completely incapable of learning in the large grid. Despite the promising cost reduction of using a direct localisation method, since it cannot perform for the large grid in the UMi channel, the direct localisation method is excluded from the further testing carried out in the following chapter.

5 Results

This chapter investigates the performance of the proposed AoA-based fingerprinting localisation method compared to the baseline both of which are presented in chapter 4. The methods are compared in three tests. The first test examines the effect of using increasing amounts of pilots for the location estimations. The second test investigates how often localisation needs to done in order to maintain a certain accuracy. The third test combines the results of the second test with an analysis of the impact of reducing the size of the data types in the location estimation, to calculate the load on the backhaul of the system. Lastly a discussion reflecting on the results in a broader context is carried out.

As mentioned in the problem scope, in section 2.4, the focus of the project is to examine the cost of localisation methods, rather than focus purely on their localisation accuracy. However, cost can be defined in many ways. In the context of ISAC, the first aspect of cost which is examined, is how much it impairs the existing communication. All of the localisation methods considered in this project, base their localisation on estimated channel coefficients from pilot symbols.



(a) Static UTs ***(b)*** Moving UTs

Figure 5.1. Figure showing the fading of the true channel and channel estimates for static and moving UTs. The velocity of the moving UTs is 1.414 ms/s.

As can be seen in figure 5.1, the true channel coefficients of the UMi channel model are flat in time as long as the UTs are not moving. With the noise, the channel coefficients do change over time, however, only based on the additive white Gaussian noise. This means that if a UT is stationary, there is no impact on the channel coefficients in waiting a long time and slowly gathering enough estimated channel coefficients to make more accurate location estimates. However, when the UTs are moving, there is a risk that gathering coefficients over too much time will introduce enough noise to negatively impact the location accuracy. In order to test the effect which movement has on the accuracy of the methods, data sets of moving UTs and static UTs are generated and used separately in testing. The movement of the UTs are set to uniformly random generated movement speeds between 0-1.667 m/s to emulate human walking speeds.

The first test conducted in this chapter will examine the impact of allocating increasing amounts of OFDM symbols as pilots over a set amount of time. In 5G OFDM communication, the communication is divided into subframes which are 1 ms in length. Within a subframe a number of slots are placed. A slot is defined to consist of 14 OFDM symbols, 2 of which are pilots, which means that the length of a slot depends on the bandwidth of the subcarriers, which can be 15, 30, 60, 120, and 240 kHz. At 15 kHz a slot takes 1 ms which leaves room for one slot per subframe. As the bandwidth increases, the length of a slot decreases leaving room for more slots in each subframe – at 30 kHz, a slot is 0.5 ms, at 60 kHz it is 0.25 ms, etc. As an example, considering the chosen bandwidth of 120 kHz and a time limit of 100 ms this allows for a total of 11200 pilots ($\frac{100}{\frac{0.125}{14}} = 11200$) if all of the communication is allocated as pilots. However, due to hardware limitations and the RAM overhead in Sionnas generation and estimation of channel coefficients, a limit of roughly 2000 consecutive symbols can be created – an amount of symbols which covers roughly 18 ms. This is considering the use of only a single subcarrier which, when considering the noise power, means that the noise will have to be generated based on an extended amount of subcarriers. In particular, the noise is based on the use of 10 subcarriers. Furthermore, due to additional hardware limitations, the baseline considers a maximum of 500 consecutive symbols. This limitation is due to high the computational complexity of the baseline method.

Besides the cost of doing single location estimations, the movement of the UTs will affect the cost of the methods in regards to the frequency of the location estimations. When the UT is not stationary, the accuracy of a single location estimation will deteriorate over time as the UT moves away from the area of the estimate. It is therefore necessary to make periodic estimates to track the UT. The frequency with which the estimates are made, will determine how far the UT can move between each estimate and as a result will impact the accuracy over time. The second test will investigate this by evaluating the relation between the cost (frequency of estimations) and continued accuracy of the location estimates.

Lastly, the cost of the estimates are evaluated in regards to the backhaul. For both the baseline and the proposed AoA-based fingerprinting methods, the angles are calculated at the BSs. This means that it is not necessary to send the channel coefficients over the backhaul. Instead, the methods only have to send the angles and an additional metric for the variance of the angle. For the baseline, the metric of the variance is the confidence factor and for the proposed method it is the actual variance value. While python uses 32 byte floats as the standard type for these values, lowering the precision of the data will reduce the load on the backhaul. This might, however, be at the cost of localisation accuracy. As such, the last test investigates the cost of the localisation methods on the backhaul by looking at the load as a function of localisation frequency, data type, and number of BSs.

The parameters used for the tests carried out in this chapter are summarised in table 5.1.

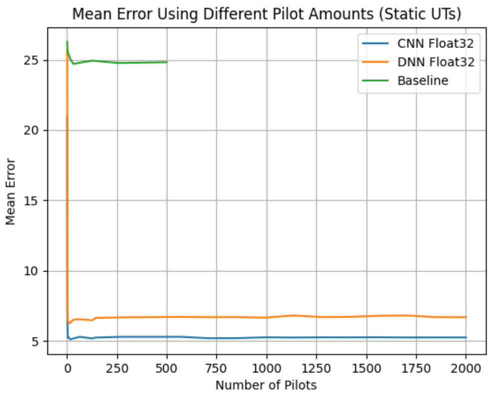
System Parameters	
Grid size: ANs & UTs	300×300 m
Anchor node spacing	4 m
Grid size: BSs	450×450 m
Base station spacing	150 m
Center frequency	28e9 Hz
Subcarrier spacing	120e3 Hz
Number of subcarriers	1
Noise power	k_B*290*120e3*10
Pilot Test Parameters	
Pilot count	1-2000 $2^p, \quad p \in [0, 7]$ and $142 * p, \quad p \in [1, 14]$
Pilot count (Baseline)	1-500 $\min(500, 2^p), \quad p \in [0, 9]$
Number of UTs	1000 moving & 1000 static
Movement speed of UTs	0-1.667 m/s
Backhaul Cost Test Parameters	
Data types	[Float32, Float16, Uint8]

Table 5.1. Table containing the system- and test parameters for the tests

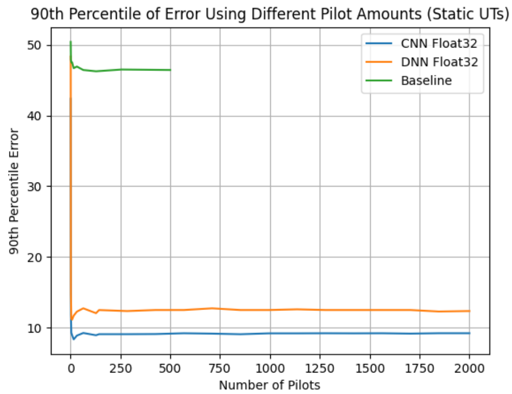
5.1 Pilot Test

The first test investigates the accuracy of the considered localisation methods as a function of the number of OFDM pilots used to estimate angles. The two proposed AoA-based fingerprinting methods – one which uses a CNN model for angle estimations and one which uses a DNN model for angle estimations – are compared against the baseline. Figure 5.2 shows the mean and 90th percentile error of the three methods when considering static UTs. Immediately apparent is the gap in localisation accuracy between the baseline and the proposed methods for all amounts of pilots. Besides this, all methods converge at a relatively stable localisation accuracy well before the maximum number of pilots is used. Looking closer at the results for the first 145 symbols, shown in figure 5.3, it can be seen that all methods reach this stable accuracy by around 32 pilots. Even considering the reduced number of 500 consecutive symbols, a total of 71 symbols are already designated as pilots in normal OFDM communication. This means that neither of the methods require more pilots than what is already being generated through normal OFDM communication in order to reach their maximal localisation accuracy. As such, it can be concluded that for communication networks with similar parameters as those used for these tests, neither method impose additional strain on the communication.

While there is a relatively slight difference between the DNN and CNN, the CNN does reach a better localisation accuracy than the DNN, both in terms of the mean error and the 90th percentile error. The CNN reaches a mean localisation error of roughly 5 meters while the DNN only reaches 6.5 meters. For the 90th percentile error, the CNN manages to get below 10 meters, while the DNN tapers off at 11 meters. The further implication of the exact localisation accuracies, will be discussed in the test regarding the cost on the backhaul in section 5.3.

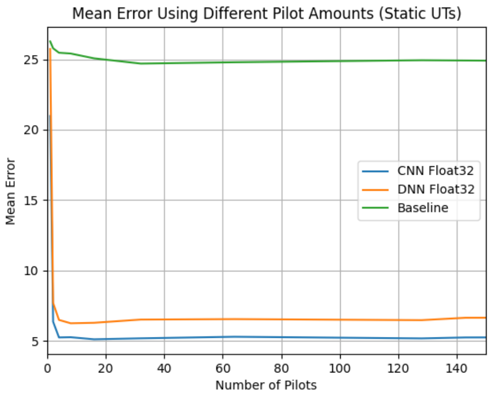


(a) Mean

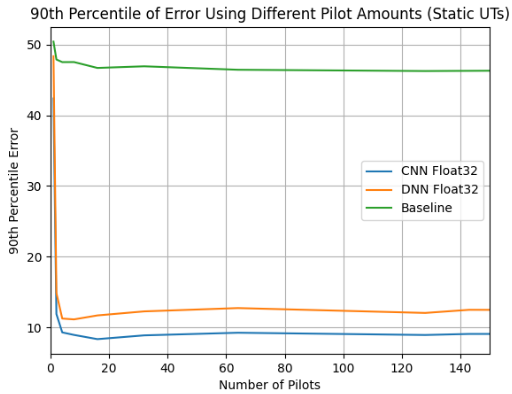


(b) 90th Percentile

Figure 5.2. Plot showing the accuracy of the localisation methods using differing amounts of pilots for static UTs.



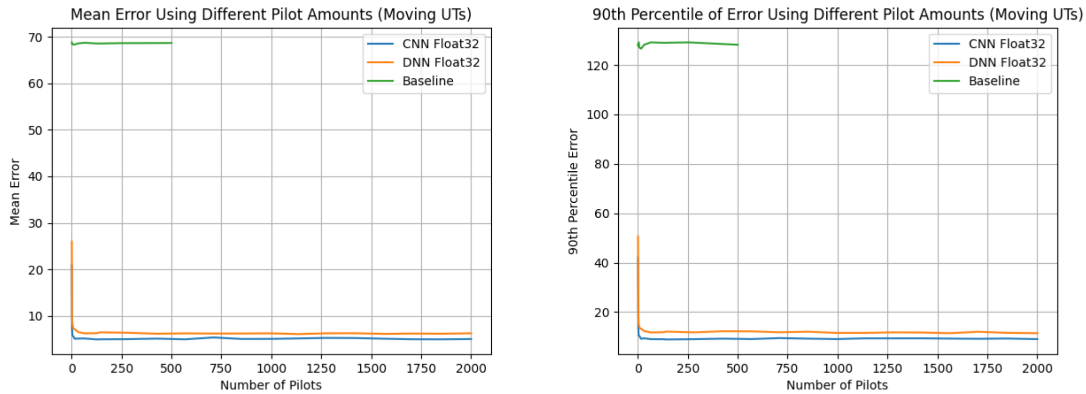
(a) Mean



(b) 90th Percentile

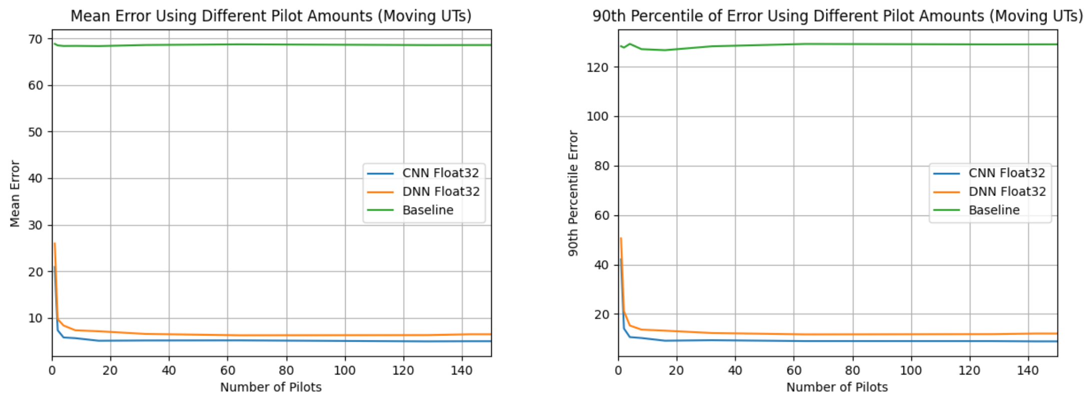
Figure 5.3. Plot showing the accuracy of the localisation methods using differing amounts of pilots for static UTs.

Considering next the case of moving UTs, as shown in figure 5.4, it is mainly the baseline method which is affected in terms of localisation accuracy. As seen in figure 5.5, all methods still converge after roughly the same amount of pilots, however, the accuracy which the baseline reaches as it converges is significantly lower than in the static case. The two proposed methods on the other hand, do not seem to lose much if any accuracy when the UTs are moving.



(a) Mean ***(b)*** 90th Percentile

Figure 5.4. Plot showing the accuracy of the localisation methods using differing amounts of pilots for moving UTs.



(a) Mean ***(b)*** 90th Percentile

Figure 5.5. Plot showing the accuracy of the localisation methods using differing amounts of pilots for moving UTs.

5.2 Localisation Update Frequency Test

For the localisation update frequency test, the mean error and 90th percentile error of the two proposed methods are used to determine a localisation error as a function of the time between each localisation estimation. The baseline is not included in this test as the accuracy for moving UTs is not considered comparable to the two proposed methods. For the test, it is assumed that the UT, which the localisation error is regarded, is moving at a constant speed of 1.667 m/s. Furthermore, it is assumed that the UT is always moving away from the latest location estimation, and that the estimation error is always equal to the mean or 90th percentile error. The mean and 90th percentile errors are based on the estimations done using 32 pilots. These errors are:

- CNN mean: 5.14 m
- DNN mean: 6.54 m
- CNN 90th percentile: 9.37 m
- DNN 90th percentile: 12.28 m

The results using these errors are shown in figure 5.6a. The figure does not give specific values for the required localisation frequency, without specifying a maximum allowed mean error or 90th percentile error. As an example of this, a limit of 15 meters is used. With this limit, figure 5.6b shows how long the UTs can go without doing another localisation estimation and still stay within the limit considering the mean localisation error and 90th percentile error. With this maximum mean and 90th percentile error limit, the combined cost on the backhaul is investigated.

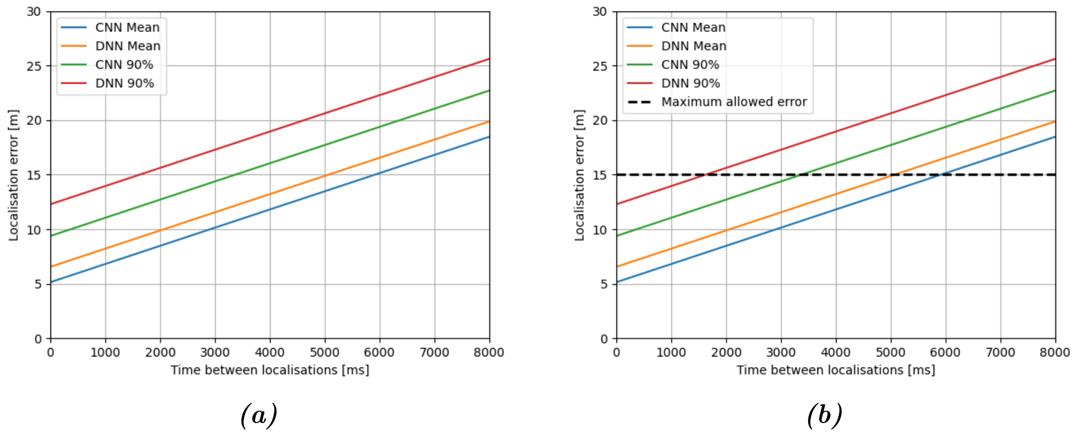


Figure 5.6. Plot depicting the development of the localisation error as more time passes without re-estimating the location. The localisation error is initialised at the mean error and 90th percentile error for the two proposed localisation methods. The error increases based on a UT moving at a constant speed of 1.667 m/s away from the last location estimate.

5.3 Cost on Backhaul

The cost on the backhaul, C_b, is based on the number of BSs which contribute angles to fingerprints, the data type of the fingerprints including supplementary values such as variance and confidence factors, and the frequency of the localisation estimations. The cost is defined as:

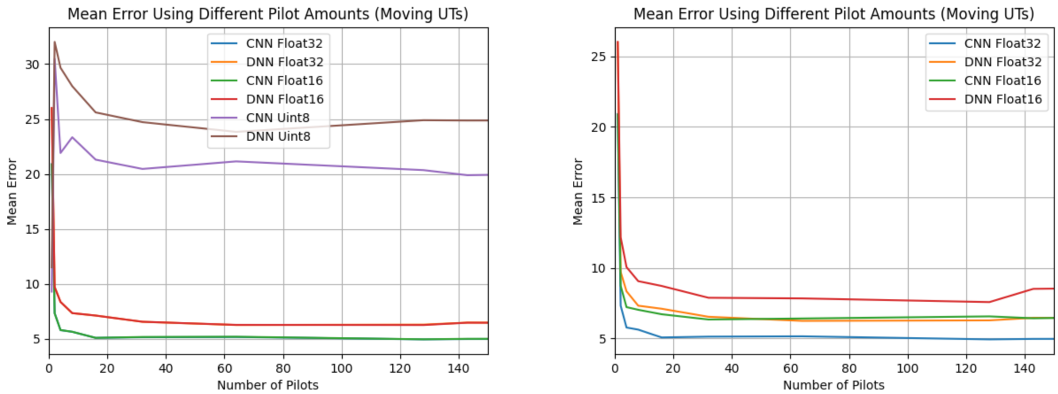
$$C_b = \frac{R * (B_f + B_\sigma)}{\Delta_t}, \tag{5.3.1}$$

where R is the number of BSs, B_f is the size of the fingerprint angle data type, B_σ is the size of the variance data type, and Δ_t is the time between each localisation estimation.

From the previous test an example of 15 meters as the upper bound of the mean or 90th percentile error was used. This value is further used in this test as the basis for the location estimation frequency.

Before the cost can be calculated, the effect of using different data types for the fingerprints is investigated. Similar to the update frequency test, the baseline is omitted from this test as neither the mean nor 90th percentile error is below 15 meters. Figure 5.7a shows the mean error of the localisation estimations for moving UTs where the data type of the angle values in the fingerprints is set to Float32, Float16, and Uint8. The figure shows that there is no difference in the accuracy for either of the float types. This can either mean that the reduction in data size does not impact the accuracy or that somewhere in the location estimation, the data is already being reduced to a Float16 type. Either way, this means that reducing the data size pre-emptively does not limit the performance of the methods. On the other hand, going to Uint8 has a detrimental effect on the accuracy. As such, the fingerprints are reduced to Float16 for the backhaul cost calculation.

Figure 5.7b shows that the accuracy does drop when the variance data type is reduced to a float16. Furthermore, the Uint8 data type is omitted from the graph, as the use of this data type did not allow the localisation methods to produce location estimates. As such, the data type for the variance is kept at Float32.



(a) Fingerprint angles ***(b)*** Variance

Figure 5.7. Plots showing the accuracy of the proposed localisation methods using differing amounts of pilots for moving UTs. The plots compare the accuracy of different versions of the proposed methods using decreasing data type precisions. Figure (a) shows localisation error when changing the AoA data type and (b) shows the localisation error when changing the variance data type.

The last step before the cost on the backhaul can be calculated is to get the exact time intervals between each localisation estimation for the two methods. With the the upper bound of the mean or 90th percentile error at 15 meters, and the individual error values for mean and 90th percentile errors, presented in section 5.2, the location update time intervals are calculated to be:

- CNN mean: 5916.91 ms
- DNN mean: 5073.07 ms
- CNN 90th percentile: 3375.12 ms
- DNN 90th percentile: 1631.08 ms

Based on 16 BSs, 16 Bytes for the fingerprint values, 32 Bytes for the variance values, and the update times presented above, the cost on the backhaul is calculated using equation (5.3.1) to be:

- CNN mean: 0.123 B/s
- DNN mean: 0.151 B/s
- CNN 90th percentile: 0.228 B/s
- DNN 90th percentile: 0.471 B/s

These results show that the proposed methods do not impose a significant strain on the backhaul communication given the 15 meter upper bound. While no specific parameters for the backhaul, such as data rate, has been presented, it is highly unlikely that the load generated by the proposed localisation methods will be significant.

5.4 Further Considerations

From the results it was shown that the two proposed fingerprinting-based methods outperformed the baseline in all cases. While it was not included in the investigation on backhaul cost due to the disparity in accuracy, the amount of data which the baseline has to transmit per location estimation is on par with the proposed fingerprinting methods. However, one area in which the baseline has a distinct advantage over the proposed fingerprinting methods is that it does not require the training of an ML model. In order to train a model for predicting AoA values, a considerably large data set, containing training data with known target AoA values, is required. Acquiring this data set is not guaranteed to be a trivial task. While this issue has not been considered for this project, there are a few ways to do it in practise. The first method is based on a large data set where the AoA targets are based on random locations in the considered localisation area. For this, data can be gathered passively over time by having UTs transmit their location which has been estimated using an external robust localisation method. Alternatively, the AoA targets can be based on the locations of the ANs used in the fingerprint training. As gathering data from these ANs is required for the fingerprinting based methods, the processes can be combined to reduce the total amount of required work.

The fingerprint training itself is a significant cost which is required for any fingerprinting based method. As a result, any method which does not require this will have an inherent advantage when considering the feasibility of real use. One such example is the direct localisation method which was introduced in section 4.5. While no promising results where obtained in the design of the localisation method, it was found that other authors have investigated the use of ML-based direct localisation specifically for indoor localisation [Jing et al., 2019]. From the results gathered in the design of the method using the LoS channel, it was shown that the method struggled with scaling to the larger grid. This supports the use of the method in indoor localisation where transmission distances are generally shorter, and access points are more densely placed than BSs would in outdoor environments. Regardless, further investigating the use of direct localisation for outdoor use, is considered worthwhile due to its significantly reduced cost in setup.

Another way of removing the cost of the fingerprint training is to use the geometry-based localisation methods by combining the ML-based angle estimators with multiangulation. However, despite the fact that the ML models trained for the fingerprinting are trained to estimated the AoA, they cannot estimate the sine of the angle. This means that a multiangulation-based localisation method is not trivial to implement. The method would need some way of estimating the sine in order to accurately determine the position of UTs. There is, however, a possibility that the disparity in variance of the AoA estimates from signals received at the front and back of the UMi antenna arrays can be used for this. If this is the case, then it is possible that the increased accuracy of angle estimates obtained with the ML estimators will allow for localisation methods which reduce the cost of manual work without lowering the accuracy of the localisations.

A point which was not investigated in the project but which can have a significant impact on the use of a localisation method is the computational complexity of the methods. For the fingerprinting based method considered in this project, the baseline had a considerably higher complexity than the ML-based methods. When estimating angles for the testing of this method, it was measured that it took 160 ms to estimate 32 AoA values. Comparing this to the ML-based methods, the CNN took 8 ms and the DNN took 4 ms for the same amount of AoA estimates. However, it must be noted that these values are based on the use of a dedicated server with hardware that is not realistically used in BSs. It is therefore likely that these values will increase when the estimations are performed on more appropriate hardware. Especially the ML estimations are likely to take longer to compute if the BSs are not equipped with a GPU or AI accelerator.

To conclude this discussion, the overall performance of the proposed methods is considered in terms of some of the requirements for localisation accuracy in 5G and beyond. In chapter 1 it was introduced that certain application of localisation require localisation errors of less than 10 meters. While the mean error of the CNN based method reached well within this point, the 90th percentile error only just managed to get below 10 meters. However, one aspect which has not been covered in this project is that of using successive localisation estimations to improve the overall accuracy. A simple approach for this is to remove the extreme errors by evaluating the probability that the UT moved far enough in the time between localisation estimations to reach the point of the new estimate. Furthermore, it has been shown in other works that when the devices which are being tracked are moving, techniques such as Kalman filters can be used to improve the overall accuracy of the location estimations [Abu Ali and Abu-Elkheir, 2015]. While the addition of these methods are likely to improve the overall reliability of the localisation methods, it is difficult to make definitive statements on the viability of the proposed methods for use in 5G and beyond applications. In the end, it comes down to the specific requirements of each application.

Conclusion 6

This project examined the accuracy of ISAC-based fingerprinting localisation methods in regards to the cost on the underlying communications network as detailed by the problem statement:

How does the accuracy of ISAC-based localisation methods relate to their cost on the underlying communications network?

As part of the process of answering this, particular focus was put on the development two localisation methods: 1) A fingerprinting method which improved the localisation accuracy of the considered AoA-based baseline method. 2) An ML based localisation method which removed the required manual work in collecting fingerprints for the necessary ANs.

For the proposed fingerprinting localisation method, the design of improvements focused on the three main aspects of AoA-based fingerprinting – the estimation of angles, the comparison of fingerprints, and the calculation of the UT locations. The improvements where based on observations made for the baseline when validated on a simple LoS channel model. For the angle estimations, ML models using a CNN and DNN structure were trained on feature extracted data. The data consisted of the calculated phase values from the channel coefficients estimated at a receiving BS – one phase value per antenna at the BS. Using these phase values, the models could predict the AoA of the received signals within a range of 0 to π – without specifying the sine of the angle. For the fingerprint comparison, a pruning step was designed to remove problematic angles from the fingerprints. Using the pruning of fingerprints aided in removing a significant amount of high error estimations. Lastly, the location calculation was changed from being the average position of the ANs with the lowest MSE calculated from the fingerprint comparison. Instead, the inverse of the MSE values are used to determine the fraction of the chosen ANs positions which are used when calculating the UT position. Using this, the accuracy of the localisation was improved by reducing the error of nearly all estimated locations.

In order to remove the need of collecting fingerprints, an ML model was trained to predict the location of a UT directly from the channel coefficients estimated at each BS. While this method did show decent results in the LoS channel for a small area with densely placed BSs, it was not able to scale to increasing distances between BSs. This was further exaggerated in the UMi model, where the ML model was incapable of learning for the BS density used in the final testing in the report. As such, this localisation method was not investigated further.

The proposed fingerprinting methods were compared against the baseline based on the relation between the cost of the methods and their accuracy. This was done using three metrics for the cost: the amount of pilots used for each location estimate, the frequency of location estimates, and the load on the backhaul. The results of the tests considering these metrics show that the proposed methods significantly outperform the baseline in terms of accuracy, regardless of the number of pilots used. Furthermore, they show that the proposed methods reaches their maximum accuracy without imposing on the existing communication between UTs and BSs. Lastly, they show that the load generated on the backhaul is relatively insignificant.

Bibliography

3GPP, September 2018. 3GPP. *Study on positioning use cases.* TR 22.872 V16.1.0, 2018.

3GPP, April 2022. 3GPP. *Study on channel model for frequencies from 0.5 to 100 GHz.* TR 38.901 V17.0.0, 2022.

Abu Ali and Abu-Elkheir, 2015. Najah Abu Ali and Mervat Abu-Elkheir. *Improving localization accuracy: Successive measurements error modeling.* Sensors, 15(7), 15540–15561, 2015.

Alamu et al., 2021. Olumide Alamu, Babatunde Iyaomolere and Abdulfatai Abdulrahman. *An overview of massive MIMO localization techniques in wireless cellular networks: Recent advances and outlook.* Ad Hoc Networks, 111, 102353, 2021.

Bishop, 2006. Christopher M Bishop. *Information science and statistics.* Pattern Recognition and Machine Learning. Springer, 2006.

Flex, 2020. Flex. *Massive MIMO antenna array*, 2020. URL `https://flex.com/downloads/massive-mimo-antenna-array-reference-design`. Date: 30/05/2023.

Ganeriwal et al., 2003. Saurabh Ganeriwal, Ram Kumar and Mani B Srivastava. *Timing-sync protocol for sensor networks.* pages 138–149, 2003.

Garcia et al., 2017. Nil Garcia, Henk Wymeersch, Erik G Larsson, Alexander M Haimovich and Martial Coulon. *Direct localization for massive MIMO.* IEEE Transactions on Signal Processing, 65(10), 2475–2487, 2017.

Goodfellow et al., 2016. Ian Goodfellow, Yoshua Bengio and Aaron Courville. *Deep Learning.* MIT Press, 2016. `http://www.deeplearningbook.org`.

Hoydis et al., 2022. Jakob Hoydis, Sebastian Cammerer, Fayçal Ait Aoudia, Avinash Vem, Nikolaus Binder, Guillermo Marcus and Alexander Keller. *Sionna: An open-source library for next-generation physical layer research.* arXiv preprint arXiv:2203.11854, 2022.

Jing et al., 2019. Yuan Jing, Jinshan Hao and Peng Li. *Learning spatiotemporal features of CSI for indoor localization with dual-stream 3D convolutional neural networks.* IEEE Access, 7, 147571–147585, 2019.

Kabiri et al., 2022. Meisam Kabiri, Claudio Cimarelli, Hriday Bavle, Jose Luis Sanchez-Lopez and Holger Voos. *A Review of Radio Frequency Based Localisation for Aerial and Ground Robots with 5G Future Perspectives.* Sensors, 23(1), 188, 2022.

Lin et al., 2018. Zhipeng Lin, Tiejun Lv and P Takis Mathiopoulos. *3-D indoor positioning for millimeter-wave massive MIMO systems*. IEEE Transactions on Communications, 66(6), 2472–2486, 2018.

Liu et al., 2022. Fan Liu, Yuanhao Cui, Christos Masouros, Jie Xu, Tony Xiao Han, Yonina C Eldar and Stefano Buzzi. *Integrated sensing and communications: Towards dual-functional wireless networks for 6G and beyond*. IEEE journal on selected areas in communications, 2022.

MacCartney et al., 2013. George R MacCartney, Junhong Zhang, Shuai Nie and Theodore S Rappaport. *Path loss models for 5G millimeter wave propagation channels in urban microcells*. pages 3948–3953, 2013.

Naseri et al., 2022. Mostafa Naseri, Adnan Shahid, Gert-Jan Gordebeke, Sam Lemey, Michiel Boes, Samuel Van de Velde and Eli De Poorter. *Machine Learning-Based Angle of Arrival Estimation for Ultra-Wide Band Radios*. IEEE Communications Letters, 26(6), 1273–1277, 2022.

Nokia, 2023. Nokia. *AirScale Massive MIMO radios*, 2023. URL `https://www.nokia.com/networks/mobile-networks/airscale-radio-access/massive-mimo/`. Date: 30/05/2023.

One6G, June 2022. One6G. *6G Technology Overview - One6G White Paper*, 2022. URL `https://doi.org/10.5281/zenodo.6630706`.

PNT, 2022. PNT. *GPS Accuracy*, 2022. URL `https://www.gps.gov/systems/gps/performance/accuracy/`. Date: 30/05/2023.

Popovski, 2020. Petar Popovski. *Wireless Connectivity: An Intuitive and Fundamental Guide*. John Wiley & Sons, 2020.

Prasad and Bhargava, 2019. KNR Surya Vara Prasad and Vijay K Bhargava. *RSS-based positioning in distributed massive MIMO under unknown transmit power and pathloss exponent*. pages 1–5, 2019.

Shen et al., 2021. Zhexian Shen, Kui Xu and Xiaochen Xia. *2D fingerprinting-based localization for mmWave cell-free massive MIMO systems*. IEEE Communications Letters, 25(11), 3556–3560, 2021.

Talvitie et al., 2020. Jukka Talvitie, Toni Levanen, Mike Koivisto, Tero Ihalainen, Kari Pajukoski and Mikko Valkama. *Beamformed radio link capacity under positioning uncertainty*. IEEE Transactions on Vehicular Technology, 69(12), 16235–16240, 2020.

Wang et al., 2019. Qing Wang, Xian Wang, Hua Chen, Xiaotian Zhu, Wei Liu, Weiqing Yan and Laihua Wang. *An effective localization method for mixed far-field and near-field strictly non-circular sources*. Digital Signal Processing, 94, 125–136, 2019.

Zhang et al., 2020. Jiayi Zhang, Emil Björnson, Michail Matthaiou, Derrick Wing Kwan Ng, Hong Yang and David J Love. *Prospective multiple antenna technologies for beyond 5G*. IEEE Journal on Selected Areas in Communications, 38 (8), 1637–1660, 2020.