UNIVERSITÉ LAVAL

Fingerprint-Based Localization in Massive MIMO Systems Using Machine Learning and Deep Learning Methods

Thèse

Seyedeh Samira Moosavi

Doctorat en génie électrique
Philosophiæ doctor (Ph. D.)

Québec, Canada

Fingerprint-Based Localization in Massive MIMO Systems Using Machine Learning and Deep Learning Methods

Thèse

Seyedeh Samira Moosavi

Sous la direction de:

Résumé

À mesure que les réseaux de communication sans fil se développent vers la 5G, une énorme quantité de données sera produite et partagée sur la nouvelle plate-forme qui pourra être utilisée pour promouvoir de nouveaux services. Parmis ceux-ci, les informations de localisation des terminaux mobiles (MT) sont remarquablement utiles. Par exemple, les informations de localisation peuvent être utilisées dans différents cas de services d'enquête et d'information, de services communautaires, de suivi personnel, ainsi que de communications sensibles à la localisation. De nos jours, bien que le système de positionnement global (GPS) des MT offre la possibilité de localiser les MT, ses performances sont médiocres dans les zones urbaines où une ligne de vue directe (LoS) aux satellites est bloqué avec de nombreux immeubles de grande hauteur. En outre, le GPS a une consommation d'énergie élevée. Par conséquent, les techniques de localisation utilisant la télémétrie, qui sont basées sur les informations de signal radio reçues des MT tels que le temps d'arrivée (ToA), l'angle d'arrivée (AoA) et la réception de la force du signal (RSS), ne sont pas en mesure de fournir une localisation de précision satisfaisante. Par conséquent, il est particulièrement difficile de fournir des informations de localisation fiables des MT dans des environnements complexes avec diffusion et propagation par trajets multiples. Les méthodes d'apprentissage automatique basées sur les empreintes digitales (FP) sont largement utilisées pour la localisation dans des zones complexes en raison de leur haute fiabilité, rentabilité et précision et elles sont flexibles pour être utilisées dans de nombreux systèmes.

Dans les réseaux 5G, en plus d'accueillir plus d'utilisateurs à des débits de données plus élevés avec une meilleure fiabilité tout en consommant moins d'énergie, une localisation de haute précision est également requise. Pour relever un tel défi, des systèmes massifs à entrées multiples et sorties multiples (MIMO) ont été introduits dans la 5G en tant que technologie puissante et potentielle pour non seulement améliorer l'efficacité spectrale et énergétique à l'aide d'un traitement relativement simple, mais également pour fournir les emplacements précis des MT à l'aide d'un très grand nombre d'antennes associées à des fréquences porteuses élevées. Il existe deux types de MIMO massifs (M-MIMO), soit distribué et colocalisé.

Ici, nous visons à utiliser la méthode basée sur les FP dans les systèmes M-MIMO pour

fournir un système de localisation précis et fiable dans un réseau sans fil 5G. Nous nous concentrons principalement sur les deux extrêmes du paradigme M-MIMO. Un grand réseau d'antennes colocalisé (c'est-à-dire un MIMO massif colocalisé) et un grand réseau d'antennes géographiquement distribué (c'est-à-dire un MIMO massif distribué). Ensuite, nous extrayons les caractéristiques du signal et du canal à partir du signal reçu dans les systèmes M-MIMO sous forme d'empreintes digitales et proposons des modèles utilisant les FP basés sur le regroupement et la régression pour estimer l'emplacement des MT. Grâce à cette procédure, nous sommes en mesure d'améliorer les performances de localisation de manière significative et de réduire la complexité de calcul de la méthode basée sur les FP.

Abstract

As wireless communication networks are growing into 5G, an enormous amount of data will be produced and shared on the new platform, which can be employed in promoting new services. Location information of mobile terminals (MTs) is remarkably useful among them, which can be used in different use cases of inquiry and information services, community services, personal tracking, as well as location-aware communications.

Nowadays, although the Global Positioning System (GPS) offers the possibility to localize MTs, it has poor performance in urban areas where a direct line-of-sight (LoS) to the satellites is blocked by many tall buildings. Besides, GPS has a high power consumption. Consequently, the ranging based localization techniques, which are based on radio signal information received from MTs such as time-of-arrival (ToA), angle-of-arrival (AoA), and received signal strength (RSS), are not able to provide satisfactory localization accuracy. Therefore, it is a notably challenging problem to provide precise and reliable location information of MTs in complex environments with rich scattering and multipath propagation. Fingerprinting (FP)-based machine learning methods are widely used for localization in complex areas due to their high reliability, cost-efficiency, and accuracy and they are flexible to be used in many systems.

In 5G networks, besides accommodating more users at higher data rates with better reliability while consuming less power, high accuracy localization is also required in 5G networks. To meet such a challenge, massive multiple-input multiple-output (MIMO) systems have been introduced in 5G as a powerful and potential technology to not only improve spectral and energy efficiency using relatively simple processing but also provide an accurate locations of MTs using a very large number of antennas combined with high carrier frequencies. There are two types of massive MIMO (M-MIMO), distributed and collocated.

Here, we aim to use the FP-based method in M-MIMO systems to provide an accurate and reliable localization system in a 5G wireless network. We mainly focus on the two extremes of the M-MIMO paradigm. A large collocated antenna array (i.e., collocated M-MIMO) and a large geographically distributed antenna array (i.e., distributed M-MIMO). Then, we extract signal and channel features from the received signal in M-MIMO systems as fingerprints and propose FP-based models using clustering and regression to estimate MT's location.

Through this procedure, we are able to improve localization performance significantly and reduce the computational complexity of the FP-based method.

Contents

List of Tables

List of Figures

List of Algorithms

List of Acronyms

3GPP	Third-Generation Partnership Project
5G	Fifth-Generation
ADCAM	Angle-Delay Channel Amplitude Matrix
AoA	Angle of Arrival
AoD	Angle of Delay
ANN	Artificial Neural Network
APC	Affinity Propagation Clustering
BFGS	Broyden-Fletcher-Goldfarb-Shannon
BS	Base Station
Cell-ID	Cell Identity
CoO	Cell of Origin
CFR	Channel Frequency Response
CH	Calinski-Harabasz
CIR	Channel Impulse Response
COO	Cell of Origin
CSI	Channel State Information
CU	Computing Unit
CM-MIMO	Collocated Massive Multiple-Input Multiple-Output
DB	Davies-Bouldin
DFT	Discrete Fourier Transform
DL	Deep Learning
DM-MIMO	Distributed Massive Multiple-Input Multiple-Output
DoA	Direction of arrival
FP	Fingerprinting
FSPL	Free Space Path Loss
GP	Gaussian Process
GPR	Gaussian Process Regression
GPS	Global Positioning Systems
GP	Gaussian Process
ID	Identity
JADSC	Joint Angle-Delay Similarity Coefficient
KD-tree	K-Dimensional tree
KM	K-Means
LBS	Location Based Service
LoB	Lines of Bearing

LoS	Line-of-Sight
LDPL	Log-Distance Path Loss
M-MIMO	Massive Multiple-Input Multiple-Output
ML	Machine Learning
MLP	Multi-Layer Perceptron
MT	Mobile Terminal
MU-MIMO	Multi-User MIMO
NLoS	Non-Line-of-Sight
NN	Neural Networks
OFDM	Orthogonal Frequency Division Multiplexing
PCA	Principal Component Analysis
ReLU	Rectified Linear Unit
RMSE	Root Mean Squared Error
RSS	Received Signal Strength
RRH	Remote Radio Head
RP	Reference Point
RTT	Round Trip Time
SI	Silhouette
t-SNE	t-distributed Stochastic Neighbor Embedding
TDD	Time-Division Duplex
ToA	Time of Arrival
UE	User Equipment
URA	Uniform Rectangular Array
ULA	Uniform Linear Array

List of Symbols

$\mathbf{a}_{f,k}$	Channel impulse response vector associated with fth path of kth user
b_0	Path loss at a reference distance
$\mathbf{b}^k$	Bias vector of kth layer
B	Number of features of $\mathbf{r}_l$
c_t	Cluster ID
$\mathbf{c}_t$	Vector of cluster ID
C_i	ith cluster head
$\mathbf{C}_t$	Covariance matrix of ADCPM fingerprints
$\mathcal{C}_{prm_j}$	Set of cluster ID corresponding to clusters created by parameter prm_j
cfm_{prm_j}	Cluster identification accuracy
cfm	Vector of cluster identification accuracy
cm	Cluster validation values
cm_{prm_j}	Validity index of cluster created with prm_j parameter
CH	Calinski-Harabasz index value
$d(.,.)$	Euclidean distance
d_0	Reference distance
d_{mk}	Distance between mth RRh and kth user
$d(\mathbf{p}_l)$	Average distance between $\mathbf{p}_l$ and all training RSS vectors in other clusters
$\text{diam}(c_t)$	Diameter of cluster c_t
D	Dimensions of data after using dimensionality reduction
DB	Davies-Bouldin index value
$\mathbf{e}_i$	ith eigenvector of covariance matrix
$\mathbf{e}(\theta)$	Array response vector
$\mathbf{E}$	Eigenvector matrix
$f_x(.)$	Function which map the RSS vector $\mathbf{p}_k$ of any user k to its x-coordinate location
$f_y(.)$	Function which map the RSS vector $\mathbf{p}_k$ of any user k to its y-coordinate location
$f(\mathbf{p}_l)$	Average distance between $\mathbf{p}_l$ and all training RSS vectors in the same cluster
F	Number of scattering paths
$\mathbf{F}$	Phase-shifted discrete Fourier transform (DFT) matrix
g_{mk}	Flat-fading channel gain between user k and RRH m
G_r	Receiver antennas' gain
G_t	Transmitter antennas' gain
$\mathbf{G}$	Channel matrix between K users and the BS antenna array
$\mathcal{GP}$	Gaussian process with zero-mean and covariance matrix $\mathbf{Q}_t$
h_{mk}	Large-scale fading coefficient

$H(p_l)$	Shannon entropy of p_l
$\mathbf{h}^k$	Output from kth layer
$\mathbf{H}_k$	Overal CFR matrix
I	Number of eigenvalues
K	Number of single-antenna users or MTs
K_{nn}	Number of nearest neighbours of a testing data point
L	Number of training data points
$\hat{L}$	Number of testing data points
M	Number of single-antenna RRHs
N	Number of validation data
N_{sc}	Number of subcarrier
N_g	Number of guard subcarriers
$\mathbf{n}_m$	Additive white Gaussian noise
$\mathbf{N}$	Standardized radio map of one-dimensional array of ADCPM fingerprints
p_0^{dB}	Uplink RSS at reference distance d_0.
p_{mk}	RSS of user k at RRH m
$\mathbf{p}_k$	Uplink RSS vector of user k
$\mathbf{p}_{zt}$	Centroid RSS vector of cluster t
$\mathbf{p}_{ztot}$	Centroid RSS vector among all training RSS vectors
$\hat{\mathbf{p}}$	Testing RSS vector
$\mathcal{P}_{prm_j}$	Set of clusters created by parameter prm_j
PL_F	Free-space path loss
PL_{LD}	Log-distance path loss
P_r	Received signal's power
P_t	Transmitted signal's power
$pref$	Preference value
$Perp$	Number of effective neighbors that each data point has
$\mathbf{prm}$	Clustering parameters
$\mathbf{P}$	Training RSS matrix
$\mathbf{P}_{High}$	Similarity matrix in the original high dimensional space
$\mathbf{P}_t$	Training RSS data points in cluster t
$\hat{\mathbf{P}}$	Testing RSS matrix
q_t	User-defined function of covariance matrix in tth cluster
$\mathbf{q}_t$	Vector of q_t between test RSS $\hat{\mathbf{p}}$ and u training RSS in tth cluster
q_{mk}	Small-scale fading
$\mathbf{Q}_t$	Covariance matrix of tth cluster
$\mathbf{Q}_{Low}$	Similarity matrix in low dimensional space
R	Distance between two BSs
$\mathbf{R}_k$	ADCPM fingerprint of user k
$\mathbf{R}$	Radio map of one-dimensional array of ADCPM fingerprint
$\mathbf{r}_l$	One-dimensional array of ADCPM fingerprint
$RMSE$	Root mean squared error
$s(\mathbf{p}_l, \mathbf{p}_{l''})$	Pairwise similarity of $\mathbf{p}_l$ and $\mathbf{p}_{l''}$
$\mathbf{s}_l$	Standardized training RSS vector
SI	Silhouette index value
$\mathbf{S}$	Standardized radio map
$\mathbf{S}_{Sim}$	Similarities matrix
$\mathbf{S}_{prmj,t}$	Training data and corresponding label of tth cluster
t_0	Receiving time measured at receiver

t_i	Sending time measured at ith sender
T	Number of clusters
T_s	Sample interval
$\mathcal{TR}$	Training set
u_t	Number of training RSS vectors in tth cluster
$u_{f,k}$	Rounding function of the ToA of each path $\tau_{f,k}$ and sample interval T_s
$\mathbf{U}$	Low dimensionality radio map
v	Speed of light
v_x	Gaussian random variables with zero mean and variance $\sigma^2_{v_x}$
v_y	Gaussian random variables with zero mean and variance $\sigma^2_{v_y}$
$\mathcal{V}$	Validation set
$\mathbf{w}_k$	Symbol vector transmitted by user k
$\mathbf{w}_{mu}$	Symbol vector transmitted by K users
$\mathbf{W}^k$	Weight matrix
$\mathbf{x}$	x-coordinates
$\hat{x}$	Estimated x-coordinate
$\mathbf{y}$	y-coordinates
$\hat{y}$	Estimated y-coordinate
$\mathbf{y}_m$	Sum of symbol vector transmitted from user k and received at RRH m
$\mathbf{y}_{mu}$	Sum of all symbols transmitted from K users and received at the BS antenna
z_{mk}	Log-normal shadowing noise coefficient between RRH m and user k
ρ	Transmission power
α	Path-loss exponent
$\sigma^2_{v_x}$	Variance of Gaussian random variable v_x
$\sigma^2_{v_y}$	Variance of Gaussian random variable v_y
$\hat{\sigma}^2_x$	Uncertainty in estimating x-coordinate
δ	Delta function
σ^2_z	Variance of shadowing noise
θ	AoA value
$\boldsymbol{\theta}_t$	Unknown GPR parameter or hyperparameter vector of cluster$t$ for RSS fingerprints
$\boldsymbol{\phi}_t$	Unknown GPR parameter or hyperparameter vector of cluster$t$ for ADCPM fingerprints
λ_i	ith eigenvalue
λ	Wavelength
$\hat{\boldsymbol{\theta}}_t$	Maximum likelihood estimator of parameter $\boldsymbol{\theta}_t$
κ	First hyperparameter of the squared-exponential function in GPR for RSS data
η	Second hyperparameter of the squared-exponential function in GPR for RSS data
β	First hyperparameter of the linear function in GPR for RSS data
γ	Second hyperparameter of the squared-exponential function in GPR for ADCPM fingerprint
ϑ	Hyperparameter of the squared-exponential function in GPR for ADCPM fingerprint
ϖ	Hyperparameter of the linear function in GPR for ADCPM fingerprint
$\delta_{kk'}$	Delta fuction between user k and user k'
$\hat{\mu}_x$	Mean of Gaussian distribution and the best estimator for $\hat{x}$
Λ	Diagonal matrix
$\phi(.)$	Activation function in DNN
$\tau_{f,k}$	ToA of the signal received from kth user through fth path
$\omega_{f,k}$	Complex attenuation of fth path
$\mathbf{\Omega}_k$	ADCPM matrix
Σ	Covariance matrix of radio map $\mathbf{S}$

To my Family

Foreword

This thesis is written based on technical articles. Chapters 2, 3, and 4, respectively, contain the materials of three journal papers that I wrote during my Ph.D. study. For each paper based chapter, the section Résumé is first added at its beginning to facilitate the reading for French-speaking audiences. Then, to form a consistent representation of the used variables, some symbols in the original papers are adjusted in this thesis. The author's contributions to these papers and the information used in this thesis are as follows.

Chapter 2 is based on the following paper: S. S. Moosavi and P. Fortier, "Fingerprinting Positioning in Distributed Massive MIMO Systems Using Affinity Propagation Clustering and Gaussian Process Regression", submitted to Wireless Personal Communication, Springer Science+Business Media, Feb., 2020. In this paper, an optimal clustering scheme based on affinity propagation clustering (APC) and Gaussian process regression (GPR) is presented to minimize the searching space of reference points, reduce the computational complexity, and increase positioning accuracy in distributed massive multiple-input multiple-output (DM-MIMO) systems. Paul Fortier and I jointly conceptualized the fundamental idea of this work. I developed the methodology and conducted related simulations and numerical conclusions. Paul Fortier provided critical guidance and comments on this work. The manuscript was written by me and revised by Paul Fortier. Also, a conference paper of this work was accepted: S. S. Moosavi and P. Fortier, "Fingerprinting Localization Method Based on Clustering and Gaussian Process Regression in Distributed Massive MIMO Systems." 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, Sep., 2020.

Chapter 3 is based on the following paper: S. S. Moosavi and P. Fortier, "An Improved and Low-dimensional Fingerprint-based Localization Method in Collocated Massive MIMO - OFDM Systems", has been accepted for publication in Wireless Personal Communication, Springer Science+Business Media, Jan., 2021. In this paper, we propose an improved and low-dimensional FP-based localization method in collocated massive MIMO (CM-MIMO) orthogonal frequency division multiplexing (OFDM) systems to decrease the computational complexity and the positioning error. I conceived the main idea of this study with the guidance of Paul Fortier. I developed the methodology and carried out related simulations and

numerical conclusions. Paul Fortier provided essential comments and suggestions on the methodology and numerical results and revised the manuscript. Also, a conference paper of this study was accepted: S. S. Moosavi and P. Fortier, "A Fingerprint Localization Method in Collocated Massive MIMO-OFDM Systems Using Clustering and Gaussian Process Regression", 92nd Vehicular Technology Conference (VTC2020-Fall), Nov., 2020.

Chapter 4 is based on the following paper: S. S. Moosavi and P. Fortier, "An Accurate, Robust and Low Dimensionslity Deep Learning Localization Approach in DM-MIMO Systems Based on RSS", submitted to Wireless Personal Communication, Springer Science+Business Media, Jan., 2021. In this paper, we proposed a low-dimensionality fingerprint-based localization method using dimensionality reduction technique, clustering and deep learning regression in DM-MIMO to increase the accuracy of localization in term of root-mean-squared error (RMSE). The basic idea of this work was conceived by me under the supervision of Paul Fortier. I developed the methodology and conducted related simulations and numerical conclusions. Paul Fortier provided essential comments and suggestion on the methodology and numerical results and revised the manuscript.

Acknowledgement

First of all, I would like to express my sincere thanks to my respected supervisor, Professor Paul Fortier, for his devotion to my doctoral study and research. His expertise and profound experience led me to the world of wireless communications. His consistent support and guidance, which directed me in the right direction to overcome all the challenges, not only helped me how to do doctoral research but also improved my ability to write high-quality technical papers. He helped to verify or improve my research methods and results, and to revise the technical papers that I wrote, and without his help, this project was so far away from the last station. Apart from providing essential research resources, he recommended to me many local and international conferences, so that I could exchange ideas with other professionals and broaden my horizons. Moreover, his encouragement kept me always confident in my project. I appreciate all he has done for me, without which I cannot finish this thesis. I could not have imagined having a better advisor for my Ph.D. study.

Second, special thanks to Professor Jean-Yves Chouinard, who collaborated in this project as co-supervisor and we could take the benefit of his help in the way of approaching to the solution for this project.

I thank the members of the doctoral supervisory committee for their time in evaluating my report for the pre-doctoral exam and my research progress. They provided me with insightful comments and recommendations, therefore, improving or accelerating my research. I should also thank the jury members, who are responsible for this thesis, for their time in reading and evaluating the thesis.

In addition, I thank Laval University and NSERC (Natural Sciences and Engineering Research Council of Canada) for supporting my doctoral study.

I also want to extend my thanks to the colleagues in the Radiocommunications and Signal Processing Laboratory at Laval University. They all made the laboratory a friendly and pleasant workplace. Particular thanks go to Professor Jean-Yves Chouinard for his wonderful lectures and insightful discussions. I also thank Professor Leslie Rusch for her outstanding lectures.

Last but not the least, my gratitude goes to my family: my parents and my husband Mr. Hadi Ghazanfari for supporting me spiritually throughout writing this thesis and my life in general.

Introduction

Nowadays, positioning has been identified as an important application in fifth-generation (5G) cellular networks. The positioning accuracy is expected to improve dramatically with 5G features such as wideband signals, massive antenna arrays, and ultra-dense network deployments. Massive multiple-input multiple-output (M-MIMO) is an enabling technology for accurate positioning besides improving communication performance. However, positioning using M-MIMO based on radio signal is still an issue in urban environments where there are multipath propagation and non-line-of-sight (NLoS) signal conditions. The fingerprinting (FP)-based method using machine learning (ML) algorithms plays an important role in overcoming this problem. This chapter introduces 5G systems, positioning in 5G using massive MIMO, and the motivations and objectives. Finally, the thesis outlines and research contributions are also presented.

5G Systems

During the last two decades, the accelerated evolution of cellular communication technologies has been observed. However, the ever-increasing mobile data volume and the lack of global bandwidth are still the principal challenges of current wireless networks. The main motivation of 5G wireless networks is to provide gigahertz (GHz) bandwidth for high data rate communication with low latency and higher localization accuracy. Recently, there have been significant developments in the research on 5G networks. Several enabling technologies are being explored for the 5G mobile systems era, such as Millimeter-wave (mm-wave) band, new waveform, and M-MIMO systems. The mm-wave band provides 5G systems with an amount of bandwidth on the order of GHz. It also can provide data rate on the order of gigabits per second (Gbps). New waveforms can provide higher spectral efficiency, lower in-band and out-band emission, lower power consumption, and lower implementation complexity. M-MIMO systems improve spectral, and energy efficiencies [1], [2]. They can also be used for localization due to many antenna arrays at the base station (BS) which can capture the signal. Therefore, they help improve positioning accuracy in addition to enhancing communication performance [3].

M-MIMO systems are considered as systems with large number of antennas at the BS which serves tens or hundreds of single-antenna MTs simultaneously with the same time-frequency resource [1], [2]. Two paradigms can be used to deploy the M-MIMO systems: co-located massive MIMO (CM-MIMO) and distributed massive MIMO (DM-MIMO). In CM-MIMO, the antenna arrays at the BS are located in a compact area, while in DM-MIMO, there is a large number of single-antenna remote radio heads (RRHs) which are geographically spread out over a large area and connected with high-speed front-haul links to a computing unit (CU) [2].

Localization in Massive MIMO Systems

Positioning has become an important application in 5G due to its significant potential in extensive commercial use cases, e.g., industrial automation, remote operation, emergency call-outs, personal tracking, and location-aware communications [4]. Therefore, besides providing higher data rates and lower latency, high accuracy positioning is also required in 5G networks. Although the Global Positioning System (GPS) [5] provides a precise estimation of mobile terminals (MTs) position, it needs a direct line-of-sight (LoS) to the satellites, which is blocked in urban areas by many tall buildings [6]. There are also ranging-based positioning techniques [7–10] which are based on radio signal information received from MTs such as time-of-arrival (ToA), angle-of-arrival (AoA), direction-of-arrival (DoA), and received signal strength (RSS). In 5G, M-MIMO can be used for localization by estimating radio signal information due to using a large number of antenna arrays. However, radio-based positioning is always a challenging task in NLoS urban environments, where the signal between the MT and the receiving antennas is blocked and reflected with high-rising buildings block. Therefore, the estimated position of the MT is not reliable when it enters an area with NLoS situations.

One approach for positioning in NLoS is to use FP-based methods, referred to herein as machine learning algorithms. Detection of NLoS is also an important feature, that makes it possible to decide when traditional positioning methods can and cannot be used. The FP-based positioning methods are attracting growing interest in combining mobile positioning requirements into the 5G wireless communication systems due to their broad applicability and high cost-efficiency without any hardware requirement on the MTs [11].

Using FP-based method in M-MIMO systems is a strategic solution to provide low-cost, continuous, and accurate positioning [12]. But the use of FP-based method for positioning in massive MIMO has some challenges that should be solved. Here, we present some challenging aspects of using FP-based method in both DM-MIMO and CM-MIMO, which are addressed in the framework of the thesis. We discuss here the most important issues.

Motivations and Objectives

Position information in massive MIMO systems may enable many applications for 5G networks such as tracking of goods or products in intralogistics process, autonomous driving, navigating emergency forces, etc. However, providing high data rates and good localization accuracy is still a challenging problem especially in NLoS conditions. We aim to apply machine learning algorithms in M-MIMO systems to perform accurate positioning in urban environments with NLoS conditions. By using the M-MIMO system, large amounts of data are received on the uplink which can be potentially exploited for positioning and location awareness. However providing positioning in 5G has still many challenges that need to be addressed. We discuss in the following the three most important issues.

- **Accuracy:** In 5G the accuracy of positioning is expected to be less than 5 meters.
- **Computational complexity:** The positioning system uses fingerprints received at predefined locations that would affect the computational complexity. This means that to avoid an excess in computational complexity, the number and the dimensions of the training points (especially if it is online) should be carefully chosen.
- **Latency:** One of the challenges of FP-based methods is the relatively large latency which becomes an issue in making a model, in which many training data points are used.

The contributions of this thesis consists in applying FP-based positioning to massive MIMO antenna arrays. Our contributions are therefore:

- Proposing a new scheme to improve positioning accuracy in DM-MIMO.
- Proposing a low-dimension scheme to improve positioning accuracy in CM-MIMO.
- Considering a low dimension scheme regarding the computational complexity problem created by positioning in M-MIMO using the FP-based method.

Thesis Outline

The structure of this thesis is as follows. In Chapter 1, we review the different radio-based localization methods used in the literature. By reviewing these methodologies, we can have a better picture of the advantages and disadvantages of each approach, which will result in better methodology selection for our work.

In Chapter 2, a clustering scheme based on affinity propagation clustering (APC) and Gaussian process regression (GPR) is presented to minimize the search space of reference points, reduce the computational complexity, and increase positioning accuracy in distributed massive multiple-input multiple-output (DM-MIMO) systems.

In Chapter 3, an improved and low-dimensional FP-based localization method in collocated massive MIMO (CM-MIMO) orthogonal frequency division multiplexing (OFDM) systems is proposed to decrease the computational complexity and the positioning error.

In Chapter 4, we proposed a low-dimensionality positioning fingerprint-based localization method using dimensionality reduction technique, clustering and deep learning regression in DM-MIMO. Using the feature extraction method, we intend to deliver a feature dataset with an acceptable level of accuracy in positioning to reduce latency and root-mean-squared error (RMSE) of positioning.

Finally, a discussion and a conclusion are presented at the end of the thesis.

Chapter 1

Literature Review

As discussed in the Introduction, massive MIMO (M-MIMO) is one of the potential technologies which propose the possibility to enhance positioning accuracy in 5G networks and 5G will be the first generation to take advantage of position information that is accurate enough to be used in wireless network design and optimization. In this chapter, at first, we will provide an overview of M-MIMO system. Afterwards, a positioning systems classification is presented, and we give a state of the art review of positioning techniques and technologies by discussing the underlying mathematical principles and cellular positioning techniques, including FP-based methods.

1.1 Massive MIMO

Massive MIMO is a form of multi-user MIMO (MU-MIMO) systems where the number of BS antennas and the number of users are large. In massive MIMO, hundreds or thousands of BS antennas simultaneously serve tens or hundreds of users in the same frequency resource. Some main points of massive MIMO are [13]:

- **Time-division duplex (TDD) operation:** In massive MIMO with TDD, the channel estimation overhead is independent of the number of BS antennas. Since in massive MIMO, the number of antenna is large, TDD operation is superior.
- **Linear processing:** In massive MIMO the signal processing at the terminal ends deals with large dimensional matrices/vectors because the number of BS antennas and users is large. Therefore, simple signal processing is required. In massive MIMO, linear processing (linear combining schemes in the uplink and linear precoding schemes in the downlink) is almost optimal.
- **Complexity:** In massive MIMO, all the complexity is at the BS.

1.1.1 System Model

In massive MIMO, TDD operation is preferable which consists of three operations: channel estimation, uplink data transmission, and downlink data transmission. The uplink transmission is described in the next section. For explaining each operation, we consider a multi-user MIMO (MU-MIMO) system which consists of one BS and K active users. The BS is equipped with M antennas, while each user has a single-antenna as shown in Fig. 1.1. In general, each user can be equipped with multiple antennas. However, for simplicity of the analysis, we consider systems with single-antenna users. We assume that all K users share the same time-frequency resource. Furthermore, we assume that the BS and the users have perfect channel state information (CSI). The channels are acquired at the BS and the users during the training phase.

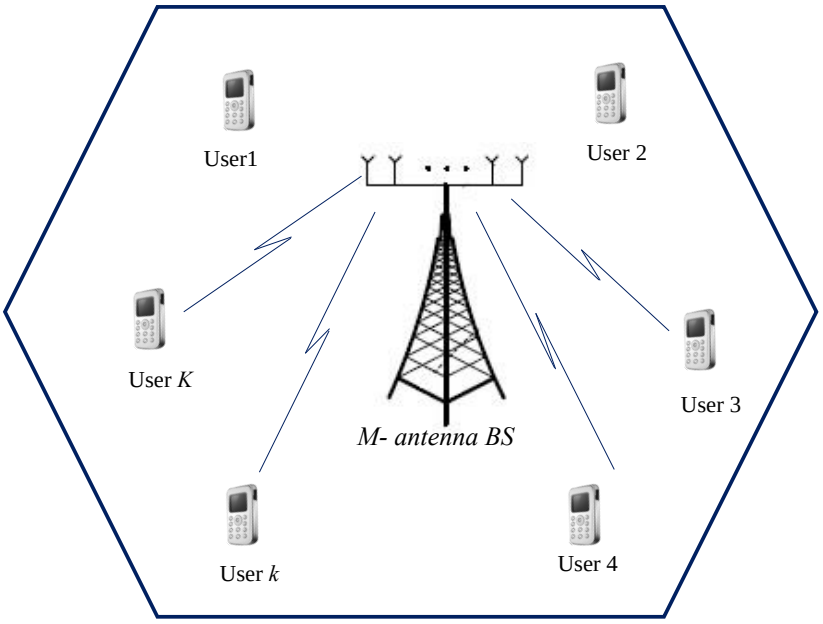


Figure 1.1: Multi-user MIMO Systems. Here, K single-antenna users are served by the M-antenna BS in the same time-frequency resource.

Uplink Transmission

In the uplink transmission, the users simultaneously transmit K symbols, $\mathbf{w}_{mu} = [w_1, \ldots, w_K]^T$, and the input signal power is normalized (i.e., $E(||\mathbf{w}_{mu}||^2) = 1$). The BS receives an $M \times 1$ signal vector which is the combination of all signals transmitted from all K users and defined

as:

$$\begin{aligned}\mathbf{y}_{mu} =& \sqrt{\rho}\sum_{k=1}^{K}\mathbf{g}_k w_k + \mathbf{n} \\ =& \sqrt{\rho}\mathbf{G}_{mu}\mathbf{w}_{mu} + \mathbf{n},\end{aligned} \tag{1.1}$$

where ρ is transmission power, $\mathbf{n} \in \mathbb{C}^M$ is the additive noise vector and $\mathbf{G}_{mu} \in \mathbb{C}^{M\times K}$ is the channel matrix between the K users and the BS antenna array, with entries $g_{mk} = q_{mk}\sqrt{h_{mk}}$ $(m = 1,\ldots,M, k = 1,\ldots,K)$ where h_{mk} and $h_{m,k}$ are, respectively, the small-scale and large-scale fadings between the k^{th} user and the m^{th} BS antenna [14].

1.2 Basic Mathematical Principles Used in Positioning

Two mathematical techniques are usually employed for calculating the position of a UE using signals received from several transmitters: triangulation and trilateration.

1.2.1 Triangulation

Triangulation is a geometric technique applying triangles to measure the position. This principle involves combining information received from measurements made by the BS to determine the user position. Distance and position are found in AoA systems (i.e., where the AoA is known) using this principle. An example is shown in Fig. 1.2. Two BSs, BS_1 and BS_2 are located on known coordinates, (x_1, y_1) and (x_2, y_2), also both BSs are separated by distance R. The AoAs at the two BSs are θ_1 and θ_2 which are known. From trigonometry, we obtain (x, y) the coordinates of MT as:

$$x = \frac{R\tan(\theta_2)}{\tan(\theta_2) - \tan(\theta_1)} \tag{1.2}$$

$$y = \frac{R\tan(\theta_1)\tan(\theta_2)}{\tan(\theta_2) - \tan(\theta_1)} \tag{1.3}$$

To find the distances between BS_1, BS_2, and the MT, we use the following equations.

$$d_1 = \|(x,y),(x_1,y_1)\| = \sqrt{(x-x_1)^2 + (y-y_1)^2} \tag{1.4}$$

$$d_2 = \|(x,y),(x_2,y_2)\| = \sqrt{(x-x_2)^2 + (y-y_2)^2} \tag{1.5}$$

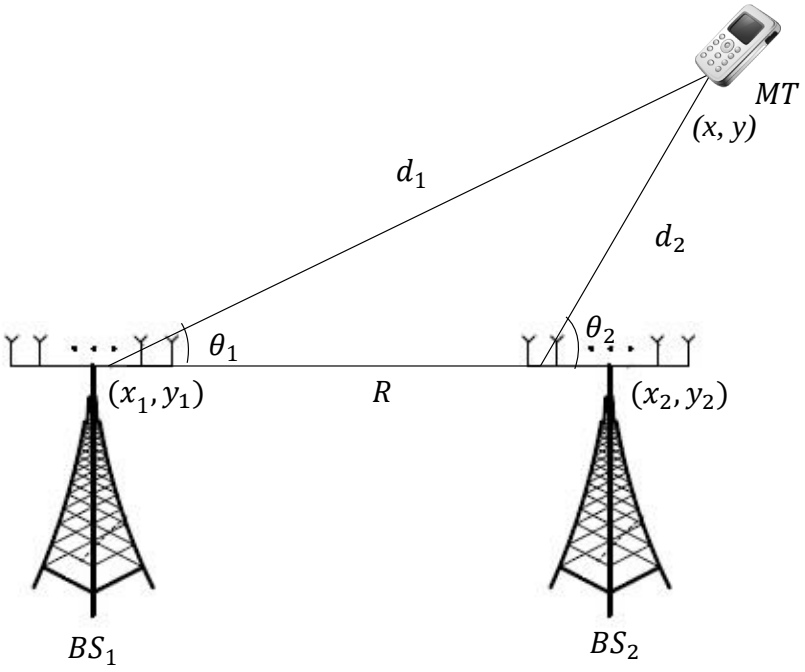


Figure 1.2: Illustration of positioning using triangulation.

1.2.2 Trilateration

Trilateration is the process of estimating the position of a user equipment (UE) given distance measurements to a set of BSs with a known position. Briefly, the concept of the trilateration technique is to measure the UE's position based on the following requirements:

1. The distances between the target point and a number of reference points whose positions are known.
2. Coordinates (positions) of reference points.

Fig 1.3 shows an ideal positioning-estimation scenario where there are three BSs with known fixed positions. The goal is to measure the two-dimensional position of the MT while the distance between the MT and all BSs are known and are obtained by measuring using some positioning method like received signal strength (RSS) or time of arrival as described later.

According to Fig. 1.3, the distance between the MT and the receivers can be estimated as:

$$d_i^2 = (x_i - x)^2 + (y_i - y)^2, i = 1, 2, ... \tag{1.6}$$

where d_i is the distance between the MT and i^{th} BS and (x_i, y_i) and (x, y) are the coordinates of the i^{th} BS and the MT, respectively. In the case of three BS, each distance is defined as follows:

$$d_1^2 = (x_1 - x)^2 + (y_1 - y)^2 \tag{1.7}$$

$$d_2^2 = (x_2 - x)^2 + (y_2 - y)^2 \tag{1.8}$$

$$d_3^2 = (x_3 - x)^2 + (y_3 - y)^2 \tag{1.9}$$

By considering BS_1 at the origin, i.e., $x_1 = y_1 = 0$ and subtracting (1.7) from (1.8) and (1.7) from (1.9), we obtain:

$$d_2^2 - d_1^2 = x_2^2 - 2x_2x + y_2^2 - 2y_2y \tag{1.10}$$

$$d_3^2 - d_1^2 = x_3^2 - 2x_3x + y_3^2 - 2y_3y \tag{1.11}$$

These two equations can be written in matrix form as:

$$\begin{bmatrix} x_2 & y_2 \\ x_3 & y_3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{2} \begin{bmatrix} A_2^2 - d_2^2 + d_1^2 \\ A_3^2 - d_3^2 + d_1^2 \end{bmatrix} \tag{1.12}$$

where $A_i^2 = x_i^2 + y_i^2$ and we thus have:

$$\mathbf{Ab} = \mathbf{f} \tag{1.13}$$

where

$$\mathbf{A} = \begin{bmatrix} x_2 & y_2 \\ x_3 & y_3 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \mathbf{f} = \frac{1}{2} \begin{bmatrix} A_2^2 - d_2^2 + d_1^2 \\ A_3^2 - d_3^2 + d_1^2 \end{bmatrix} \tag{1.14}$$

Therefore

$$\mathbf{b} = \mathbf{A}^{-1}\mathbf{f} = -\frac{1}{2(x_3y_2 - x_2y_3)} \begin{bmatrix} y_3 & -y_2 \\ -x_3 & x_2 \end{bmatrix} \begin{bmatrix} A_2^2 - d_2^2 + d_1^2 \\ A_3^2 - d_3^2 + d_1^2 \end{bmatrix} \tag{1.15}$$

Finally (x, y) can be obtained as

$$x = -\frac{1}{2} \left(\frac{y_2(d_3^2 - x_3^2 - d_1^2 - y_3^2) + y_3(d_1^2 - d_2^2 + y_2^2 + x_2^2)}{x_3y_2 - x_2y_3} \right) \tag{1.16}$$

$$y = \frac{1}{2} \left(\frac{x_2(d_3^2 - x_3^2 - d_1^2 - y_3^2) + x_3(d_1^2 - d_2^2 + y_2^2 + x_2^2)}{x_3y_2 - x_2y_3} \right) \tag{1.17}$$

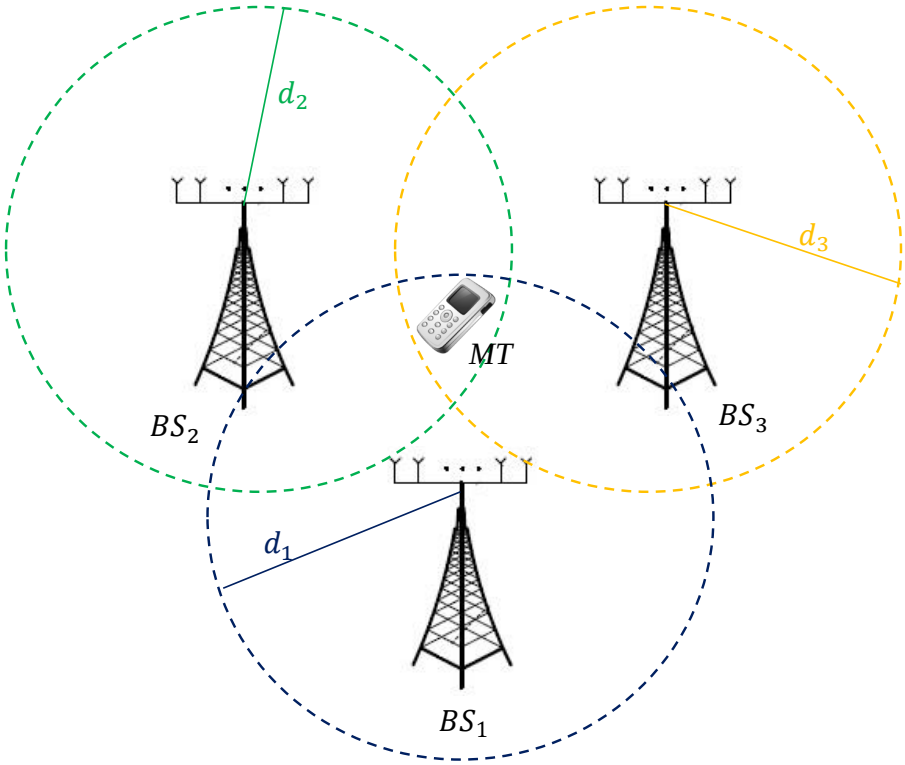


Figure 1.3: Illustration of positioning using trilateration.

1.3 Cellular positioning

Cellular positioning refers to the positioning techniques implemented in cellular networks. Such mechanisms can be applied for both outdoor and indoor situations.

1.3.1 Cell Identity (Cell-ID)

Cell-ID positioning, also known as cell of origin (CoO) positioning, is a proximity-based method which is the most straightforward approach for positioning [15]. In this method, which is shown in Fig. 1.4, by using the position of the base stations and their signal strength range, we can easily estimate the position of the UE at the cell level. Since the UE can be anywhere in the cell coverage area, the precision of the cell-ID method depends on the network topology and can vary from a few meters (i.e., in a pico-cellular or selected indoor scenario) to kilometers (i.e., in macrocellular settings) [16]. The best estimation is obtained in urban areas where the cell sizes are the smallest. Therefore, this method can be quickly ruled out since many BSs would be needed, which is not possible, for example in a suburban environment. To enhance the accuracy of this technique, additional parameters, such as round trip time (RTT) and RSS, can be used [15], [17], [18].

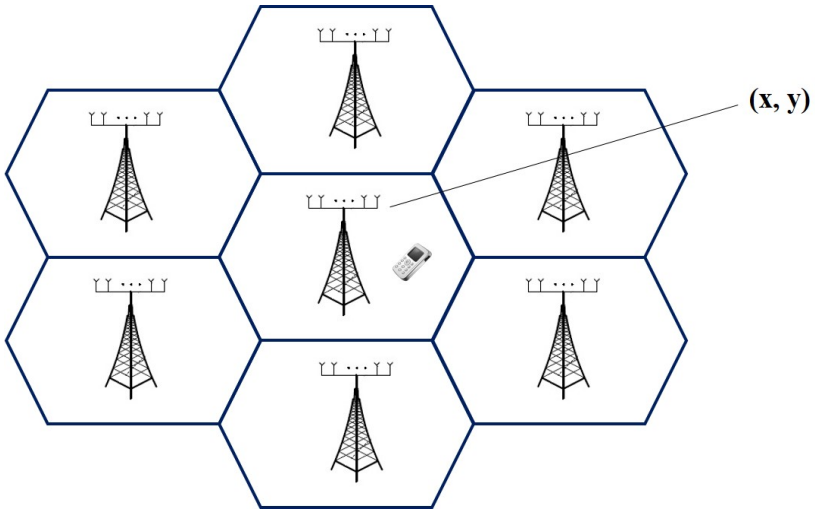


Figure 1.4: Cell-ID positioning technique.

1.3.2 Range-based methods

The foundation of many positioning techniques is based on the estimation of the physical distance between two nodes. Estimations are achieved by measuring the specific features of the signals transferred between the nodes, including signal propagation times, angle of arrival or signal strengths.

Angle of Arrival

The angle of arrival (AoA) is a range-based positioning technique which is used to estimate the Direction of Arrival (DOA). In AoA, the UE's position is estimated by measuring the angle of incidence at which signals arrive at the receiving antenna. Geometric principles such as triangulation can then be used to determine the position from the intersection of two lines of bearing (LoB) formed by a radial line to each receiving antenna, as shown in Fig. 1.5. In a two-dimensional plane, at least two receiving antennas are needed for position estimation. while with at least three or more receiving antennas accuracy is improved. This method works well in line-of-sight (LoS) situations, but suffers from decreased accuracy when faced with signal reflections from surrounding objects. Therefore, a conventional shortcoming that AoA shares with some of the other methods is its sensitivity to multipath interference. Unfortunately, AoA becomes hardly usable in dense urban areas [19].

Time of Arrival

Time of Arrival (ToA) systems are based on the accurate estimation of the arrival time of a signal transmitted from a MT to several receiving antennas [20]. Because signals propagate

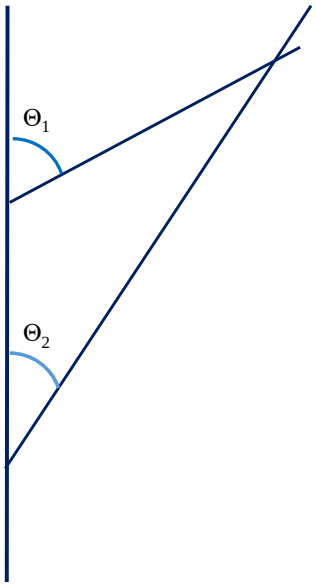


Figure 1.5: The AoA principle.

with a known speed (the speed of light (v) or about 300 meters per microsecond), the distance between the MT and each receiving antenna can be defined as:

$$d_i = (t_i - t_0) \times v \tag{1.18}$$

where t_i and t_0 are the sending and receiving times of the signal (measured at the sender and the receiver, respectively) and v is the signal velocity. Then, the position of the MT can be measured by using trilateration [19]. Since this method needs exact time synchronization between the MT and the receivers clocks, the involved MTs must be tightly synchronized, and time-stamp of the transmission signal is essential. ToA techniques can estimate position in two-dimensional as well as three-dimensional planes. By forming spherical instead of circular models, 3D resolution can be achieved. A shortcoming of the ToA method is the requirement for precise time synchronization of all MTs. Therefore, the position estimation accuracy is affected by the high propagation speeds or very small mistakes in time synchronization. For instance, a time measurement error as small as 100 nanoseconds can lead to a positioning error of 30 meters. ToA-based positioning approaches usually have challenges in environments where a large amount of multipath, noise or interference exist. The GPS is a common example of a ToA systems where precision timing is produced by atomic clocks [19], [20].

Received Signal Strength

A commonly observed feature in wireless devices is a received signal strength indicator (RSSI), which can be used to measure the power of the incoming radio signal at the receiver. The concept of RSS is that a signal decays and lose its energy with the distance travelled. Therefore, as the distance between transmitter and receiver increases, the RSSI value decreases. There are two different strategies for positioning using RSS. The first solution is trilateration, which calculates the position by distances obtained using RSS. In this method, at least three reference points are required to accurately determine the position of the user in two-dimensional coordinates. The second strategy is fingerprinting (FP) which is explained in the next section and is based on a database that stores the power of the various points. Several theoretical and empirical models could be used to state the correlation between RSSI and distance [21], [22]. One of these models is the path loss model which is used for large-scale fading. In a Free Space Path Loss (FSPL) model, the power of the received signal in the LOS environment where there is no obstacle between the transmitter and receiver [23], [24] is given by:

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2} \tag{1.19}$$

where P_r and P_t are the received and transmitted signal's power, and G_t and G_r are the receiver and transmitter antennas' gains, respectively. λ is the wavelength obtained from the transmitted signal's central frequency and d is the distance separating the transmitter and the receiver. All parameters are known in the FSPL model except d which can be calculated by solving (1.19). The free-space path loss, PL_F, without any system loss can be directly derived from (1.19) as:

$$PL_F(d)[dB] = 10\log\left(\frac{P_t}{P_r}\right) = -10\log\left(\frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2}\right) \tag{1.20}$$

without antenna gains (i.e., $G_t = G_t = 1$), (1.20) is reduced to:

$$PL_F(d)[dB] = 10\log\left(\frac{P_t}{P_r}\right) = 20\log\left(\frac{4\lambda d}{\pi}\right) \tag{1.21}$$

But, in LoS with multipath and NLoS conditions, channels are more complex because many reflections, refractions, and attenuations happen for transmitted signals on their way to the receiver. As a consequence, signal strength will be lost with increased distance [25]. In fact, a more generalized form of the path loss model can be formed by modifying the FSPL with the path loss exponent α varying with the propagation environments. This is known as the Log-distance Path Loss (LDPL) model and is defined as:

$$P_r(d) = P_t - PL_{LD}(d) \tag{1.22}$$

$$PL_{LD}(d)[dB] = PL_F(d_0) + 10\alpha \log\left(\frac{d}{d_0}\right) \tag{1.23}$$

where d_0 is the reference distance below which the path loss inherits the characteristics of free-space loss in (1.20). d_0 must be suitably estimated for various propagation environments. For instance, the value of d_0 typically varies from 1 km (for a cellular system with a cell radius greater than 10 km) to 100 m or 1 m (for a microcellular system with a cell radius of 1 km or a very small radius) [26]. It is clear that the path loss increases with the path loss exponent. Even if the distance between the transmitter and receiver is equal, every path may have different path loss since the surrounding environments may vary with the position of the receiver in practice. However, all the aforementioned path loss models do not take this particular situation into account. A log-normal shadowing model is useful when dealing with a more realistic situation. If z denotes a Gaussian random variable with a zero mean and a standard deviation of σ, then, the log-normal shadowing model is defined as:

$$PL_{LD}(d)[dB] = PL_F(d_0) + 10\alpha \log\left(\frac{d}{d_0}\right) + z \tag{1.24}$$

In other words, this particular model allows the receiver at the same distance d to have a different path loss, which varies with the random shadowing effect z.

Fingerprinting

The fingerprinting (FP) technique is an enhancement of the Cell-ID method, where the information of neighbor cells are also gathered to find the position of the MT. In the literature database correlation, pattern recognition and pattern matching can be used instead of FP. The theory of FP is that each particular position in the site of interest has a unique fingerprint. The FP method is simpler to use compared to other techniques such as AoA and TOA which require LOS between the transmitter and the receiver to estimate the position. In a multipath environment, the signal would be affected by multipath, and as a result, the position accuracy would be decreased. FP techniques have the potential to solve the problems stated above. They estimate the position of a mobile device very precisely using signal properties. These characteristics may be obtained from radio signals (i.e., RSS) [27], [28] and any other information that is position-dependent to create a radio map for the MT. When the radio map is made, MTs enter the second step where the position of a mobile can be recognized by matching the received signal to the radio map entries. Therefore in machine learning (ML) terminology, positioning by fingerprinting has two steps described as follows: the training phase and the positioning phase.

- **Training phase**
  The training phase can also be considered as the calibration phase which assists in

creating the database before the positioning phase. Through the training phase a radio database is constructed which includes a set of radio measurements made at a number of known positions over the specified area. These parameters are available from radio interface technologies, such as RSS, RTT, or the results of more complex processing such as path loss profiles. The radio measurements stored in the database are called fingerprints [29]. Each fingerprint stored in the database may be achieved by averaging several radio measurements produced at the same position but at different moments, or by averaging several measurements produced at different positions (as mentioned in [30]). So, the database is created by using either experimental measurements or theoretical modeling tools. The latter is used in the case of outdoor positioning where large surfaces are to be covered by the fingerprinting system. Therefore, a large-scale database is generated.

- **Positioning phase**
  In the positioning phase, the real-time measurements received by the MT are used to compare with the fingerprints existing in the database. Therefore, the coordinates of the reference point whose fingerprint returns the closest match should be the estimated position of the MT. At this stage, various positioning algorithms or matching algorithms may also be used to determine the position of the mobile by comparing the actual measurements (FP) to the ones stored in the database.

Chapter 2

Fingerprinting Positioning in Distributed Massive MIMO Systems Using Affinity Propagation Clustering and Gaussian Process Regression

Résumé

Les systèmes massifs à entrées multiples et sorties multiples (M-MIMO) améliorent la précision de positionnement en plus d'améliorer les performances de communication. La méthode d'empreinte digitale (FP) est largement utilisée pour les applications de positionnement en raison de sa fiabilité, de sa rentabilité et de sa précision élevées. La méthode FP basée sur la régression de processus gaussien (GPR) pourrait potentiellement être utilisée dans les systèmes M-MIMO pour améliorer la précision de positionnement. Cependant, elle est limitée par une complexité de calcul élevée. Dans ce chapitre, une méthode de positionnement FP basée sur le clustering de propagation d'affinité (APC) et le GPR est présentée pour estimer la position de l'utilisateur dans un système MIMO massif distribué (DM-MIMO) à partir de la force du signal reçu (RSS) sur la liaison montante. Dans le procédé proposé, un schéma de regroupement optimal basé sur l'APC est présenté pour diviser la zone cible en plusieurs petites régions, ce qui minimise l'espace de recherche des points de référence et réduit la complexité de calcul et l'erreur d'estimation de position. Ensuite, un modèle GPR est créé pour chaque région sur la base de la distribution de données RSS dans chaque région pour fournir une précision de positionnement supplémentaire. Une méthode améliorée basée sur l'arbre K-dimensionnel (KD-tree) est également présentée pour que les utilisateurs de test trouvent leur région la plus probable. Ensuite, leurs positions sont estimées sur la base du modèle GPR de cette région. Les résultats de la simulation révèlent que

le schéma proposé améliore considérablement la précision de positionnement par rapport à l'utilisation du seul GPR pour toute la zone cible. Cette approche a une couverture élevée et améliore les performances de la racine de l'erreur quadratique moyenne (RMSE) à quelques mètres, ce qui est attendu dans les réseaux 5G. Par conséquent, il contribue également à réduire la complexité de calcul du GPR dans les systèmes de positionnement.

Abstract

Massive multiple-input multiple-output (M-MIMO) systems improve positioning accuracy besides enhancing communication performance. Fingerprinting (FP) method is widely used for positioning applications due to its high reliability, cost-efficiency, and accuracy. The FP method based on Gaussian process regression (GPR) could potentially be used in M-MIMO systems to improve positioning accuracy. However, it is limited by high computational complexity. In this chapter, an FP positioning method based on the affinity propagation clustering (APC) and GPR is presented to estimate the user's position in a distributed massive MIMO (DM-MIMO) system from the uplink received signal strength (RSS). In the proposed method, an optimal clustering scheme based on APC is presented to split up the target area into several small regions, which minimizes the searching space of reference points and reduces the computational complexity and position estimation error. Then, a GPR model is created for each region based on the RSS data distribution within each region to provide further positioning accuracy. An improved method based on the K-dimensional tree (KD-tree) is also presented for test users to find their most likely region. Then their positions are estimated based on the GPR model of that region. Simulation results reveal that the proposed scheme improves positioning accuracy significantly compared to using only GPR for the whole target area. This approach has high coverage and improves average root-mean-squared error (RMSE) performance to a few meters, which is expected in 5G networks. Consequently, it also helps to reduce the computational complexity of GPR in the positioning systems.

2.1 Introduction

Increasing demand for location information makes positioning an important characteristic in fifth-generation (5G) networks [3]. Positioning has been applied in different use cases of inquiry and information services, community services, personal tracking, as well as location-aware communications [4]. Therefore, besides providing higher data rates and lower latency, high accuracy positioning is also required in 5G networks. Nowadays, the Global Positioning System (GPS) [5] is commonly used due to its availability. Although GPS provides a

precise estimation of mobile terminals (MTs) position, it needs a direct line-of-sight (LoS) to the satellites, which is blocked in urban areas with many tall buildings [6]. Therefore, it suffers loss accuracy in most situations. Besides, GPS has high power consumption which can quickly drain the battery on the mobile devices [31]. Consequently, there are ranging-based positioning techniques [7–10] which are based on radio signal information received from MTs such as time-of-arrival (ToA), angle-of-arrival (AoA), and received signal strength (RSS). Among these, the RSS-based methods are effective because RSS is not only suitable for line-of-sight (LoS) situations (AoA) [9], but can also be used for non-line-of-sight (NLoS) conditions and it doesn't need extra expensive hardware at the base station (BS) such as high accuracy clocks for time synchronization (ToA) [10]. Moreover, RSS data can be easily measured for most wireless systems [32]. But, in complex environments, it suffers from coarse range estimates that can be solved using fingerprint (FP) positioning methods based on machine learning (ML) algorithms [14].

FP-based positioning methods have two phases, namely offline and online. In the offline phase, a radio map of RSS information from the MT and their position is created and is stored. Afterward, a new RSS data is compared with the stored data in the online phase to estimate its position. Since FP-based positioning methods have a good performance in complex environments [14], [33], they are flexible to be used in many systems such as WiFi networks [34–36]. Apart from received signal information, channel state information (CSI) [37], [38] is used as the position fingerprint. The FP-based positioning methods are attracting growing interest in combining mobile positioning requirements into the 5G wireless communication systems due to their broad applicability and high cost-efficiency without any hardware requirement on the MTs [11].

In recent years massive multiple-input multiple-output (M-MIMO) has been introduced in 5G networks to help improve positioning accuracy in addition to enhancing communication performance [3]. M-MIMO is an emerging technology which serves tens or hundreds of UEs simultaneously with the same time-frequency resource to improve spectral and energy efficiencies [1], [2]. Two paradigms can be used to deploy the M-MIMO systems: collocated massive MIMO (CM-MIMO) and distributed massive MIMO (DM-MIMO). In CM-MIMO, the antenna arrays at the BS are located in a compact area, while in DM-MIMO, there is a large number of single-antenna remote radio heads (RRHs) which are geographically spread out over a large area and connected with high-speed front-haul links to a computing unit (CU) [2]. DM-MIMO gives higher spectral efficiency, average throughput rate [39] and bit-per-joule energy efficiency [40] and enhances coverage area compared to CM-MIM [39]. Therefore, to obtain such performance gains, DM-MIMO systems are useful for positioning [14]. User positioning with M-MIMO is still in its nascent stage. There are several research works in this area. In [41], [42], and [43], AoA is estimated precisely in the M-MIMO systems employing very large uniform rectangular arrays (URAs) to position users.

In [44], a compressed sensing approach is proposed in order to estimate user location directly from data acquired such as ToA information at multiple M-MIMO BSs. The authors in [45] and [46] consider the combined information of time delay, angle of delay (AoD), and AoA for positioning users in a M-MIMO system. The Gaussian process regression (GPR) method is employed in [14] to estimate position users using a vector of RSS which is considered as a fingerprint in a DM-MIMO system.

Several machine learning methods have been investigated for wireless user positioning, such as GP methods [14], and, more recently, deep learning techniques [47]. Also, clustering of the fingerprints has already been studied in several works. Since we consider a cluster-based method in our work, we choose the affinity propagation clustering (APC) algorithm [48] to divide the fingerprints of the target area into several clusters and the GPR method to make a non-linear regression model for each cluster for reasons that will be explained later in the chapter.

Studying positioning methods proposed for WiFi systems [49], [50], [37], we can see that they are not applicable for massive MIMO systems, because WiFi-based positioning methods do not consider the associated inter-user interference, which are present in multi-user MIMO transmissions on the uplink. Also, the positioning in M-MIMO systems is operated on the uplink, where the BS estimates the user's position. In contrast, most RSS-based WiFi positioning schemes [49], [50] focus on the downlink, where the users estimate their positions by handling the computational cost.

In this chapter, we consider a DM-MIMO system where first, the RRHs receive large amounts of data, which are large vectors of RSS, on the uplink, and then the received data is recorded at the BS [14]. Later, an affinity propagation clustering (APC) and Gaussian process regression (GPR)-based fingerprinting positioning system is presented which consists of two phases: the training phase and the positioning phase. In the first phase, a large area is divided into small clusters using an optimal clustering method, which is based on the APC algorithm, and the data distribution within each cluster, is precisely modeled using the GPR method [51]. Then, in the positioning phase, the user's cluster is first specified by a proposed cluster identification algorithm which is based on finding nearest neighbors using K-dimensional tree (KD-tree) and its position is estimated using the GPR model of that cluster. Simulation results of the proposed method indicate a significant improvement in position estimation accuracy by reducing the average root-mean-square error (RMSE) to a meter which is expected in 5G networks. It also exhibits better performance than the existing methods owing to RSS estimation using GPR [14]. Consequently, clustering helps to minimize the searching space of reference points on the testbed and thus reduces the online computational complexity of the positioning system.

The structure of this chapter is as follows. In Section 2.2, the system model for user

positioning is presented. In Section 2.3, we describe the details of the proposed positioning method. The results obtained through simulations are shown in Section 2.4. Finally, conclusions are provided in Section 2.5.

2.2 Multi-User DM-MIMO System Model

Since we want to estimate the positions of users using the RSS at the BS, we consider the uplink of a single-cell multi-user DM-MIMO system with K single-antenna users transmitting signals to M single-antenna RRHs, which are connected to a central computing unit (CU) by high-speed front haul links (Fig. 2.1). The RSS values from each RRH are gathered to form an $M \times 1$ RSS vector by the CU when users transmit on the uplink. Then, a machine learning model, which gets the uplink RSS vectors as input and gives the position coordinates of the K transmitting users as the output, is used in the CU [52], [53].

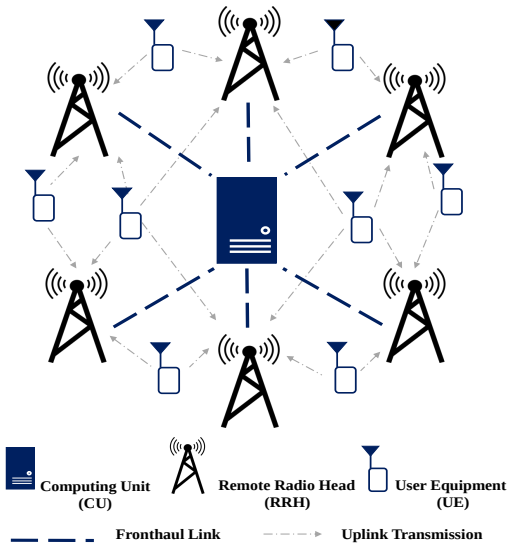


Figure 2.1: Multi-user DM-MIMO system model for position estimation.

To explain the uplink of a multi-user DM-MIMO system with more detail, let $\mathbf{w}_k$ be the symbol vector transmitted by user k with transmission power ρ. If g_{mk} is the flat-fading channel gain between user k and RRH m, the sum symbol vector $\mathbf{y}_m$ received at RRH m is given by

$$\mathbf{y}_m = \sqrt{\rho} \sum_{k=1}^{K} g_{mk} \mathbf{w}_k + \mathbf{n}_m \tag{2.1}$$

In (2.1), $g_{mk} = q_{mk}\sqrt{h_{mk}}$ is a flat-fading channel where q_{mk} denotes small-scale fading represented by an independent and identically distributed (i.i.d.) zero mean complex Gaussian random variable with unit variance, i.e., $q_{mk} \sim \mathcal{CN}(0,1)$, h_{mk} is the large-scale fading coef-

ficient, and $\mathbf{n}_m \sim \mathcal{N}(0, \sigma_n^2 \mathbf{I})$ is the additive white Gaussian noise. Note that the large-scale fading coefficient h_{mk} can be modeled [54] as

$$h_{mk} = b_0 d_{mk}^{-\alpha} 10^{z_{mk}/10} \tag{2.2}$$

where b_0 is the path loss at a reference distance d_0, d_{mk} is the distance between user k and RRH m, α is the path-loss exponent (typically dependent on the environment and the range), and z_{mk} is the log-normal shadowing noise coefficient with $10 \log_{10} z_{mk} \sim \mathcal{N}(0, \sigma_z^2)$.

For measuring the RSS, we consider the power of the received signal at RRH m which is given by $||\mathbf{y}_m||^2$ according to (2.1). But we should note that $||\mathbf{y}_m||^2$ at RRH m is in fact the multiuser RSS because the symbol vectors which are transmitted by all K users are combined at RRH m. Consequently, $||\mathbf{y}_m||^2$ cannot be directly used to estimate the position of user k. So the RSS of each user is not separately distinguishable. To overcome this, the symbol vectors $\mathbf{w}_k$ in (2.1) should be mutually orthogonal and should be already known at the RRH [14].

So, we need users to transmit an orthogonal set of pilot signals during channel estimation [55]. The RSS p_{mk} of user k can then be obtained from (2.1) [53] as

$$p_{mk} = \rho h_{mk} |q_{mk}|^2 \tag{2.3}$$

From (4.3), we can see that RSS is varied due to the small-scale fading and shadowing of the wireless channel. The variation of small-scale fading can be decreased by averaging it over multiple time-slots according to the channel hardening effect [56]. But the shadowing effect, which is position-dependent and therefore depends on the user location, cannot be averaged out [7]. Therefore, the RSS between user k and RHH m, which is obtained from (2.2) and (2.3), when converted to dB scale, is given [53] by

$$p_{mk}^{dB} = p_0^{dB} - 10\alpha \log_{10}(d_{mk}) + z_{mk} \tag{2.4}$$

where $p_0^{dB} = 10 \log_{10}(\rho b_0)$ is the uplink RSS at the reference distance d_0. Once the per-user RSS values p_{mk}, $m = 1,\ldots,M$ and $k = 1,\ldots,K$, are extracted as above, the CU uplink RSS vector $\mathbf{p}_k$ is given by

$$\mathbf{p}_k = [p_{1k}^{dB}, p_{2k}^{dB}, \ldots, p_{Mk}^{dB}]^T \tag{2.5}$$

which is considered as the fingerprint.

2.3 Fingerprinting Positioning Method Based on Clustering and GPR Model

The proposed position estimation scheme has two phases: a training phase and a positioning phase, as shown in Fig. 2.2. Briefly, in the training phase, the large target area is first divided into small regions using an APC algorithm based on collected RSS samples. Then the outcome of the clustering process is validated by a cluster validity index to obtain a set of number of clusters that best fits natural partitions without any prior information. Also, an improved cluster identification algorithm based on nearest neighbor is applied to find the optimal number of clusters. The data distribution within each cluster is then accurately modeled with GPR.

When a new vector of RSS is received in the positioning phase, its corresponding cluster (cluster ID) is first determined using the cluster identification algorithm. Therefore, the small region in which the new user is most probably located is distinguished. Within this region, further position estimation is applied using the GPR model of related clusters to increase the precision of the estimation.

2.3.1 Training Phase

In this phase, we aim to find optimal clustering results and create a GPR model based on RSS distribution in each cluster. Therefore, the RSS samples collected in the target area are analysed. First, the APC algorithm is used, and the clustering results are evaluated using a cluster validity index. Then the cluster identification algorithm using the nearest neighbor is applied. According to this, we can handle the trade-off between the accuracy of cluster identification and the number of clusters to identify the optimal number of clusters. Finally, a GPR model is fitted to each cluster.

Let us say we have L training locations. The $L \times M$ training RSS matrix $\mathbf{P}$ and the corresponding $L \times 1$ training x-coordinates $\mathbf{x}$ and $L \times 1$ training y-coordinates $\mathbf{y}$ are defined as follows

$$\begin{aligned} \mathbf{P} &= [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_L]^T, \\ \mathbf{x} &= [x_1, x_2, \ldots, x_L]^T, \\ \mathbf{y} &= [y_1, y_2, \ldots, y_L]^T \end{aligned} \tag{2.6}$$

where each row in the training matrix $\mathbf{P}$ indicates the training RSS vector $\mathbf{p}_l$ related to the training x-coordinates x_l and the the training y-coordinates y_l, $\forall l = 1, \ldots, L$.

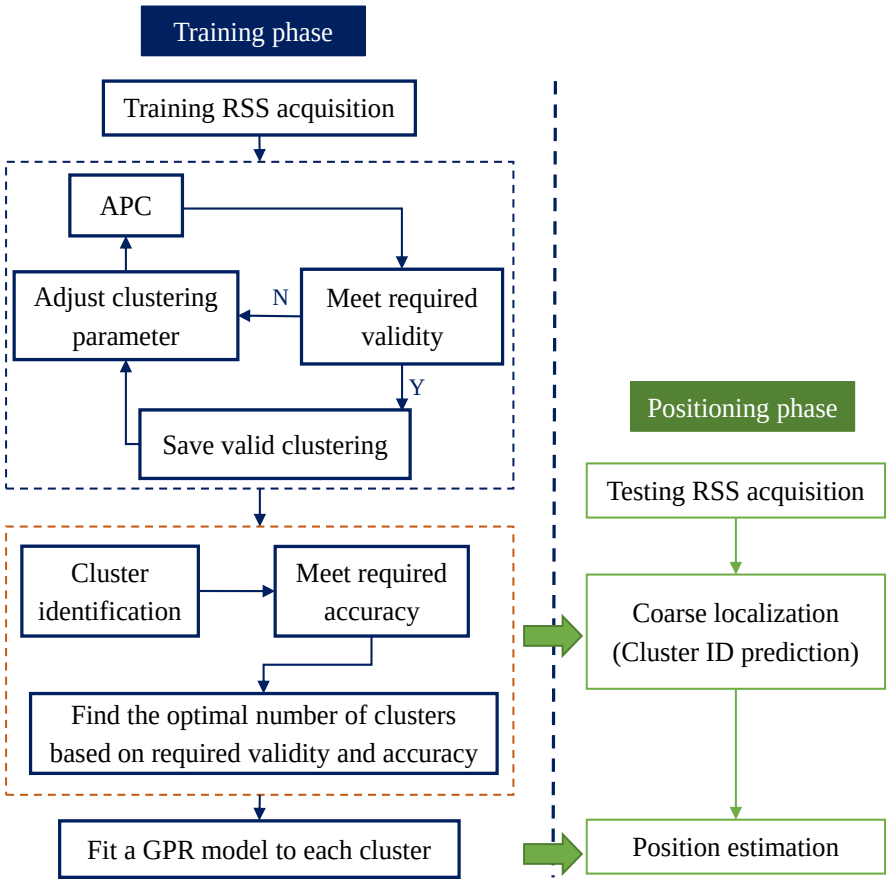


Figure 2.2: Overview of the proposed position estimation scheme.

RSS clustering using the APC and cluster validation

From among several clustering methods, we choose APC for our study because (i) it starts by allocating the same chance to each data point to become an exemplar, while k-means (KM) clustering begins by selecting a random set of initial exemplars [57], (ii) contrary to the KM algorithm, it learns the number of clusters so there is no need to set the number of clusters in advance, and (iii) it is stable and deterministic over many runs [58].

The APC divides the training matrix $\mathbf{P}$ into clusters by considering all $\mathbf{p}_l$, $l = 1, 2, \ldots, L$ as potential cluster exemplars and then exchanges two kinds of messages, named responsibility and availability, between them until a convergence is achieved [48]. When the exemplar of each cluster is selected, the clusters are formed by allocating each $\mathbf{p}_l$, $l = 1, 2, \ldots, L$ to its most similar exemplar. APC has two kinds of real-valued input: (i) the similarities matrix $\mathbf{S}_{Sim}$; (ii) the preference $pref$. APC uses a pairwise similarity $s(\mathbf{p}_l, \mathbf{p}_{l''})$ (for $l \neq l''$) to represent the fitness of $\mathbf{p}_l$ to be selected as the exemplar with respect to $\mathbf{p}_{l''}$. Since, in this work, we aim to minimize the squared error, the similarity calculation in the APC process is based on negative squared error (Euclidean distance) and is defined as

$$s(\mathbf{p}_l, \mathbf{p}_{l''}) = -||\mathbf{p}_l - \mathbf{p}_{l''}||^2 \tag{2.7}$$

where $1 \leq l, l'' \leq L$ and $l \neq l''$.

In the APC, the number of clusters is affected by the value of $pref$, i.e., a high value of $pref$ forms many exemplars (clusters), while a low value leads to a small number of exemplars (clusters). A good initial selection for the preference value is the median of the similarities matrix $\mathbf{S}_{Sim}$ which is defined as

$$pref = \text{median}(s(\mathbf{p}_l, \mathbf{p}_{l''})) \tag{2.8}$$

where $1 \leq l, l'' \leq L$ and $l \neq l''$. To get the desired number of valid clusters, a set of $pref$ values is needed to be set. Therefore, we increase the value of the initial preference to create a set of preference values.

The training RSS clustering and cluster validation procedures are presented in detail in Algorithm 2.1.

First, choose a parameter (such as the preference value) from the clustering parameters **prm** (step 1). Use selected parameter prm_j in the APC algorithm to split the training matrix $\mathbf{P}$ into sub-matrices as clusters. Therefore, $\mathbf{P}$ is converted to T clusters and a set of clusters $\mathcal{P}_{prm_j} = \{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_T\}$ and corresponding cluster ID set $\mathcal{C}_{prm_j} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_T\}$ are created (step 2). Assign a cluster ID to each training RSS $\mathbf{p}_l$. Therefore, each cluster can be denoted as $\mathbf{S}_{prmj,t} = [\mathbf{P}_t, \mathbf{c}_t]$ and each training RSS vector $\mathbf{p}_l$ is specified by a cluster ID such that $\mathcal{S}_{prm_j} = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_L\}$, where $\mathbf{s}_l = [\mathbf{p}_l, c_t]$, $c_t \in \mathbf{c}_t$ (step 3). Then calculate the cluster validity index such as the silhouette (SI) [59], the Davies-Bouldin (DB) [60], and the Calinski-Harabasz (CH) [61] index, which are explained in Section 2.4.1, to validate the quality of clustering (step 4). If the quality of clustering is not sufficient, ignore the created sets in steps 2 and 3, and repeat the clustering with the next parameter prm_{j+1} (steps 5-6); otherwise, save all sets and clusters obtained from clustering with the prm_j parameter in steps 2-3 (step 8) and put the cm_{prm_j} value in the **cm** list in descending order (step 9). Then, for the cluster identification which is explained in the next section, use the validation set approach [62] to randomly select a percentage of observations in each cluster and put them in set $\mathcal{V}$ as a validation set and put the remaining observations in $\mathcal{TR}$ as the training set. Consequently, the validation set, $\mathcal{V}$, has N elements and a the training set $\mathcal{TR}$ has $L - N$ elements. Note that the cluster ID is assumed unknown in the validation set (steps 10-12). Put the cluster ID of all observation of set $\mathcal{V}$ in **realID** list (step 13).

Algorithm 2.1 RSS clustering and cluster validation

Require:
Similarity matrix $\mathbf{S}_{Sim}$
Clustering parameters $\mathbf{prm} = [prm_1, prm_2, \ldots, prm_J]$
The training matrix $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_L]^T$

1: **for** $j = 1 \rightarrow J$ **do**
2: Apply APC algorithm to $\mathbf{P}$ using prm_j
3: Assign a cluster ID to each training RSS $\mathbf{p}_l$
4: cm_{prm_j} = Compute cluster validity index
5: **if** $cm_{prm_j} \leq threshold$ **then**
6: Clustering with prm_j is not valid and ignore all sets created in steps 2-3
7: **else**
8: Save all the sets created in steps 2-3
9: Put cm_{prm_j} in **cm** list in descending order
10: **for** $t = 1 \rightarrow T$ **do**
11: Select a percentage of observations in $\mathbf{S}_{prm_{j,t}}$ and put them in $\mathcal{V}_{prm_j}$ set and the remaining in $\mathcal{TR}_{prm_j}$ set
12: **end for**
13: Put clusterID of observation in V_{prm_j} in **realID**
14: **end if**
15: **end for**

Cluster identification based on nearest neighbor

In this section, we focus on finding the cluster identification of the new data point based on the nearest neighbors. For this purpose, KD-tree [63] is employed to find the nearest neighbors. Then the accuracy of cluster identification is estimated. Based on this, we can manage the trade-off between the accuracy of cluster identification and the number of clusters. The process of cluster identification is described in detail in Algorithm 2.2.

First, choose the clustering which has the maximum quality and an *element* from the validation set $\mathcal{V}$ (steps 1-2). Find K_{nn} nearest neighbors of *element* from all the observations in each cluster using KD-tree and calculate the RSS distance between them (steps 4-5). Select the minimum distance and assign the cluster ID of that RSS vector to *element* as the estimated cluster ID and put it into the **estimatedID** vector (step 7). Calculate the classification identification accuracy by comparing the **realID** and the **estimatedID** vectors (step 9). If the accuracy of the cluster identification is not sufficient, ignore the clustering with cm_{prm_j}; otherwise, put the cfm_{prm_j} value in the **cfm** vector (steps 11-14). Select the item which has the highest quality of clustering and highest cluster identification accuracy as the optimal number of clustering and make the GPR model for them (step 17).

Algorithm 2.2 Cluster identification based on nearest neighbor

for each cm_{prm_j} **in cm do**
  for each *element* **in** $\mathcal{V}_{prm_j}$ **do**
    for $t = 1 \rightarrow T$ **do**
      Find nearest neighbors of *element* from all observations in $\mathbf{S}_{prm_j,t}$ using KD-tree algorithm
      Calculate RSS distance from *element* to its nearest neighbors and put them in the **distance** vector
    end for
    Select minimum value of **distance** and find its cluster ID as estimated cluster ID and put it into **estimatedID** vector
  end for
  cfm = Calculate cluster identification accuracy by comparing **realID** and **estimatedID** vectors
  if $cfm \leq threshold$ **then**
    Ignore clustering with cm_{prm_j} selected
    Delete cm_{prm_j} from **cm**
  else
    Put cfm_{prm_j} in **cfm** vector
  end if
end for
Select clustering with highest cm_{prm_j} and highest cfm_{prm_j} as optimal clustering

Making the GPR model

Until now, an optimal number of clusters is determined. We aim to make a regression model for each cluster based on their RSS vectors in matrix $\mathbf{P}_t, t = 1, \ldots, T$ and their known locations $[\mathbf{x}_t, \mathbf{y}_t]$. Then, the trained regression model gets the RSS vector of a test user as input and gives an estimate of the test user's location as output. For simplicity, we focus on the RSS vectors of a cluster, but the same procedure can be applied for other clusters as well. Also, we consider the t^{th} cluster has L' training RSS samples.

In this case, $f_x(.)$ and $f_y(.)$ are defined as the functions which map the RSS vector $\mathbf{p}_k$ of any user k in cluster $\mathbf{P}_t$ to its 2-dimensional location coordinates (x_k, y_k), such that

$$\begin{aligned} x_k &= f_x(\mathbf{p}_k) + v_x, \\ y_k &= f_y(\mathbf{p}_k) + v_y \end{aligned} \tag{2.9}$$

where the noise terms, v_x and v_y, are i.i.d. Gaussian random variables with zero mean and variance $\sigma_{v_x}^2$ and $\sigma_{v_y}^2$ respectively. From (2.9), we can see that, the x-coordinate x_k (and y-coordinate y_k) for any user k is a non-linear function of its uplink RSS vector $\mathbf{p}_k$ with the additional presence of shadowing noise z_{mk}. Therefore, estimating the x-coordinate x_k (and y-coordinate y_k) from $\mathbf{p}_k$ is a non-linear regression problem in machine learning. There are several non-linear regression methods. Among them, we focus on GPR as our method, be-

cause (i) GP methods are non-parametric [64], (ii) the performance of GP methods is as good as the other machine learning algorithm in term of multiple performance metrics, including the squared prediction error [65], and (iii) include deep learning methods. The following presents the detail of GPR model on the training. For simplicity, details of the GPR model are presented only for the x-coordinates of the users, but the same procedure is employed for the y-coordinates as well.

In (2.9), the function $f_x(.)$ is supposed to follow a Gaussian process with zero-mean and covariance matrix $\mathbf{Q}_t$, whose elements are a user-defined function $q_t(.)$, such that

$$f_x(.) \sim \mathcal{GP}\big(0, q_t(.)\big) \tag{2.10}$$

We choose the covariance function $q_t(.)$ between user k and user k' in cluster $\mathbf{P}_t$, as a weighted sum of the squared-exponential, the linear and the delta function [66] which is defined as

$$q_t(\mathbf{p}_k, \mathbf{p}_{k'}) = \kappa e^{-\frac{1}{2\eta^2}\|\mathbf{p}_k - \mathbf{p}_{k'}\|^2} + \beta \mathbf{p}_k{}^T \mathbf{p}_{k'} + \sigma^2_{v_x} \delta_{kk'} \tag{2.11}$$

where the delta function $\delta_{kk'}$ is one if $k = k'$ and zero otherwise.

From (2.11), we have an unknown GPR parameter or hyperparameter vector $\boldsymbol{\theta}_t = [\kappa, \eta, \beta]$ that needs to be optimized. According to the GP assumption, we know that $\mathbf{x}_t$ is Gaussian distributed, $\mathbf{x}_t|\mathbf{P}_t, \boldsymbol{\theta}_t \sim \mathcal{N}\big(0, \mathbf{Q}_t\big)$ [51]. The log-likelihood function is used to derive the maximum likelihood estimator of parameter $\boldsymbol{\theta}_t$. The estimator $\hat{\boldsymbol{\theta}}_t$ is obtained by solving

$$\hat{\boldsymbol{\theta}}_t = \arg\max_{\boldsymbol{\theta}_t} \log(p(\mathbf{x}_t|\mathbf{P}_t, \boldsymbol{\theta}_t)) \tag{2.12}$$

The optimization problem in (2.12) can be solved using a limited memory BFGS-B algorithm [67].

2.3.2 Positioning Phase

In this phase, suppose there are $\hat{L}$ test users whose location coordinates are unknown and needs to be estimated. Therefore, there is an $\hat{L} \times M$ test user RSS matrix $\hat{\mathbf{P}}$ which is used to estimate the $\hat{L} \times 1$ x-coordinates vector $\hat{\mathbf{x}}$ which are unknown. For each test user RSS $\hat{\mathbf{p}}$, the process of location estimation is described below.

- **Step 1: Cluster Identification**
  The cluster ID c_t of test user RSS $\hat{\mathbf{p}}$ is estimated by using the cluster identification based on nearest neighbor Algorithm 2.

- **Step 2: Position estimation**
  The GPR model related to cluster c_t is used for estimating the x-coordinate $\hat{x}$ of test

user RSS $\hat{\mathbf{p}}$.

Now we would like to determine the posterior density of the position, i.e. $\hat{x}|\mathbf{x}_t, \mathbf{P}_t, \hat{\mathbf{p}}$. According to [68], this distribution is Gaussian and the best estimate for $\hat{x}$ is the mean of this distribution, which is defined as

$$\hat{\mu}_x = \mathbf{q}_t^T [\mathbf{Q}_t + \sigma_{v_x}^2 \mathbf{I}_{L'}]^{-1} \mathbf{x}_t \tag{2.13}$$

where $\mathbf{q}_t = (q_t(\hat{\mathbf{p}}, \mathbf{p}_1)), \ldots, q_t(\hat{\mathbf{p}}, \mathbf{p}_u))$ and u is the number of training RSS vectors in the cluster t. Also, the uncertainty in our estimate is given by its variance which is given as

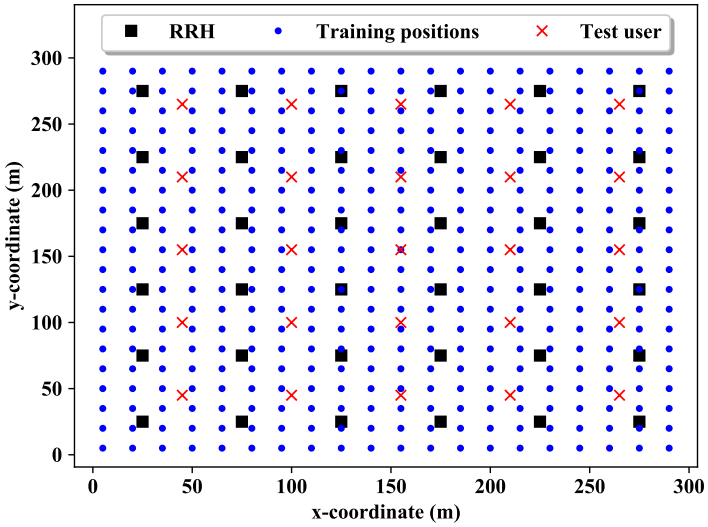
$$\hat{\sigma}_x^2 = \sigma_{v_x}^2 - q_t(\hat{\mathbf{p}}, \hat{\mathbf{p}}) - \mathbf{q}_t^T [\mathbf{Q}_t + \sigma_{v_x}^2 \mathbf{I}_{L'}]^{-1} \mathbf{q}_t \tag{2.14}$$

2.4 Results and Discussion

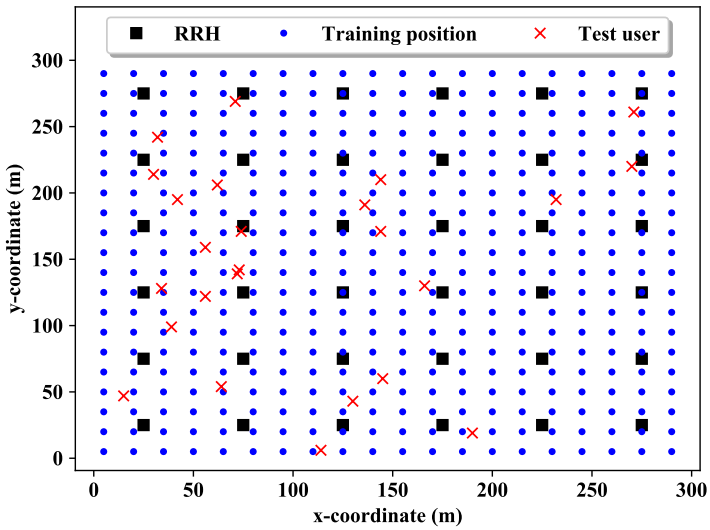
In this section, we compare the position estimation performance of using GPR and the proposed method (APC-GPR) in an DM-MIMO system. Therefore, GPR and APC-GPR positioning methods are simulated in order to be evaluated. A DM-MIMO system setup, with $M = 36$ single antenna RRHs, $L = 400$ training locations, and $\hat{L} = 25$ test users is considered. Training users are distributed every 15 m in a grid configuration over the whole area of 300 m $\times$ 300 m. Test users are also distributed in a grid and a random configurations, as shown in Fig. 2.3. For training, the RSS matrix $\mathbf{P}$ is generated using (2.4) with user power transmit $\rho = 21$ dBm, reference path loss $b_0 = -47.5$ dB and different shadowing noise variance $\sigma_z^2 = 1, 3, 5, 7$ dB. Also, we set the path loss exponent to $\alpha = 0$ for $0 < d_{mk} < 10$ m, $\alpha = 2$ for 10 m $< d_{mk} <$ 50 m, and $\alpha = 6.7$ for 50 m $< d_{mk}$, according to the 3GPP Urban Micro propagation model [69].

2.4.1 Clustering and Cluster Validation Results

As mentioned in the training phase, the APC algorithm is used to cluster the training RSS matrix $\mathbf{P}$. Before applying APC, according to (2.8), we select the median value of the similarity matrix in (2.7) as the initial preference value. Then by increasing this value, we generate a range of preference values because changes in the preference value can result in quite different clustering results. Therefore, APC takes these values as an input to cluster the training RSS matrix $\mathbf{P}$. For evaluating the performance of the proposed method, first, the valid number of clusters should be specified. It should be noted that the clustering results would affect the cluster identification accuracy, the computational complexity, and further affect the position estimation accuracy, as shown later in this section. For this purpose, first, the valid



(a) Scenario A: Grid configuration test user positions.



(b) Scenario B: Random configuration test user positions.

Figure 2.3: Simulation setup with $M = 36$ single antenna RHHs, $L = 400$ training positions, and $\hat{L} = 25$ test users,

clusterings, which has high quality, are identified. To evaluate the quality of clustering results, clustering index validity such as the silhouette (SI) [59], the Davies-Bouldin (DB) [60], and the Calinski-Harabasz (CH) [61] index are considered [70]. Then, the results of validity indexes are averaged over 100 Monte-Carlo runs.

The SI is a well-known measure of how similar a training RSS vector $\mathbf{p}_l, l = 1, \ldots, L$ is to its own cluster (cohesion) compared to other clusters (separation). It is then averaged over all training RSS vectors and is defined [59] as

$$SI = \frac{1}{L}\sum_{l=1}^{L}\frac{d(\mathbf{p}_l) - f(\mathbf{p}_l)}{\max\{d(\mathbf{p}_l), f(\mathbf{p}_l)\}} \tag{2.15}$$

where $d(\mathbf{p}_l)$ is the average distance between $\mathbf{p}_l$ and all training RSS vectors in other clusters and $f(\mathbf{p}_l)$ is the average distance between $\mathbf{p}_l$ and all training RSS vectors in the same cluster. In Fig. 2.4a, the red line demonstrates the relation between the preference value and the number of created clusters. We can see that changing the preference value of the APC algorithm changes the clustering results and increasing the preference values almost increases the number of clusters. i.e., the number of clusters is almost monotonically increasing with preference values. Also, the blue line represents the relation between the number of clusters and the quality of clustering based on the SI value. It should be noted that the value of SI is in the range of 0 to 1. A larger value of SI represents higher clustering quality. Each cluster can be separated significantly when $SI > 0.5$. We have an unsuitable cluster structure when $SI < 0.2$. Among clustering results, the 4 highest silhouette values correspond to the preference values which create 17, 19, 20 and 25 clusters.

Fig. 2.4b shows the validity of clustering using the DB index, which is the ratio of within-cluster distances to between-cluster distances and is defined [60] as

$$DB = \frac{1}{T}\sum_{t=1}^{T}\max_{t'=1,\ldots,T, t\neq t'}\frac{\mathrm{diam}(c_t) + \mathrm{diam}(c_{t'})}{d(\mathbf{p}_{zt}, \mathbf{p}_{zt'})} \tag{2.16}$$

where in this case, $d(.,.)$ is the Euclidean distance and the diameter of a cluster is defined as

$$\mathrm{diam}(c_t) = \sqrt{\frac{1}{u_t}\sum_{\mathbf{p}\in c_t} d(\mathbf{p}, \mathbf{p}_{zt})^2} \tag{2.17}$$

with u_t the number of RSS vectors and $\mathbf{p}_{zt}$ the centroid of cluster c_t. Since finding the clusters with minimum intra-cluster distances is the objective in DB, small values for DB are taken to be suitable. It is revealed that the lower value of the DB index belongs to the cases with 17, 19, 20 and 25 clusters.

The CH index is defined [61] as

$$CH = \frac{\sum_{t=1}^{T} u_t.d(\mathbf{p}_{zt}, \mathbf{p}_{ztot})^2}{T-1}\frac{L-T}{\sum_{t=1}^{T}\sum_{\mathbf{p}\in c_t} d(\mathbf{p}, \mathbf{p}_{zt})} \tag{2.18}$$

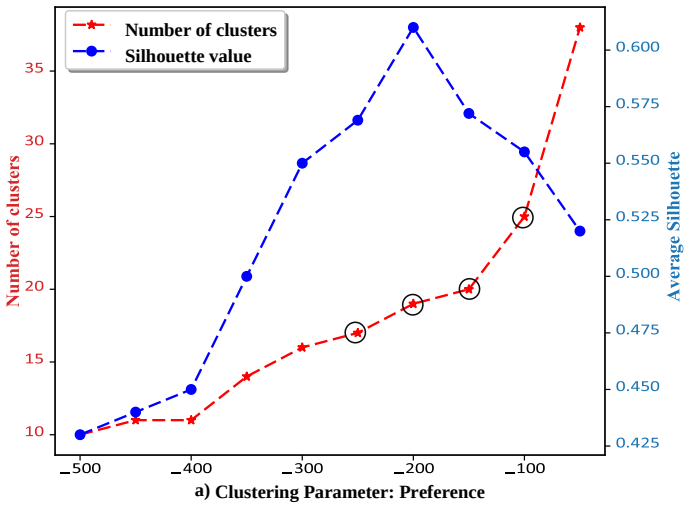
where $\mathbf{p}_{ztot}$ is the centroid of all training RSS vectors. Since the goal is to obtain well-separated and compact clusters, the maximum value for CH presents a proper separation for the training RSS vectors. Fig. 2.4c indicates that the 4 highest values for the CH index relates to the preference values with 17, 19, 20 and 25 clusters. By considering the results of

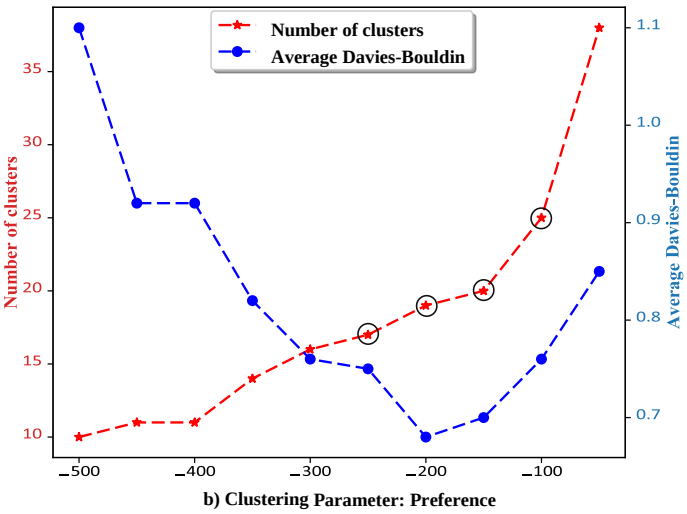
validity indexes, we see that when there are $17, 19, 20$ and 25 clusters, the training matrix is partitioned well. Therefore, these partitioning are chosen as valid clustering.

2.4.2 Accuracy of Cluster Identification

According to the results in Section 2.4.1, the quality of clustering is sufficient when the number of created clusters are $17, 19, 20$ and 25. Therefore, in this part, cluster identification Algorithm 2.2 is applied in these valid cases to allocate validation data $\mathcal{V}$ to each cluster. We consider 20% of all training data as validation data which is equal to 80 samples. To find the nearest neighbor as a part of the cluster identification Algorithm 2.2, the KD-tree algorithm with different values of K_{nn} is considered and evaluated. Then, the accuracy of cluster identification is measured based on the error of cluster membership, which is done by comparing the predicted and the real cluster ID of the validation RSS samples. Since the accuracy of cluster identification is an important part of our method, a threshold is considered for the accuracy of the cluster identification algorithm. If the accuracy of cluster identification in each valid clustering case is less than the defined threshold, that clustering is ignored. Otherwise, its validity check and the number of clusters of the clustering, which has the highest validity and sufficient accuracy requirements for cluster identification, is selected as the optimal number of clusters. In our work, we consider 95% accuracy as threshold.

Fig. 2.5 shows the the cluster identification accuracy of Algorithm 2.2. It can be seen that there is a relation between the number of clusters and cluster identification accuracy. We can see that by increasing the number of clusters, i.e., 25 clusters, the cluster identification accuracy decreases. In contrast, a lower number of clusters, i.e., 17 and 19 clusters, increases the cluster identification accuracy. However, if there is only one cluster, the cluster identification accuracy is 100 percent, but we will have poor accuracy of position estimation. But, if there are many clusters, identifying the cluster will be difficult, and the clustering method will finally become incapable for position estimation. Therefore, for accurate position estimation, an optimal number of clusters is a critical factor. Also, as shown in Fig. 2.5, there is a relationship between the accuracy of cluster identification and the number of nearest neighbors K_{nn}. Therefore, we need to choose the best number of nearest neighbors. Because a lower value of K_{nn} gives more weight to data, and the model learns to predict more locally, while a higher value of K_{nn} neglects outliers to the data, and the model learns to predict more globally. If the value of K_{nn} is too high, the algorithm will not be able to classify the data. Therefore K_{nn} needs to be relatively small. Therefore, we set $K_{nn} = 1, 3$, and 5. We can see that when K_{nn} is equal to 3, the accuracy is maximum in all cases, and when K_{nn} is equal to 1 and 5, the accuracy is under the threshold. Therefore, we select the KD-tree with $K_{nn} = 3$ nearest neighbors. Also, it is observed that when the number of clusters is 17 and 19, the accuracy is almost the same and is the maximum. However, the validity of 19 clusters





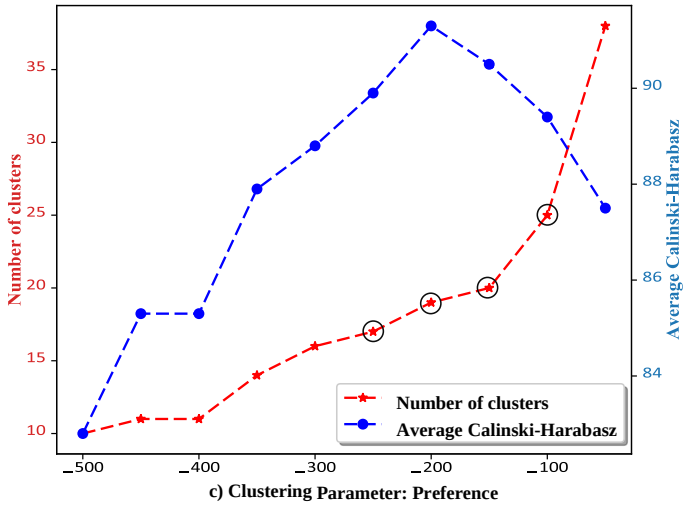


Figure 2.4: Cluster validation results using cluster validity indexes; a) Silhouette index, b) Davies-Bouldin index, c) Calinski-Harabasz index.

is better than clustering with 17 clusters. Therefore, 19 is selected as the optimal number of clusters, which has 96.25 percent correct cluster-ID estimation accuracy. Therefore, we can conclude that the optimal number of clusters is equal to 19.

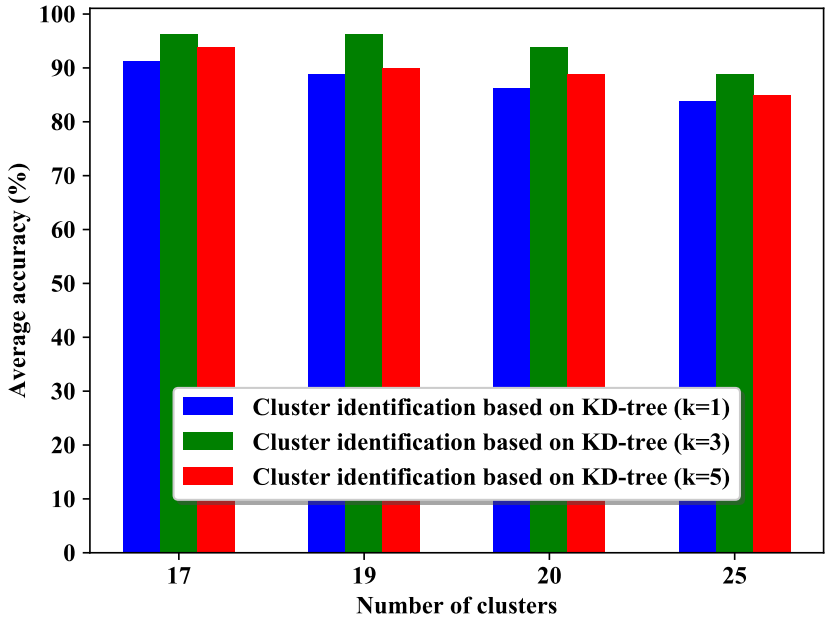


Figure 2.5: Average accuracy of cluster identification when the number of clusters is 17, 19, 20, and 25.

2.4.3 RMSE Performance

To evaluate the accuracy of the positioning of the proposed method, we measure the position estimation performance in terms of the root mean squared error (RMSE) between the real coordinates (x_l, y_l) of the test users and their estimates $(\hat{x}_l, \hat{y}_l)$ as follows:

$$RMSE = \sqrt{\frac{\sum_{l=1}^{\hat{L}} (x_l - \hat{x}_l)^2 + (y_l - \hat{y}_l)^2}{\hat{L}}} \tag{2.19}$$

The simulation results of the APC-GPR positioning method are compared with the method which only uses GPR [14] for positioning in DM-MIMO system. The RMSE results are averaged over 100 Monte-Carlo runs. Several other investigations are also considered to justify the impacts of the variations of important parameters in position estimation performance, such as the effects of the number of single antenna RRHs, the effects of the number of training samples, and the shadowing noise variance during the training phase.

Impact of the number of RRHs

The average RMSE of the test user positions is evaluated as a function of the number of RRHs. We plot the average RMSE performance of the proposed method for $M = 25, 36, 49, 64, 81$ and a shadowing noise variance of 5 dB. Fig. 2.6 shows that the average RMSE decreases by increasing the number of RRHs. It can be also observed that increasing the number of BS antennas results in performance improvement in positioning, which in turn motivates the DM-MIMO usage. Although a similar decrease in average RMSE is observed in the GPR method [14], by increasing the number of RRHs, the APC-GPR in the current study has a significantly lower average RMSE compared to using only GPR in both grid and random configuration. Also, we can see that we have better performance in scenario A compared to scenario B. We can claim that there are two significant reasons that RMSE decreases with increasing M: (i) there are extra RSS information that becomes available to the GPR models because of the addition of the new RRHs, (ii) the increased average signal strength is a result of the reduced RRH-to-MT distance.

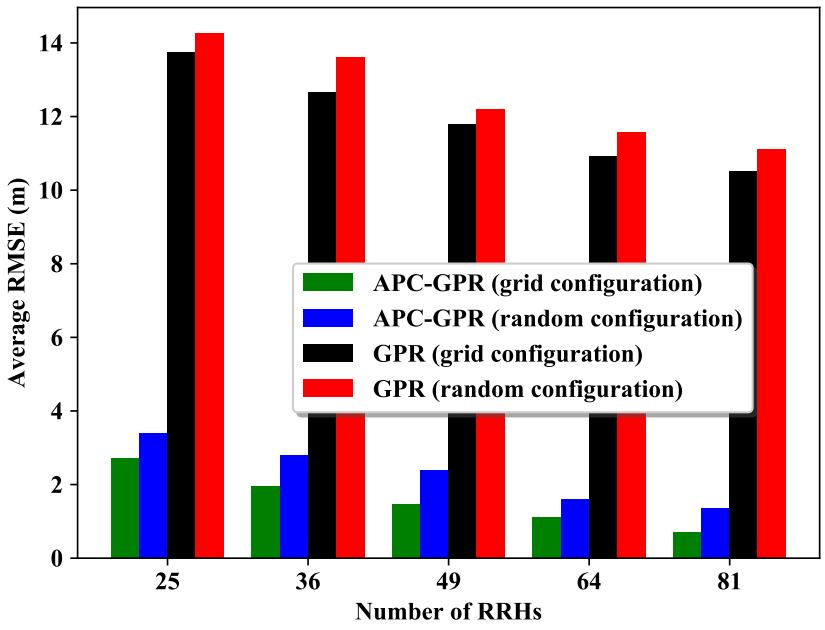


Figure 2.6: Average RMSE of the GPR and the APC-GPR methods for different numbers of RRHs $M = 25, 36, 49, 64, 81$, when the shadowing noise variance is 5 dB and $L = 400$.

Impact of the number of training samples

Fig. 2.7 shows the average RMSE of positioning as a function of the number of training samples. It can be seen that the average RMSE decreases when the number of training samples is

increased because there is more available RSS information in the whole environment, while the lack of training samples leads to loss of position information.

It is also seen that the average RMSE in scenario A is better than scenario B. In both scenarios, the APC-GPR method has a better performance than the GPR method.

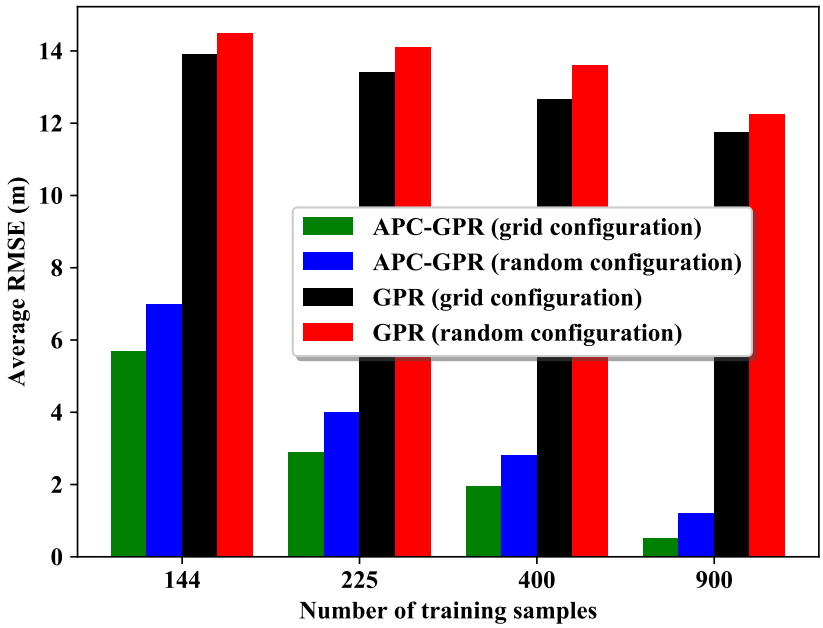


Figure 2.7: Average RMSE of the GPR and the APC-GPR methods for different numbers of training samples $L = 225, 400, 900$, when the shadowing noise variance is 5 dB and $M = 36$.

Impact of shadowing noise variance

Fig. 2.8 shows the average RMSE performance of the APC-GPR method in the current study and the GPR method as a function of shadowing noise variance [14] when $M = 36$ for both scenarios, i.e., A and B. The RMSE performance is averaged over 100 Monte-Carlo simulations for shadowing noise ranging from 1 dB to 7 dB. As seen in both methods, the average RMSE increases by increasing the shadowing noise variance. However, the former has lower average RMSE compared to the latter and scenario A has lower average RMSE compared to scenario B.

Impact of channel correlation

Until now, it has been supposed that the shadowing noise coefficients of different uplink channels are mutually uncorrelated. Since the RRH antennas that are close to each other may share a common set of obstacles, the above assumption may not always hold in practice.

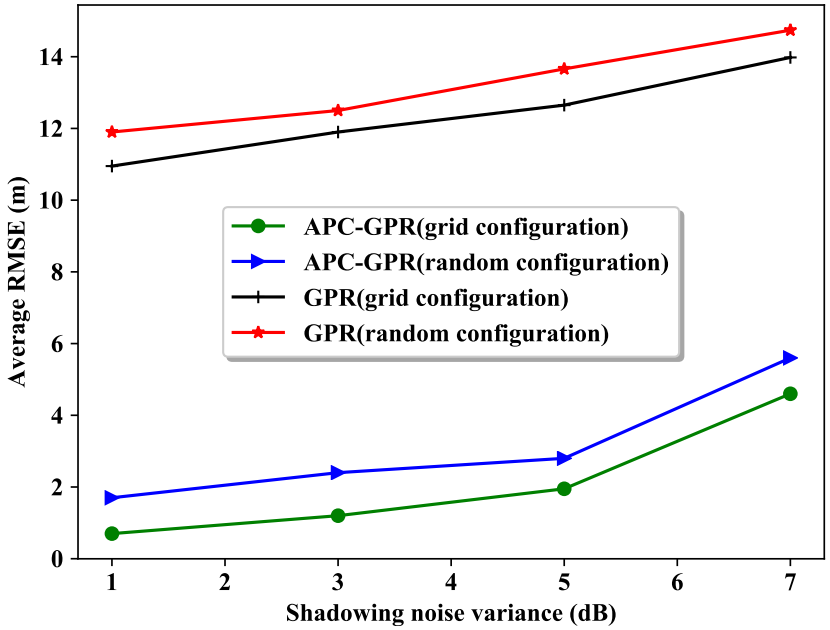


Figure 2.8: Average RMSE performance of the GPR and the APC-GPR methods for different shadowing noise levels, when $M = 36$ and $L = 400$.

Therefore, in this part, we discuss how the correlated shadowing among the test RSS vectors affects the localization performance. The GP method employs the function $q_t(.)$ to map the correlation among inputs to the correlation among outputs. When there is a notable correlation among the inputs, the GP methods would require to make a smaller number of independent predictions on the test datasets. That means, when the GPR model has an accurate prediction on a test RSS vector, it tends to make accurate predictions on other test vectors that have a high correlation with the considered test RSS vector. This leads to an improvement in the RMSE performance.

2.4.4 Computational Complexity

Computational complexity is one of the problems found in many positioning methods. When we use GPR for positioning, it has a high computational complexity according to (2.13) and (2.14). Therefore, it is not efficient to use only GPR for positioning as in [14]. To overcome this problem, we proposed an efficient positioning algorithm based on the APC and GPR to decrease the GPR computational complexity in the positioning phase. The proposed method uses APC to divide all the training data points in the target area into separate sets based on the correlation of locations. In each cluster, the GPR is used to estimate new position. When the GPR model is made, the training data that belongs to the same cluster is considered. When all L training data points in the target area are considered in GPR, the complexity is $\mathcal{O}(L^3)$ [51]. But in the proposed method, the L training data points are divided into T clus-

ters. When the number of training data in a cluster is L', (3.34) and (3.35) are simplified in the proposed method by only considering L' training data points. Therefore, the complexity is $\mathcal{O}(L'^3)$ for a cluster, and the total complexity is $T.\mathcal{O}(L'^3)$ for all T clusters. In general, $T.\mathcal{O}(L'^3) \ll \mathcal{O}(L^3)$ because $L' \ll L$. So, the proposed method can efficiently decrease the computational complexity of the GPR predictive model.

2.5 Conclusion

We proposed a cluster-based approach to estimate user's position from their uplink RSS data in a DM-MIMO system. In the proposed method of the current study, the whole testbed was first divided into clusters using an optimal clustering algorithm based on the APC algorithm, which reduces the computational cost of online positioning. In the proposed method, APC was chosen for clustering due to its initialization-independent property and better selection of cluster heads compared with K-means clustering. KD-tree was used for cluster identification to allow for a quick finding of the related cluster. Also, GPR was used for further location estimation within each cluster.

To evaluate the proposed method, which is based on clustering and GPR, we compared it with the method which only uses GPR to estimate the position. Through simulation results, we presented several new insights on RSS-based user positioning in DM-MIMO system: (i) the average RMSE values of the proposed method decrease with increasing the number of RRHs, (ii) the average RMSE values of the proposed method decrease with increasing the number of training samples, (iii) clustering the target area into several clusters helps to minimize the computational complexity of online positioning, and (iv) the performance of the proposed method is better than using only GPR for position estimation.

Chapter 3

An Improved and Low-dimensional Fingerprint-based Localization Method in Collocated Massive MIMO-OFDM Systems

Résumé

La localisation a attiré une attention significative dans la 5G en raison de la demande croissante de service basé sur la localisation (LBS). Le MIMO massif (Massive Multiple-Input Multiple-Output) a été introduit dans la 5G en tant que technologie puissante en raison de son potentiel évident pour l'amélioration des performances de communication et la localisation dans des environnements complexes. La localisation basée sur les empreintes digitales (FP) est une méthode prometteuse pour les environnements de diffusion riches grâce à sa fiabilité et sa précision élevées. La méthode de régression de processus gaussien (GPR) pourrait être utilisée comme méthode de localisation basée sur le FP pour faciliter la localisation et fournir une grande précision. Cependant, cette méthode présente une complexité de calcul élevée, en particulier dans les environnements à grande échelle. Dans cette étude, nous proposons une méthode de localisation améliorée et de faible dimension basée sur la FP dans un systéme basé sur le multiplexage par division de fréquences orthogonales et le MIMO massif colocalisé utilisant l'analyse en composantes principales (PCA), l'algorithme de clustering de propagation d'affinité (APC) et la régression de processus gaussien (GPR) pour estimer l'emplacement de l'utilisateur. Les empreintes digitales sont d'abord extraites sur la base des informations d'état instantané du canal (CSI) en tirant pleinement parti des domaines d'angle et de retard à haute résolution. Tout d'abord, le PCA est utilisé pour prétraiter les données et réduire la dimension de l'entité. Ensuite, les empreintes digitales

d'entraînement sont regroupées à l'aide de l'algorithme APC pour augmenter la précision de la prédiction et réduire la complexité du calcul. Enfin, la distribution des données de chaque "cluster" est modélisée avec précision à l'aide d'un GPR pour fournir un support pour une localisation ultérieure. Les résultats de la simulation révèlent que la méthode proposée améliore considérablement les performances de localisation en réduisant l'erreur d'estimation de l'emplacement. En outre, cela réduit la complexité de la correspondance et la complexité de calcul.

Abstract

Localization has drawn significant attention in 5G due to the fast-growing demand for location-based service (LBS). Massive multiple-input multiple-output (M-MIMO) has been introduced in 5G as a powerful technology due to its evident potentials for communication performance enhancement and localization in complex environments. Fingerprint-based (FP) localization are promising methods for rich scattering environments thanks to their high reliability and accuracy. The Gaussian process regression (GPR) method could be used as an FP-based localization method to facilitate localization and provide high accuracy. However, this method has high computational complexity, especially in large-scale environments. In this study, we propose an improved and low-dimensional FP-based localization method in collocated massive MIMO orthogonal frequency division multiplexing (OFDM) systems using principal component analysis (PCA), the affinity propagation clustering (APC) algorithm, and Gaussian process regression (GPR) to estimate the user's location. Fingerprints are first extracted based on instantaneous channel state information (CSI) by taking full advantage of the high-resolution angle and delay domains. First, PCA is used to pre-process data and reduce the feature dimension. Then, the training fingerprints are clustered using the APC algorithm to increase prediction accuracy and reduce computation complexity. Finally, each cluster's data distribution is accurately modelled using GPR to provide support for further localization. Simulation results reveal that the proposed method improves localization performance significantly by reducing the location estimation error. Additionally, it reduces the matching complexity and computational complexity.

3.1 Introduction

For supporting users with high quality of service, recognizing their location is essential. Therefore, location information has recently become an important characteristic to drive location and context-aware services in wireless communications [71]. Nevertheless, it is a notoriously challenging problem to provide precise and reliable location information of user

equipments (UEs) by using multipath propagation in wireless communications [72].

Currently, with the advent of the fifth-generation (5G) of wireless communications, multiple-input multiple-output (MIMO) orthogonal frequency division multiplexing (OFDM) systems have received great attention from the localization community due to their potential for improving user-localization accuracy [73]. Indeed, by employing a very large number of antennas at the base station, massive MIMO-OFDM, besides improving spectral and energy efficiency [74], can obtain a higher multipath resolution in the angle and delay domains to provide high-accuracy localization for location-based services [73]. Furthermore, the resilience against small scale fading is provided through processing measurements across the massive array [47]. Therefore, the potential of massive MIMO-OFDM to support location-based services is one of the main economic drivers of 5G wireless communications [75].

Global positioning system (GPS) is the most well-known localization technique. Although it provides a precise estimation of mobile terminal's (MT's) position, it suffers a loss of accuracy in areas where there is no direct line of sight (LoS) between transmitter and receiver [76]. Consequently, there are range-based localization methods [7–10], which are based on radio signal information received from MTs such as angle-of-arrival (AoA), time-of-arrival (ToA), and received signal strength (RSS). However, the AoA-based methods deal with the non-line-of-sight (NLoS) error [9]; in ToA-based methods, the base stations (BSs) need to be synchronized [10] and using RSS has coarse range estimation error in complex environments [32].

Another approach is fingerprinting (FP), which has achieved extensive attention for localization in recent years due to its promising performance in complex multipath environments. It is flexible and can be used in many systems such as WiFi networks [35] and in systems where channel state information (CSI) [38] is used as the location fingerprint. Unlike traditional methods, it uses massive data to train a model and then it is employed for localization [73]. For this purpose, many machine learning (ML) and deep learning (DL) algorithms are used. However, the ML and DL algorithms often have to deal with data complexity due to the high dimensionality of the data. Some solutions such as using dimensionality algorithms have been proposed to reduce complexity [77]. By considering all benefits of the massive MIMO-OFDM system and the FP methods, integrating 5G and fingerprints can be a good solution for localization in rich scattering environments.

Localization with massive MIMO is still in its nascent stage. In [41], AoA is estimated precisely in massive MIMO systems employing very large uniform rectangular arrays (URAs). The authors in [45] consider the combined information of time delay, angle of delay (AoD), and AoA for localization of MTs in a massive MIMO system. In [44], a compressed sensing approach is proposed to determine MT's location directly from data acquired, such as ToA information at multiple massive MIMO BSs. The Gaussian process regression (GPR)

method is employed in [14] to estimate location of users using a vector of RSS, which is considered as a fingerprint in a DM-MIMO system. A fingerprinting method is presented in [78], wherein an angle-delay channel power matrix (ADCPM) is first extracted as a fingerprint. Then a system is employed for clustering the fingerprints with a mathematical model, such as the joint angle-delay similarity coefficient (JADSC). However, efficiency is lost because the effective range of the JADSC is dependent on the scatterers' density.

In this chapter, we consider a collocated massive MIMO-OFDM system where the BS is equipped with a large array of antennas to serve single antenna UEs over their coverage area. We propose a machine learning localization method using the CSI of collocated massive MIMO-OFDM system to achieve accurate localization resolution. For this purpose, the fingerprints are extracted from the known channel estimations. Then a combination of principal component analysis (PCA) and t-distributed stochastic neighbor embedding (t-SNE) is used to pre-process data, reduce the dimensions of the data features and visualize the data. Later, an affinity propagation clustering (APC) and Gaussian process regression (GPR)-based fingerprinting positioning system is presented, which consists of two phases: the training phase and the online positioning phase. In the first phase, a large area is divided into small clusters using an optimal clustering method based on the APC algorithm. The data distribution within each cluster is precisely modeled using the GPR method. Then, in the positioning phase, the user's cluster is first specified by a cluster identification algorithm based on a multi-layer perceptron (MLP) neural network. Its location is estimated using the GPR model of that cluster.

The remainder of this chapter is structured as follows. In Section 3.2, we present the system model of collocated massive MIMO-OFDM. In Section 3.3, we describe the proposed machine learning localization method in detail. The results obtained through simulations are shown in Section 3.4. Finally, conclusions are provided in Section 3.5.

3.2 System Description

3.2.1 Massive MIMO-OFDM Channel Model

In this section, we consider the uplink of a multi-user collocated massive MIMO-OFDM system. In this system, we aim to localize the K single-antenna UEs, which are randomly distributed in the coverage area. The UEs transmit signals to the BS, which is equipped with M antennas in the form of a uniform linear array (ULA), through $F \gg 1$ different scattering paths, as shown in Fig. 3.1.

According to Fig. 3.1, the wireless signals are propagated through multiple scattering paths with AoA $\theta_{f,k} \in (0, \pi)$, and the distance $d_{f,k}$ between the UE's antenna and the first

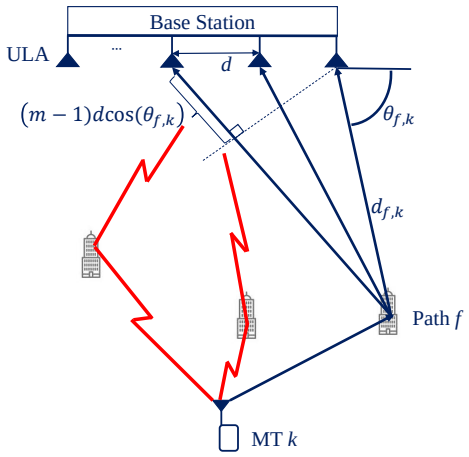


Figure 3.1: The wireless channel from an arbitrary user k to the BS.

receive antenna of the fth path. Assume that at this multipath wireless channel, the CSI is known at the BS through the uplink channel. The channel impulse response (CIR) vector associated with the fth path of the kth user is given by [78]

$$\mathbf{a}_{f,k} = \omega_{f,k}\mathbf{e}(\theta_{f,k}) \tag{3.1}$$

where $\omega_{f,k} \sim \mathcal{CN}(0, \sigma_{f,k})$ is the complex attenuation of the fth path and $\mathbf{e}(\theta) \in \mathbb{C}^{M\times 1}$ is the array response vector related to the AoA, θ, and is given [78] by

$$\mathbf{e}(\theta) = \left[1, e^{-j2\pi\frac{d\cos(\theta)}{\lambda}}, \ldots, e^{-j2\pi\frac{(M-1)d\cos(\theta)}{\lambda}}\right]^T \tag{3.2}$$

where λ and d represent the carrier wavelength and the space between two antennas, respectively. For OFDM systems, the channel frequency response (CFR) of the nth subcarrier can be defined as the summation of the time domain CIRs with different delays [78] as

$$\mathbf{h}_{k,n} = \sum_{f=1}^{F} \omega_{f,k}\mathbf{e}(\theta_{f,k})e^{-j2\pi\frac{nu_{f,k}}{N_{sc}}} \tag{3.3}$$

where $u_{f,k} = \lfloor\frac{\tau_{f,k}}{T_s}\rceil$ and $\frac{u_{f,k}}{N_{sc}}$ indicates the temporal propagation delay corresponding to the fth path. $\tau_{f,k} = \frac{d_{f,k}}{v}$ is the ToA of each path, v is the speed of light, T_s and N_{sc} are the sample interval and the number of subcarriers in the OFDM system, respectively. Then the overall CFR matrix is defined as

$$\mathbf{H}_k = [\mathbf{h}_{k,0}, \mathbf{h}_{k,1}, \ldots, \mathbf{h}_{k,N_{sc}-1}] \tag{3.4}$$

3.2.2 Fingerprint Extraction

For creating a fingerprint, it is required to extract several characteristics that are constant in the FP method. Therefore, wide-sense stationary features from the instantaneous CSI are

considered as a fingerprint. Discrete Fourier transform (DFT) operations are employed to establish a simple mapping from the CFR matrix to a sparse structure [78], and the angle-delay channel power matrix (ADCPM) is given as

$$\mathbf{\Omega}_k = \mathbf{F}^H \mathbf{H}_k \mathbf{G}^* \tag{3.5}$$

where $\mathbf{F} \in \mathbb{C}^{M \times M}$ indicates a phase-shifted discrete Fourier transform (DFT) matrix represented by $[\mathbf{F}]_{i,j} \triangleq \frac{1}{\sqrt{M}} e^{-j2\pi \frac{i(j-M/2)}{M}}$. $\mathbf{G} \in \mathbb{C}^{N_{sc} \times N_g}$ is defined as $[\mathbf{G}]_{i,j} \triangleq \frac{1}{\sqrt{N_{sc}}} e^{-j2\pi \frac{ij}{N_{sc}}}$ and N_g is the number of guard subcarriers. The left multiplication operator, $\mathbf{F}^H$, and the right multiplication operator, $\mathbf{G}^*$, cause the frequency domain CFR map to the angle and the delay domain, respectively. The complex gain related to the ith AoA and the jth ToA is represented by the (i, j)th element of the ADCPM fingerprint which is define as

$$\mathbf{R}_k = \mathbb{E}\{\mathbf{\Omega}_k \odot \mathbf{\Omega}_k^*\} = \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,N_g} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,N_g} \\ \vdots & \vdots & \ddots & \vdots \\ r_{M,1} & r_{M,2} & \cdots & r_{M,N_g} \end{bmatrix} \tag{3.6}$$

where $\odot$ denotes the Hadamard product so that $[\mathbf{R}_k]_{i,j} = \mathbb{E}\{|[\mathbf{\Omega}_k]_{i,j}|^2\}$. As shown in Fig. 3.2, the ADCPM represents the AoA, the ToA, and the channel power of each path related to the scattering environment of user k.

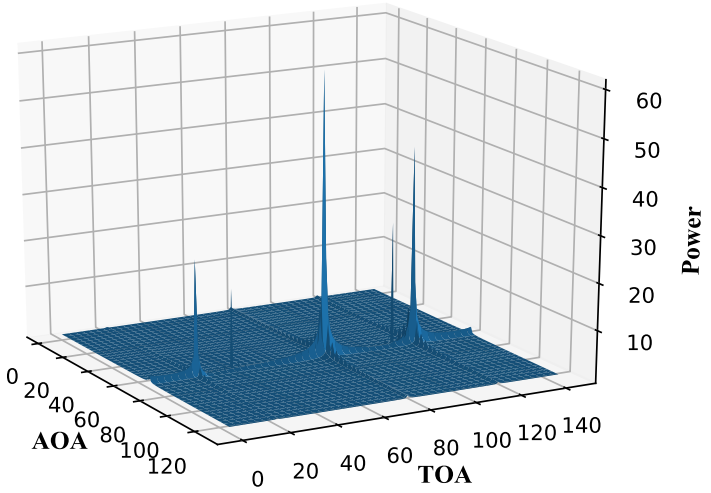


Figure 3.2: Example of ADCPM with $M = 128$ and $N_g = 144$.

3.3 Fingerprinting Localization Method Based on Clustering and Regression

Fig. 3.3 shows an overview of the architecture of the proposed localization method. The ADCPM is extracted as a fingerprint from the channel estimation result known to the BS. The remaining structure of the proposed location estimation method consists of two phases: the training phase and the estimation phase.

In the training phase, the training data are collected from a grid of known location reference points (RPs). Then, the fingerprints are labeled with their corresponding location coordinates, and the original dataset is created. The proposed method is based on four principal component blocks: i) pre-processing, which is used to standardize, reduce dimensions of the features, and visualize data, ii) clustering, where similar data are grouped and stored in the database, iii) cluster identification, which is employed for coarse localization, and iv) regression, where an accurate model is created for each cluster based on their similar data distribution. We will describe each block in detail in the following subsections.

During the estimation phase, we randomly select some unknown locations to collect the testing data. Similar to the training phase, the collected data is processed using the same method to generate fingerprints. Then, its corresponding cluster (cluster ID) is first determined using a cluster identification algorithm. Accordingly, the small region in which the new user is most probably located is identified. Lastly, accurate position estimation is applied using the GPR model of the corresponding cluster to improve the accuracy of the location estimation.

3.3.1 Offline Phase

Let's assume there are L training RPs. The ADCPM fingerprint of each RP is first converted to a one-dimensional array such that we have

$$\mathbf{r}_l = \left[r_{1,1}, \cdots, r_{1,N_g}, r_{2,1}, \cdots, r_{2,N_g}, \cdots, r_{M,1}, \cdots, r_{M,N_g} \right] \tag{3.7}$$

Therefore, each sample $\mathbf{r}_l$ has $B = M \times N_g$ attributes, i.e., $A_b, b = 1, \ldots, B$, and the radio map is defined as

$$\mathbf{R} = \left[\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_L \right]^T \tag{3.8}$$

where each row of $\mathbf{R}$ indicates a B-dimensional training vector $\mathbf{r}_l$ related to the training x-coordinates x_l and the the training y-coordinates y_l, $l = 1, \ldots, L$. Therefore, the corresponding $L \times 1$ training x-coordinates $\mathbf{x}$ and $L \times 1$ training y-coordinates $\mathbf{y}$ are defined as

follows

$$\begin{aligned} \mathbf{x} &= [x_1, x_2, \ldots, x_L]^T, \\ \mathbf{y} &= [y_1, y_2, \ldots, y_L]^T \end{aligned} \tag{3.9}$$

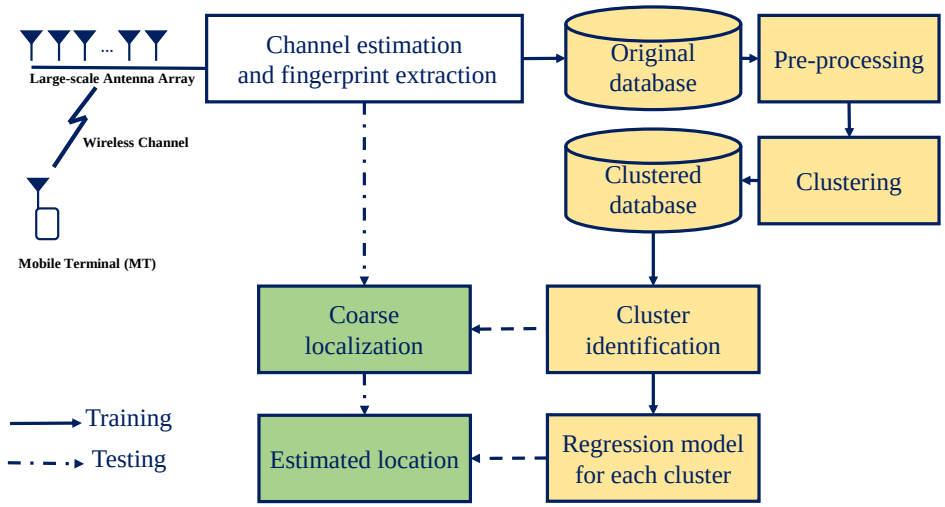


Figure 3.3: Overview of the proposed position estimation scheme.

Pre-processing

Data pre-processing is the core stage in data mining and machine learning. Data transformation, dimensionality reduction, and data visualization are well-known techniques in this step [79].

- **Data transformation**

  Standardization is the main pre-processing step in data mining, where attributes values are standardized from different dynamic ranges to a particular range [79]. Standardized datasets learn the training model faster and have better quality, efficiency, and accurate clustering results [80]. Based on the nature of the datasets for the analysis, it is essential to select an appropriate standardization method. In our case, for each training vector $\mathbf{r}_z$, standardization of each value, $r_{l,b}, b = 1, \ldots, B$, of attribute $A_b, b = 1, \ldots, B$ is done as follows

$$s_{l,b} = \frac{r_{l,b} - \mu_b}{\sigma_b} \tag{3.10}$$

  where μ_b and σ_b represent the mean and the standard deviation of attribute A_b. After standardization, the training vectors $\mathbf{r}_l$ converts to $\mathbf{s}_l$ where $l = 1, \ldots, L$, and the radio map $\mathbf{R}$ is changed into $\mathbf{S} \in \mathbb{R}^{l \times B}$.

- **Dimensionality reduction**

  The number of attributes that describe the dataset is defined as its dimensionality. These dimensions are represented as columns. Dimension reduction involves reducing the number of these columns to obtain a reduced or "compressed" representation of the original data, especially with the presence of the curse dimensionality. If the columns are correlated, there is some redundant information that affects the training model results [81]. Therefore, it is crucial to use dimensionality reduction techniques to avoid overfitting and reduce the model's complexity. In our case study, we applied principal components analysis (PCA) for dimensionality reduction.

  PCA is a well-established method for dimensionality reduction. PCA is an orthogonal linear transformation that maps the given dataset $\mathbf{R}$ present in a B-dimensional space to a D-dimensional space such that $D < B$, while retaining the most relevant information [82]. Before clustering, using PCA is a powerful method for clustering high dimensional datasets. The procedure of PCA is explained in the following. After data standardilation, which was explained in the previous section, i.e., the data transformation section, the covariance matrix of radio map $\mathbf{S}$ is computed by

$$\mathbf{\Sigma} = (\mathbf{S})^T \mathbf{S} \tag{3.11}$$

  where $\mathbf{\Sigma} \in \mathbb{R}^{B \times B}$. In order to find the principal components in the new feature space, we need to compute the eigenvalues λ_i and eigenvectors $\mathbf{e}_i$ of the covariance matrix $\mathbf{\Sigma}$, satisfying $\mathbf{\Sigma}\mathbf{e}_i = \lambda_i \mathbf{e}_i$. For computing eigenvalues λ_i, we have

$$(\mathbf{\Sigma} - \lambda_i \mathbf{I})\mathbf{e}_i = 0 \tag{3.12}$$

  Since $\mathbf{e}_i$ is a non-zero vector, (3.12) can be equal to zero if $\det(\mathbf{\Sigma} - \lambda_i \mathbf{I}) = 0$. Let I eigenvalues in descending order form the diagonal matrix $\mathbf{\Lambda}$:

$$\mathbf{\Lambda} = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_I] \tag{3.13}$$

  To discover the eigenvectors of $\mathbf{\Sigma}$ using eigenvalue decomposition, the matrix of eigenvectors is defined as

$$\mathbf{E} = [\mathbf{e}_1, \mathbf{e_2}, \ldots, \mathbf{e}_I] \tag{3.14}$$

  where $\mathbf{E} \in \mathbb{R}^{B \times I}$. With the D largest eigenvalues of $\mathbf{\Sigma}$, the eigenvector matrix is defined as

$$\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_D] \tag{3.15}$$

  where $\mathbf{E} \in \mathbb{R}^{B \times D}$ and its columns represent the principal components (the new dimensions) which are orthogonal to each other and arranged in decreasing order of variance. More precisely, the first eigenvector $\mathbf{e}_1$ is the direction in which the data varies the most, the second eigenvector $\mathbf{e}_2$ is the direction of greatest variance among those that are orthogonal (perpendicular) to the first eigenvector, and so on. The last step

is transforming the standardized radio map to a new dimensions radio map which is computed by

$$\mathbf{N} = \mathbf{SE} \tag{3.16}$$

where $\mathbf{N} \in \mathbb{R}^{L \times D}$.

- **Data visualization**

  The t-distributed stochastic neighbor embedding (t-SNE) algorithm is an innovative probabilistic method for data visualization. It is well known in machine learning due to its remarkable ability to transform high dimensional data to lower dimensions by preserving the neighborhood structure of the dataset [83, 84]. If there are L standardized training data points, in the t-SNE algorithm the similarity of data point $\mathbf{s}_l$ to $\mathbf{s}_{l'}$, where $1 \leq l, l' \leq L$ and $l \neq l'$, is a conditional probability given [83] by

$$p_{l'|l} = \frac{\exp\left(\frac{-\|\mathbf{s}_l - \mathbf{s}_{l'}\|^2}{2\sigma_l^2}\right)}{\sum_{l \neq l''} \exp\left(\frac{-\|\mathbf{s}_l - \mathbf{s}_{l''}\|^2}{2\sigma_l^2}\right)} \tag{3.17}$$

  $p_{l'|l}$ is high for nearby data points, whereas it will be relatively small for widely separated data points. σ_l is associated with a predefined input parameter $Perp$ known as "perplexity" and can be loosely interpreted as the number of effective neighbors that each data point has and is defined [85] as

$$Perp(p_l) = 2^{(H(p_l))} \tag{3.18}$$

  where $H(p_l)$ is the Shannon entropy, which is given by

$$H(p_l) = -\sum_{l'} p_{l'|l} \log_2 p_{l'|l} \tag{3.19}$$

  Then the $L \times L$ similarity matrix $\mathbf{P}_{High}$ in the original high dimensional space is formed and its entries are defined [83] as

$$p_{l,l'} = \frac{p_{l|l'} + p_{l'|l}}{2L} \tag{3.20}$$

  The t-SNE algorithm tends to learn a D-dimensional map $\{\mathbf{n}_1, \ldots, \mathbf{n}_L\}, \mathbf{n}_l \in \mathbb{R}^{1 \times d}$ of the original data that reflects the similarities $p_{l,l'}$ as well as possible. For this purpose, a Student t-distribution with one degree of freedom is used. Using this distribution, a $L \times L$ similarity matrix $\mathbf{Q}_{Low}$ in low dimensional space is defined whose entries are given by

$$q_{l,l'} = \frac{\left(1 + \|\mathbf{n}_l - \mathbf{n}_{l'}\|^2\right)^{-1}}{\sum_{l \neq l''} \left(1 + \|\mathbf{n}_l - \mathbf{n}_{l''}\|^2\right)^{-1}} \tag{3.21}$$

The t-SNE for finding the projections of the input data $\mathbf{s}_l$ in lower dimension as $\mathbf{n}_l$, employs a gradient-based technique to minimizes the Kullback–Leibler distance, which is defined as the cost function between $\mathbf{P}_{High}$ and $\mathbf{Q}_{Low}$ [83] and is given by

$$C = KL(\mathbf{P}_{High}||\mathbf{Q}) = \sum_{l=1}^{L}\sum_{l'=1}^{L} p_{l,l'} \log \frac{p_{l,l'}}{q_{l,l'}} \tag{3.22}$$

The gradient of the Kullback-Leibler distance between $\mathbf{P}_{High}$ and the Student-t based joint probability distribution $\mathbf{Q}_{Low}$ is given [83] by

$$\frac{\delta C}{\delta \mathbf{n}_l} = 4\sum_{l'}(p_{l,l'} - q_{l,l'})(\mathbf{n}_l - \mathbf{n}_{l'})\left(1 + \|\mathbf{n}_l - \mathbf{n}_{l'}\|^2\right)^{-1} \tag{3.23}$$

Clustering and Clustering Validation

The training data clustering and cluster validation procedure are presented in Fig. 3.4. After preprocessing, let us define $\mathbf{N}$ as the compressed radio map whose D-dimensions rows are $\mathbf{n}_l, l = 1, \ldots, L$. First a clustering algorithm is employed. Then, the quality of clustering is evaluated by a cluster validity index. In our method, we consider a different clustering algorithm which is explained in the following.

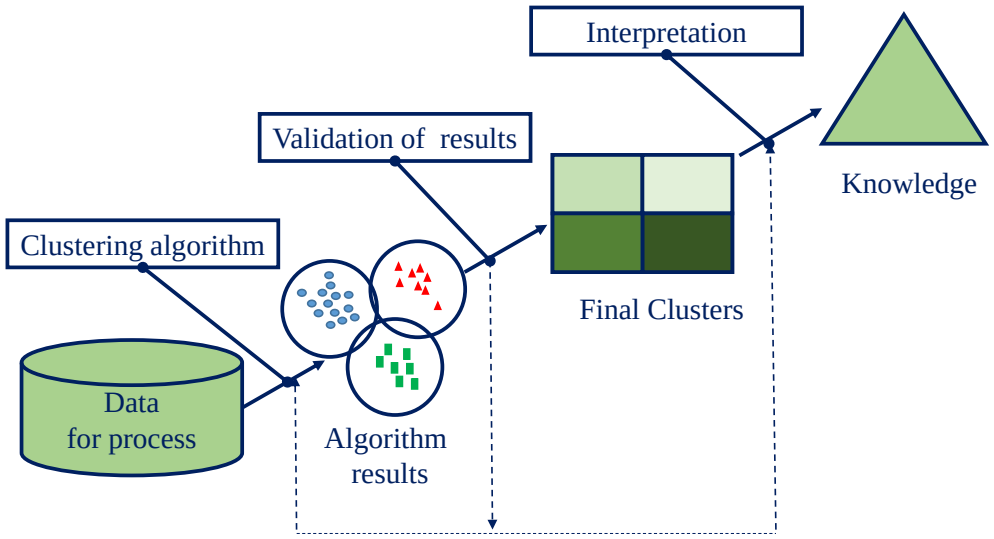


Figure 3.4: The pre-processing scenarios.

- **Affinity propagation clustering**
  The affinity propagation clustering (APC) algorithm [48] divides the training fingerprints into clusters by allocating each fingerprint sample an equal chance to become a

cluster-head (CH) [57]. Let us assume that the radio map $\mathbf{N}$ obtained after preprocessing on the training phase is represented as follows

$$\mathbf{N} = \left[\mathbf{n}_1, \mathbf{n}_2, \cdots, \mathbf{n}_L\right]^T \tag{3.24}$$

where, each row of $\mathbf{N}$ is the fingerprint of the lth RP.

To divide the training fingerprints into several clusters, the affinity propagation clustering (APC) algorithm [48] is employed by allocating each fingerprint sample an equal chance to become an exemplar. In K-means clustering, the number of output clusters and the corresponding random set of initial exemplars must be identified in advance [57]. Therefore, APC outperforms K-means clustering because it benefits from the initialization independent feature and better selection of the cluster exemplar.

The APC algorithm requires two types of real-valued input to divide $\mathbf{N}$ into clusters: the similarities matrix $\mathbf{S}_{Sim}$ and the preference $pref$. A pairwise similarity $s(\mathbf{n}_l, \mathbf{n}_{l'})$ (for $l \neq l'$) is used to indicate the appropriateness of $\mathbf{n}_l$ to be selected as the exemplar with respect to $\mathbf{n}_{l'}$. Since we aim to minimize the squared error, the similarity calculation in the APC is based on the negative squared error (Euclidean distance) as follows:

$$s(\mathbf{n}_l, \mathbf{n}_{l'}) = -||\mathbf{n}_l - \mathbf{n}_{l'}||^2 \tag{3.25}$$

where $1 \leq l, l' \leq L$ and $l \neq l'$. Also $pref$ is defined as follows

$$pref = \text{median}(s(\mathbf{n}_l, \mathbf{n}_{l'})) \tag{3.26}$$

To evaluate the quality of clustering results, a clustering index validity such as the silhouette (SI) is considered [70]. The SI is a well-known measure of how similar a training vector $\mathbf{n}_l, l = 1, \ldots, L$ is to its own cluster (cohesion) compared to other clusters (separation). It is then averaged over all training vectors and is defined [59] as

$$SI = \frac{1}{L} \sum_{l=1}^{L} \frac{d(\mathbf{n}_l) - f(\mathbf{n}_l)}{\max\{d(\mathbf{n}_l), f(\mathbf{n}_l)\}} \tag{3.27}$$

where $d(\mathbf{n}_l)$ is the average distance between $\mathbf{n}_l$ and all training vectors in other clusters and $f(\mathbf{n}_l)$ is the average distance between $\mathbf{n}_l$ and all training vectors in the same cluster.

Cluster Identification

In this section, to identify the corresponding cluster of each fingerprint, several algorithms can be employed such as KD-tree which was used in Chapter 2 and the artificial neural network (ANN). The KD-tree is not suitable for finding nearest neighbors in high dimensions

due to the curse of dimensionality. The ANN that we use is the multi-layer perceptron (MLP) which is employed and evaluated for coarse localization. For this purpose, a subset of the clustered training data points is selected as a validation dataset, and it is supposed that their cluster IDs are unknown. This subset is used to obtain the accuracy of the cluster identification model. Let us suppose there are D attributes after using dimensionality reduction and 20% of the training data are selected as validation dataset. The details of this cluster identification algorithm are described in the following.

- **Multi-layer Perceptron (MLP) Neural Networks (NN)**
  MLP is a supervised learning algorithm that learns a function $h(.) : R^D \sim R^T$ by training on a dataset, where D is the number of dimensions of attributes for the input and T is the number of cluster IDs for the output. Given a set of attributes and a target, MLP can learn a non-linear function for classification. MLP consists of three layers: input, hidden, and output. As shown in Fig. 3.5, we associate the input nodes with reference node $A_1 \sim A_D$ and the output nodes with predefined cluster-ID. The leftmost layer, known as the input layer, consists of a set of neurons $\{A_1, A_2, \ldots, A_D\}$ representing the input attributes. The hidden layers are between the input and output layers, where the transitional computations are performed. Each hidden layer uses the output of the previous layer to perform a non-linear operation, which is defined as:

$$\mathbf{h}^k = \phi(\mathbf{W}^k \mathbf{h}^{k-1} + \mathbf{b}^k) \tag{3.28}$$

  where $\mathbf{W}^k$ is a fully connected weight matrix that represents all the connections between each node of the $(k-1)$th layer and each node of the kth layer. $\mathbf{b}^k$ is the bias vector of kth layer, $\mathbf{h}^{k-1}$ represents the output from the previous layer. The weights and biases in a neural network are initially set to random values but the model is trained using the back-propagation (BP) method and the Adam optimizer [86] to minimize the loss function and updates the network parameters (i.e., weights and biases) iteratively until a convergence is achieved. $\phi(.)$ is the activation function. In our case, we use Rectified Linear Unit (ReLU) [87] (i.e., $\phi(n) = \max(n; 0)$) as an activation function in hidden layers and the softmax function is used as the activation function of the output layer so that the sum of the output values of all output neurons is equal to 1 and is defined as

$$\phi(\mathbf{n})_i = \frac{e^{n_i}}{\Sigma_{t=1}^{T} e^{n_t}} \tag{3.29}$$

  The output layer receives the values from the last hidden layer and transforms them into output values. Then the model is evaluated by comparing the real cluster-ID and the estimated cluster-ID.

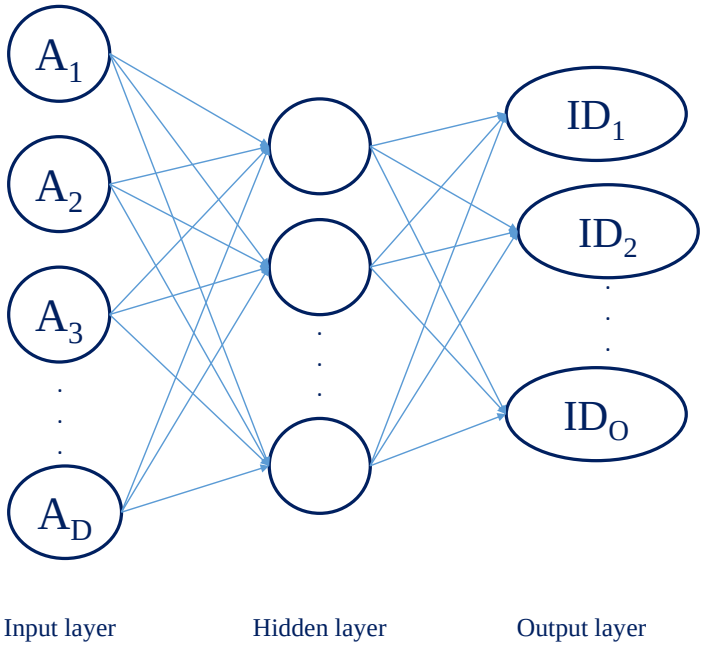


Figure 3.5: Structure of MLP classifier for cluster identification.

Making a Regression Model

Let us suppose matrix $\mathbf{N}$ is divided into T clusters $\mathbf{N}'_t, t = 1, \ldots, T$ with known locations $[\mathbf{x}_t, \mathbf{y}_t]$. Therefore, we have $\mathbf{N} = \{\mathbf{N}'_1, \mathbf{N}'_2, \ldots, \mathbf{N}'_T\}$. Consequently, the training data of each cluster are modeled using a GPR model, which takes the fingerprint as an input and provides the UE's location as an output. For simplicity, the data of a cluster is only considered in this part, but a similar procedure is applied for other clusters as well. Therefore, we consider the t^{th} cluster which has L' training samples.

Let us define $f_x(.)$ and $f_y(.)$ as the functions which map the fingerprint vector $\mathbf{n}_k$ of any user k into cluster $\mathbf{N}'_t$ to provide the 2-dimensional user's location coordinates (x_k, y_k), such that

$$\begin{aligned} x_k &= f_x(\mathbf{n}_k) + v_x, \\ y_k &= f_y(\mathbf{n}_k) + v_y \end{aligned} \tag{3.30}$$

where v_x and v_y are error terms modeled as i.i.d. Gaussian random variables with zero mean and variance $\sigma^2_{v_x}$ and $\sigma^2_{v_y}$, respectively. From (3.30), we can see that, estimating the x-coordinate x_k (and y-coordinate y_k) from $\mathbf{n}_k$ is a non-linear regression problem in machine learning. Among non-linear regression methods, we choose GPR because it is a powerful, Bayesian non-parametric approach that provides a probability distribution over all possible values [64]. Also, GPR methods have a good performance in terms of multiple metrics, including the squared prediction error [65]. For simplicity, details of the GPR model are

presented for the x-coordinates of the users and the same procedure can be applied for the y-coordinates.

In (3.30), function $f_x(.)$ is supposed to be random and follows a Gaussian process with zero-mean and covariance matrix $\mathbf{C}_t$ (also known as a kernel matrix), such that

$$f_x(.) \sim \mathcal{GP}\left(0, \mathbf{C}_t\right) \tag{3.31}$$

The covariance function between user k and user k' in cluster $\mathbf{N}'_t$, is the weighted sum of the squared-exponential, the linear and the delta function [66] and is defined by

$$c_t(\mathbf{n}_k, \mathbf{n}_{k'}) = \gamma e^{-\frac{1}{2\vartheta^2}\left\|\mathbf{n}_k - \mathbf{n}_{k'}\right\|^2} + \varpi \mathbf{n}_k^T \mathbf{n}_{k'} + \sigma_{v_x}^2 \delta_{kk'} \tag{3.32}$$

where $c_t(\mathbf{n}_k, \mathbf{n}_{k'})$ is an element of $\mathbf{C}_t$ and the delta function $\delta_{kk'}$ is 1 if $k = k'$ and 0 otherwise. According to (3.32), we need to estimate and optimize an unknown GPR hyperparameter vector $\boldsymbol{\phi}_t = [\gamma, \vartheta, \varpi]$ from the training data. Based on the GP assumption, i.e. data can be represented as a sample from a multivariate Gaussian distribution, we know that $\mathbf{x}_t$ is Gaussian distributed, $\mathbf{x}_t|\mathbf{N}'_t, \boldsymbol{\phi}_t \sim \mathcal{N}\left(0, \mathbf{C}_t\right)$ [51]. The log-likelihood function is used to derive the maximum likelihood estimator of parameter $\boldsymbol{\phi}_t$. The estimator $\hat{\boldsymbol{\phi}}_t$ is obtained by solving

$$\hat{\boldsymbol{\phi}}_t = \arg\max_{\boldsymbol{\phi}_t} \log(p(\mathbf{x}_t|\mathbf{N}'_t, \boldsymbol{\phi}_t)) \tag{3.33}$$

The optimization problem in (3.33) can be solved using a limited memory Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm which is an optimization algorithm in the family of quasi-Newton methods that approximates the BFGS algorithm using a limited amount of computer memory. It is a well-known algorithm for parameter estimation in machine learning that is explained in detail in [67].

3.3.2 Online Positioning Phase

In this phase, the position of a test user whose location is unknown is estimated. Let us suppose there are $\hat{L}$ test users. For this purpose, the $\hat{L} \times D$ testing matrix $\hat{\mathbf{N}}$ is used to estimate the $\hat{L} \times 1$ x-coordinate vector $\hat{\mathbf{x}}$. For each test user data $\hat{\mathbf{n}}$, the process of location estimation is described below.

- **Step 1: Cluster identification**
  Based on the cluster identification algorithm, the cluster ID t of testing data point $\hat{\mathbf{n}}$ is estimated.

- **Step 2: Location estimation**
  By using the GPR model of related cluster t, the x-coordinate $\hat{x}$ of test user fingerprint $\hat{\mathbf{n}}$ is estimated.

Now we would like to determine the posterior density of the location, i.e. $\hat{x}|\mathbf{x}_t, \mathbf{N}'_t, \hat{\mathbf{n}}$. According to [68], this distribution is Gaussian and the best estimate for $\hat{x}$ is the mean of this distribution, which is defined as

$$\hat{\mu}_x = \mathbf{c}_t^T [\mathbf{C}_t + \sigma_v^2 \mathbf{I}_{L'}]^{-1} \mathbf{x}_t \tag{3.34}$$

where $\mathbf{c}_t = (c_t(\hat{\mathbf{n}}, \mathbf{n}_1)), \ldots, c_t(\hat{\mathbf{n}}, \mathbf{n}_{\mathbf{L}'}))$ and L' is the number of training data in the cluster t. Also, the uncertainty in our estimate is given by its variance which is given as

$$\hat{\sigma}_x^2 = \sigma_v^2 - c_t(\hat{\mathbf{n}}, \hat{\mathbf{n}}) - \mathbf{c}_t^T [\mathbf{C}_t + \sigma_v^2 \mathbf{I}_{L'}]^{-1} \mathbf{c}_t \tag{3.35}$$

3.4 Results and Discussion

In this section, we compare the position estimation performance of the two-stage fingerprint clustering method in [78], which is considered as a benchmark, and the proposed method. To simulate a common urban wireless propagation scenario, a 120° sector with radius $R = 500$ m is considered. At the center of the sector, we place a BS which is equipped with a ULA where the number of antennas M is 128. The major wireless parameters which are employed in the simulation are set to those typical in LTE [69], [88], as listed in Table 3.1. Also, to have a simple analysis, it is assumed that the location of a UE is estimated based on the 20 nearest scatterers. The RPs are uniformly distributed every 10 m in a grid configuration over the whole target area. However, the test users are randomly distributed in the sector. In our case, the number of training RPs L is 893 and the number of testing data points $\hat{L}$ is 157. For each path, the location point of scattering is used to compute the AoA, the CFR of each UE is calculated according to (3.1) and (3.3).

Table 3.1: Wireless Parameters

Parameters	Value
Sampling interval	8.138 ns
Sampling frequency	122.88 MHz
Transmission bandwidth	100 MHz
Guard interval	4.7 MHz
Sub-carrier spacing	15 kHz
Carrier frequency	4650 MHz

3.4.1 Preprocessing Results

As mentioned in the offline phase, the fingerprints of the RPs are first extracted and stored into the dataset with their corresponding coordinates. The training data $\mathbf{R}$ are standardized

according to (4.8) and the pre-processing scenarios are employed. For evaluating the performance of the proposed method, first, the new dimensions of the training data should be specified using dimensionality reduction PCA. In PCA, a vital part is to estimate how many principal components which explain the variance of the data are needed. The relation between the number of principal components and the percentage of data variance is shown in Table 3.2. We can see that with the first 20 components, approximately 80% of the variance is contained, while we need 58 components to describe close to 95% of the variance. Since we aim to have 90% of the variance, the number of principal components is set to 35 and thus the dimensions of the original data is reduced to 35.

Table 3.2: The projection loss of the principal components.

Principal components	Variance percentage
20	80
30	85
35	90
58	95

In Fig. 3.6, shows the t-SNE visualization of applying the PCA on our localization dataset with 35 principal components. We can see that the data are very clearly separated into subgroups. We can clearly see how all the samples are nicely spaced apart and grouped together with their respective locations. If we now use a clustering algorithm to pick out the separate clusters, we could probably automatically assign new points to a label.

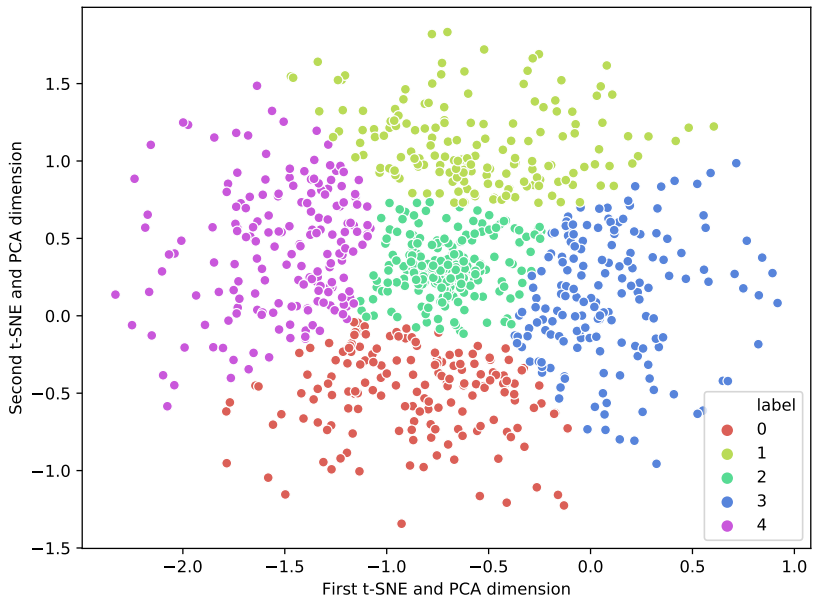


Figure 3.6: Preprocessing data using PCA and t-SNE.

3.4.2 Clustering and Cluster Validation Results

For clustering the compressed training data, k-means, APC and agglomerative algorithms are applied. Before applying APC, according to (3.26), we select the median value of the similarity matrix in (4.10) as the initial preference value. Then by increasing this value, we generate a range of preference values because changes in the preference value can result in quite different clustering results. Therefore, APC takes these values as an input to cluster the compressed training matrix $\mathbf{N}$. The maximum number of iterations in APC is $num\_iterations = 200$. For evaluating the performance of the proposed method, first, the valid number of clusters should be specified. It should be noted that the clustering results affect the cluster identification accuracy, the computational complexity, and further affect the position estimation accuracy, as shown later in this section. For this purpose, first, the valid clusterings which has high quality, are identified. Then, the results of validity indexes are averaged over 100 Monte-Carlo runs.

Fig. 3.7 shows the quality of clustering of the training data points using k-means, affinity propagation and agglomerative clustering algorithms based on the silhouette, SI, value. It should be noted that the value of SI is in the range of 0 to 1. A larger value of SI represents higher clustering quality. Each cluster can be separated significantly when $SI > 0.5$. We have an unsuitable cluster structure when $SI < 0.2$. According to the clustering results, we can see that when we use affinity propagation to cluster the training data points, we have better average silhouette which means that the data are well separated. Also, we have a good clustering quality when we use agglomerative clustering, especially when we have 5 clusters.

3.4.3 Accuracy of Cluster Identification

As mentioned in the offline phase, the MLP algorithm is applied for cluster identification. To evaluate the accuracy of the MLP model, 20% of all training data are considered as a validation dataset and it is supposed that their cluster-ID is unknown. Then, by comparing the estimated cluster ID and the real one, the accuracy is obtained. In our analysis, the accuracy of the model is 92%.

3.4.4 Positioning Performance

For estimating the position, the GPR model is trained by solving the log-likelihood maximization problem in (3.33).

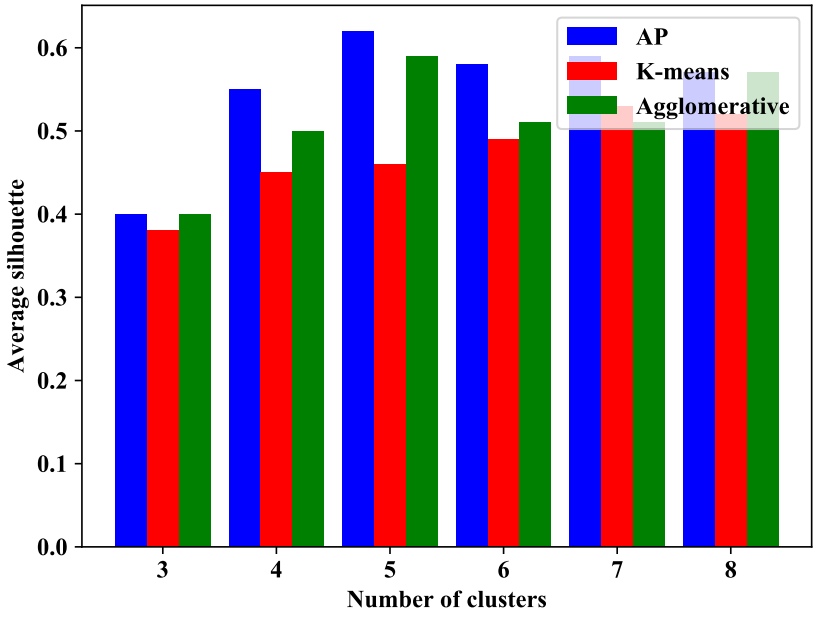


Figure 3.7: Comparison of clustering algorithms based on average silhouette.

During the online positioning phase, the fingerprints of the new UEs are extracted and their position is estimated with our positioning method. Then the estimated positions are compared with the true positions and the performance of the proposed fingerprint wireless positioning system is evaluated.

In this part, by considering the two-stage fingerprint clustering method in [78] as a benchmark, we first present the performance of the localization accuracy of the proposed method. Simulation results are obtained to indicate that the proposed method is suitable in massive MIMO-OFDM systems. Also, the effect of the number of BS antennas in location estimation performance is evaluated. For evaluation, a massive MIMO-OFDM system with 128 BS antennas is considered. Fig. 3.8 shows the cumulative distribution of the estimation errors for different methods. We can see that the accuracy of the proposed method is better than the benchmark with 93% reliability for 10-meter accuracy. For comparison, in [78], the reliability for 10-meter accuracy was 70%.

The impact of the number of BS antennas on the localization accuracy is demonstrated in Fig. 3.9. When the number of antennas is increased from 64 to 128, the reliability for 10-meter accuracy is increased from 81% to 90%. Also, we can see that increasing the number of BS antennas increases the localization accuracy.

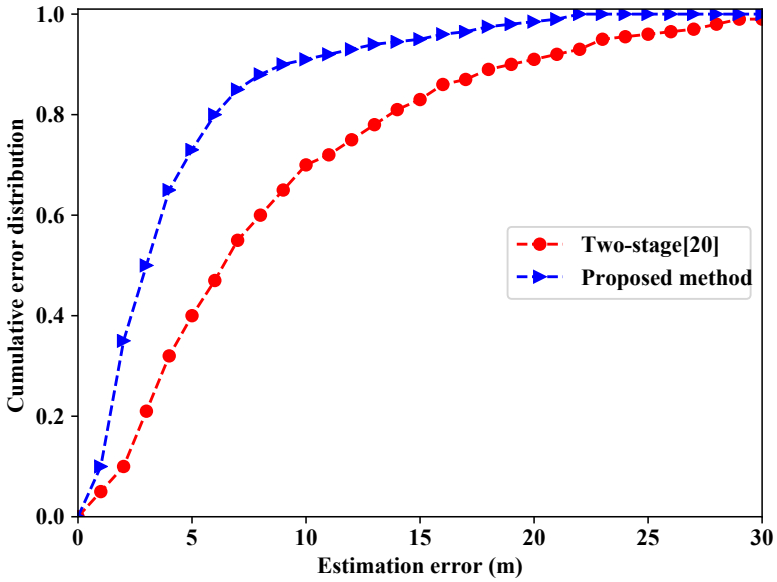


Figure 3.8: The CDF of the location errors.

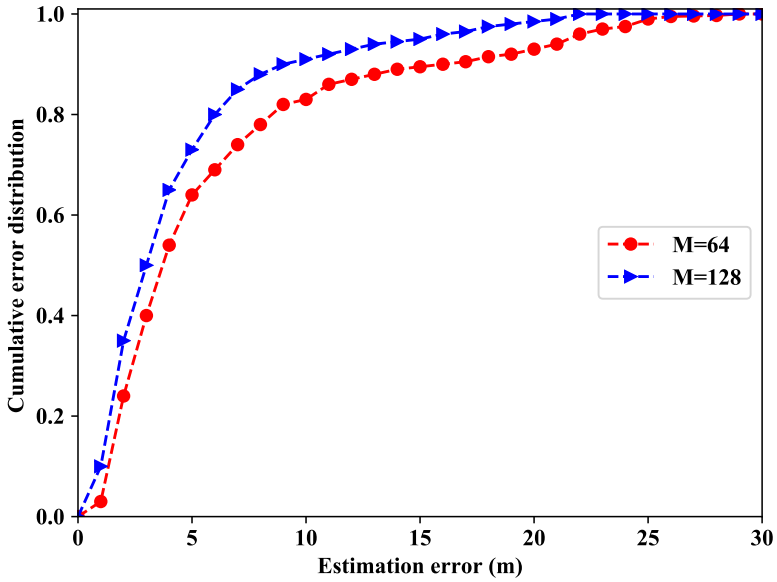


Figure 3.9: The CDF of the location errors with different number of antennas M.

3.5 Conclusion

We proposed a low dimensional cluster-based approach to estimate the user's location from the CSI in a collocated massive MIMO-OFDM system. In the proposed method, first, all high

dimension training data were map into a lower dimensional space. Then the whole testbed was divided into clusters using different clustering algorithms, which reduces the computational cost of online positioning. APC was chosen for clustering due to its initialization-independent property and better selection of cluster representative compared with k-means and agglomerative clustering. A MLP was used for cluster identification to allow for a quick finding of the related cluster. Also, GPR was applied for further location estimation within each cluster. The proposed method was compared with a previous work in terms of localization accuracy. Also, through simulation results, we showed that with increasing the number of antennas, the localization performance is improved.

Chapter 4

An Accurate, Robust and Low Dimensionality Deep Learning Localization Approach in DM-MIMO Systems Based on RSS

Résumé

Actuellement, la localisation dans des systèmes MIMO massifs distribués (DM-MIMO) basée sur l'approche d'empreintes digitales (FP) suscite un grand intérêt. Cependant, cette méthode souffre de trajets multiples et d'une grace dégradation du signal, de sorte que sa précision est détériorée dans des environnements de propagation complexes, ce qui entraîne des variations de la force du signal reçu (RSS). Par conséquent, le but de ce travail est de fournir une localisation robuste et précise. Dans ce chapitre, nous proposons une approche basée sur la FP pour améliorer la précision de la localisation en réduisant le bruit et les dimensions des données RSS. Dans l'approche proposée, les empreintes digitales reposent uniquement sur le RSS terminal mobile (MT) à antenne unique collecté au niveau de chacun des éléments d'antenne de réception de la station de base MIMO massif. Après avoir créé une carte radio, une analyse en composantes principales (PCA) est effectuée pour réduire le bruit et la redondance. La PCA réduit la dimension des données, ce qui conduit à la sélection des antennes appropriées et réduit la complexité. Un algorithme de clustering basé sur K-means et le clustering de propagation d'affinité (APC) est utilisé pour diviser la zone entière en plusieurs régions, ce qui améliore la précision du positionnement et réduit la complexité et la latence. Enfin, afin d'avoir une estimation de localisation précise élevée, toutes les données similaires dans chaque "cluster" sont modélisées à l'aide d'une régression de réseau neuronal profond (DNN) bien conçue. Les résultats de la simulation montrent que le

schéma proposé améliore considérablement la précision de positionnement. Cette approche a une couverture élevée et améliore les performances de la racine de l'erreur quadratique moyenne (RMSE) à quelques mètres, ce qui est attendu dans les réseaux 5G et au-delà. Par conséquent, cela prouve également la supériorité de la méthode proposée sur les précédents schémas d'estimation de localisation.

Abstract

Currently, localization in distributed massive MIMO (DM-MIMO) systems based on the fingerprinting (FP) approach has attracted great interest. However, this method suffers from severe multipath and signal degradation such that its accuracy is deteriorated in complex propagation environments, which results in variable received signal strength (RSS). Therefore, providing robust and accurate localization is the goal of this work. In this chapter, we propose an FP-based approach to improve the accuracy of localization by reducing the noise and the dimensions of the RSS data. In the proposed approach, the fingerprints rely solely on the RSS from the single-antenna MT collected at each of the receive antenna elements of the massive MIMO base station. After creating a radio map, principal component analysis (PCA) is performed to reduce the noise and redundancy. PCA reduces the data dimension which leads to the selection of the appropriate antennas and reduces complexity. A clustering algorithm based on K-means and affinity propagation clustering (APC) is employed to divide the whole area into several regions which improves positioning precision and reduces complexity and latency. Finally, in order to have high precise localization estimation, all similar data in each cluster are modeled using a well-designed deep neural network (DNN) regression. Simulation results show that the proposed scheme improves positioning accuracy significantly. This approach has high coverage and improves average root-mean-squared error (RMSE) performance to a few meters, which is expected in 5G and beyond networks. Consequently, it also proves the superiority of the proposed method over the previous location estimation schemes.

4.1 Introduction

Location-based services (LBSs) have recently attracted significant attention in wireless network applications. Today, several consumer goods are equipped with user location features which provide immediate and accurate localization of lost, delayed, or damaged assets [3]. Location information has a great application potential in industry, medicine, emergency management, surveillance, controlling autonomous vehicles, and many other vari-

ous fields [89]. At the same time, the demand for recognition of a mobile terminal's (MT) location has greatly increased so that numerous researches have been conducted on MTs' location [81]. Therefore, developping localization technology is becoming more and more important. The source of most localization systems are based in urban settings [81]. Today, the global positioning system (GPS) is the most used technology for outdoor localization due to its availability [5]. However, its accuracy deteriorates in shadowed locations and in the vicinity of high-rise buildings due to diminished satellite signals in the absence of line-of-sight (LoS) propagation [6]. Also, it requires too much power on an MT. Therefore, LBSs significantly need accurate and real-time localization to obtain notable performance improvement over existing cellular localization networks.

Currently, with the advent of 5G, the use of massive multiple-input multiple-output (M-MIMO) systems is drawing attention from the localization research community [89]. M-MIMO systems have been introduced as an enabling technology for 5G networks to improve localization accuracy in addition to enhancing communication performance [3]. In our work, we consider positioning multiple users simultaneously in a distributed massive MIMO (DM-MIMO) system, wherein the users are served on the same time-frequency resource by a large number of spatially-separated remote radio heads (RRHs) distributed over the whole area [2].

DM-MIMO systems potentially provide higher spectral efficiency [39], energy efficiency [90], average throughput rate, and coverage probability compared to conventional co-located massive MIMO (CM-MIMO) systems, where the base station (BS) is equipped with an array of co-located antennas [39]. Although DM-MIMO systems have significant benefits, user localization has not yet been as well established as with CM-MIMO systems [91]. A variety of wireless signal properties have been considered in M-MIMO systems for MT's localization. Among them, we concentrate on the received signal strength (RSS) since it has lower cost and complexity [91]. Machine learning (ML) approaches are then employed on the extracted signal properties, which are considered as to be a unique fingerprint of a specific location. Therefore, the localization problem can be solved using pattern recognition, which consists of fingerprint extraction, fingerprint matching, and ultimately location estimation [91].

In this chapter, we propose a robust and precise localization method using a dimension reduction technique, clustering, and regression, which are accomplished via two modes: the offline mode and the online mode. The extracted RSS samples from the whole area are analyzed during the offline mode. First, the dimensions of RSS samples are reduced using principal component analysis (PCA). The whole area is then split into several sub-areas using a combination of clustering algorithms based on the K-means and affinity propagation clustering (APC) algorithms. Ultimately, a deep neural network (DNN) regression is applied to the RSS samples of each cluster. The accuracy of the model is estimated using a validation dataset. When a new fingerprint is given in the online mode, it is first preprocessed, then

its cluster is specified, and finally, its location is estimated. We now summarize our major contributions in four aspects.

1. In the preprocessing step, we apply the PCA technique on all data samples to denoise the RSS sample and extract effective features and reduced unimportant features from RSS vectors. Preprocessing leads to speed up and improve the accuracy of our proposed machine learning-based method since the training time and complexity are reduced significantly with fewer dimensions (features). Also, it helps us to select a proper set of RRHs.

2. In the clustering step, a fast convergence, and initial value independent clustering method relying on a combination of K-means and AP clustering algorithms is proposed. This method reduces latency and computational complexity and helps to improve localization accuracy.

3. We propose a DNN regression for each cluster using all the data of the corresponding cluster to estimate the location more precisely.

4. The performance of the proposed localization method is evaluated in terms of root-mean-squared-error (RMSE) via simulations and compared to the works in [91].

The rest of the chapter is organized as follows. Section 4.2 overviews existing techniques and related works for localization. In Section 4.3, we present the system model. The positioning method is proposed in Section 4.4. Simulations results are presented and discussed in Section 4.5. A conclusion is presented in Section 4.6.

4.2 Related Work

4.2.1 User Positioning in Massive MIMO System

In recent years, user positioning in M-MIMO has attracted much attention and there are several research works in this area. The authors in [41], [42], and [43] use angle-of-arrival (AoA) information to estimate UE position in M-MIMO systems. In [45] and [46], the combined information of AoA, angle-of-delay (AoD), and time delay is used for user positioning in M-MIMO, where in [45] a mm-Wave M-MIMO system including LOS scenarios is considered. In [44], a compressed sensing approach is proposed to estimate the location of a MT from time-of-arrival (ToA) data recorded at multiple M-MIMO BSs. In [92], an environment sensing method is employed in a highly directional 60 GHz mm-Wave network to estimate MT's positions. However, the localization in all of the above techniques is based on the information obtained from a CM-MIMO system configuration, where the BS hosts

an array of antennas. But, these methods are not applicable in DM-MIMO systems, where single-antenna remote radio heads are considered. In [14] and [93], the Gaussian process regression (GPR) ML algorithm is employed based on RSS measurements in DM-MIMO systems. In [81], the performance of several ML algorithms, which are used in conjunction with fingerprint-based MT localization for DM-MIMO wireless systems configurations, is investigated and evaluated. In [91], RSS-based positioning using a machine learning method relies on the affinity propagation clustering algorithm and the GPR algorithm. Among the relevant works, the study of [91] is the most pertinent for our investigation, wherein the focus of the analysis is based on GPR. We expand on the work presented in [91], using data compression and deep learning algorithms to provide higher localization accuracy and less computational complexity.

4.2.2 Machine Learning and Deep Learning for User Positioning

Localization techniques are classified into four main categories: proximity-based, angle-based, range-based, and fingerprinting-based. The proximity-based is the most straightforward technique where the location is provided approximately in a particular radio coverage area based on the locations of the BSs. Therefore, BSs are required, which is not suitable for large areas [94]. The angle-based technique, which is based on the AoA of the received signal, is not efficient in non-line-of-sight (NLoS) situations because it produces a coarse error for positioning [9], [95]. In range-based techniques, one must compute the distance between the MT and at least three BSs. Then the MT location is estimated using trilateration. This can be accomplished through radio signal information received from MTs such as ToA and received signal strength (RSS). The ToA method is known for its complexity because it requires very expensive hardware at the BS, such as high accuracy clocks for time synchronization [10]. In addition, it has low performance in NLoS environments. It has been demonstrated that the RSS method is appropriate in non-urban environments because by increasing the distances the path loss is expected to decrease steadily [81]. This issue can be mitigated when the RSS method is employed in conjunction with a fingerprinting (FP) based method [14].

In a FP-based method, the location of MTs is estimated based on a pre-recorded data, called fingerprint, using ML and deep learning (DL) algorithms [14]. Since FP-based positioning methods have a good performance in highly-cluttered multipath environments [14], [33], they can be used in many systems such as WiFi networks [34–36, 96–98]. In addition to received signal information, channel state information (CSI) [37], [38] is used as the position fingerprint. In recent years, the FP-based localization method has attracted significant interest by combining mobile positioning requirements into 5G wireless communication systems due to its broad applicability and high cost-efficiency without any hardware requirement on the MTs [11].

Several machine learning methods including GP methods [14], and more recently, deep learning methods [47], [37] have been applied and investigated for wireless user positioning. However, the proposed methods for WiFi systems [37,49,50] are not applicable for M-MIMO systems because they do not consider the associated inter-user interference. In addition, they concentrate on the downlink, where the MTs estimate their positions by managing the computational cost while in M-MIMO systems, positioning is performed on the uplink, where the BS estimates the MTs' position.

4.3 System Description

In the considered single-cell DM-MIMO system (Fig. 4.1), there are K single-antenna users that transmit signals to M single-antenna RRHs on the same time-frequency resource. The high-speed front-haul links connect RRHs to a central processor unit (CU). When the RRHs receive signals transmitted by the users on the uplink, individually record their own multi-user RSS values and send them to the CU. The CU handles the multi-user interference and extracts the per-user RSS values from the multi-user RSS values. Then the CU from each user forms an $M \times 1$ RSS vector to perform localization [52], [53]. Details are as follows.

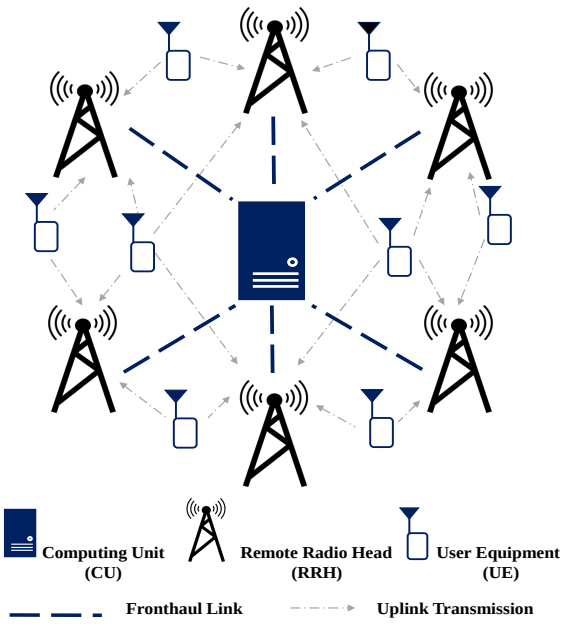


Figure 4.1: Multi-user DM-MIMO system model for location estimation.

4.3.1 Propagation Model

To explain the uplink of a multi-user DM-MIMO system in more detail, let $\mathbf{w}_k$ be the symbol vector transmitted by user k with transmission power ρ. If g_{mk} is the flat-fading channel gain between user k and RRH m, the sum symbol vector $\mathbf{y}_m$ received at RRH m is given by

$$\mathbf{y}_m = \sqrt{\rho} \sum_{k=1}^{K} g_{mk} \mathbf{w}_k + \mathbf{n}_m \tag{4.1}$$

In (4.1), $g_{mk} = q_{mk}\sqrt{h_{mk}}$ is a flat-fading channel where $q_{m,k}$ denotes small-scale fading represented by an independent and identically distributed (i.i.d.) zero mean complex Gaussian random variable with unit variance, i.e., $q_{mk} \sim \mathcal{CN}(0,1)$, h_{mk} is the large-scale fading coefficient, and $\mathbf{n}_m \sim \mathcal{N}(0, \sigma_n^2 \mathbf{I})$ is the additive white Gaussian noise. Note that the large-scale fading coefficient h_{mk} can be modeled [54] as

$$h_{mk} = b_0 d_{mk}^{-\alpha} 10^{z_{mk}/10} \tag{4.2}$$

where b_0 is the path loss at reference distance d_0, d_{mk} is the distance between user k and RRH m, α is the path-loss exponent (typically dependent on the environment and the range), and z_{mk} is the log-normal shadowing noise coefficient with $10 \log_{10} z_{mk} \sim \mathcal{N}(0, \sigma_z^2)$.

4.3.2 Mitigating Multi-user Interferences

For measuring the RSS, we consider the power of the received signal at RRH m which is given by $||\mathbf{y}_m||^2$ according to (4.1). But we should note that $||\mathbf{y}_m||^2$ at RRH m is in fact the multiuser RSS because the symbol vectors which are transmitted by all K users are combined at RRH m. Consequently, $||\mathbf{y}_m||^2$ cannot be directly used to estimate the position of user k. So the RSS of each user is not separately distinguishable. To overcome this, the symbol vectors $\mathbf{w}_k$ in (4.1) should be mutually orthogonal and should be already known at the RRH [14].

So, we need users to transmit an orthogonal set of pilot signals during channel estimation [55]. The RSS p_{mk} of user k can then be obtained from (4.1) [53] as

$$p_{mk} = \rho h_{mk} |q_{mk}|^2 \tag{4.3}$$

From (4.3), we can see that the RSS varies due to small-scale fading and shadowing of the wireless channel. The variation of small-scale fading can be decreased by averaging it over multiple time-slots according to the channel hardening effect [56]. But the shadowing effect, which is position-dependent and therefore depends on the user location, cannot be averaged out [7]. Therefore, the RSS between user k and RHH m, which is obtained from (4.2) and (4.3), when converted to dB scale, is given [53] by

$$p_{mk}^{dB} = p_0^{dB} - 10\alpha \log_{10}(d_{mk}) + z_{mk} \tag{4.4}$$

where $p_0^{dB} = 10\log_{10}(\rho b_0)$ is the uplink RSS at reference distance d_0. Once the per-user RSS values p_{mk}, $m = 1,\ldots,M$ and $k = 1,\ldots,K$, are extracted as above, the CU uplink RSS vector $\mathbf{p}_k$ is given by

$$\mathbf{p}_k = [p_{1k}^{dB}, p_{2k}^{dB}, \ldots, p_{Mk}^{dB}]^T \tag{4.5}$$

which is considered as the fingerprint.

4.4 A Clustering and Deep Learning Approach-Based Fingerprinting

An overview of the structure of the proposed localization method is shown in Fig. 4.2, which consists of two distinct modes: the offline mode and the online mode.

During the offline mode, the system captures the RSS fingerprints from a grid of known location reference points (RPs). Then, each fingerprint is labeled with corresponding location coordinates. The labeled data is divided into training, validation, and testing datasets. In the proposed method, learning is done with the training dataset and the performance is checked with the validation dataset. The accuracy of the localization system is then presented based on the testing dataset. Then PCA is applied for dimension reduction. After that, only a subset of dimensions (features) that have the maximum variance is selected. Therefore, an efficient feature set is produced. In the clustering step, the reduced-dimension training data is divided into several clusters using an efficient clustering method, which is based on K-means and AP. Later a cluster identification is employed for cluster matching and coarse localization. Ultimately, a DNN regression is trained for each cluster based on their similar data distributions. The accuracy of the model is evaluated using the validation dataset. If the accuracy of the proposed model is not sufficient, the clustering and regression parameters are modified in each iteration until convergence is achieved.

During the online mode, we provide the learned model to estimate the unknown locations of the test data. We first input the test data to the preprocessing level to transform the test data and reduce its dimensions. Then, its cluster or regions is identified using the cluster identification algorithm. Finally, the DNN regression of the related cluster is used to estimate the location.

4.4.1 Offline Mode

Let's assume there are L training data points which correspond to different RPs. Therefore, the radio map is formed as

$$\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_L]^T = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & p_{1,M} \\ p_{2,1} & p_{2,2} & \cdots & p_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ p_{L,1} & p_{L,2} & \cdots & p_{L,M} \end{bmatrix} \tag{4.6}$$

where each row $\mathbf{p}_l$ of radio map $\mathbf{P}$ is an M-dimensional fingerprint that corresponds to the training x-coordinates x_l and the training y-coordinates y_l, $l = 1, \ldots, L$.

$$\begin{aligned} \mathbf{x} &= [x_1, x_2, \ldots, x_L]^T, \\ \mathbf{y} &= [y_1, y_2, \ldots, y_L]^T \end{aligned} \tag{4.7}$$

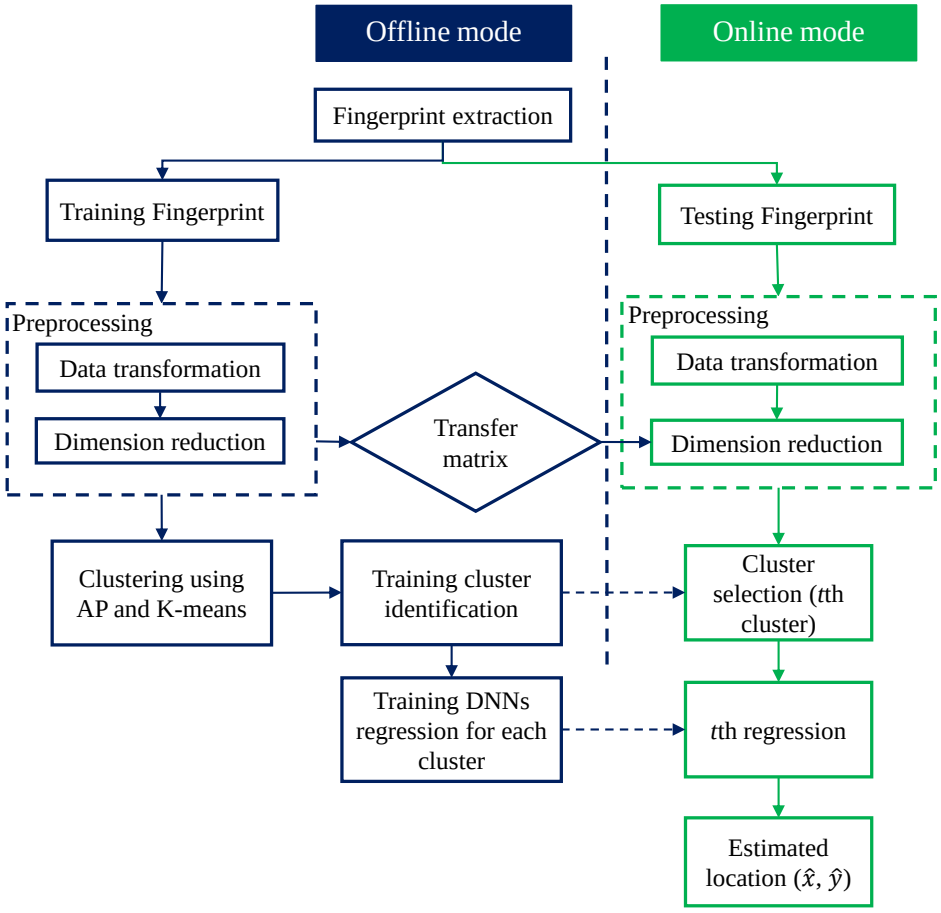


Figure 4.2: Overview of the proposed position estimation scheme.

Pre-processing

The core step in data mining and machine learning is data pre-processing, which consists of data transformation and noise reduction, and dimensionality reduction with PCA in our method [79]. Each step is explained in detail as follows.

- **Data Transformation**

  In data transformation, each feature value $p_{l,m}, m = 1, \ldots, M$ of each training fingerprint $\mathbf{p}_l$ is standardized [79] using

  $$s_{l,m} = \frac{p_{l,m} - \mu_m}{\sigma_m} \tag{4.8}$$

  where μ_m and σ_m are the mean and the standard deviation of the mth feature. With standardization, the training vector $\mathbf{p}_l$ converts to $\mathbf{s}_l$ where $l = 1, \ldots, L$, and the radio map $\mathbf{P}$ is changed into $\mathbf{S} \in \mathbb{R}^{L \times M}$.

- **Dimensionality reduction**

  PCA is employed for denoising and dimension reduction in order to map the standardized radio map $\mathbf{S}$, which is in an M-dimensional space, to an D-dimensional space such that $D < M$, while the most relevant information is maintained [82]. In PCA, in order to find the principal components (PCs) in the new feature space, we need to compute the eigenvalues λ_i and eigenvectors $\mathbf{e}_i$ of the covariance matrix of $\mathbf{S}$, $\mathbf{\Sigma} = \mathbf{S}^T\mathbf{S}$, where $\mathbf{\Sigma} \in \mathbb{R}^{M \times M}$ and satisfying $\mathbf{\Sigma}\mathbf{e}_i = \lambda_i \mathbf{e}_i$. The details for computing the eigenvalues λ_i are presented in [82].
  Let us assume there are I eigenvalues which are placed in descending order form the diagonal matrix $\mathbf{\Lambda} = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_I]$. Choosing the D largest eigenvalues of $\mathbf{\Sigma}$, we form eigenvector matrix $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_D]$ where $\mathbf{E} \in \mathbb{R}^{M \times D}$. Each column of $\mathbf{E}$ outlines the PCs, which are orthogonal to each other and in decreasing order. More precisely, the first eigenvector $\mathbf{e}_1$ is the direction that captures the maximum variance of data. The second eigenvector $\mathbf{e}_2$ is the direction that has the greatest variance among those that are orthogonal to the first eigenvector, and so on. Therefore, the low dimensionality radio map is $\mathbf{U} = (\mathbf{S})\mathbf{E}$, where $\mathbf{U} \in \mathbb{R}^{L \times D}$ and is defined as

  $$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_L]^T \tag{4.9}$$

  where $l = 1, 2, \ldots, L$.

Clustering

A clustering algorithm is required to split the whole area into several regions based on the collected RSS data. K-means is a very popular clustering algorithm that is extensively used

due to its fast convergence. However, it is sensitive to the initial condition wherein the number of clusters is predefined, and a random set of initial exemplars is selected in advance. Therefore, in K-means, many runs are needed to get a good clustering result. However, it does not guarantee i) that an appropriate initialization will occur during the repetitious running, and ii) unique clustering because we get different results with randomly chosen initial clusters.

In contrast, the affinity propagation (AP) clustering algorithm [48], has the initialization-independent property wherein all RSS samples have an equal chance to be a cluster head (CH). The optimal number of clusters is then obtained by passing iteratively two kinds of messages, named validity and responsibility, to maximize a fitness function until a good set of CHs emerges [48]. Therefore, APC can provide a good set of CHs with high speed. However, it can sometimes fail to converge, particularly for large similarity matrices. Considering the convergence property of K-means and the good performance of affinity propagation, a new clustering method is employed. The APC algorithm is first used to determine the optimal number of clusters and the initial CHs. Then, K-means is employed to create the final clustering results by iteration based on the initial CHs.

As mentioned in [91], the AP clustering algorithm requires two inputs to divide $\mathbf{U}$ into clusters: the similarities matrix $\mathbf{S}_{Sim}$ and the preference $pref$, which are defined as follows [48]:

$$\begin{aligned} s_{Sim}(\mathbf{u}_l, \mathbf{u}_{l'}) &= -||\mathbf{u}_l - \mathbf{u}_{l'}||^2 \\ pref &= \text{median}(s_{Sim}(\mathbf{u}_l, \mathbf{u}_{l'})) \end{aligned} \tag{4.10}$$

where $1 \leq l, l' \leq L$ and $l \neq l'$.

The cluster algorithm used in our research is summarized in Algorithm 4.1. The validity of clustering is measured by the silhouette which is a well-known measure of how similar a training RSS vector $\mathbf{u}_l$, $l = 1, \ldots, L$ is to its own cluster (cohesion) compared to other clusters (separation). It is averaged over all training RSS vectors and is defined [59] as

$$SI = \frac{1}{L} \sum_{l=1}^{L} \frac{d(\mathbf{u}_l) - f(\mathbf{u}_l)}{\max\{d(\mathbf{u}_l), f(\mathbf{u}_l)\}} \tag{4.11}$$

where $d(\mathbf{u}_l)$ is the average distance between $\mathbf{u}_l$ and all training RSS samples in other clusters and $f(\mathbf{u}_l)$ is the average distance between $\mathbf{u}_l$ and all training RSS samples in the same cluster. A clustering which has sufficient SI value is regarded as a valid clustering.

Cluster selection

To determine the cluster corresponding to a new data point, a cluster identification based on KD-tree is used, and its accuracy is evaluated based on the valid clustering. For this purpose, a subset of clustered training RSS samples is selected as a validation dataset, and it is supposed that their cluster ID are unknown. This subset is used to obtain the accuracy of the cluster identification algorithm. First, the KD-tree algorithm, which finds similar data quickly, is employed for each cluster. Then the KD-tree uses each RSS sample of the validation dataset to find their K_{nn} nearest neighbors from each cluster; among them, the one that has the minimum distance is selected. Therefore, the predicted cluster ID of the validation RSS sample will be the cluster ID of its nearest neighbor.

To estimate the accuracy of clustering, we need to determine the error of cluster membership by comparing the predicted and the real cluster ID of the validation RSS samples. A threshold is considered for the accuracy of the cluster identification algorithm. If the accuracy of the valid clustering is less than the threshold, that clustering is ignored. Otherwise, its validity check and the number of clusters of the clustering which has the highest validity and sufficient accuracy requirements for cluster identification is selected as the best number of clusters.

Algorithm 4.1 Clustering based on APC and K-means

Require:
Preference value $pref$
The training matrix $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_L]^T$
1: Compute similarity matrix $\mathbf{S}_{Sim}$
2: Run APC algorithm to get CHs $C = \{C_1, C_2, ..., C_T\}$
3: Calculate the number of clusters T and the initial centers for the K-means clustering
4: Run K-means clustering

Regression

In order to have very precise localization, we use a DNN to solve the regression problem in each cluster.

Fig. 4.3 shows the fully connected multi-layer neural network structure in our method, which consists of an input layer, hidden layers, and an output layer. The input layer consists of artificial input nodes and receives the initial data for further processing. After dimension reduction, each RSS vector is composed of D RSS values. Therefore, the number of input nodes is equal to the number of dimensions of each RSS vector. The output layer produces the required output. The position information of the MT is set as output. Therefore, the number of output nodes is two. The hidden layers are between the input and output layers,

where the transitional computations are performed. Each hidden layer uses the output of the previous layer to perform a non-linear operation, which is defined as:

$$\mathbf{h}^k = \phi(\mathbf{W}^k \mathbf{h}^{k-1} + \mathbf{b}^k) \tag{4.12}$$

where $\mathbf{W}^k$ is a fully connected weight matrix that represents all the connections between each node of the $(k-1)$th layer and each node of the kth layer. $\mathbf{b}^k$ is the bias vector of the kth layer, $\mathbf{h}^{k-1}$ represents the output from the previous layer. The weights and biases in a neural network are initially set to random values but the model is trained using the back-propagation (BP) method and the Adam optimizer [86] to minimize the loss function and the network parameters (i.e., weights and biases) are updated iteratively until convergence is achieved. $\phi(.)$ is the activation function and in our case, we use a Rectified Linear Unit (ReLU) [87] (i.e., $\phi(x) = \max(x; 0)$) in the hidden layers and a linear function (i.e., $\phi(x) = x$) in the output layer, since the localization is a regression problem.

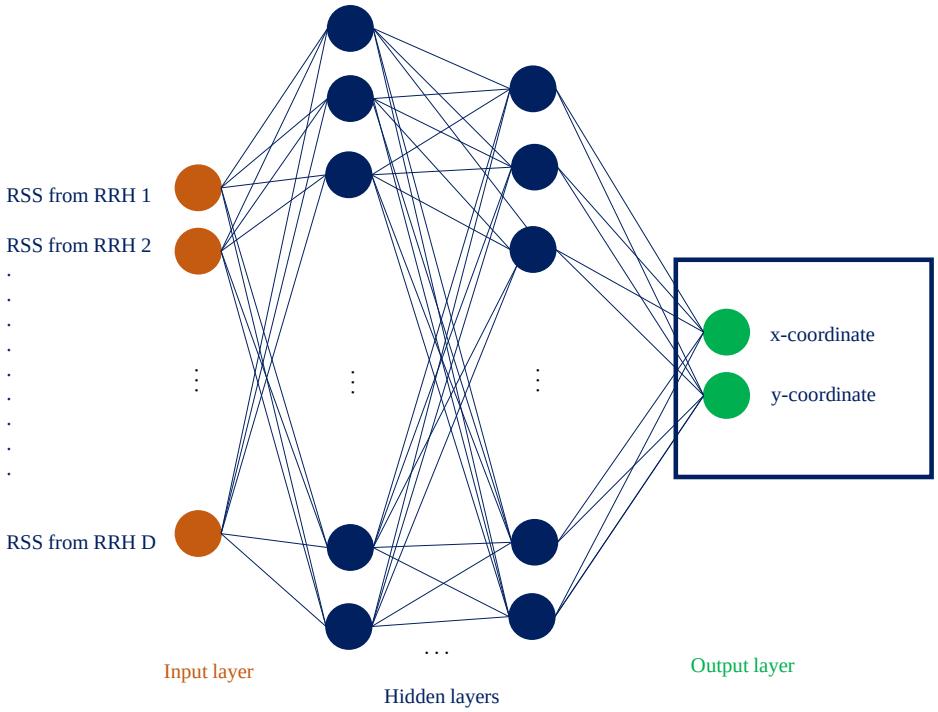


Figure 4.3: The DNN structure.

4.4.2 Online Mode

In this phase, the position of a test user whose location is unknown is estimated. Let us suppose there are $\hat{L}$ test users. After using dimensionality reduction, the $\hat{L} \times D$ testing matrix $\hat{\mathbf{U}}$ is used to estimate the $\hat{L} \times 2$ location coordinates $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$. For each test user data $\hat{\mathbf{u}}$, the process of location estimation is described below.

- *Step 1: Cluster selection*
  The cluster ID t of testing data point $\hat{\mathbf{u}}$ is determined using the cluster selection algorithm.

- *Step 2: Location estimation*
  The DNN regression model of cluster t is used to estimate the location $(\hat{x}, \hat{y})$ of fingerprint $\hat{\mathbf{u}}$.

4.5 Performance Evaluation

In this section, we compare using simulations, the location estimation using GPR [14], APC-GPR [91], and the proposed method in a DM-MIMO system with $M = 36$ single antenna RRHs, $L = 400$ training locations, and $\hat{L} = 16$ test users. Training users are distributed every 10 m in a grid configuration over the whole area of 200 m $\times$ 200 m. Test users are distributed in a random configuration, as shown in Fig. 4.4. For training, the RSS matrix $\mathbf{P}$ is generated using (4.4) with user transmit power $\rho = 21$ dBm, reference path loss $b_0 = -47.5$ dB and different shadowing noise variance $\sigma_z^2 = 1, 3, 5$ dB. Also, we set the path loss exponent to $\alpha = 0$ for $0 \leq d_{mk} < 10$ m, $\alpha = 2$ for 10 m $\leq d_{mk} < 50$ m, and $\alpha = 6.7$ for 50 m $\leq d_{mk}$, according to the 3GPP urban micro propagation model [69]. The PCA technique is applied on $\mathbf{P}$ to reduce the dimensions of RSS vectors and to generate the transformed RSS radio map $\mathbf{U}$. Then the clustering algorithm is applied to cluster all the training data using a similarity matrix and preference values which are generated by (4.10). When the optimal number of clusters is obtained, K-means algorithm is run for 100 times where the number of clusters is equal to 6. Also, the KD-tree is evaluated with different K. Finally, the proposed DNN model is trained with different hidden layers and activation functions. The root-mean-squared error (RMSE) between the real coordinates (x_l, y_l) of the test users and their estimates $(\hat{x}_l, \hat{y}_l)$ is considered as a performance metric, which is defined as

$$RMSE = \sqrt{\frac{\sum_{l=1}^{\hat{L}} (x_l - \hat{x}_l)^2 + (y_l - \hat{y}_l)^2}{\hat{L}}} \tag{4.13}$$

The RMSE, is averaged over the Monte-Carlo realizations. Lower RMSE values indicate better location estimation performance.

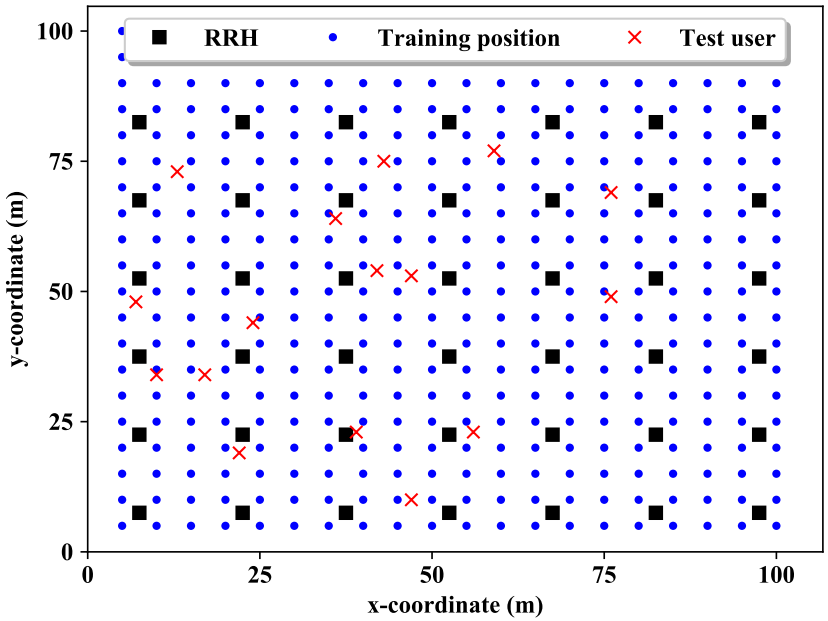


Figure 4.4: Simulation setup with $M = 36$ single antenna RHHs, $L = 400$ training positions, and $\hat{L} = 16$ test users.

4.5.1 Preprocessing

For an efficient and accurate clustering algorithm, we need to extract the essential RSS values received by RRHs by reducing the noise and the high dimensionality of the data. As mentioned, the PCA is employed in the offline mode to extract the more important feature set from the original RSS data set, while assuring the same level of positioning accuracy. Also, each RSS sample vector in the online mode is transformed into its low-dimension representation and is then compared with the corresponding low-dimension radio map. Using PCA, the optimal number of components that capture the greatest variance in the data are found. In this work, a 98% variance criterion is considered. Note that different variance thresholds may be chosen depending on the applications specific requirements. Fig. 4.5 shows how the variance is captured by principal components. We see that the first three components explain the majority of the variance in our data.

From Fig. 4.6, we can see that with the first 27 components, 98% of the variance is contained. Therefore, the number of principal components is set at 27, and the dimension of the original data is reduced.

4.5.2 Clustering

By running AP, the optimal number of clusters is equal to 6, which is considered in the k-means clustering as the input. In the case of 6 clusters, we have a maximum of silhouette

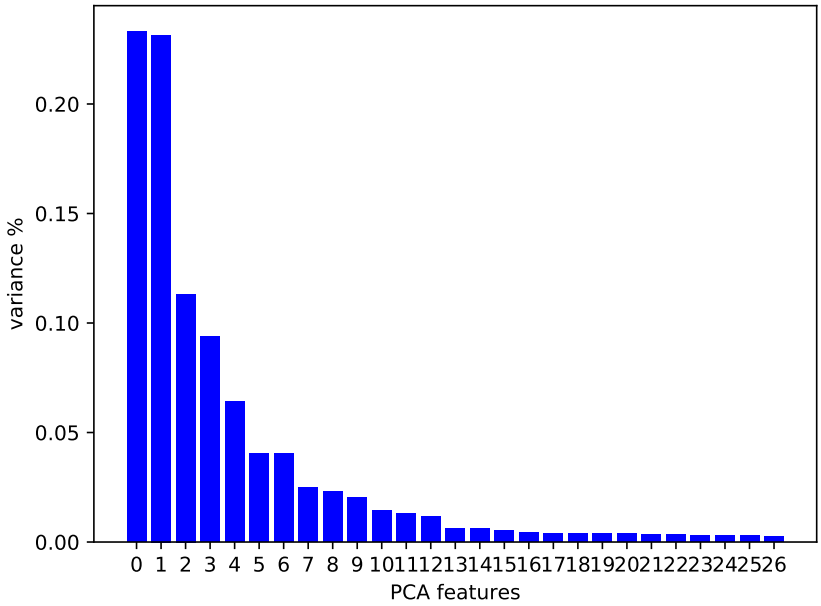


Figure 4.5: Variance of the principal components.

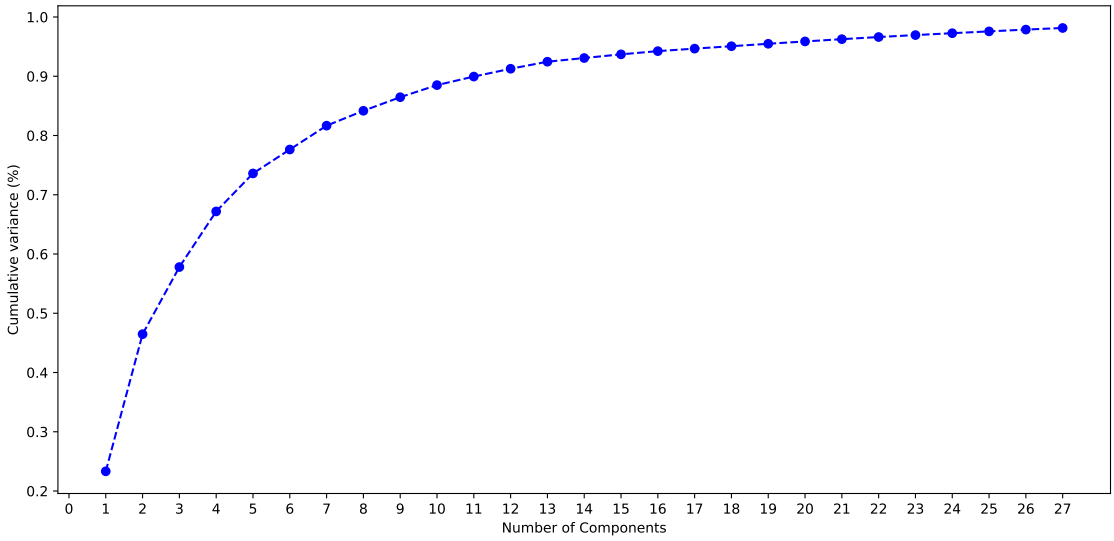


Figure 4.6: The cumulative sum of PCA components' variance. The first component already contains more than 20% of the total variance, 27 components take into account 98% of the RSS.

value. Also, the KD-tree algorithm with $K_{nn} = 3$ is used for cluster identification as considered in [91].

4.5.3 Comparison of Different Localization Methods

Fig. 4.7 shows the average RMSE of the test user's location estimation as a function of different shadowing noise variance ranging from 1 dB to 5 dB for different methods. We can see that the average RMSE is increased by increasing the shadowing noise variance in all methods. When we apply PCA in GPR and AP-GPR methods in [14] and [91] respectively, although a similar increase in average RMSE is observed by increasing the shadowing noise variance, the methods where PCA is used have lower average RMSE than those with no PCA and also the proposed method in the current study has a significantly lower average RMSE compared to the others. Also, we can see that the proposed method has a superior performance compared to the previous methods. Using PCA reduces noise and the number of dimensions of the data, that leads to increase stability, and reduces the computational complexity.

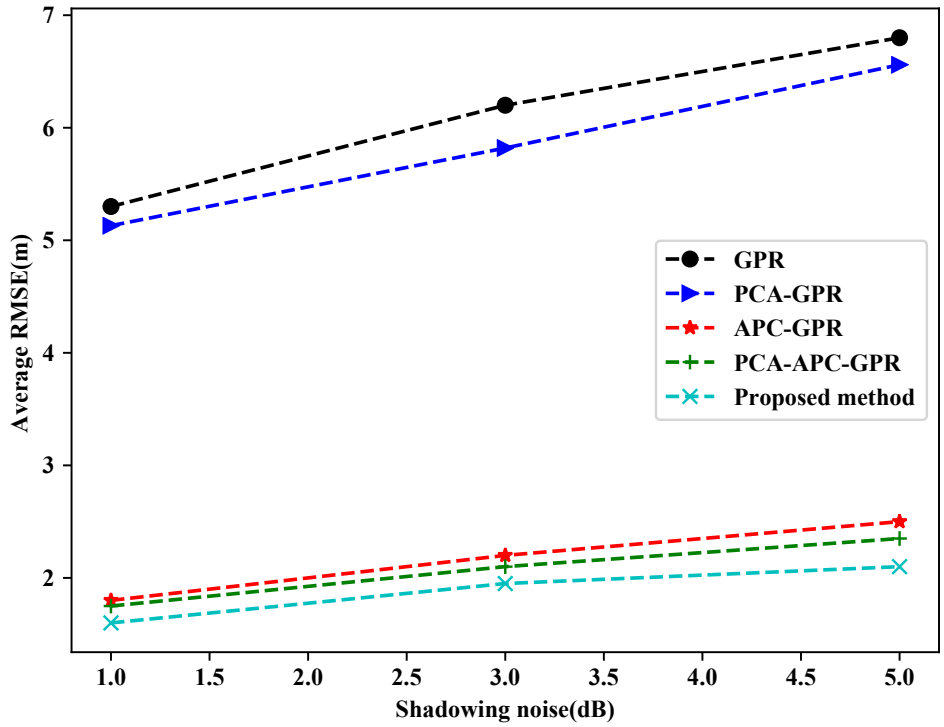


Figure 4.7: Average RMSE of using GPR [14], APC-GPR [91] and the proposed methods with $M = 36$, when the shadowing noise variance is 1, 3, and 5 dB and $L = 400$.

4.6 Conclusion

We proposed an efficient and low dimension FP-based method using PCA, APC and k-means, and DNN to estimate the user's location based on RSS values in a DM-MIMO sys-

tem. In the proposed method, after preprocessing the data such as denoising and dimension reduction, the whole testbed was first divided into clusters using the AP and k-means algorithms, which reduces the computational cost of online positioning. AP was chosen for clustering due to its initialization-independent property and a better selection of CHs and k-means was combined with AP due to its great convergence. Then, KD-tree was used for cluster identification to allow for a quick finding of the related cluster. Also, DNN was applied for further location estimation within each cluster. The proposed method was compared to previous works in terms of localization accuracy. Numerical results have justified our proposed localization system over previous schemes. Also, through simulations, we showed that increasing the shadowing noise variance decreases localization performance.

Conclusion

Summary of the thesis

The main goal of this doctoral research was to present accurate localization methods in M-MIMO systems such as DM-MIMO and CM-MIMO. In other words, a M-MIMO system should be able to localize the MTs with high accuracy. Different researches have been done in this field in the past to provide localization systems. However, providing more accurate localization was a problem in the majority of these researches. In this thesis, the main focus was on presenting a method that overcomes this problem. In other words, the presented localization system has the ability to estimate the location of MTs with high accuracy while having low computational complexity and high speed. A cluster-based approach using affinity propagation clustering (APC) and Gaussian process regression (GPR) was presented in Chapter 2 to estimate MT's location from their uplink RSS data in a DM-MIMO system. The modified version of this structure combined with PCA was presented in Chapter 3 to estimate the user's location from the CSI in a collocated massive MIMO-OFDM system. Also, low dimension FP-based method using PCA and DNN was presented in Chapter 4 to estimate the user's location based on RSS values in a DM-MIMO system.

Contributions

The first contribution presented in this thesis was to propose a method using clustering based on APC, and regression based on GPR in DM-MIMO. Since the FP-based techniques in localization systems have high computational complexity, we used clustering to divide the whole environment into several regions, which leads to reducing computational complexity and average searching cost. APC was chosen for clustering due to its initialization-independent property and better selection of cluster heads compared with k-means clustering. KD-tree and GPR were selected for cluster identification and location estimation within each cluster, respectively. The goal of the proposed model was first to distinguish the region or cluster of MTs, then estimating their location. The main idea in this contribution was to increase the accuracy of location estimation while reducing the computational complexity

of using GPR. Referring to the results of Chapters 3, the presented localization system provides better performance in estimating MT's location. Also, its performance was evaluated as a function of different parameters. We proved that i) the average RMSE values of the proposed method decrease by increasing the number of RRHs and the number of training samples, (ii) clustering the whole area into several clusters helps to minimize the computational complexity of online positioning.

Then, a low dimensional cluster-based approach was presented for a collocated massive MIMO-OFDM system to estimate the user's location from the CSI. A collocated massive MIMO with OFDM modulation was selected for the system model. Also, the CSI was used for extracting ADCPM fingerprints in multipath situations. PCA and APC were selected for dimensionality reduction and clustering, respectively. MPL was chosen for cluster identification to allow for a quick finding of the related cluster or region. Also, GPR was applied for further location estimation within each cluster. Referring to the results of Chapter 4, the proposed method provides higher accuracy compared with a previous work and by increasing the number of antennas, the localization performance is improved.

Finally, an efficient and low dimension FP-based method using PCA, a combination of APC and K-means, and DNN regression was proposed to improve the localization accuracy of the approach offered in the first contribution. In this method, preprocessing using PCA was applied to denoise the RSS data and reduce the dimensions of the RSS vectors which lead to selecting antennas that are essential for DM-MIMO systems implementation. Then, a clustering based on APC and K-means was employed to hold the convergence property of K-means and the good performance of affinity propagation. For this purpose, the APC algorithm was first used to determine the optimal number of clusters and the initial CHs. Then, K-means was applied to form the final clustering results by iteration based on the initial CHs. The DNN regression was selected for more accuracy of location estimation. The proposed method was compared to previous works in terms of localization accuracy. Through numerical results, it was proved that the proposed localization system is better than previous schemes.

Future works

As localization in massive MIMO using ML and DL techniques is still a relatively new research area in wireless communications, there are many topics for future works, including propagation models, big datasets, deep learning algorithms, system designs, and implementation issues. Related to the work in this thesis, we can consider the following topics for future studies.

1. Assuming the channel is unknown to the BS and estimating the channel considering there is pilot contamination.

2. Using uniform rectangular antenna arrays and 3D propagation model for the M-MIMO systems.

3. Combining mmWave and M-MIMO technologies to have a broad spectrum of advantages such as i) high multiplexing gains due to a large number of antenna arrays in M-MIMO, ii) huge available bandwidth that is found at millimeter frequencies, and iii) reduced interference due to narrow beamforming. These combinations make an opportunity to support a high-speed service such as localization. For this purpose, 5G heterogeneous networks (HetNets) can be considered as the candidate system for localization which is made up of macrocell and small cell BSs, all with massive MIMO and mmWave communication capacities.

4. Even if the results of this thesis are promising, the data set used for evaluation was limited. More data would give the machine learning algorithms a larger learning set which could boost the performance of the positioning methods. A larger data set having more overlapping data points could make it possible to evaluate the machine learning algorithms.

5. Using longer sequences of estimated positions from the machine learning algorithms, for tracking the MTs in urban areas.

6. Trying a variational autoencoder (VAE), which has been used to remove noise, and convolutional neural network (CNN), which has been used to obtain promising positioning accuracy from earlier research in both CM-MIMO and DM-MIMO.

7. Developing new detection algorithms to confirm that the LoS and NLoS classifications made in this thesis are of interest.

Bibliography

[1] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 742–758, Oct. 2014.

[2] C.-M. Chen, S. Blandino, Steve, A. Gaber, C. Desset, A. Bourdoux, L. Van der, and S. Pollin, "Distributed massive MIMO: A diversity combining method for TDD reciprocity calibration," in *IEEE Global Communications Conference*, pp. 1–7, Dec. 2017.

[3] Y. Liu, X. Shi, S. He , and Z. Shi, "Prospective Positioning Architecture and Technologies in 5G Networks," *IEEE Network*, vol. 31, pp. 115–121, Aug. 2017.

[4] P. Zhang, J. Lu, Y. Wang , and Q. Wang, "Cooperative localization in 5G networks: A survey," *ICT Express*, vol. 3, pp. 27–32, Mar. 2017.

[5] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global positioning system: theory and practice*. Springer Science & Business Media, Dec. 2012.

[6] Y. Cui and S. S. Ge, "Autonomous vehicle positioning with GPS in urban canyon environments," *IEEE Transactions on Robotics and Automation*, vol. 19, pp. 15–25, Feb. 2003.

[7] A. Zanella, "Best practice in RSS measurements and ranging," *IEEE Communications Surveys & Tutorials*, vol. 18, pp. 2662–2686, Apr. 2016.

[8] H. C. So and L. Lin, "Linear least squares approach for accurate received signal strength based source localization," *IEEE Transactions on Signal Processing*, vol. 59, pp. 4035–4040, May. 2011.

[9] M. Li and Y. Lu, "Angle-of-arrival estimation for localization and communication in wireless networks," in *16th European Signal Processing Conference*, pp. 1–5, IEEE, 2008.

[10] Y. Chan and K. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE Transactions on signal processing*, vol. 42, pp. 1905–1915, Aug. 1994.

[11] G. Mao, *Localization Algorithms and Strategies for Wireless Sensor Networks: Monitoring and Surveillance Techniques for Target Tracking*. IGI Global, May 2009.

[12] X. Sun, C. Wu, X. Gao, and G. Y. Li, "Fingerprint-based localization for massive MIMO-OFDM system with deep convolutional neural networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 10846–10857, 2019.

[13] H. Q. Ngo, *Massive MIMO: Fundamentals and system designs*, vol. 1642. Linköping University Electronic Press, 2015.

[14] V. Savic and E. G. Larsson, "Fingerprinting-based positioning in distributed massive MIMO systems," in *Vehicular Technology Conference (VTC Fall)*, pp. 1–5, Sept. 2015.

[15] R. S. Campos, "Evolution of Positioning Techniques in Cellular Networks, from 2G to 4G," *Wireless Communications and Mobile Computing*, vol. 2017, Jan. 2017.

[16] J. Stefanski, "Accuracy analysis of mobile station location in cellular networks," in *Information Technology (ICIT)*, pp. 101–102, IEEE, Jun. 2010.

[17] M. Thorpe, M. Kottkamp, A. Rössler , and J. Schütz, "LTE Location Based Services: Technology Introduction," *Rohde & Schwarz*, Apr. 2013.

[18] M. Bolic, M. Rostamian , and P. M. Djuric, "Proximity detection with RFID: a step toward the internet of things," *IEEE Pervasive Computing*, vol. 14, pp. 70–76, Apr. 2015.

[19] A. Headquarters, "Wi-Fi location-based services 4.1 design guide," *Cisco: San Jose, CA, USA*, May. 2008.

[20] B. Singh, S. Pallai , and S. K. Rath, "A Survey of Cellular Positioning Techniques in GSM Networks," in *National Conference in Mobile Computing-NCMC'12*, 2012.

[21] A. Payal, C. Rai , and B. Reddy, "Experimental analysis of some radio propagation models for smart wireless sensor networks applications," in *SAI Intelligent Systems Conference (IntelliSys)*, pp. 338–342, IEEE, Nov. 2015.

[22] T. K. Sarkar, Z. Ji, K. Kim, A. Medouri , and M. Salazar-Palma, "A survey of various propagation models for mobile communication," *IEEE Antennas and propagation Magazine*, vol. 45, pp. 1–82, Jun. 2003.

[23] Y. S. Cho, J. Kim, W. Yang , and C. G. Kang, *MIMO-OFDM wireless communications with MATLAB*. John Wiley & Sons, Aug. 2010.

[24] W. Dargie and C. Poellabauer, *Fundamentals of wireless sensor networks: theory and practice*. John Wiley & Sons, Nov. 2010.

[25] S. Gezici, "A survey on wireless position estimation," *Wireless personal communications*, vol. 44, pp. 263–282, Feb. 2008.

[26] W. C. Lee, *Mobile communications engineering*. McGraw-Hill Professional, May 1982.

[27] D. Dardari, P. Closas, and P. M. Djuri, "Indoor tracking: Theory, methods, and technologies," *IEEE Transactions on Vehicular Technology*, vol. 64, pp. 1263–1278, Apr. 2015.

[28] D. Carrillo, V. Moreno, B. Úbeda, and A. F. Skarmeta, "Magicfinger: 3d magnetic fingerprints for indoor location," *Sensors*, vol. 15, pp. 17168–17194, July 2015.

[29] S. Frattasi and R. F. Della, *Mobile positioning and tracking: from conventional to cooperative techniques*. John Wiley & Sons, Aug. 2017.

[30] B. Lakmali, W. Wijesinghe, K. De Silva, K. Liyanagama, and S. Dias, "Design, implementation & testing of positioning techniques in mobile networks," in *International Conference on Information and Automation for Sustainability (ICIAFS)*, pp. 94–99, IEEE, Dec. 2007.

[31] G. Shravan, L. Jack, C. Romit Roy, C. Landon, and S. Al, "Micro-blog: sharing and querying content through mobile phones and social participation," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, pp. 174–186, ACM, June. 2008.

[32] C. Ning, R. Li , and K. Li, "Outdoor location estimation using received signal strength-based fingerprinting," *Wireless Personal Communications*, vol. 89, pp. 365–384, July 2016.

[33] D. Vo and P. De, "A survey of fingerprint-based outdoor localization," *IEEE Communications Surveys & Tutorials*, vol. 18, pp. 491–506, Jun. 2015.

[34] S. C. Ergen, H. S. Tetikol, M. Kontik, R. Sevlian, R. Rajagopal, and P. Varaiya, "RSSI-fingerprinting-based mobile phone localization with route constraints," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 1, pp. 423–428, 2013.

[35] S. Yiu, M. Dashti, H. Claussen, and F. Perez-Cruz, "Wireless RSSI fingerprinting localization," *Signal Processing*, vol. 131, pp. 235–244, Feb. 2017.

[36] A. Zhang, Y. Yuan, Q. Wu, S. Zhu, and J. Deng, "Wireless localization based on RSSI fingerprint feature vector," *International Journal of Distributed Sensor Networks*, vol. 11, p. 528747, Nov. 2015.

[37] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 66, pp. 763–776, Mar. 2016.

[38] H. Chen, Y. Zhang, W. Li, X. Tao, and P. Zhang, "ConFi: Convolutional neural networks based indoor Wi-Fi localization using channel state information," *IEEE Access*, vol. 5, pp. 18066–18074, Sep. 2017.

[39] G. N. Kamga, M. Xia, Minghua, and S. Aïssa, , "Spectral-efficiency analysis of massive MIMO systems in centralized and distributed schemes," *IEEE Transactions on Communications*, vol. 64, pp. 1930–1941, Jan. 2016.

[40] J. Joung, Y. K. Chia, and S. Sun, "Energy-efficient, large-scale distributed-antenna system (L-DAS) for multiple users," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 954–965, Mar. 2014.

[41] A. Hu, T. Lv, H. Gao, Z. Zhang, and S. Yang, "An ESPRIT-based approach for 2-D localization of incoherently distributed sources in massive MIMO systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 996–1011, Oct. 2014.

[42] S. A. Shaikh and A. M. Tonello, "Localization based on angle of arrival in EM lens-focusing massive MIMO," in *IEEE 6th International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*, pp. 124–128, IEEE, Sep. 2016.

[43] T. Lv, F. Tan, H. Gao, and S. Yang, "A beamspace approach for 2-D localization of incoherently distributed sources in massive MIMO systems," *Signal Processing*, vol. 121, pp. 30–45, Apr. 2016.

[44] N. Garcia, H. Wymeersch, E.G.Larsson, A. M. Haimovich, and M. Coulon, "Direct localization for massive MIMO," *IEEE Transactions on Signal Processing*, vol. 65, pp. 2475–2487, May 2017.

[45] A. Shahmansoori, G. E. Garcia, G. Destino, G. Seco-Granados, and H. Wymeersch, "5G position and orientation estimation through millimeter wave MIMO," in *IEEE Globecom Workshops*, pp. 1–6, IEEE, Dec. 2015.

[46] A. Guerra, F. Guidi, and D. Dardari, "Position and orientation error bound for wideband massive antenna arrays," in *IEEE International Conference on Communication Workshop (ICCW)*, pp. 853–858, IEEE, Jun. 2015.

[47] J. Vieira, E. Leitinger, M. Sarajlic, X. Li, and F. Tufvesson, "Deep convolutional neural networks for massive MIMO fingerprint-based positioning," in *IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–6, IEEE, Oct. 2017.

[48] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *science*, vol. 315, pp. 972–976, Feb 2007.

[49] S. Kumar, R. M. Hegde, and N. Trigoni, "Gaussian process regression for fingerprinting based localization," *Ad Hoc Networks*, vol. 51, pp. 1–10, Nov. 2016.

[50] S. Yiu and K. Yang, "Gaussian process assisted fingerprinting localization," *IEEE Internet of Things Journal*, vol. 3, pp. 683–690, Sep. 2015.

[51] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*, vol. 2. MIT Press Cambridge, MA, 2006.

[52] A. Yang, Y. Jing, C. Xing, Z. Fei, and J. Kuang, "Performance analysis and location optimization for massive MIMO systems with circularly distributed antennas," *IEEE Transactions on Wireless Communications*, vol. 14, pp. 5659–5671, Oct. 2015.

[53] K. S. V. Prasad, E. Hossain, and V. K. Bhargava, "A numerical approximation method for RSS-based user positioning in distributed massive MIMO," in *IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pp. 1–6, IEEE, Dec. 2017.

[54] K. Zheng, S. Ou, and X. Yin, "Massive MIMO channel models: A survey," *International Journal of Antennas and Propagation*, 2014.

[55] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO versus small cells," *IEEE Transactions on Wireless Communications*, vol. 16, pp. 1834–1850, Mar. 2017.

[56] S. Gunnarsson, J. Flordelis, L. Van der Perre, and F. Tufvesson, "Channel hardening in massive MIMO - A measurement based analysis," in *IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, IEEE, Jun. 2018.

[57] E. Gokcay and J. Principe, "Information theoretic clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 158–171, Aug. 2002.

[58] R. Refianti, A. Mutiara, and A. Syamsudduham,, "Performance evaluation of affinity propagation approaches on data clustering," *International Journal of Advanced Computer Science and Applications*, vol. 7, pp. 420–429, Mar. 2016.

[59] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, Nov. 1987.

[60] D. David and B. Donald, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 224–227, Apr. 1979.

[61] R. Calinski and J. Harabasz, "A dendrite method for cluster analysis," *Communications in Statistics-theory and Methods*, vol. 3, pp. 1–27, Jan. 1974.

[62] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*, vol. 112. Springer, 2013.

[63] S. Dhanabal and S. Chandramathi, "A review of various k-nearest neighbor query processing techniques," *International Journal of Computer Applications*, vol. 31, pp. 14–22, Oct. 2011.

[64] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions," *Journal of Mathematical Psychology*, vol. 85, pp. 1–16, Aug. 2018.

[65] C. E. Rasmussen, *Evaluation of Gaussian processes and other methods for non-linear regression*. PhD thesis, University of Toronto, Canada, 1997.

[66] K. S. V. Prasad, E. Hossain, and V. K. Bhargava, "Low-dimensionality of noise-free RSS and its application in distributed massive MIMO," *IEEE Wireless Communications Letters*, vol. 7, no. 4, pp. 486–489, 2017.

[67] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, 1995.

[68] F. Pérez-Cruz, S. Van Vaerenbergh, J. J. Murillo-Fuentes, M. Lázaro-Gredilla, Miguel, and I. Santamaria, "Gaussian processes for nonlinear signal processing: An overview of recent advances," *IEEE Signal Processing Magazine*, vol. 30, pp. 40–50, June 2013.

[69] 3GPP, "Further advancements for e-utra physical layer aspects," *TR 36.814*, Mar. 2010.

[70] S. Saitta, B. Raphael, and I. F. Smith, "A comprehensive validity index for clustering," *Intelligent Data Analysis*, vol. 12, no. 6, pp. 529–548, 2008.

[71] A. Dammann, R. Raulefs, and S. Zhang, "On prospects of positioning in 5G," in *IEEE International Conference on Communication Workshop (ICCW)*, pp. 1207–1213, June. 2015.

[72] R. Mendrzik, F. Meyer, G. Bauch, and M. Win, "Localization, mapping, and synchronization in 5G millimeter wave massive MIMO systems," in *20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, IEEE, July 2019.

[73] F. Wen, H. Wymeersch, B. Peng, W. P. Tay, H. C. So, and D. Yang, "A survey on 5G massive mimo localization," *Digital Signal Processing*, vol. 94, pp. 21–28, Nov. 2019.

[74] H. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Transactions on Communications*, vol. 61, pp. 1436–1449, Feb 2013.

[75] H. Pirzadeh, C. Wang, and H. Papadopoulos, "Machine-learning assisted outdoor localization via sector-based fog massive MIMO," in *IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, May 2019.

[76] V. C. Prakash and G. Nagarajan, "A hybrid RSS-TOA based localization for distributed indoor massive MIMO systems," in *International Conference on Emerging Current Trends in Computing and Expert Technology*, pp. 1359–1370, Springer, Mar. 2019.

[77] B. Berruet, O. Baala, A. Caminada, and V. Guillet, "Delfin: a deep learning based CSI fingerprinting indoor localization in IoT context," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8, IEEE, Sep. 2018.

[78] X. Sun, X. Gao, Y. G. Li, and W. Han, "Fingerprint based single-site localization for massive MIMO-OFDM systems," in *IEEE Global Communications Conference*, pp. 1–7, IEEE, Dec. 2017.

[79] H. S. Obaid, S. A. Dheyab, and S.S. Sabry, "The impact of data pre-processing techniques and dimensionality reduction on the accuracy of machine learning," in *Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)*, (Jaipur), pp. 279–283, IEEE, Mar. 2019.

[80] I. B. Mohamad and D. Usman, "Standardization and its effects on k-means clustering algorithm," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 6, pp. 3299–3303, Sep 2013.

[81] M. Alkhayrat, M. Aljnidi and K. Aljoumaa, "A comparative dimensionality reduction study in telecom customer segmentation using deep learning and PCA," *Journal of Big Data*, vol. 7, p. 9, Dec. 2020.

[82] I. Jolliffe, *Principal Component Analysis, Second Edition*. 2nd ed. Springer series in statistics, 2002.

[83] L. V. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, Nov. 2008.

[84] F. C. Kingman and S. Kullback, "Information theory and statistic," *Neurocomputing*, vol. 54, Jul. 2007.

[85] A. Gisbrecht, A. Schulz, and B. Hammer, "Parametric nonlinear dimensionality reduction using kernel t-SNE," *Neurocomputing*, vol. 147, pp. 71–82, Jan. 2015.

[86] Z. Zhang, "Improved Adam optimizer for deep neural networks," in *26th International Symposium on Quality of Service (IWQoS)*, pp. 1–2, IEEE, 2018.

[87] M. Soltanolkotabi, "Learning ReLUs via gradient descent," in *Advances in neural information processing systems*, pp. 2007–2017, 2017.

[88] N. Miyazaki, S. Nanba, and S. Konishi, "MIMO-OFDM throughput performances on MIMO antenna configurations using LTE-based testbed with 100 MHz bandwidth," in

72nd Vehicular Technology Conference-Fall (VTC-fall), (Ottawa, Canada), pp. 1–5, IEEE, Sep. 2010.

[89] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.

[90] J. Joung, Y. K. Chia and S. Sun, "Energy-efficient, large-scale distributed-antenna system (L-DAS) for multiple users," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, pp. 954–965, Mar. 2014.

[91] S. S. Moosavi and P. Fortier, "Fingerprinting localization method based on clustering and Gaussian process regression in distributed massive MIMO systems," in *International Symposium on Personal, Indoor and Mobile Radio Communications*, IEEE, Aug. 2020.

[92] T. Wei, A. Zhou and X. Zhang, "Facilitating robust 60 GHz network deployment by sensing ambient reflectors," in *14th USENIX Symposium on Networked Systems Design and Implementation NSDI 17*, pp. 213–226, 2017.

[93] K. S. Prasad, E. Hossain, V. K. Bhargava and S. Mallick, "Analytical approximation-based machine learning methods for user positioning in distributed massive MIMO," *IEEE Access*, vol. 6, pp. 18431–18452, Feb. 2018.

[94] J. Larsson, "Distance estimation and positioning based on Bluetooth low energy technology," Master's thesis, KTH Royal Institute of Technology, 2015.

[95] C. Gao, G. Zhao and H. Fourati, *Cooperative Localization and Navigation: Theory, Research and Practice*. CRC Press, 1st ed., 2019.

[96] Y. Yang, P. Dai, H. Huang, M. Wang, and Y. Kuang, "A semi-simulated RSS fingerprint construction for indoor Wi-Fi positioning," *Electronics*, vol. 9, p. 1568, Oct. 2020.

[97] Z. Wang, Z. Wang, L. Fan, and Z. Yu, "A hybrid Wi-Fi fingerprint-based localization scheme achieved by combining Fisher Score and stacked sparse autoencoder algorithms," *Mobile Information Systems*, vol. 2020, Apr. 2020.

[98] J. Bai, Y. Sun, W. Meng, and C. Li, "Wi-Fi fingerprint-based indoor mobile user localization using deep learning," *Wireless Communications and Mobile Computing*, vol. 2021, Jan. 2021.