REPRESENTATION LEARNING BASED DOWNLINK CHANNEL COVARIANCE MATRIX ESTIMATION FOR FDD MASSIVE MIMO

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MELİH CAN ZERİN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

SEPTEMBER 2024

Approval of the thesis:

REPRESENTATION LEARNING BASED DOWNLINK CHANNEL COVARIANCE MATRIX ESTIMATION FOR FDD MASSIVE MIMO

submitted by **MELİH CAN ZERİN** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İlkay Ulusoy
Head of Department, **Electrical and Electronics Engineering** _____

Prof. Dr. Ali Özgür Yılmaz
Supervisor, **Electrical and Electronics Engineering, METU** _____

Assoc. Prof. Dr. Elif Vural
Co-supervisor, **Electrical and Electronics Engineering, METU** _____

Examining Committee Members:

Assoc. Prof. Dr. Ayşe Melda Yüksel Turgut
Electrical and Electronics Engineering, METU _____

Prof. Dr. Ali Özgür Yılmaz
Electrical and Electronics Engineering, METU _____

Assist. Prof. Dr. Özlem Tuğfe Demir
Electrical and Electronics Engineering, TOBB ETU _____

Date:03.09.2024

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Melih Can Zerin

Signature :

ABSTRACT

REPRESENTATION LEARNING BASED DOWNLINK CHANNEL COVARIANCE MATRIX ESTIMATION FOR FDD MASSIVE MIMO

Zerin, Melih Can
M.S., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Ali Özgür Yılmaz
Co-Supervisor: Assoc. Prof. Dr. Elif Vural

September 2024, 64 pages

In this thesis, we propose an algorithm for downlink (DL) channel covariance matrix (CCM) estimation for frequency division duplexing (FDD) massive multiple-input multiple-output (MIMO) communication systems with base station (BS) having a uniform linear array (ULA) antenna structure. We make use of the inherent similarity between the uplink (UL) CCM and the DL CCM due to angular reciprocity. We consider a setting where the UL CCM is mapped to the DL CCM by a mapping function. We first present a theoretical error analysis of learning a nonlinear embedding by constructing a mapping function, which points to the importance of the Lipschitz regularity of the mapping function for achieving high estimation performance. Then, based on the theoretical ground, we propose a representation learning algorithm as a solution for the estimation problem, where Gaussian RBF kernel interpolators are chosen to map UL CCMs to their DL counterparts. The proposed algorithm is based on the optimization of an objective function that fits a regression model between the DL CCM and UL CCM samples in the training dataset and preserves the local geometric structure of the data in the UL CCM space, while explicitly regulating the

Lipschitz continuity of the mapping function in light of our theoretical findings. The proposed algorithm surpasses benchmark methods in terms of three error metrics as shown by simulations.

Keywords: Channel Covariance Matrix, Massive MIMO, FDD, Gaussian RBF Interpolation, Representation Learning

ÖZ

FDD KİTLESEL MIMO İÇİN GÖSTERİM ÖĞRENİMİ TABANLI AŞAĞI BAĞLANTILI KANAL KOVARYANS MATRİSİ KESTİRİMİ

Zerin, Melih Can
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Ali Özgür Yılmaz
Ortak Tez Yöneticisi: Doç. Dr. Elif Vural

Eylül 2024 , 64 sayfa

Bu tezde, baz istasyonunun (BS) düzgün doğrusal dizi (ULA) anten yapısına sahip olduğu frekans bölmeli çift yönlü (FDD) kitlesel çok girdili çok çıktılı (MIMO) haberleşme sistemleri için aşağı bağlantılı (DL) kanal kovaryans matrisi (CCM) kestirimi için bir algoritma önerilmektedir. Açısal karşılıklılık nedeniyle yukarı bağlantılı (UL) CCM ile DL CCM arasındaki doğal benzerlikten faydalanılmıştır. UL CCM'nin bir eşleme fonksiyonu ile DL CCM'ye eşlendiği bir ortam ele alınmıştır. Öncelikle, yüksek tahmin performansı elde etmek için eşleme fonksiyonunun Lipschitz düzenliliğinin önemine dikkat çeken bir eşleme fonksiyonu oluşturarak doğrusal olmayan bir gömülümü öğrenmenin teorik hata analizi sunulmuştur. Daha sonra, kestirim probleminin çözümü için, bu teorik temele dayanan bir gösterim öğrenme algoritması önerilmektedir. Bu algoritmada, UL CCM'leri DL karşılıklarına eşlemek için Gaussian RBF kernel interpolasyon fonksiyonları eşleme fonksiyonu olarak seçilmiştir. Önerilen algoritma, eğitim veri setindeki DL CCM ve UL CCM örnekleri arasında bir regresyon modeline uyan ve UL CCM uzayındaki verilerin yerel geometrik yapısını

koruyan, aynı zamanda eşleme fonksiyonunun Lipschitz sürekliliğini teorik bulgularımız ışığında açık bir şekilde düzenleyen bir amaç fonksiyonun optimizasyonuna dayanmaktadır. Önerilen algoritma, simülasyonların gösterdiği üzere üç hata metriği açısından literatürdeki yöntemleri geride bırakmaktadır.

Anahtar Kelimeler: Kanal Kovaryans Matrisi, Kitlesel MIMO, FDD, Gaussian RBF İnterpolasyonu, Gösterim Öğrenimi

To my dear family and loved ones

ACKNOWLEDGMENTS

I would like to thank my supervisor Prof. Dr. Ali Özgür Yılmaz and my co-supervisor Assoc. Prof. Dr. Elif Vural for their invaluable guidance throughout this research.

I would like to thank Assist. Prof. Dr. Gökhan Muzaffer Güvensen and Res. Asst. Anıl Kurt for always allocating time for my questions and meaningful discussions.

I would like to thank my family and friends for always supporting me and believing in me.

The help of Dr. -Ing. Bitan Banerjee during the implementation of the CGAN-based benchmark method in [1] is acknowledged.

This work was supported by Turkcell Technology under 5G and Beyond Joint Graduate Support Program run by the Information and Communication Technologies Authority of Türkiye.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

LIST OF ABBREVIATIONS

ABBREVIATIONS

ACF	Autocorrelation Function
AoA	Angle of Arrival
AS	Angular Spread
BS	Base Station
CCM	Channel Covariance Matrix
CDF	Cumulative Distribution Function
CMD	Correlation Matrix Distance
CNN	Convolutional Neural Network
CSI	Channel State Information
DL	Downlink
DM	Deviation Metric
DNN	Deep Neural Network
DSC	Distributed Source Coding
DTL	Deep Transfer Learning
EVD	Eigenvalue Decomposition
FDD	Frequency Division Duplexing
i.i.d.	Independent Identically Distributed
MIMO	Multiple Input Multiple Output
ML	Machine Learning
MMSE	Minimum Mean Squared Error
mmWave	Millimeter Wave
MSE	Mean Squared Error
NMSE	Normalized Mean Square Error

NN	Neural Network
PAS	Power Angular Spectrum
PSD	Positive Semidefinite
RBF	Radial Basis Function
SINR	Signal-to-Interference-plus-Noise Ratio
SLNR	Signal-to-Leakage-and-Noise Ratio
SNR	Signal-to-Noise Ratio
TDD	Time Division Duplexing
UE	User Equipment
UL	Uplink
ULA	Uniform Linear Array
URA	Uniform Rectangular Array

CHAPTER 1

INTRODUCTION

Massive MIMO is a favorable technology for 5G and beyond networks in terms of achieving high spectral efficiency and energy efficiency [2]. In this technology, the base station (BS) has a high number of antennas, which is much more than the number of active user terminals [3]. The conventional operation mode for massive MIMO is taken as time division duplexing (TDD) due to channel reciprocity [4]. Sharing the same wireless medium and frequency band, uplink and downlink channels are said to be reciprocal in TDD systems [5], which means that learning the uplink channel state information (CSI), the base station can infer the downlink CSI as well, and does not require any additional pilot training for downlink channel estimation. On the other hand, it is of great importance to implement massive MIMO on FDD systems due to the fact that most of the wireless networks operate on FDD mode, meaning that the infrastructure is ready for its implementation [1, 6]. Also, FDD operation results in higher data rates and greater coverage than TDD mode does [1, 6].

However, there is a drawback of this technology for frequency division duplexing (FDD) systems: the excessive impractical pilot and feedback overhead [4, 7]. The reciprocity of uplink and downlink channels does not hold for FDD systems, because even though they share the same wireless medium, they operate using different carrier frequencies [8]. Since the uplink and downlink channels are not reciprocal in FDD systems, the channel estimation process consumes too much resource for pilot and feedback symbols due to the high number of antennas at the base station [4].

One solution to loosen the pilot and feedback overhead is to use the DL CCM instead of the DL CSI [7]. The channel covariance matrix provides second order channel statistics, which may be useful for channel estimation and beamforming [9]. There-

fore, it is an important parameter to know for the implementation of massive MIMO in FDD systems and knowing it as accurately as possible is highly important. In literature, there are numerous studies where the DL CCM is estimated by using the UL CCM, as in [1, 10–15]. The motivation behind this choice is the following: Even though there is no channel reciprocity between the uplink and downlink channels, there is spatial reciprocity between them [16]. Therefore, their power distribution in the angular domain, i.e., their power angular spectrum (PAS), is commonly taken as the same. Thus, one can say that the UL CCM is quite informative about the DL CCM. Many of the fundamental approaches for the solution of this problem benefits from this property.

Some of the earlier works propose simple signal processing methods for the DL CCM estimation problem, such as [10, 11, 15]. On the other hand, much more complex tools like deep learning can also be incorporated into the solution of the UL-to-DL CCM transformation problem, as in [1]. Signal processing-based methods may experience performance degradation in practical cases where the UL CCM is not perfectly known. Their performance heavily depends on the availability of an accurate UL CCM and error propagation may occur in the cases where these methods are used to predict the DL CCM from noisy UL CCM estimates. Deep learning solutions are more robust to noise compared to classical methods due to the process of learning from the data. Nonetheless, in order to learn an accurate deep learning model with good generalization ability, one needs to use excessive amounts of data. Especially, as the number of base station antennas increases, the size of the matrix to be learned will increase, which increases the need for more training data.

In this thesis, for the DL CCM estimation problem, we propose to learn a nonlinear interpolation function which maps an arbitrary user's UL CCM to its DL CCM. In view of the above discussions, our idea is to seek a trade-off between simple signal processing-based methods and the complex deep learning solutions. We thus propose to learn a nonlinear interpolator that possesses the rich representation power of nonlinear methods with successful generalization capabilities, while involving fewer parameters to be optimized compared to neural networks in order to require much less training data. To the best of our knowledge, the estimation of DL CCMs from their UL counterparts via nonlinear interpolators has not yet been studied thoroughly

in the current literature, due to which we aim to address both the theoretical and methodological aspects of this problem.

Inspired by the nonlinear supervised manifold learning approach in [17] that pays particular attention to the generalization of the learned embedding to previously unseen data, in this thesis, we propose to solve the DL CCM estimation problem with representation learning with graph regularization. Representation learning is an established and mature machine learning method dealing with learning the inherent structure of data [18]. Learning representations from data simplifies obtaining the information useful for building an estimator [19]. In [17], the authors give a theoretical performance analysis of a classification algorithm that jointly learns an embedding and an interpolation function that generalizes the embedding to previously unseen data. Although the method in [17] addresses a quite different application, i.e., data classification, some of the findings in [17] provide useful insights for the regression problem we consider in this thesis. In particular, the principle of the preservation of the local geometry, which is employed in [12] as well, suggests that making the neighborhood relations between the points in the UL CCM space match those in the DL CCM space may have a positive impact on the estimation performance.

We first present a detailed theoretical analysis in order to explore the effect of the preservation of local geometry in the context of our regression problem along with the impact of the properties of the interpolation function used. Our theoretical results show that we should learn an embedding for DL CCMs from UL CCMs considering the following properties: (1) The local neighborhoods of the points in the UL CCM space should be preserved in the DL CCM space. (2) The embeddings of the UL CCMs in the training dataset should be close to the corresponding ground truth DL CCMs. (3) The interpolation function that maps UL CCMs to DL CCM estimates should have a low Lipschitz constant so that the function generalizes successfully to new UL CCMs that were not available during training.

Next, in the light of our theoretical findings, we propose an objective function, where a term related to the preservation of the local neighborhood structure and two terms related to the Lipschitz constant of the interpolator are optimized together with a data fitting term. We choose Gaussian radial basis function (RBF) kernels for our inter-

polator, which provides a smooth interpolation of training data points by preventing sudden changes in the embedding space, i.e., overfitting, thanks to the Lipschitz regularity of the Gaussian kernel. We use an alternating optimization method to minimize the objective function in an iterative fashion, in order to jointly learn the embedding and the parameters of the RBF interpolation function.

In this thesis, our main contributions to the field of DL CCM estimation from UL CCM are the following:

- We first present a theoretical analysis of learning interpolation functions that map UL CCMs to their DL counterparts, with the purpose of identifying the main factors that affect the estimation error of the DL CCM. Our analysis shows that the error is essentially influenced by: (i) the average estimation error of the nearest neighbors of the point in the training dataset, (ii) the Lipschitz constant of the interpolation function, and (iii) the maximum value of the ratio of the distance between two DL CCMs to the distance between their UL counterparts.
- We next propose a novel representation learning method for DL CCM estimation, which builds on our theoretical results and relies on a model with much fewer parameters compared to other methods such as deep-learning based ones. The proposed method thus achieves considerably higher estimation performance in settings with limited availability of training data. Meanwhile, the nonlinear structure of the learnt model allows for successfully capturing the particular geometry of the data, making it favorable against simpler solutions such as linear transformations.

This thesis is organized as follows: Chapter 2 summarizes the significant earlier works in the literature for the UL-DL CCM conversion problem. In Chapter 3, the system model for the communication scenario is explained and the theoretical motivation behind our method is presented. A representation learning method for the problem of DL CCM estimation from UL CCMs is proposed in Chapter 4. In Chapter 5, the performance of the proposed algorithm is compared to benchmark methods via simulations in terms of several performance metrics, and a stability and sensitivity analysis is presented for the proposed algorithm. Finally, the concluding remarks are given in Chapter 6.

CHAPTER 2

RELATED WORK

In this chapter, we present some general background information about the methods offered in the literature to solve the implementation problems of massive MIMO in FDD systems. These methods are divided into two groups, and each section presents one group of the methods.

In Section 2.1, different methods proposed to help lighten the burden of FDD massive MIMO systems are explained, which generally target CSI estimation, efficient pilot and/or feedback design, precoder/beamformer design etc.. Then, in Section 2.2, uplink-to-downlink channel covariance transformation methods are described.

2.1 Different Methods for the Implementation of Massive MIMO in FDD Systems

In this section, various methods in the literature are summarized, which are developed for solving the problem of impractical pilot and feedback overhead of FDD massive MIMO.

Some works suggest DL CSI estimation methods as in [20–22]. There are also methods that design feedback signals in an efficient way, some of which are [23–25]. Some studies make joint designs such as [7, 26, 27]. [28] makes DL CCM estimation, but instead of using the corresponding UL CCM, the pilot weighting matrix is optimized.

In [20], an uplink-to-downlink CSI mapping is conducted assuming that the user position-to-channel mapping is bijective. With this assumption, one can express the uplink-to-downlink channel mapping function as a composite of the function that

maps the uplink channel to the user position and the function that maps the user position to the downlink channel. Motivated by a theoretical analysis, a complex-valued deep neural network architecture is proposed to approximate the composite function mentioned above.

In [21], a neural network-based channel estimator is proposed, where the network is trained using solely the UL CSIs, but is used for both uplink and downlink CSI prediction. This setup is proposed based on the conjecture named "UL-DL distributional reciprocity". Basically, the main claim is that the small difference between the uplink and the downlink carrier frequencies can be treated as phase differences. If enough UL CSI data are collected, their distribution can be approximated as the distribution of the DL CSI in the same environment by considering each UL CSI as a DL CSI originating from a user position different than that of the UL CSI. Based on this idea, a convolutional neural network (CNN) is trained, whose input and output are compressed UL CSIs and whole UL CSIs, respectively. Then, the trained network is used to find the DL CSI of a user from the compressed version of its DL CSI. The advantages of this setup are that no DL CSI is needed to train the network and that a compressed version of the DL CSI is sufficient to estimate the whole DL CSI.

In [22], a neural network architecture is trained for DL CSI estimation and DL beamforming by extracting the joint long term properties of a wireless channel that is shared by both the UL and the DL channels due to the "partial reciprocity" of the UL and the DL channels.

In [23], a deep transfer learning (DTL) algorithm is proposed to obtain the DL CSI feedback. First, a CNN is trained by using a high number of data in one wireless channel environment. Then, this pre-trained model is adapted to different wireless channel environments by DTL via fine-tuning with little data obtained from these environments. This way, a feedback scheme is obtained with low training cost.

In [24], a deep neural network (DNN)-based feedback method is proposed. DL CSI is first sampled in time and frequency domains. Then, the sampled CSI is compressed in the space domain and fed back to the BS. At the BS, the received "compressive sampled CSI" is first decompressed by a network. Next, the decompressed CSI is interpolated to return to the original dimension of the DL CSI. Finally, a refinement

is applied via another neural network (NN) to obtain the final estimation of the DL CSI.

A machine learning (ML)-based variable length feedback design for DL CSI is proposed in [25]. First, the DL CSI is projected onto a lower dimensional latent space, which is found via the truncated version of the unitary matrix obtained from the eigenvalue decomposition (EVD) of the sample covariance matrix of the channel matrix. Then, different number of bits are allocated to each principal component. Finally, the quantization of each principal component is conducted via k-means clustering.

[26] proposes a method named "Joint Spatial Division and Multiplexing (JSDM)". In this study, a MIMO scenario with multiple single antenna-users is considered. First, the users are divided into groups based on their channel covariance matrices so that the users in the same group are close in terms of their angle of arrivals (AoAs). Next, a two-stage precoding design is conducted. First, one common "prebeamforming matrix" per each group is designed based on the channel covariance matrix that is assumed to be the same for all users in a group. Using the prebeamforming matrix, users' instantaneous channel realizations are reduced in dimensions. The second stage precoder matrix is designed based on the effective instantaneous channel obtained after the prebeamforming stage for the users to be served, which are selected via scheduling. In [7], a similar joint design setup is studied, where different user grouping (clustering) methods and several novel similarity measures are proposed. Also, the balance between different user groups is considered via a joint load balancing and precoding design algorithm.

In [27], a joint pilot, feedback and precoder design is proposed. In this study, rather than estimating the CSI and sending feedback to the BS for each user independently, a joint channel estimation, feedback and precoding matrix design is developed for multiple users. The main idea forming this design is to consider the multiuser channel estimation and the feedback problem as a distributed source coding (DSC) problem. The channel estimation and the quantization phases take place at the users in a distibuted fashion, while the feedback bits of all users are then processed at the BS jointly for precoding design. A deep learning model is designed for the implementation of the DSC-based idea, where each user and the BS has a DNN structure. The DNNs

at the users map the pilot signals received into the feedback bits, while the DNN at the BS maps the collected feedback bits coming from all the users into the precoder matrix.

In [28], authors design an algorithm to find a pilot weighting matrix to shrink the feasible set of DL CCMs and find the center of the set in an FDD massive MIMO system with limited feedback and Type I codebook.

2.2 UL-to-DL CCM Transformation

Estimation of the DL CCM from the UL CCM is a well-studied problem with numerous solutions proposed in the literature. The UL-to-DL CCM conversion methods are here divided into two categories based on the PAS estimation: indirect mehods and direct methods.

2.2.1 Indirect Methods

Numerous studies in the literature benefit from the angular reciprocity concept between the uplink and the downlink channels by estimating the PAS from the UL CCM and use this estimate as the DL CCM's PAS in order to find the DL CCM estimate. [9, 13, 14] and [29] present methods, which find a discrete approximation of the PAS from the UL CCM and use it in DL CCM. The methods in [9, 13] estimate the power distribution in certain angles, which can be seen as sampling the PAS at those angles. [14] and [29] express the UL CCM by a system of equations in order to estimate the PAS and find the corresponding DL CCM.

In [9], the CCM is written approximately as a weighted summation of non-orthogonal basis vectors. The array response vector of the DL channel at an arbitrary angle is expressed as a linear transformation of that of the UL channel, and it is combined with the approximate discretized CCM formula to relate the DL CCM to UL CCM. Then, the discrete PAS is approximated by using the instantaneous UL channel gains at certain angles, and it is used in the discretized DL CCM formula to estimate the DL CCM. In [13], the PAS is estimated in a uniform grid consisting of discrete angle

points, whose number is much higher than the number of BS antennas. Dealing with Toeplitz CCMs, the first row of the UL CCM is enough to find the whole matrix. Therefore, the norm of the difference between the first row of the UL CCM estimate at hand and its approximation via the discrete PAS is minimized to find the PAS at the grid points. Using the PAS estimate and Fourier transform resampling, the first row of the DL CCM is found, which fully describes the whole matrix.

In [14] and [29], the main idea is to express the CCM as a system of linear equations by inner products in an infinite dimensional Hilbert space. Writing the UL CCM this way, the PAS estimate is found. Then, it is used in the system of linear equations representing the DL CCM to find an estimate of the DL CCM.

2.2.2 Direct Methods

One can also find the DL CCM from its UL counterpart directly. The following studies suggest methods that map the UL CCM directly to the DL CCM without the intermediate step of finding the shared PAS: [1, 10–12, 15].

In [10], a frequency calibration matrix is suggested to convert the UL CCM to its DL counterpart by taking the carrier frequency gap between UL and DL into account. The PAS is approximated by first expressing it as a periodic function and then truncating its Fourier series expansion. In this way, UL CCM and DL CCM can be written of the matrix-vector multiplication form. Due to the shared PAS, a linear transformation matrix can be obtained to map the UL CCM to the DL CCM, namely "frequency calibration matrix".

[11] suggests a cubic splines method is suggested to interpolate the magnitude and the phase of the DL CCM's elements from the corresponding UL CCM's elements. First, a spatio-temporal function is introduced, which is obtained from the CCM. Next, it is separated into its absolute value and phase via the Bessel function of first kind and of order zero. Then, the absolute value and the unwrapped phase of the spatio-temporal function are resampled by a factor of the DL/UL carrier frequency ratio via splines method. In this way, DL CCM is obtained.

[12] proposes to create a dictionary of UL/DL CCM pairs. This work points out

the similarity between the neighborhood relationships in the UL CCM and the DL CCM spaces. When a new user with an unknown DL CCM comes, one can fit its UL CCM to the barycenter of the UL CCMs in the dictionary and find nonnegative weights for all the UL CCMs in the dictionary to represent the new UL CCM. Then, using the same weights for each DL CCM in the dictionary as the weights of their UL counterparts in the dictionary, one can find the DL CCM estimate of the new user point via its UL CCM.

In [1], the image-to-image translation method named "pix2pix" presented in [30] is adopted to the UL-DL CCM conversion problem. The CCMs are represented as RGB images, where the real parts of the matrices are placed in the "R" channels of the images, the imaginary parts of the matrices are put into the "G" channels of the images, and the "B" channels are fixed to a constant since they are not used. Then, training a conditional generative adversarial network (CGAN), a fake image generator is trained so that it can find the corresponding DL CCM of a new UL CCM.

[15] proposes linear transformations for the solution of the UL/DL CCM conversion problem for systems with base station having either uniform linear array (ULA) or uniform rectangular array (URA) antenna structure. The fundamental idea behind this solution is the following: The elements of the UL CCM are seen as some sort of a nonlinear transformation of the PAS. Then, a continuous approximation of the PAS is found. This approximate PAS is used to find the closed form estimation of the DL CCM, which is a linear transformation of the UL CCM [15].

In this thesis, we develop a method falling into the latter category. Previous works in the literature using machine learning methods for the UL/DL CCM conversion problem generally adopt deep learning as their means of solution. Although deep learning seems promising for capturing complex input-output relationships learning highly complex models, their success generally depends on the access to massive datasets. On the other hand, our method learns a simple nonlinear function with much less parameters needed during neural network training while grasping the geometry of the output space sufficiently well.

CHAPTER 3

PERFORMANCE BOUNDS FOR DL CCM ESTIMATION VIA GAUSSIAN RBF KERNELS

In this chapter, first, the system model of the communication scenario is described. Next, the representation learning setting considered for the problem of DL CCM estimation from UL CCM is presented. Then, an upper bound on the error of an arbitrary test sample is provided.

3.1 System Model

We consider an FDD single cell massive MIMO system, in which a base station (BS) containing M antennas forming a uniform linear array (ULA) serves single-antenna user equipments (UE). The UL channel operates at the carrier frequency of f_{UL}, and the DL channel operates at the carrier frequency of f_{DL}. Their respective wavelengths are denoted as λ_{UL} and λ_{DL}. The ratio of carrier frequencies is denoted by $f_R = \frac{f_{DL}}{f_{UL}} = \frac{\lambda_{UL}}{\lambda_{DL}}$. The UL and the DL channels are considered to be frequency-flat.

The UL and the DL channel vectors ($\mathbf{h}_{UL}$ and $\mathbf{h}_{DL}$, respectively) are formulated as follows [1], [15]:

$$\mathbf{h}_x = \int_{\bar{v}-\Delta}^{\bar{v}+\Delta} \gamma_x(\phi)\mathbf{a}_x(\phi)\, d\phi, x \in \{UL, DL\}, \tag{3.1}$$

where $\gamma_x(\phi)$ is the complex channel gain corresponding to the AoA ϕ, $\bar{v}$ is the mean AoA, Δ is the spread of AoAs and $\mathbf{a}_x(\phi)$ is the array response vector at the angle ϕ. The array response vectors of the UL and the DL channels ($\mathbf{a}_{UL}(\phi)$ and $\mathbf{a}_{DL}(\phi)$,

respectively) are given by:

$$\mathbf{a}_x(\phi) = [1\ e^{j2\pi\frac{d}{\lambda_x}\sin\phi}\ ...\ e^{j2\pi\frac{d}{\lambda_x}(M-1)\sin\phi}]^T, x \in \{UL, DL\}, \tag{3.2}$$

where $d = \frac{\lambda_{UL}}{2}$.

We consider the wide sense stationary uncorrelated scattering (WSSUS) model for our communication scenario as in [15]. In this model, the autocorrelation function (ACF) of the channel gain is time-invariant, and scattering at different AoAs are uncorrelated. Considering the UL and the DL channels as zero mean channels, the UL CCM and the DL CCM ($\mathbf{R}_{UL}$ and $\mathbf{R}_{DL}$, respectively) can then be formulated as [15]:

$$\begin{aligned}\mathbf{R}_x = \mathbb{E}\left\{(\mathbf{h}_x - \mathbb{E}\{\mathbf{h}_x\})(\mathbf{h}_x - \mathbb{E}\{\mathbf{h}_x\})^H\right\} &= \mathbb{E}\{\mathbf{h}_x\mathbf{h}_x^H\} \\ &= \int_{\bar{v}-\Delta}^{\bar{v}+\Delta} p(\phi)\mathbf{a}_x(\phi)\mathbf{a}_x^H(\phi)\,d\phi, x \in \{UL, DL\},\end{aligned} \tag{3.3}$$

where $p(\phi)$ is the power angular spectrum (PAS). The PAS is the same for uplink and downlink, and it is normalized to 1, i.e., $\int_{\bar{v}-\Delta}^{\bar{v}+\Delta} p(\phi)d\phi = 1$. From (3.3) and (3.2), one can conclude that CCMs are Hermitian, i.e., $\mathbf{R}_x = \mathbf{R}_x^H$, for $x \in \{UL, DL\}$. The ULA antenna structure and the WSSUS model lead CCMs to be Toeplitz. Due to its Hermitian and Toeplitz structure, the $\mathbf{R}_x$ matrix given in (3.3) can be represented by its first row.

3.2 Notation and Setting

Let $\{\mathbf{r}_{UL}{}^i, \mathbf{r}_{DL}{}^i\}_{i=1}^N$ be a training dataset with N training UL/DL CCM sample pairs, where $\mathbf{r}_x{}^i \in \mathbb{R}^{1\times 2M-1}$ is a row vector obtained by the concatenation of the real and imaginary parts of the first row vector of the i^{th} CCM in the training dataset for $x \in \{UL, DL\}$. The first element of the first row of a CCM is always real, so it has no imaginary part. Hence, the vectors in the dataset are of length $2M-1$. Let the UL data samples be drawn i.i.d. from a probability measure υ on $\mathbb{R}^{1\times 2M-1}$. The training samples are embedded into $\mathbb{R}^{1\times 2M-1}$ such that each training sample $\mathbf{r}_{UL}{}^i$ is mapped to a vector $\hat{\mathbf{r}}_{DL}^i \in \mathbb{R}^{1\times 2M-1}$.The mapping is assumed to be extended to the

whole data space through an interpolation function $f : \mathbb{R}^{1\times 2M-1} \rightarrow \mathbb{R}^{1\times 2M-1}$ such that each training sample is mapped to its embedding as $f(\mathbf{r}_{UL}{}^{i}) = \hat{\mathbf{r}}^{i}_{DL}$. Let $\mathbf{r}_{UL}{}^{test}$ be the concatenated vector of an arbitrary UL CCM test point and $B_\delta(\mathbf{r}_{UL}{}^{test})$ be an open ball of radius δ around it:

$$B_\delta(\mathbf{r}_{UL}{}^{test}) := \left\{\mathbf{r}_{UL} \in \mathbb{R}^{2M-1} : \left\|\mathbf{r}_{UL}{}^{test} - \mathbf{r}_{UL}\right\| < \delta\right\}. \tag{3.4}$$

Let A^{UL} be the set of the training samples within a δ-neighborhood of $\mathbf{r}_{UL}{}^{test}$ in $\mathbb{R}^{1\times 2M-1}$

$$A^{UL} := \left\{\mathbf{r}_{UL}{}^{i} : \mathbf{r}_{UL}{}^{i} \in B_\delta\left(\mathbf{r}_{UL}{}^{test}\right)\right\}. \tag{3.5}$$

Denoting the support of the probability measure υ as $\mathcal{M} \subset \mathbb{R}^{1\times 2M-1}$, we define

$$\eta_\delta := \inf_{\mathbf{r}_{UL}{}^{test}\in\mathcal{M}} \upsilon\left(B_\delta\left(\mathbf{r}_{UL}{}^{test}\right)\right) \tag{3.6}$$

which is a lower bound on the measure of the open ball $B_\delta\left(\mathbf{r}_{UL}{}^{test}\right)$ around any test point.

3.3 Theoretical Analysis for Motivation Behind the Proposed Method

We now present a theoretical analysis of the regression problem of UL-to-DL CCM conversion via a mapping function $f(\cdot)$. We consider a setting with the following assumptions:

1. The function $f : \mathbb{R}^{1\times 2M-1} \rightarrow \mathbb{R}^{1\times 2M-1}$ is Lipschitz continuous with constant L; i.e., for any $\mathbf{r}_1, \mathbf{r}_2 \in \mathbb{R}^{1\times 2M-1}$, $\|f(\mathbf{r}_1) - f(\mathbf{r}_2)\| \leq L\|\mathbf{r}_1 - \mathbf{r}_2\|$.
2. The probability measure υ has a bounded support $\mathcal{M} \subset \mathbb{R}^{1\times 2M-1}$.
3. For any $\delta > 0$, the probability measure lower bound η_δ is strictly positive, i.e., $\eta_\delta > 0$.

We examine the relation between the local geometries of the UL CCM and the DL CCM spaces with the following lemma:

Lemma 1. *Let $\boldsymbol{p}_{UL}{}^{i}$ and $\boldsymbol{p}_{UL}{}^{j}$ be two different points in $\mathbb{R}^{1\times 2M-1}$ drawn i.i.d. from υ, and let $\boldsymbol{p}_{DL}{}^{i}$ and $\boldsymbol{p}_{DL}{}^{j}$ be their DL counterparts. If $\|\boldsymbol{p}_{UL}{}^{i} - \boldsymbol{p}_{UL}{}^{j}\| \leq 2\delta$, then,*

there exists a constant $K > 0$ *such that* $\|\boldsymbol{p}_{DL}{}^i - \boldsymbol{p}_{DL}{}^j\| \leq K \|\boldsymbol{p}_{UL}{}^i - \boldsymbol{p}_{UL}{}^j\| \leq 2K\delta$*, under the following assumptions:*

- *The PAS,* $p(\phi)$*, is uniform.*
- δ *is so small that two points in a* δ *ball of a test point, say point* i *and point* j*, have very close mean AoA values, i.e.,* $\bar{v}_i - \bar{v}_j \approx 0$*.*
- *The spread of AoA,* Δ*, of each data point in the dataset is constant and the same.*

Proof of Lemma 1 is given in Appendix A.

Remark 1. Motivated by Lemma 1, for the given special case where the PAS is uniform and the angular spread (AS) of each user in a dataset is the same, one can say that if two points are close to each other in the UL CCM space, they should be close to each other in the DL CCM space as well. In practice, the constant K takes values close to f_R in realistic settings. We demonstrate this with a numerical analysis in Appendix D. Overall, Lemma 1 provides useful insight for settings where a mapping function is to be learned between the spaces of UL CCMs and DL CCMs.

For a sufficiently large dataset, i.e., for a sufficiently high N value, the distance between a point in the dataset and its nearest neighbors becomes considerably small, so that one can think of the ball radius parameter δ as a small constant. In Theorem 1, we consider such a setting and provide an upper bound on the test error of the estimate of an arbitrary test point obtained via the interpolation function $f(\cdot)$.

Theorem 1. *Let the training sample set contain at least* N *training samples* $\{\boldsymbol{r}_{UL}{}^i\}_{i=1}^N$ *with* $\boldsymbol{r}_{UL}{}^i \sim \upsilon$*. Let* $\boldsymbol{r}_{UL}{}^{test}$ *be a test sample drawn from* υ *independently of the training samples. Assume that the interpolation function* $f : \mathbb{R}^{1\times 2M-1} \to \mathbb{R}^{1\times 2M-1}$ *is a Lipschitz continuous function with Lipschitz constant* L*. Let* $\epsilon > 0$*,* $\frac{1}{N\eta_\delta} \leq a < 1$ *and* $\delta > 0$ *be arbitrary constants. Then, for a dataset with users having uniform PAS (*$p(\phi)$*) with the same AS (*Δ*), for sufficiently large* N*, the following inequality holds*

$$\begin{aligned}&\left\|\boldsymbol{r}_{DL}{}^{test} - f(\boldsymbol{r}_{UL}{}^{test})\right\| \\ &\qquad \leq \frac{1}{|A^{UL}|} \sum_{i:\boldsymbol{r}_{UL}{}^i \in A^{UL}} \left\|\boldsymbol{r}_{DL}{}^i - f(\boldsymbol{r}_{UL}{}^i)\right\| + (L+K)\,\delta + \sqrt{2M-1}\epsilon, \quad (3.7)\end{aligned}$$

with probability at least

$$\left(1-\exp\left(-2N((1-a)\,\eta_\delta)^2\right)\right)\left(1-2\sqrt{2M-1}\exp\left(-\frac{aN\eta_\delta\epsilon^2}{2L^2\delta^2}\right)\right). \tag{3.8}$$

The proof of Theorem 1 is given in Appendix B.

Remark 2. Setting the probability parameters $\delta > 0$ and $\epsilon > 0$ to sufficiently small constant values, one can see that the probability expression given in (3.8) approaches 1 at an exponential rate, as $N \to \infty$. Thus, it can be concluded that as $N \to \infty$, with probability approaching 1, the difference between a test point's estimation error and the average estimation error of training points within the δ-neighborhood of the test point can be made as small as desired. (One can choose the δ parameter sufficiently close to 0, as $N \to \infty$.) From this result, one can conclude the following:

- The smaller the average estimation error of the training points in the δ-neighborhood of the test point can be made via the algorithm used to learn the $f(\cdot)$ function, the smaller the upper bound on the estimation error of the test point gets. This can be achieved by arranging the objective function of the algorithm accordingly.

- Learning a function $f(\cdot)$ with a low Lipschitz constant L leads to a faster decrease in the upper bound. This can also be achieved by proper adjustments in the objective function of the algorithm. In practice, our result puts forward the following trade-off between the Lipschitz constant L and the training error: While one may reduce the training error to arbitrarily small values by increasing the complexity of $f(\cdot)$, this may come at the cost of learning a too irregular function with high Lipschitz constant L. Consequently, this causes poor generalization to new test data. A better strategy is to seek a trade-off between the minimization of the training error and the regularity of the learned interpolator $f(\cdot)$.

CHAPTER 4

UPLINK TO DOWNLINK CCM CONVERSION VIA GAUSSIAN RBF KERNEL FUNCTIONS

In this chapter, we propose a representation learning algorithm motivated by the theoretical analysis in the previous section for the problem of DL CCM estimation from UL CCM.

4.1 Problem Formulation

Let $\mathbf{X} = [(\mathbf{r}_{UL}{}^1)^T \ldots (\mathbf{r}_{UL}{}^N)^T]^T \in \mathbb{R}^{N\times(2M-1)}$ be the input training data matrix. Let $\mathbf{Y} = [\mathbf{y}_1^T \ldots \mathbf{y}_N^T]^T \in \mathbb{R}^{N\times(2M-1)}$ be the embedding matrix, where $\mathbf{y}_i = f(\mathbf{x}_i)$. Let $\mathbf{R}_{true} = [(\mathbf{r}_{DL}{}^1)^T \ldots (\mathbf{r}_{DL}{}^N)^T]^T \in \mathbb{R}^{N\times(2M-1)}$ be the output training data matrix. $\mathbf{R}_{true}$ is the DL counterpart of $\mathbf{X}$.

Our aim is to find a function $f(\cdot)$ that approximates the training data sufficiently well, i.e., $f(\mathbf{x}_i) = \mathbf{y}_i \approx \mathbf{r}_{DL}{}^i$, and preserves the nearest neighbors of each input vector in the embedding space, while mapping previously unseen UL CCMs (test data) to DL CCMs with low error. The interpolation problem can be formulated considering the following objectives.

Lipschitz regularity of the interpolation function: The interpolation function is of the form

$$f(\mathbf{r}_{UL}) = [f^{(1)}(\mathbf{r}_{UL})\ f^{(2)}(\mathbf{r}_{UL}) \ldots f^{(2M-1)}(\mathbf{r}_{UL})]. \tag{4.1}$$

Here $f(\cdot)$ is chosen as a radial basis function (RBF) interpolator due to its well-studied properties [31]. Specifically, the Gaussian RBF kernel is chosen for the extension of the embedding, where

$$f^{(k)}(\mathbf{r}_{UL}) = \sum_{i=1}^{N} C_{ik}\, e^{-\frac{\|\mathbf{r}_{UL} - \mathbf{r}_{UL}{}^{i}\|^2}{\sigma^2}} \tag{4.2}$$

is the k^{th} element of $f(\mathbf{r}_{UL})$, for $k \in \{1, ..., 2M-1\}$. C_{ik} are the interpolator coefficients, and σ is the scale parameter of the Gaussian RBF kernel.

The Lipschitz constant of the Gaussian RBF interpolation function is provided in [17] as

$$L = \sqrt{2}e^{-1/2}\sqrt{N}\sigma^{-1}\|\mathbf{C}\|_F, \tag{4.3}$$

where $\mathbf{C} \in \mathbb{R}^{N\times(2M-1)}$ is the matrix containing the interpolator coefficient C_{ik} in its $(i,k)^{th}$ element, $i \in \{1, ..., N\}$, $k \in \{1, ..., 2M-1\}$. The matrix $\mathbf{C}$ is obtained as

$$\mathbf{C} = \mathbf{\Psi}^{-1}\mathbf{Y}, \tag{4.4}$$

by learning a mapping $\mathbf{Y}$ from the training data matrix $\mathbf{X}$, where $\mathbf{\Psi} \in \mathbb{R}^{N\times N}$ is the RBF kernel matrix, whose $(i,j)^{th}$ element is $e^{-\frac{\|\mathbf{r}_{UL}{}^{i} - \mathbf{r}_{UL}{}^{j}\|^2}{\sigma^2}}$.

From Theorem 1, the Lipschitz constant, L, of the interpolator, $f(\cdot)$, should be small so as to reduce the error upper bound in (3.7), which leads to a good generalization of the embedding to the test data. With that in mind, according to (4.3), the following terms should be minimized while learning embedding coordinates and the function parameters of the RBF interpolator:

- σ^{-2}
- $\|\mathbf{C}\|_F^2 = \|\mathbf{\Psi}^{-1}\mathbf{Y}\|_F^2 = tr(\mathbf{Y}^T\mathbf{\Psi}^{-2}\mathbf{Y})$

Preservation of the local geometry between the UL/DL CCM spaces: Due to the angular reciprocity, there is an inherent similarity between the UL CCM and DL CCM of the same user, even though there is no explicit function that relates one another. On the other hand, from Lemma 1, one can see that with enough training data, the points in the UL space become so close that the distance between nearest neighbors is bounded proportionally to the distance between their corresponding points in the DL space. Therefore, in order to preserve the local geometry of UL CCMs in the embedding space, the following term should be minimized

$$\sum_{i,j=1}^{N} (\mathbf{W})_{ij}\|\mathbf{y}_i - \mathbf{y}_j\|^2 = tr(\mathbf{Y}^T\mathbf{L}\mathbf{Y}), \tag{4.5}$$

where $\mathbf{W}$ is a weight matrix whose $(i,j)^{th}$ entry is given by $(\mathbf{W})_{ij} = e^{-\frac{\|\mathbf{r}_{UL}{}^i - \mathbf{r}_{UL}{}^j\|^2}{\theta^2}}$ (for a scale parameter θ), $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the Laplacian matrix, and $\mathbf{D}$ is the diagonal degree matrix with i^{th} diagonal entry $(\mathbf{D})_{ii} = \sum_j (\mathbf{W})_{ij}$. The weights in the weight matrix are selected according to the pairwise distances between data pairs, i.e., $\|\mathbf{r}_{UL}{}^i - \mathbf{r}_{UL}{}^j\|$ for $i, j \in \{1, ..., N\}, i \neq j$. In this way, for nearby $(\mathbf{r}_{UL}{}^i, \mathbf{r}_{UL}{}^j)$ pairs with strong edge weights, a high penalty is applied to the action of mapping $\mathbf{y}_i$ and $\mathbf{y}_j$ far from one another, which preserves the structure of the local neighborhoods between the UL and the DL domains [32]. The equality in (4.5) is shown in [32].

UL/DL CCM pairs in the training dataset: Since the task is to learn a function that maps UL CCMs to their corresponding DL CCMs, the UL-DL CCM pairs in the training dataset are also incorporated into our optimization problem. Instead of employing hard data fidelity constraints, in order to achieve better noise tolerance we prefer the quadratic penalty term given by

$$\|\mathbf{Y} - \mathbf{R}_{true}\|_F^2.$$

Overall problem: We finally combine the above terms to form our overall objective function as

$$\min_{\mathbf{Y},\sigma} tr(\mathbf{Y}^T\mathbf{L}\mathbf{Y}) + \mu_1 tr(\mathbf{Y}^T\mathbf{\Psi}^{-2}\mathbf{Y}) + \mu_2\sigma^{-2} + \mu_3\|\mathbf{Y} - \mathbf{R}_{true}\|_F^2, \tag{4.6}$$

where μ_1, μ_2 and μ_3 are positive weights to determine the relative importance of each term in the objective function.

4.2 Solution of the Problem

The optimization problem defined above is not jointly convex in $\mathbf{Y}$ and σ. We employ an alternating optimization method, where one of the parameters is fixed while the other one is optimized in an alternative fashion at each iteration. This alternation is continued until convergence or the maximum number of iterations is reached.

Optimization of Y: When σ is fixed, the optimization problem in (4.6) becomes the following:

$$\min_{\mathbf{Y}} tr(\mathbf{Y}^T\mathbf{L}\mathbf{Y}) + \mu_1 tr(\mathbf{Y}^T\mathbf{\Psi}^{-2}\mathbf{Y}) + \mu_3\|\mathbf{Y} - \mathbf{R}_{true}\|_F^2 \tag{4.7}$$

This minimization problem is a quadratic and convex problem. The closed form solution of the problem in (4.7) is given by

$$\mathbf{Y}^* = \mu_3 \left(\mathbf{A} + \mu_3 \mathbf{I}\right)^{-1} \mathbf{R}_{true}, \tag{4.8}$$

where $\mathbf{A} = \mathbf{L} + \mu_1 \mathbf{\Psi}^{-2}$. The eigenvalues of a graph Laplacian matrix are always non-negative, i.e., the Laplacian matrix is a positive semidefinite (PSD) matrix. Therefore, the $(\mathbf{A} + \mu_3 \mathbf{I})$ is always invertible.

Optimization of σ: When $\mathbf{Y}$ is fixed, the optimization problem is the following:

$$\min_{\sigma} \mu_1 tr(\mathbf{Y}^T \mathbf{\Psi}^{-2} \mathbf{Y}) + \mu_2 \sigma^{-2}, \tag{4.9}$$

since Ψ has terms related to σ^{-2}.

Although nonconvex, this problem involves the optimization of a single scalar variable σ, which can be solved via an exhaustive search of σ in a reasonable interval.

Our solution algorithm is summarized in Algorithm 1.

Algorithm 1: DL CCM Interpolation via Gaussian RBF Kernel

input : Training data matrices $\mathbf{X}$ and $\mathbf{R}_{true}$

Initialization:

Construct the graph Laplacian matrix $\mathbf{L}$ and the RBF kernel matrix $\mathbf{\Psi}$

Assign weight parameters μ_1, μ_2 and μ_3 and initial values of σ and $\mathbf{Y}$

repeat

  Fix σ and optimize $\mathbf{Y}$ via (4.8);

  Fix $\mathbf{Y}$ and optimize σ via (4.9);

until *convergence of the objective function or the maximum iteration number is reached*;

output: Kernel scale parameter σ, embedding matrix $\mathbf{Y}$

After learning the embedding matrix $\mathbf{Y}$ and the kernel scale parameter σ with Algorithm 1, one can calculate the interpolator coefficient matrix $\mathbf{C}$ by using equation (4.4). Thus, using (4.1) and (4.2), one can estimate the DL CCM of a new test sample that is not in the training dataset by using its UL CCM.

The integral of PAS over all angles is known to be 1; however we have no constraints forcing such a normalization while learning the embedding and the kernel scale parameter. For this reason, once we obtain the estimate $\hat{\mathbf{r}}_{DL}$, we normalize it such that its first entry becomes $\hat{\mathbf{r}}_{DL}(1) = 1$.

4.3 Complexity Analysis

The main factors that determine the complexity of our algorithm are the optimization problems given in (4.7) and (4.9), which are solved in an alternating fashion. The complexity of constructing the matrices $\mathbf{L}$ and $\mathbf{\Psi}$ is $\mathcal{O}(MN^2)$, where N is the number of training data in the dataset. The matrix inversion operations in (4.8) and (4.9) are of complexity $\mathcal{O}(N^3)$, which is the determining part of the complexity analysis, in a typical scenario where $M < N$. Hence, the overall complexity of our algorithm is $\mathcal{O}(N^3)$.

After the training is completed, the Gaussian RBF interpolation function can be directly used to find the DL CCMs of new data. The complexity of finding an estimate of the DL CCM using our function is $\mathcal{O}(M^2N)$ since for each element of the mapping vector of size $(2M-1)$, $(2M-1)$-dimensional vectors are used for calculation at N center locations [33].

CHAPTER 5

SIMULATIONS

In this chapter, we evaluate the performance of our algorithm with simulations. We first observe the behavior of the objective function and that of the estimation performance of our method throughout the iterations. Next, we conduct tests to study how the performance of our method varies with algorithm hyperparameters. Finally, we compare the performance of our method to the performances of some baseline methods in the literature.

5.1 Simulation Parameters and Setup

The simulation setup reported in Table 5.1 is used for the experiments.

Table 5.1: Simulation Parameters

Carrier Frequencies	$f_{UL} = 1.95$ GHz, $f_{DL} = 2.14$ GHz
Base Station Antenna Number (M)	One of the following: $\{32, 64, 128, 256\}$
Dataset Size (Train and Test)	500
Train/Test Data Ratio	$80\%/20\%$
μ_1, μ_2, μ_3	$0.1, 3 \times 10^5, 100$
SNR	20 dB

Users are considered to have uniform PAS with mean AoAs uniformly distributed in $[-\pi, \pi]$. The spread of AoAs of users are drawn from $[5°, 15°]$ uniformly. The carrier frequencies of uplink and downlink channels in Table 5.1 are chosen according to [34].

Let us denote the true value of a DL CCM by $\mathbf{R}_{DL}$ and its estimate by $\hat{\mathbf{R}}_{DL}$. The following three error metrics are used to compare the performance of the proposed algorithm with benchmark methods:

1. Normalized Mean Square Error (NMSE): NMSE is used to measure the average error in each entry of a CCM, which is defined as

$$\text{NMSE} = \mathbb{E}\left\{ \frac{\|\mathbf{R}_{DL} - \hat{\mathbf{R}}_{DL}\|_F^2}{\|\mathbf{R}_{DL}\|_F^2} \right\}. \tag{5.1}$$

2. Correlation Matrix Distance (CMD): This metric defined in [35] is used to quantify the deviation between the direction of the true DL CCM and that of its estimate. The CMD is given by

$$\text{CMD} = \mathbb{E}\left\{ 1 - \frac{tr(\mathbf{R}_{DL}\hat{\mathbf{R}}_{DL})}{\|\mathbf{R}_{DL}\|_F \|\hat{\mathbf{R}}_{DL}\|_F} \right\}. \tag{5.2}$$

3. Deviation Metric (DM): In [15], the following deviation metric is used to measure the deviation in the principal eigenvector of the estimated DL CCM, which is useful in beamforming applications:

$$\text{DM} = 1 - \frac{tr(\mathbf{v}^H \mathbf{R}_{DL} \mathbf{v})}{\Gamma_{max}}, \tag{5.3}$$

   where Γ_{max} is the largest eigenvalue of $\mathbf{R}_{DL}$ and $\mathbf{v}$ is the eigenvector corresponding to the largest eigenvalue of $\hat{\mathbf{R}}_{DL}$.

The dataset is constructed similarly to the setting in [15] as described below. The following steps are followed for all UL CCMs in the dataset and for DL CCMs in the training dataset. DL CCMs in the test set are constructed via only Step 1, so that they form an ideal ground truth data set for performance comparisons of our algorithm with the benchmark methods.

1. CCMs are calculated using the formula in (3.3).

2. Using the generated CCMs, UL and DL channel realizations are constructed as follows:

$$\left(\mathbf{h}_x^k\right)^c = \left(\mathbf{R}_x^k\right)^{1/2} \left(\mathbf{w}_x^k\right)^c, c = 1, ..., N_{ch}, x \in \{UL, DL\}, \tag{5.4}$$

where $\left(\mathbf{w}_x^k\right)^c \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$, $\mathbf{R}_x^k$ is the CCM of user k (either UL or DL, specified by x) and N_{ch} is the number of channel realizations. N_{ch} is taken as $2M$ in the simulations.

3. The noisy channel estimates obtained after the training phase with pilot signals are modeled and generated as follows:

$$\left(\hat{\mathbf{h}}_x^k\right)^c = \left(\mathbf{h}_x^k\right)^c + \left(\mathbf{n}_x^k\right)^c, c = 1, ..., N_{ch}, x \in \{UL, DL\} \tag{5.5}$$

where $\left(\mathbf{n}_x^k\right)^c \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$ and $\left(\hat{\mathbf{h}}_x^k\right)^c$ is the noisy channel estimate of the c^{th} channel realization. The signal-to-noise ratio (SNR) for this pilot signaling setup is taken to be the same as in [15], which is $\frac{tr(\mathbf{R}_{UL}^k)}{\sigma^2} = 20$ dB, unless it is explicitly said to be taken differently.

4. The sample covariance for user k is then given by

$$\hat{\mathbf{R}}_x^k = \frac{1}{N_{ch}} \sum_{c=1}^{N_{ch}} \left(\hat{\mathbf{h}}_x^k\right)^c \left(\hat{\mathbf{h}}_x^k\right)^{c^H} - \sigma^2\mathbf{I}, x \in \{UL, DL\}. \tag{5.6}$$

5. Due to the ULA antenna structure at the BS and the WSSUS model, the CCMs are Toeplitz, Hermitian and PSD, which is used for the correction of the sample covariance found in (5.6). The projection of the sample covariance onto the set of Toeplitz, Hermitian and PSD matrices is done by the alternative projection method proposed in [36]. The projection method solves the following optimization problem:

$$\tilde{\mathbf{R}}_x^k = \arg\min_{\mathbf{X} \in T_+^M} \|\mathbf{X} - \hat{\mathbf{R}}_x^k\|^2 \tag{5.7}$$

where T_+^M is the set of $M \times M$ Toeplitz, Hermitian and PSD matrices.

6. The matrices estimated in the previous step are normalized so that their $(1,1)^{th}$ element is 1. This is done due to the fact that the PAS of the CCMs are normalized to 1.

Remark 3. Fraunhofer distance specifies the border between the near field and the far field, and its formula is given as follows [37]:

$$d = \frac{2D^2}{\lambda}, \tag{5.8}$$

where D is the largest dimension of the antenna and λ is the wavelength. For the ULA antenna configuration, $D = (M-1)\frac{\lambda_{UL}}{2}$ and $\lambda = \lambda_{DL}$, while d can be rewritten as

$$d = \frac{2((M-1)\frac{\lambda_{UL}}{2})^2}{\lambda_{DL}} = \frac{f_R \lambda_{UL}(M-1)^2}{2}. \tag{5.9}$$

With the parameters given in Table 5.1, the Fraunhofer distance can be given as a function of the base station antenna number M as

$$d = \frac{\frac{2.14}{1.95} \times \frac{3\times10^8}{1.95\times10^9} \times (M-1)^2}{2} = 8.4418 \times 10^{-2}(M-1)^2 \tag{5.10}$$

For the M values in Table 5.1, the Fraunhofer distance values are computed as in Table 5.2, which should be taken into consideration in practice. The underlying assumption in our model is that the users are in the far-field of the antenna array. Hence, their distances to the base station are considered to be greater than the Fraunhofer distance. From Table 5.2, one can see that for base stations with $M = 32$ or $M = 64$ antennas placed in a ULA structure, Fraunhofer distances seem relatively more reasonable than those with $M = 128$ or $M = 256$ antennas, since the near-field covers great distances for the higher number of BS antenna cases. One may also note that the length of the ULA with $M = 128$ or $M = 256$ antennas becomes too large for practical cases.

Table 5.2: Fraunhofer Distance Values

$M = 32$	81.13 m
$M = 64$	335.06 m
$M = 128$	1361.58 m
$M = 256$	5489.29 m

However, the UL-to-DL CCM conversion problem in this study is not dependent on the UL and the DL carrier frequencies separately, rather it depends on their ratio. Therefore, one can apply the proposed method for systems using higher carrier frequencies, e.g., millimeter wave (mmWave) bands, as well. In mmWave systems, the

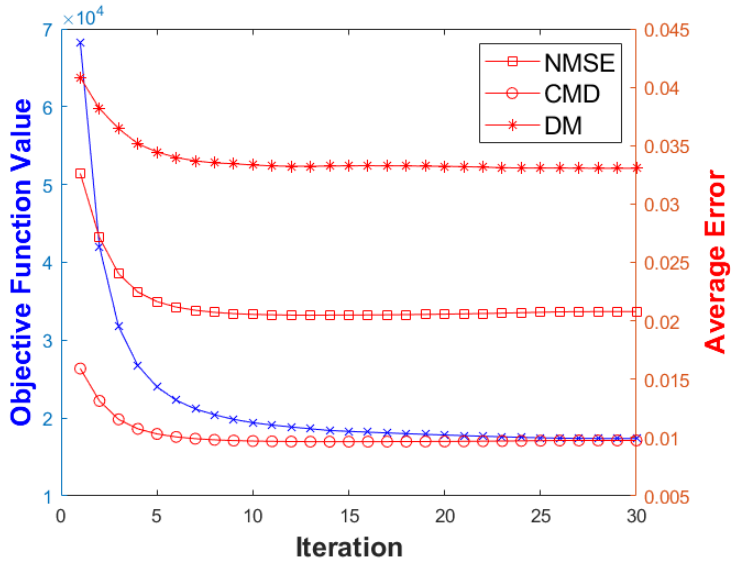


Figure 5.1: The changes on the objective function and the average error performance throughout the iterations

carrier frequencies generally lie in the range of $30 - 300$ GHz [38]. For frequencies near 30 GHz, the wavelength decreases roughly 15-fold compared to the sub-6 GHz system with the carrier frequencies in Table 5.1. Therefore, both the Fraunhofer distances and the lengths of the ULAs at the base stations with $M = 128$ and $M = 256$ antennas become more reasonable for practical cases.

5.2 Stability and Sensitivity Analysis

First, we study the change in the objective function and the change in the average NMSE of DL CCMs learned by our algorithm throughout the iterations. For $M = 64$ base station antennas, we repeat the experiments for 25 i.i.d. datasets. The average objective function and error values are presented in Figure 5.1. From Figure 5.1, one can see that the objective function decreases throughout the iterations, which is expected because the algorithm updates both the embedding and the kernel scale parameter in such a way that the objective function never increases. The average NMSE, CMD and DM follow a similar trend to decrease as the objective function, which suggests that our proposed objective function well captures the performance goal of our algorithm.

Next, we conduct a sensitivity analysis in order to examine the effect of the hyperparameters (μ_1, μ_2, μ_3) on the performance of our algorithm. Tables 5.3 and 5.4 show the NMSE values of the DL CCM estimates of our algorithm for several (μ_1, μ_2, μ_3) combinations. For each (μ_1, μ_2, μ_3), we repeat the experiments for 10 i.i.d. datasets, where the base station has $M = 64$ antennas. The average NMSE values are presented in Tables 5.3 and 5.4. Table 5.3 shows the necessity of the Lipschitz constant-related terms, since the performance gets better as μ_1 and μ_2 increase together from 0 up to around $\mu_1 = 10^{-1}$ and $\mu_2 = 3 \times 10^5$. After that point, the performance deteriorates, since the Lipschitz consant-related terms start to dominate the objective function in (4.6), which decreases the significance of the data fidelity terms in the objective function. This causes the mappings of the training points to deviate from their true values and eventually results in a performance degradation.

Table 5.4 reports the performance for different weight combinations for the Lipschitz continuity of the interpolator and the data fidelity. The ratio between μ_1 and μ_2 is fixed to a suitable number chosen based on Table 5.3. Looking at Table 5.4, one can see that as μ_3 gets smaller, the average NMSE increases drastically. However, it also shows that μ_1 (and also μ_2) should be chosen as positive numbers to improve the performance. The performance seems to be more sensitive to the data fidelity term than the Lipschitz continuity terms.

5.3 Algorithm Performance

In this section, we compare the performance our method to those of the following three benchmark methods: (1) The dictionary based method in [12], (2) the sinc transformation-based method in [15], (3) the CGAN-based method in [1]. We first calculate average errors in terms of the error metrics defined in Chapter 5.1. Then, we examine the algorithm performances by using their outputs (DL CCM estimates) in two application areas.

Table 5.3: The Variation of the NMSE with the Hyperparameters μ_1 and μ_2 for fixed $\mu_3 = 100$

μ_2 \ μ_1	0	10^{-4}	10^{-3}	10^{-2}	10^{-1}	1	10^{1}	10^{2}
0	0.0463	0.0390	0.0390	0.0389	0.0376	0.0345	0.0309	0.0402
3×10^{-1}	0.0463	0.0348	0.0378	0.0387	0.0376	0.0345	0.0309	0.0402
3×10^{1}	0.0463	0.0313	0.0320	0.0344	0.0361	0.0343	0.0309	0.0402
3×10^{3}	0.0463	0.0349	0.0325	0.0307	0.0297	0.0298	0.0300	0.0403
3×10^{5}	0.0463	0.0265	0.0221	0.0194	0.0201	0.0238	0.0319	0.0452
3×10^{7}	0.0463	0.0265	0.0221	0.0194	0.0208	0.0313	0.0540	0.0796
3×10^{9}	0.0463	0.0265	0.0221	0.0194	0.0208	0.0313	0.0540	0.1022
3×10^{11}	0.0463	0.0265	0.0221	0.0194	0.0208	0.0313	0.0540	0.1022

Table 5.4: The Variation of the NMSE with the Hyperparameters μ_1 and μ_3 for $\mu_2 = 3 \times 10^{6}\mu_1$

μ_3 \ μ_1	0	10^{-4}	10^{-3}	10^{-2}	10^{-1}	1	10^{1}	10^{2}
10^{-1}	0.2093	0.2248	0.2448	0.2550	0.2739	0.3643	0.6225	0.8106
1	0.0722	0.0704	0.0682	0.0730	0.0858	0.1252	0.2766	0.6016
10^{1}	0.0357	0.0315	0.0277	0.0241	0.0347	0.0566	0.1043	0.2657
10^{2}	0.0463	0.0324	0.0325	0.0275	0.0201	0.0311	0.0540	0.1022
10^{3}	0.0496	0.0330	0.0331	0.0335	0.0283	0.0199	0.0308	0.0538
10^{4}	0.0500	0.0331	0.0331	0.0332	0.0336	0.0284	0.0198	0.0307
10^{5}	0.0500	0.0331	0.0331	0.0331	0.0332	0.0336	0.0284	0.0198

5.3.1 Algorithm Performance in terms of the Error Metrics

In this section, we conduct three different experiments. First, we calculate the DL CCM estimation errors with a perfect dataset to study the performances of the compared methods. Then, we compare the errors of the algorithms for different M values. Finally, we calculate the DL CCM estimation errors for different SNR values.

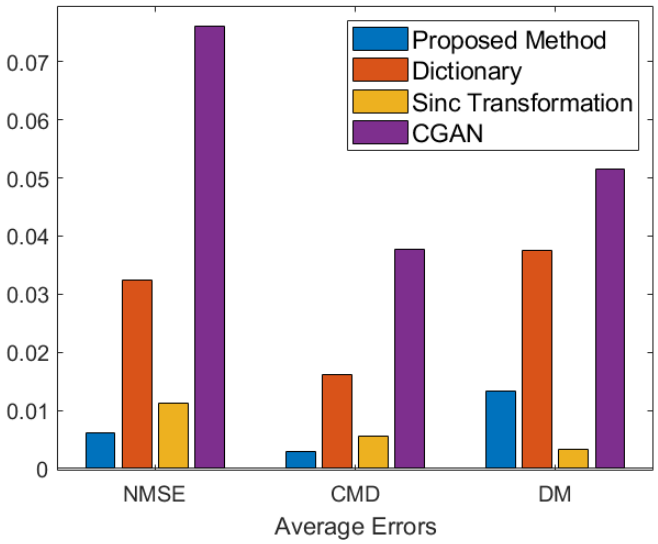
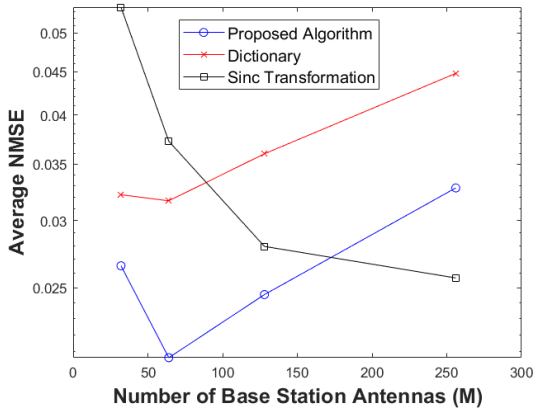
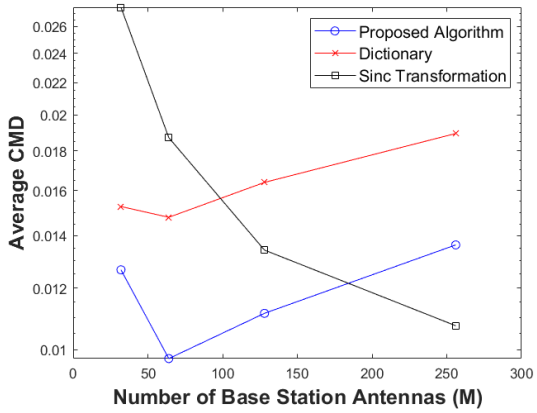


Figure 5.2: Average error values for our method and the benchmark methods for a perfect dataset with CCMs of a $M = 256$ base station antenna system

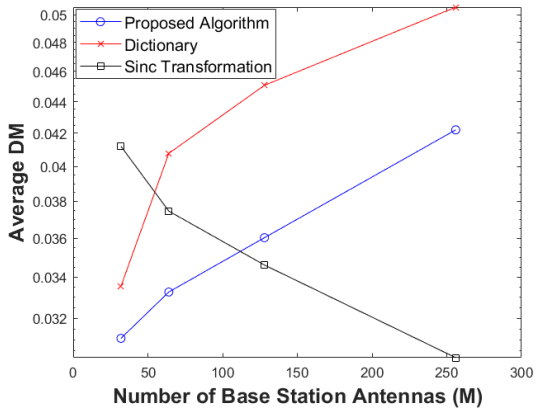
In Figure 5.2, we compare the performances of all benchmark methods for $M = 256$ base station antennas where the CCMs in both the training and the test datasets are perfectly known. The results are averaged over 10 i.i.d. datasets. For this experiment, the hyperparameters are taken as $\mu_1 = 10,\ \mu_2 = 3 \times 10^8$ and $\mu_3 = 10^7$. One can see from Figure 5.2 that our algorithm mostly outperforms the dictionary based method and the sinc transformation method, while the CGAN-based method has relatively higher error values than the other methods. In particular, our method yields the smallest average NMSE value of 6.1×10^{-3} among all methods, while its closest competitor algorithms (dictionary and sinc transformation methods) result in average NMSE values of 0.0324 and 0.0112, respectively. On the other hand, the average NMSE of the CGAN based method for this setup is 0.0761. One can interpret this finding as follows: Even though deep learning based methods can learn highly complex functions quite well, they need a large amount of data to achieve this. In settings with a limited availability of training data, such methods may fail to learn a network that can generalize to new data well. Considering also the long training processes, in the rest of our experiments we compare our algorithm with the dictionary based method and the sinc transformation method, since they are closer to our method in terms of performance.



(a)



(b)

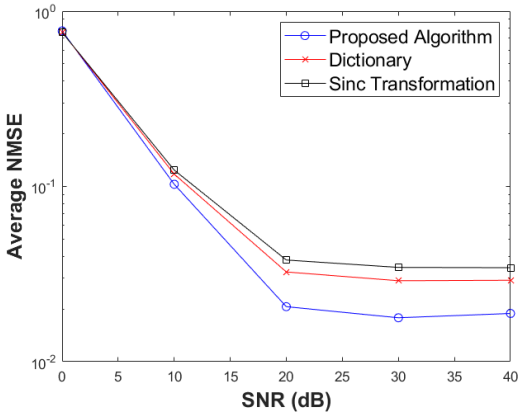


(c)

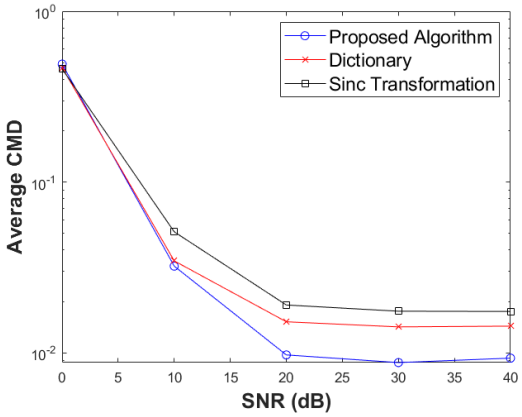
Figure 5.3: The errors of the algorithms at different BS antenna numbers for SNR $=$ 20 dB (a) NMSE (b) CMD (c) DM

Figure 5.3 demonstrates the average error values of each algorithm for the base station antenna numbers of $M \in \{32, 64, 128, 256\}$. For 25 i.i.d. datasets, the experiments are repeated and the average errors are presented in Figure 5.3. One can see that the proposed algorithm surpasses the performance of dictionary-based method for each error metric for all antenna numbers. However, the sinc transformation method has better average error performance than our method for high number of antennas, for example $M = 256$ antennas. This result is expected, since the methods that rely on training data, i.e., the dictionary method and our algorithm, both try to estimate more matrix parameters with the same dataset size, which gets more difficult as the number of base station antennas increases. On the other hand, the sinc transformation method has an error upper bound that decreases with the antenna number, M, which is presented in [15]. Even though the average error of the sinc transformation method is lower than that of our method for $M = 256$ antennas, we have observed the standard deviations of the NMSE values for our method, dictionary method and the sinc transformation method to be 0.0161, 0.0377 and 0.0343, respectively. One can deduce from these results that even though our algorithm may yield higher average error than the sinc transformation method at a high number of antennas, its performance is more stable than that of the sinc transformation method, i.e., it is less likely to encounter outliers with significantly high error values. It should also be noted that for the higher number of antennas, the far-field is further and the ULA is impractically large as stated previously. Hence, our algorithm outperforms the other methods in the more feasible scenarios.

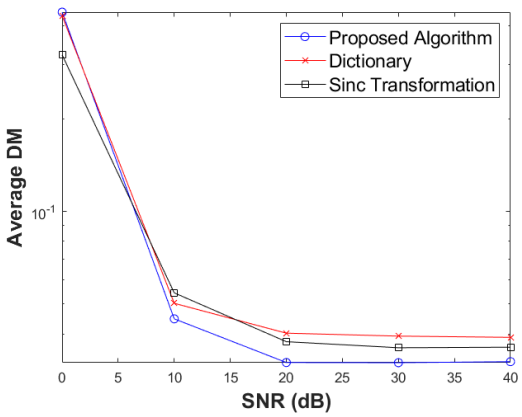
Figure 5.4 shows the performances of the algorithms when the base station has $M = 64$ antennas. The experiments are repeated for 25 i.i.d. datasets. In this scenario, the CCMs have been constructed for several different SNR values ranging from 0 dB to 40 dB and the effect of the SNR on the performance is observed. One can see that all algorithms yield high estimation error at 0 dB SNR as expected, where the CCMs are corrupted with severe noise. As the SNR increases, the estimates obtained from each algorithm improves and our algorithm outperforms the benchmark methods in all performance metrics.



(a)



(b)



(c)

Figure 5.4: The errors of the algorithms at different SNR values when the base station antenna number is $M = 64$ (a) NMSE (b) CMD (c) DM

5.3.2 Algorithm Performance in the Applications Using DL CCM

In this section, the DL CCMs estimated using our algorithm and the benchmark methods are used in DL CSI estimation and beamforming designs to see how much performance degradation they cause compared to the case where the DL CCM is known perfectly. Also, their performances are compared to each other.

Remark 4. In order to prevent any error amplification due to the inverse operations in (5.16) and (5.30), a similar projection operation done in the CCM dataset construction is applied as a post-processing operation to the CCMs obtained via the CCM estimation algorithms. This way, the resulting DL CCMs are projected onto the Toeplitz, Hermitian PSD matrices set. The dictionary-based method does not require this post-processing, since it takes weighted average of DL CCMs in a dictionary that has already been produced using this projection operation. However, since the sinc transformation method and our algorithm deal with the first row of the CCMs rather than the whole matrices, they do not impose positive semi-definiteness property on the covariance matrices predicted.

5.3.2.1 DL CSI Prediction via MMSE Channel Estimation

In this part, DL CSI estimation is conducted via the minimum mean squared error (MMSE) channel estimation method, which benefits from the DL channel covariance information.

The received signal after the pilot transmission in a Gaussian zero mean DL channel $\mathbf{h}$ with a covariance matrix $\mathbf{R}$, i.e., $\mathbf{h} \sim \mathcal{CN}(\mathbf{0},\ \mathbf{R})$, is expressed as follows:

$$y_p = \mathbf{h}^T \mathbf{x}_p + n_p,\ p \in \{1, ..., N_p\} \tag{5.11}$$

where y_p is the received signal, $\mathbf{h}$ is the DL channel, $\mathbf{x}_p$ is the pilot symbol transmitted through the elements of the BS antenna array at time instant p, $n_p \sim \mathcal{CN}(0,\ \sigma^2)$ is the noise and N_p is the number of pilot symbols transmitted.

Expressing all the received signals after pilot transmission in the form of matrix-

vector multiplication, the following formula is obtained:

$$\mathbf{y} = \mathbf{X}\mathbf{h} + \mathbf{n}, \tag{5.12}$$

where $\mathbf{y} := [y_1 \ \dots \ y_{N_p}]^T$, $\mathbf{X} := [\mathbf{x}_1 \ \dots \ \mathbf{x}_{N_p}]^T$ and $\mathbf{n} := [n_1 \ \dots \ n_{N_p}]^T$.

The MMSE channel estimator for this setup is given as follows [39]:

$$\hat{\mathbf{h}}_{MMSE} = \mathbf{R}\mathbf{X}^H \left(\mathbf{X}\mathbf{R}\mathbf{X}^H + \sigma^2\mathbf{I}\right)^{-1} \mathbf{y}. \tag{5.13}$$

The mean squared error (MSE) of this channel estimate is given in the closed form formula as follows [39]:

$$\text{MSE} = tr\left(\mathbf{R} - \mathbf{R}\mathbf{X}^H \left(\mathbf{X}\mathbf{R}\mathbf{X}^H + \sigma^2\mathbf{I}\right)^{-1} \mathbf{X}\mathbf{R}\right). \tag{5.14}$$

A total pilot power constraint is employed for pilot transmission scheme, which is $tr(\mathbf{X}\mathbf{X}^H) \leq P$, where P is the total power allocated for the pilot symbols.

The pilot matrix $\mathbf{X}$ is created so that it has orthonormal rows, i.e., the following equation is satisfied:

$$\mathbf{x}_i^H \mathbf{x}_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}, i, j \in \{1, ..., N_p\}. \tag{5.15}$$

Then, it is scaled to have $tr(\mathbf{X}\mathbf{X}^H) = P$ so that it obeys the power constraint rule provided above.

We conduct experiments for $M = 64$ BS antennas, where a CCM dataset is generated with SNR $= 20\ dB$. After learning DL CCMs from their UL counterparts via the algorithms examined, we use the DL CCM estimates in the MMSE channel estimator given in (5.13) instead of the true DL CCM values. The imperfect MMSE channel estimate of a DL channel realization $\mathbf{h}$ whose true CCM is $\mathbf{R}$ is given as follows:

$$\hat{\mathbf{h}}_{MMSE}^{imp} = \hat{\mathbf{R}}\mathbf{X}^H \left(\mathbf{X}\hat{\mathbf{R}}\mathbf{X}^H + \sigma^2\mathbf{I}\right)^{-1} \mathbf{y}, \tag{5.16}$$

where $\hat{\mathbf{h}}_{MMSE}^{imp}$ is the MMSE channel estimation of $\mathbf{h}$ obtained by using an estimate of $\mathbf{R}$, say $\hat{\mathbf{R}}$, instead of using the true CCM, $\mathbf{R}$. $\hat{\mathbf{R}}$ is obtained from one of the algorithms considered (either our algorithm or one of the benchmarks).

NMSE is used as the performance metric to measure the normalized channel estimation errors caused by the imperfect CCM estimations. The NMSE is calculated as follows:

$$\text{NMSE} = \mathbb{E}\left\{ \frac{\|\mathbf{h} - \hat{\mathbf{h}}\|^2}{\|\mathbf{h}\|^2} \right\}, \tag{5.17}$$

where $\mathbf{h}$ is the true value of a DL channel realization and $\hat{\mathbf{h}}$ is the MMSE channel estimation found with the help of the DL CCM.

The same 25 i.i.d. datasets in the previous experiments where the number of BS antennas was $M = 64$ and the SNR was $20\ dB$ are used. In order to test the average performance of the imperfect MMSE channel estimator given in (5.16) with the CCM estimation methods considered and that of the MMSE channel estimator using the true DL CCMs, the following procedure is applied:

1. First, for each dataset, the DL CCMs are estimated for the test points.
2. Using each test point's DL CCM, say $\mathbf{R}$, 100 different DL channel realizations are constructed as follows:

$$\mathbf{h}_r = \mathbf{R}^{1/2}\mathbf{w}_r, r = 1, ..., 100, \tag{5.18}$$

   where $\mathbf{w}_r \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$.
3. Next, for each channel realization given by (5.18), the MMSE estimate is computed using (5.13). Let's name this estimate as $\hat{\mathbf{h}}_r$.
4. Then, using the DL CCM estimates found from each DL CCM estimation method, say $\hat{\mathbf{R}}$, the imperfect MMSE channel estimates are calculated using

(5.16) for all 100 channel realizations for each method. Let's denote this estimate as $\hat{\mathbf{h}}_r^{imp}$.

5. Finally, the NMSE is calculated via (5.17) for $\hat{\mathbf{h}}_r$ and for $\hat{\mathbf{h}}_r^{imp}$ of each DL CCM method.

Each of the 25 datasets contains 100 test points. For each test point, 100 channel realizations are calculated as explained above, and their NMSE values are found as described. The expectation in the NMSE formula in (5.17) is found numerically by averaging first over the 100 channel realizations, then over the 100 test points, and finally over the 25 test datasets.

Two different experiments are conducted. The first setup considers the effect of the SNR in the pilot signaling on the channel estimation performance, which is plotted in Figure 5.5. The pilot transmit power is set to P and the noise power is given as σ^2 as explained previously. Thus, we can define the pilot transmit SNR as SNR $= \frac{P}{\sigma^2}$. For pilot transmit SNR values from $0\ dB$ to $50\ dB$, and for a constant pilot signaling time, which is chosen as the rank of the true DL CCM value, the MMSE channel estimators with the perfect CCM and with the CCM estimates are compared in terms of the NMSE given in (5.17). The second setup considers the effect of the number of pilot symbols on the channel estimation, which is depicted in Figure 5.6. The number of pilot symbols used ranges from 10 to 40. The pilot transmit SNR is set to $20\ dB$ for this setting.

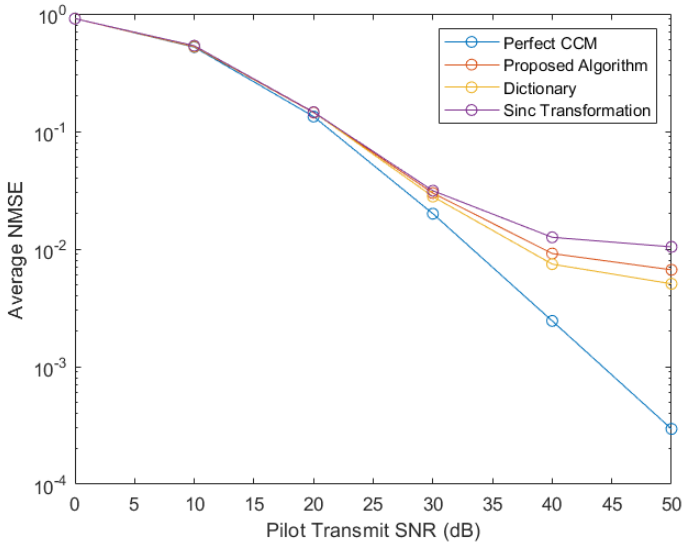


Figure 5.5: Average NMSE values of the MMSE channel estimator for different pilot transmit SNR values

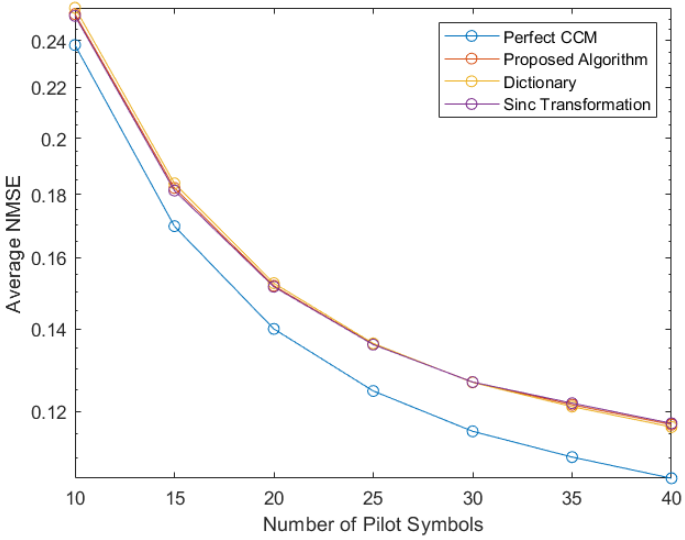


Figure 5.6: Average NMSE values of the MMSE channel estimator for different number of pilot symbols

From Figure 5.5, one can see that the error of the MMSE channel estimate with perfect CCM approaches zero while those of the MMSE channel estimates with the imperfect CCMs saturate, as the pilot transmit SNR increases. This is not a surprising result, since as the pilot transmit SNR increases, the power of the noise in the received

signals after pilot signaling approaches zero. This means that the MMSE channel estimate using the true DL CCM approaches the true channel realization considering that there is no noise in the channel observations and the CCM of the channel is known perfectly. Also, the number of pilot symbols is chosen as equal to the rank of the CCM, which leads the MSE given in (5.14) to approach zero with the diminishing noise [39]. On the other hand, even though the channel observations become noise-free for the imperfect CCM cases as well, this does not remove the imperfections in the CCM estimates. Therefore, due to these residual errors, the improvements in the MMSE channel estimates for these cases start to slow down as the pilot transmit SNR increases, especially at high values where the noise could be considered as almost zero.

If we compare the MMSE channel estimation performances for the cases with imperfect CCMs, the following result could be drawn from Figure 5.5: As one can see from Figure 5.4, our algorithm for DL CCM estimation beats the benchmark methods in terms of all three error metrics used to measure the performance of the DL CCM estimation. However, there is no direct parallel between the estimation accuracy of the DL CCM and the performance of the imperfect MMSE channel estimation employing it. In fact, for the high values of the pilot transmit SNR, the performance of the MMSE estimate using dictionary method's DL CCM is slightly better than that using our algorithm's DL CCM. This result may seem surprising at first. However, it may be possible due to the fact that the design of the DL CCM estimates do not consider minimizing the MSE of the MMSE channel estimator. Considering the fact that the dictionary method offers a solution where the DL CCM estimate is a weighted average of the DL CCMs of other user points that are in a dictionary, the resulting estimate could provide a viable DL CCM estimate. However, the other two methods (our method and the sinc transformation method) are only interested in the first row of the DL CCM rather than the whole matrix. Our best guess is the following: Due to the reason above, the dictionary method provides DL CCM estimates that are closer to being a valid CCM, which leads to stronger structural integrity of the estimated matrices. Thus, the dictionary method may result in better MMSE channel estimates even though it provides worse CCM estimates in terms of the error metrics. The errors of the DL CCM estimates found by our method and those found by the

sinc transformation method may have been amplified more than those found by the dictionary method, most probably due to the inverse operation in the MMSE channel estimation formula given in the (5.16).

If we look at Figure 5.6, we can see a similar result. In this case, for a fixed pilot transmit SNR $= 20\ dB$, we examine the effect of changing the number of the pilot symbols used in the pilot signaling process on the performance of the MMSE channel estimates. One can see that even though our algorithm outperforms the other methods in terms of DL CCM estimation, the performance gap between them almost vanishes in MMSE channel estimation, as in the case with constant pilot time and changing pilot transmit SNR.

A slightly more positive interpretation of these results may be the following: Our algorithm has been shown to be better than the other methods in terms of DL CCM estimation. Even though it may not show the same superiority in MMSE channel estimation, it achieves almost the same performance as the other methods. This shows that our algorithm promises to be useful in this application area. In order to achieve better MMSE channel estimates, the MSE formula of the MMSE channel estimate given in (5.14) can be added to the objective function given in (4.6), which may be seen as a future work.

5.3.2.2 SLNR-Based DL Beamforming Design

In this part, a statistical beamforming scheme named "Signal-to-Leakage-and-Noise Ratio (SLNR)-Based Beamforming" is employed, which uses the DL channel covariance information.

In a multi-user scenario with K single-antenna users, the beamformer for user k is designed based on maximizing its statistical SLNR, whose formula is given as follows [40]:

$$\text{SLNR}_k^{\text{avg}} = \frac{\rho \mathbf{w}_k^H \mathbf{R}_k \mathbf{w}_k}{\rho \mathbf{w}_k^H \left(\sum_{j \neq k} \mathbf{R}_j \right) \mathbf{w}_k + 1}, \tag{5.19}$$

where $\mathbf{R}_k$ is the DL CCM of the user k, ρ is the transmit SNR and $\mathbf{w}_k$ is the beam-

forming vector for user k. The transmit SNR is defined as follows:

$$\rho = \frac{P}{\sigma^2}, \tag{5.20}$$

where ρ is the transmit SNR value, P is the power allocated to the transmitted symbol and σ^2 is the noise power. The beamformer for user k is given as follows [40]:

$$\mathbf{w}_k = \mathbf{u}_{\max}\left(\left(\rho^{-1}\mathbf{I} + \sum_{j\neq k}\mathbf{R}_j\right)^{-1}\mathbf{R}_k\right), \tag{5.21}$$

where $\mathbf{u}_{\max}(.)$ is the operation of taking the eigenvector corresponding to the highest eigenvalue of a matrix.

The achievable ergodic sum rate of the K users is given as follows [40]:

$$R_{\text{sum}} = \mathbb{E}\left\{\sum_{k=1}^{K} R_k\right\} = \mathbb{E}\left\{\sum_{k=1}^{K}\log(1+\text{SINR}_k)\right\}, \tag{5.22}$$

where R_k is the achievable ergodic rate of user k and SINR_k is the signal-to-interference-plus-noise ratio (SINR) of user k, which can be expressed as follows [40]:

$$\text{SINR}_k = \frac{\rho|\mathbf{h}_k^H\mathbf{w}_k|^2}{\rho\sum_{j\neq k}|\mathbf{h}_k^H\mathbf{w}_j|^2 + 1}, \tag{5.23}$$

where $\mathbf{h}_k$ represents an instantaneous channel realization between the BS and the user k.

The performance of the SLNR-based beamformer is tested in terms of the ergodic sum rate given in (5.22) for different numbers of users and for different transmit SNR values.

In the multi-user scenario setup for the SLNR-based beamforming, for every test point in all 25 datasets (the same datasets in the MMSE channel estimation part), $K-1$ additional users are randomly generated for $K \in \{2,3,4,5\}$. Let the main user's mean AoA be $\bar{v}$ and its AS be Δ. Let the additional users' mean AoA and AS values be denoted as $\bar{v}_i$ and Δ_i, respectively, for $i \in \{2,3,4,5\}$. The mean AoAs are generated as follows:

$$\bar{v_2} = \bar{v} + \left(\Delta + \Delta_{\text{max}} + \tilde{\Delta}_{2,3}\right) \tag{5.24}$$

$$\bar{v_3} = \bar{v} - \left(\Delta + \Delta_{\text{max}} + \tilde{\Delta}_{2,3}\right) \tag{5.25}$$

$$\bar{v_4} = \bar{v_2} + \left(\Delta + \Delta_{\text{max}} + \tilde{\Delta}_{4,5}\right) \tag{5.26}$$

$$\bar{v_5} = \bar{v_3} - \left(\Delta + \Delta_{\text{max}} + \tilde{\Delta}_{4,5}\right), \tag{5.27}$$

where $\Delta_{\text{max}} = 15°$ is the maximum AS value taken in the CCM dataset creation and $\tilde{\Delta}_{2,3}$ and $\tilde{\Delta}_{4,5}$ are drawn from $[0°, 5°]$ uniformly and independently.

The AS values satisfy the following equalities:

$$\Delta_2 = \Delta_3 \tag{5.28}$$

$$\Delta_4 = \Delta_5, \tag{5.29}$$

where Δ_2 and Δ_4 are drawn from $[5°, 15°]$ uniformly and independently.

Figure 5.7 and Figure 5.8 show the performance of the SLNR-based beamforming scheme with perfect DL CCM according to (5.21) as well as those with the imperfect CCMs. The beamformer for user k with imperfect DL CCMs obtained by the DL CCM estimators can be found using (5.21) as follows:

$$\mathbf{w}_k = \mathbf{u}_{\text{max}} \left(\left(\rho^{-1}\mathbf{I} + \sum_{j \neq k} \hat{\mathbf{R}}_j \right)^{-1} \hat{\mathbf{R}}_k \right), \tag{5.30}$$

where $\hat{\mathbf{R}}_k$ is the estimate of the DL CCM of user k obtained by using any one of the estimators.

In Figure 5.7, one can see the beamformer performances in the scenario with $K = 3$ users for transmit SNR values ranging from $0\ dB$ to $30\ dB$. In Figure 5.8, the performances are observed for transmit SNR of $20\ dB$ and for different number of users, i.e., K values, in the set $\{2, 3, 4, 5\}$.

The expectation in (5.22) is calculated numerically over 100 channel realizations for

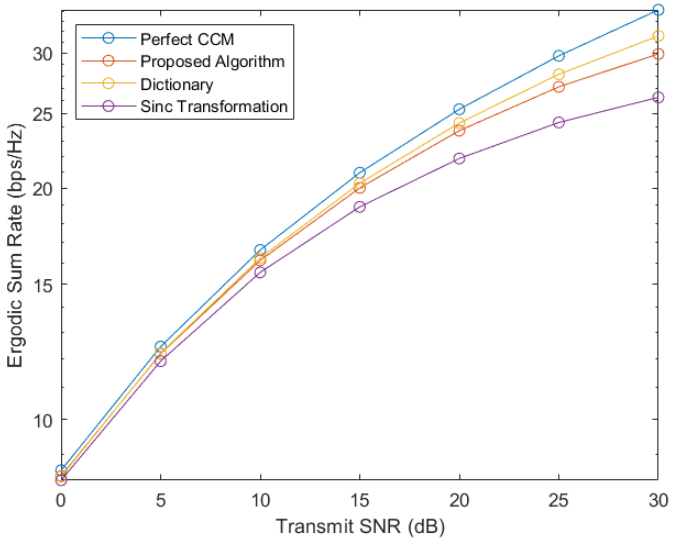


Figure 5.7: Ergodic sum rate values achieved by the SLNR-based beamformer for $K = 3$ users at different transmit SNR values

each point in the 25 i.i.d. datasets with 100 test points, similar to the expectation calculation in the MMSE channel estimation part.

Similar to the MMSE performance, it is shown in Figure 5.7 and Figure 5.8 that better CCM estimation performance does not necessarily lead to better SLNR-based beamforming performance. The dictionary method's CCMs provide slightly better performance than our method in terms of the ergodic sum rate. Similarly to the MMSE channel estimator case, the SLNR-based beamformer given in (5.30) contains a matrix inversion operation, which may have caused an imbalance between the error amplifications of the different DL CCM estimates after having been used in the SLNR-based beamformer.

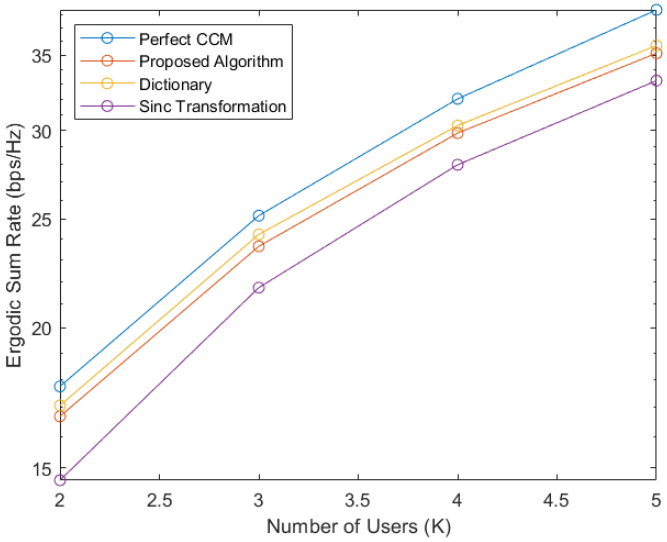


Figure 5.8: Ergodic sum rate values achieved by the SLNR-based beamformer for different numbers of users using $\rho = 20\ dB$ transmit SNR value

In order to interpret the DL CCM estimation methods' performances in terms of SLNR-based beamforming better, we also examine the empirical cumulative distribution functions (CDFs) of the minimum rates for each DL CCM estimates and for the perfect CCM case for $K = 3$ users and transmit SNR $= 20$ dB, which are depicted in Figure 5.9. From Figure 5.9, one can see that the CDFs of the minimum rates of the 3 users generally follow the same order as the ergodic sum rates given in Figure 5.7. This means that our algorithm tends to provide higher minimum rates than sinc transformation method and slightly worse minimum rates than dictionary method.

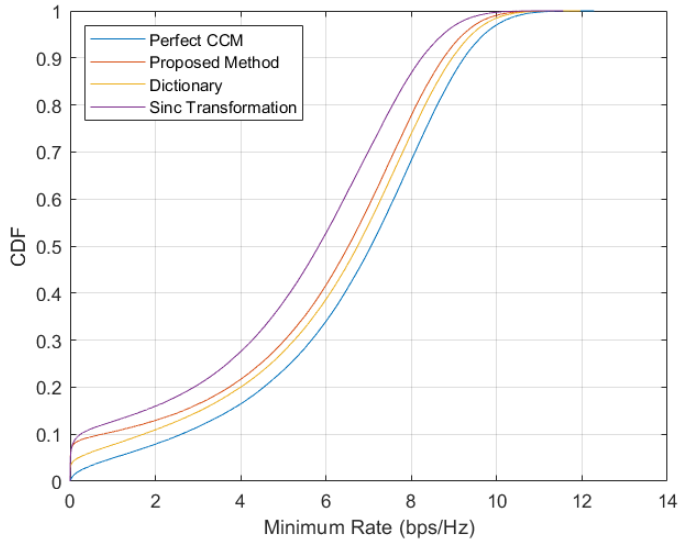


Figure 5.9: Empirical CDF of minimum rate achieved by the SLNR-based beamformer for $K = 3$ users using $\rho = 20\ dB$ transmit SNR value

Remark 5. The experimental results in this part have shown us that dictionary-based method performs slightly better than our method in MMSE channel estimation and SLNR-based beamforming. However, its computational complexity is given as $\mathcal{O}(M^4N)$ in [12]. Our algorithm has a computational complexity of $\mathcal{O}(M^2N)$ (in the test phase). Even though dictionary-based method does not require any training, it is much more computationally costly than our method. Considering the small performance difference between our method and the dictionary method in the application areas considered, one can say that our method performs almost the same as the dictionary method with a lower complexity. Hence, it may be preferable to implement the low complexity-algorithm (our method) at the cost of a slight performance degradation, especially at the systems with a high number of base station antennas, like massive MIMO systems, that is of interest in this study.

CHAPTER 6

CONCLUSIONS

In this thesis, we have proposed a novel DL CCM estimation method for FDD massive MIMO systems where the base station is equipped with ULA antennas. We have first presented a theoretical analysis that gives an upper bound on the estimation error of the DL CCM from UL CCMs. We have then proposed a representation learning-based method in order to learn a mapping function from UL CCMs to their DL CCM counterparts. The proposed method aims at learning an interpolation function from datasets relatively smaller than those needed for training deep neural networks, while capturing the richness of nonlinear learning methods so that the learned mapping is more robust to nonlinearities/discrepancies in the system parameters than simple signal processing based methods. The experimental results show that the proposed algorithm achieves better estimation performance than the benchmark methods in most of the scenarios. The proposed method can especially be useful in practical applications with limited access to training data. Our algorithm shows promising performance in such applications as it provides quite accurate downlink channel covariance estimates with a simple nonlinear learning setup. Our method also achieves results highly comparable with those of the benchmark methods in the application areas considered, which are MMSE channel estimation and SLNR-based beamforming. On the other hand, it has also been seen that the DL CCM estimation performance is not necessarily a direct indicator of the performances in the applications using the DL CCM, such as channel estimation and beamforming. Therefore, an important conclusion drawn from this study is that designing the DL CCM estimator considering its future application (channel estimation, beamforming etc.) may be beneficial. The extension of our method to other base station antenna structures such as a URA and the design of the estimator with an objective function including terms related to the potential

future applications, such as MSE of channel estimation, are left as future research directions.

REFERENCES

[1] B. Banerjee, R. C. Elliott, W. A. Krzymień, and H. Farmanbar, "Downlink channel estimation for FDD massive MIMO using conditional generative adversarial networks," *IEEE Transactions on Wireless Communications*, vol. 22, no. 1, pp. 122–137, 2023.

[2] E. Björnson, J. Hoydis, L. Sanguinetti, *et al.*, "Massive MIMO networks: Spectral, energy, and hardware efficiency," *Foundations and Trends® in Signal Processing*, vol. 11, no. 3-4, pp. 154–655, 2017.

[3] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 40–60, 2013.

[4] E. Björnson, E. G. Larsson, and T. L. Marzetta, "Massive MIMO: Ten myths and one critical question," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 114–123, 2016.

[5] E. Björnson, J. Hoydis, M. Kountouris, and M. Debbah, "Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits," *IEEE Transactions on Information Theory*, vol. 60, no. 11, pp. 7112–7139, 2014.

[6] W. Peng, W. Li, W. Wang, X. Wei, and T. Jiang, "Downlink channel prediction for time-varying FDD massive MIMO systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 5, pp. 1090–1102, 2019.

[7] Y. Xu, G. Yue, and S. Mao, "User grouping for massive MIMO in FDD systems: New design methods and analysis," *IEEE Access*, vol. 2, pp. 947–959, 2014.

[8] Z. Zhong, L. Fan, and S. Ge, "FDD massive MIMO uplink and downlink channel reciprocity properties: Full or partial reciprocity?," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*, pp. 1–5, IEEE, 2020.

[9] H. Xie, F. Gao, S. Jin, J. Fang, and Y.-C. Liang, "Channel estimation for TDD/FDD massive MIMO systems with channel covariance computing," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4206–4218, 2018.

[10] Y.-C. Liang and F. P. S. Chin, "Downlink channel covariance matrix (DCCM) estimation and its applications in wireless DS-CDMA systems," *IEEE Journal on Selected Areas in Communications*, vol. 19, no. 2, pp. 222–232, 2001.

[11] M. Jordan, A. Dimofte, X. Gong, and G. Ascheid, "Conversion from uplink to downlink spatio-temporal correlation with cubic splines," in *VTC Spring 2009-IEEE 69th Vehicular Technology Conference*, pp. 1–5, IEEE, 2009.

[12] A. Decurninge, M. Guillaud, and D. T. Slock, "Channel covariance estimation in massive MIMO frequency division duplex systems," in *2015 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, IEEE, 2015.

[13] M. B. Khalilsarai, S. Haghighatshoar, X. Yi, and G. Caire, "FDD massive MIMO via UL/DL channel covariance extrapolation and active channel sparsification," *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 121–135, 2019.

[14] L. Miretti, R. L. G. Cavalcante, and S. Stańczak, "Channel covariance conversion and modelling using infinite dimensional Hilbert spaces," *IEEE Transactions on Signal Processing*, vol. 69, pp. 3145–3159, 2021.

[15] S. Bameri, K. A. Almahrog, R. H. Gohary, A. El-Keyi, and Y. A. E. Ahmed, "Uplink to downlink channel covariance transformation in FDD systems," *IEEE Transactions on Signal Processing*, vol. 71, pp. 3196–3212, 2023.

[16] K. Hugl, K. Kalliola, J. Laurila, *et al.*, "Spatial reciprocity of uplink and downlink radio channels in FDD systems," in *Proc. COST*, vol. 273, p. 066, 2002.

[17] C. Örnek and E. Vural, "Nonlinear supervised dimensionality reduction via smooth regular embeddings," *Pattern Recognition*, vol. 87, pp. 55–66, 2019.

[18] G. Zhong, L.-N. Wang, X. Ling, and J. Dong, "An overview on data representation learning: From traditional feature learning to recent deep learning," *The Journal of Finance and Data Science*, vol. 2, no. 4, pp. 265–278, 2016.

[19] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[20] Y. Yang, F. Gao, G. Y. Li, and M. Jian, "Deep learning-based downlink channel prediction for FDD massive MIMO system," *IEEE Communications Letters*, vol. 23, no. 11, pp. 1994–1998, 2019.

[21] W. Utschick, V. Rizzello, M. Joham, Z. Ma, and L. Piazzi, "Learning the CSI recovery in FDD systems," *IEEE Transactions on Wireless Communications*, vol. 21, no. 8, pp. 6495–6507, 2022.

[22] Y. Liu and O. Simeone, "Learning how to transfer from uplink to downlink via hyper-recurrent neural network for FDD massive MIMO," *IEEE Transactions on Wireless Communications*, vol. 21, no. 10, pp. 7975–7989, 2022.

[23] J. Zeng, J. Sun, G. Gui, B. Adebisi, T. Ohtsuki, H. Gacanin, and H. Sari, "Downlink CSI feedback algorithm with deep transfer learning for FDD massive MIMO systems," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 4, pp. 1253–1265, 2021.

[24] J. Wang, G. Gui, T. Ohtsuki, B. Adebisi, H. Gacanin, and H. Sari, "Compressive sampled CSI feedback method based on deep learning for FDD massive MIMO systems," *IEEE Transactions on Communications*, vol. 69, no. 9, pp. 5873–5885, 2021.

[25] M. Nerini, V. Rizzello, M. Joham, W. Utschick, and B. Clerckx, "Machine learning-based CSI feedback with variable length in FDD massive MIMO," *IEEE Transactions on Wireless Communications*, vol. 22, no. 5, pp. 2886–2900, 2023.

[26] A. Adhikary, J. Nam, J.-Y. Ahn, and G. Caire, "Joint spatial division and multiplexing—the large-scale array regime," *IEEE Transactions on Information Theory*, vol. 59, no. 10, pp. 6441–6463, 2013.

[27] F. Sohrabi, K. M. Attiah, and W. Yu, "Deep learning for distributed channel feedback and multiuser precoding in FDD massive MIMO," *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, pp. 4044–4057, 2021.

[28] K. Li, Y. Li, L. Cheng, Q. Shi, and Z.-Q. Luo, "Downlink channel covariance matrix reconstruction for FDD massive MIMO systems with limited feedback," *IEEE Transactions on Signal Processing*, vol. 72, pp. 1032–1048, 2024.

[29] L. Miretti, R. L. G. Cavalcante, and S. Stanczak, "FDD massive MIMO channel spatial covariance conversion using projection methods," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3609–3613, IEEE, 2018.

[30] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1125–1134, 2017.

[31] C. Piret, *Analytical and numerical advances in radial basis functions*. PhD thesis, University of Colorado at Boulder, 2007.

[32] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[33] G. Roussos and B. J. Baxter, "Rapid evaluation of radial basis functions," *Journal of Computational and Applied Mathematics*, vol. 180, no. 1, pp. 51–70, 2005.

[34] "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) Radio Transmission and Reception (3GPP TS 36.101 Version 14.3.0 Release 14), document ETSI TS 136 101 V14.3.0," Apr 2017.

[35] M. Herdin and E. Bonek, "A MIMO correlation matrix based metric for characterizing non-stationarity," in *Proceedings IST Mobile & Wireless Communications Summit*, 2004.

[36] K. M. Grigoriadis, A. E. Frazho, and R. E. Skelton, "Application of alternating convex projection methods for computation of positive Toeplitz matrices," *IEEE Transactions on Signal Processing*, vol. 42, no. 7, pp. 1873–1875, 1994.

[37] K. T. Selvan and R. Janaswamy, "Fraunhofer and Fresnel distances: Unified derivation for aperture antennas," *IEEE Antennas and Propagation Magazine*, vol. 59, no. 4, pp. 12–15, 2017.

[38] Y. Niu, Y. Li, D. Jin, L. Su, and A. V. Vasilakos, "A survey of millimeter wave communications (mmWave) for 5g: opportunities and challenges," *Wireless Networks*, vol. 21, pp. 2657–2676, 2015.

[39] J. Fang, X. Li, H. Li, and F. Gao, "Low-rank covariance-assisted downlink training and channel estimation for FDD massive MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1935–1947, 2017.

[40] T. Kim and S. Park, "Statistical beamforming for massive MIMO systems with distinct spatial correlations," *Sensors*, vol. 20, no. 21, p. 6255, 2020.

[41] S. Kaya and E. Vural, "Learning multi-modal nonlinear embeddings: Performance bounds and an algorithm," *arXiv preprint arXiv:2006.02330v2*, 2020.

APPENDICES

A Proof of Lemma 1

The square of the norm of the difference between ${\mathbf{p}_{UL}}^i$ and ${\mathbf{p}_{UL}}^j$, which are arbitrary two points in $\mathbb{R}^{1\times 2M-1}$ drawn i.i.d. from υ, is given as follows:

$$\begin{aligned}
\left\|{\mathbf{p}_{UL}}^i - {\mathbf{p}_{UL}}^j\right\|^2 &= \sum_{m=1}^{M} \left|[{\mathbf{p}_{UL}}^i]_m - [{\mathbf{p}_{UL}}^j]_m\right|^2 \\
&= \sum_{m=1}^{M} \left| \int_{\bar{v_i}-\Delta}^{\bar{v_i}+\Delta} p(\phi) \exp\left(j\pi(m-1)\sin(\phi)\right)\, d\phi \right. \\
&\quad \left. - \int_{\bar{v_j}-\Delta}^{\bar{v_j}+\Delta} p(\phi) \exp\left(j\pi(m-1)\sin(\phi)\right)\, d\phi \right|^2 \\
&= \sum_{m=1}^{M} \left| \int_{\bar{v_i}-\Delta}^{\bar{v_i}+\Delta} \frac{1}{2\Delta} \exp\left(j\pi(m-1)\sin(\phi)\right)\, d\phi \right. \\
&\quad \left. - \int_{\bar{v_j}-\Delta}^{\bar{v_j}+\Delta} \frac{1}{2\Delta} \exp\left(j\pi(m-1)\sin(\phi)\right)\, d\phi \right|^2 \\
&= \sum_{m=1}^{M} \left| \int_{\bar{v_j}+\Delta}^{\bar{v_i}+\Delta} \frac{1}{2\Delta} \exp\left(j\pi(m-1)\sin(\phi)\right)\, d\phi \right. \\
&\quad \left. - \int_{\bar{v_j}-\Delta}^{\bar{v_i}-\Delta} \frac{1}{2\Delta} \exp\left(j\pi(m-1)\sin(\phi)\right)\, d\phi \right|^2 \qquad \text{(A.1)}
\end{aligned}$$

Let us define $\theta := \pi\sin(\phi)$. The limits of the integrals in Equation (A.1) cover very narrow intervals due to the condition $\bar{v_i} - \bar{v_j} \approx 0$ given in Lemma 1. Therefore, one can approximate θ as a linear function of ϕ in these intervals using a first order Taylor approximation.

For $\phi \in [\bar{v_j}+\Delta, \bar{v_i}+\Delta]$,

$$\begin{aligned}
\sin(\phi) \approx \sin&\left(\frac{\bar{v_i}+\bar{v_j}}{2}+\Delta\right) \\
&+ \left(\frac{\cos\left(\frac{\bar{v_i}+\bar{v_j}}{2}+\Delta\right)}{1!}\right)\left(\phi - \left(\frac{\bar{v_i}+\bar{v_j}}{2}+\Delta\right)\right). \qquad \text{(A.2)}
\end{aligned}$$

Therefore, one can approximate θ as $\theta \approx \alpha_1\phi + \beta_1$,
where

$$\alpha_1 = \pi \cos\left(\frac{\bar{v_i} + \bar{v_j}}{2} + \Delta\right) \tag{A.3}$$

and

$$\beta_1 \;=\; \pi\left[\sin\left(\frac{\bar{v_i} + \bar{v_j}}{2} + \Delta\right) \;-\; \left(\frac{\bar{v_i} + \bar{v_j}}{2} + \Delta\right)\cos\left(\frac{\bar{v_i} + \bar{v_j}}{2} + \Delta\right)\right]. \tag{A.4}$$

Similarly, for $\phi \in [\bar{v_j} - \Delta, \bar{v_i} - \Delta]$,

$$\sin(\phi) \;\approx\; \sin\left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right) + \left(\frac{\cos\left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right)}{1!}\right)\left(\phi - \left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right)\right). \tag{A.5}$$

Therefore, one can approximate θ as $\theta \approx \alpha_2\phi + \beta_2$,
where

$$\alpha_2 = \pi \cos\left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right) \tag{A.6}$$

and

$$\beta_2 \;=\; \pi\left[\sin\left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right) \;-\; \left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right)\cos\left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right)\right]. \tag{A.7}$$

The following approximations, which will be useful later, can be inferred from the approximations above:

$$\frac{\sin(\bar{v_i} + \Delta) + \sin(\bar{v_j} + \Delta)}{2} \approx \sin\left(\frac{\bar{v_i} + \bar{v_j}}{2} + \Delta\right) \tag{A.8}$$

$$\frac{\sin(\bar{v_i} - \Delta) + \sin(\bar{v_j} - \Delta)}{2} \approx \sin\left(\frac{\bar{v_i} + \bar{v_j}}{2} - \Delta\right) \tag{A.9}$$

$$\pi\left(\sin(\bar{v_i} + \Delta) - \sin(\bar{v_j} + \Delta)\right) \approx \alpha_1(\bar{v_i} - \bar{v_j}) \tag{A.10}$$

$$\pi\left(\sin(\bar{v_i} - \Delta) - \sin(\bar{v_j} - \Delta)\right) \approx \alpha_2(\bar{v_i} - \bar{v_j}) \tag{A.11}$$

Using the approximations above, one can write $\left|[\mathbf{p}_{UL}{}^{i}]_m - [\mathbf{p}_{UL}{}^{j}]_m\right|^2$ as the following:

$$\begin{aligned}\left|[\mathbf{p}_{UL}{}^{i}]_m - [\mathbf{p}_{UL}{}^{j}]_m\right|^2 \\ \approx \left| \int_{\pi\sin(\bar{v}_j+\Delta)}^{\pi\sin(\bar{v}_i+\Delta)} \frac{1}{2\Delta\alpha_1} \exp\left(j(m-1)\theta\right)\, d\theta \right. \\ \left. - \int_{\pi\sin(\bar{v}_j-\Delta)}^{\pi\sin(\bar{v}_i-\Delta)} \frac{1}{2\Delta\alpha_2} \exp\left(j(m-1)\theta\right)\, d\theta \right|^2 \end{aligned} \tag{A.12}$$

$$= \left| \frac{1}{2\Delta} \left[\frac{\exp\left(j(m-1)\theta\right)}{j(m-1)\alpha_1)} \Bigg|_{\pi\sin(\bar{v}_j+\Delta)}^{\pi\sin(\bar{v}_i+\Delta)} - \frac{\exp\left(j(m-1)\theta\right)}{j(m-1)\alpha_2)} \Bigg|_{\pi\sin(\bar{v}_j-\Delta)}^{\pi\sin(\bar{v}_i-\Delta)} \right] \right|^2 \tag{A.13}$$

$$\begin{aligned} = \left| \frac{1}{2\Delta j(m-1)\alpha_1} \left[\exp\left(j(m-1)\pi\sin(\bar{v}_i+\Delta)\right) \right.\right. \\ \left. - \exp\left(j(m-1)\pi\sin(\bar{v}_j+\Delta)\right) \right] \\ - \frac{1}{2\Delta j(m-1)\alpha_2} \left[\exp\left(j(m-1)\pi\sin(\bar{v}_i-\Delta)\right) \right. \\ \left.\left. - \exp\left(j(m-1)\pi\sin(\bar{v}_j-\Delta)\right) \right] \right|^2 \end{aligned} \tag{A.14}$$

$$\begin{aligned} = \left| \frac{\exp\left(j(m-1)\pi\frac{\sin(\bar{v}_i+\Delta)+\sin(\bar{v}_j+\Delta)}{2}\right)}{2\Delta j(m-1)\alpha_1} \right. \\ 2j\sin\left((m-1)\pi\frac{\sin(\bar{v}_i+\Delta)-\sin(\bar{v}_j+\Delta)}{2}\right) \\ - \frac{\exp\left(j(m-1)\pi\frac{\sin(\bar{v}_i-\Delta)+\sin(\bar{v}_j-\Delta)}{2}\right)}{2\Delta j(m-1)\alpha_2} \\ \left. 2j\sin\left((m-1)\pi\frac{\sin(\bar{v}_i-\Delta)-\sin(\bar{v}_j-\Delta)}{2}\right) \right|^2 \end{aligned} \tag{A.15}$$

$$\approx \Bigg| \frac{\exp\left(j(m-1)\pi \sin\left(\frac{\bar{v}_i+\bar{v}_j}{2}+\Delta\right)\right)}{2\Delta j(m-1)\alpha_1} 2j\sin\left((m-1)\frac{\alpha_1(\bar{v}_i-\bar{v}_j)}{2}\right) - \frac{\exp\left(j(m-1)\pi \sin\left(\frac{\bar{v}_i+\bar{v}_j}{2}-\Delta\right)\right)}{2\Delta j(m-1)\alpha_2} 2j\sin\left((m-1)\frac{\alpha_2(\bar{v}_i-\bar{v}_j)}{2}\right)\Bigg|^2 \quad \text{(A.16)}$$

Note that for $\bar{v}_i - \bar{v}_j \approx 0$, by using first order Taylor expansion, we arrive at

$$\sin\left((m-1)\frac{\alpha_1(\bar{v}_i-\bar{v}_j)}{2}\right) \approx (m-1)\frac{\alpha_1(\bar{v}_i-\bar{v}_j)}{2}$$

and

$$\sin\left((m-1)\frac{\alpha_2(\bar{v}_i-\bar{v}_j)}{2}\right) \approx (m-1)\frac{\alpha_2(\bar{v}_i-\bar{v}_j)}{2}$$

for all $m \in \{1, ..., M\}$. The expression in (A.16) can then be approximated as

$$\approx \Bigg| \frac{\exp\left(j(m-1)\pi \sin\left(\frac{\bar{v}_i+\bar{v}_j}{2}+\Delta\right)\right)}{2\Delta j(m-1)\alpha_1} 2j\left((m-1)\frac{\alpha_1(\bar{v}_i-\bar{v}_j)}{2}\right) - \frac{\exp\left(j(m-1)\pi \sin\left(\frac{\bar{v}_i+\bar{v}_j}{2}-\Delta\right)\right)}{2\Delta j(m-1)\alpha_2} 2j\left((m-1)\frac{\alpha_2(\bar{v}_i-\bar{v}_j)}{2}\right)\Bigg|^2$$

$$= \left(\frac{\bar{v}_i-\bar{v}_j}{2\Delta}\right)^2 \Bigg| \exp\left(j(m-1)\pi \sin\left(\frac{\bar{v}_i+\bar{v}_j}{2}+\Delta\right)\right) - \exp\left(j(m-1)\pi \sin\left(\frac{\bar{v}_i+\bar{v}_j}{2}-\Delta\right)\right)\Bigg|^2$$

$$= \left(\frac{\bar{v}_i-\bar{v}_j}{2\Delta}\right)^2 \left|\exp\left[\frac{j(m-1)\pi}{2}\big(\sin(\frac{\bar{v}_i+\bar{v}_j}{2}+\Delta)+\sin(\frac{\bar{v}_i+\bar{v}_j}{2}-\Delta)\big)\right] 2j\sin\left[\frac{(m-1)\pi}{2}\Big(\sin(\frac{\bar{v}_i+\bar{v}_j}{2}+\Delta)-\sin(\frac{\bar{v}_i+\bar{v}_j}{2}-\Delta)\Big)\right]\right|^2$$

$$= \left(\frac{\bar{v}_i-\bar{v}_j}{2\Delta}\right)^2 \left[2\sin\left(\frac{(m-1)\pi}{2}\Big(\sin(\frac{\bar{v}_i+\bar{v}_j}{2}+\Delta)-\sin(\frac{\bar{v}_i+\bar{v}_j}{2}-\Delta)\Big)\right)\right]^2$$

$$= \left(\frac{\bar{v_i} - \bar{v_j}}{\Delta}\right)^2 \left[\sin\left(\frac{(m-1)\pi}{2}\left(\sin(\frac{\bar{v_i}+\bar{v_j}}{2}+\Delta) - \sin(\frac{\bar{v_i}+\bar{v_j}}{2}-\Delta)\right)\right)\right]^2.$$

Let us define $\Delta_{\sin} := \sin\left(\frac{\bar{v_i}+\bar{v_j}}{2}+\Delta\right) - \sin\left(\frac{\bar{v_i}+\bar{v_j}}{2}-\Delta\right)$. Then, one can write

$$\left\|\mathbf{p}_{UL}{}^i - \mathbf{p}_{UL}{}^j\right\|^2 = \sum_{m=1}^{M} \left|[\mathbf{p}_{UL}{}^i]_m - [\mathbf{p}_{UL}{}^j]_m\right|^2$$
$$\approx \left(\frac{\bar{v_i} - \bar{v_j}}{\Delta}\right)^2 \sum_{m=1}^{M}\left(\sin\left((m-1)\pi\frac{\Delta_{\sin}}{2}\right)\right)^2. \quad \text{(A.17)}$$

Similarly, one can approximate $\|\mathbf{p}_{DL}{}^i - \mathbf{p}_{DL}{}^j\|^2$ as the following:

$$\|\mathbf{p}_{DL}{}^i - \mathbf{p}_{DL}{}^j\|^2 = \sum_{m=1}^{M} |[\mathbf{p}_{DL}{}^i]_m - [\mathbf{p}_{DL}{}^j]_m|^2$$
$$\approx \left(\frac{\bar{v_i} - \bar{v_j}}{\Delta}\right)^2 \sum_{m=1}^{M}\left(\sin\left(f_R(m-1)\pi\frac{\Delta_{\sin}}{2}\right)\right)^2, \quad \text{(A.18)}$$

where $f_R := \frac{f_{DL}}{f_{UL}}$ is the ratio between the DL and UL carrier frequencies. We have

$$\frac{\|\mathbf{p}_{DL}{}^i - \mathbf{p}_{DL}{}^j\|^2}{\|\mathbf{p}_{UL}{}^i - \mathbf{p}_{UL}{}^j\|^2}$$
$$\approx \frac{(\frac{\bar{v_i}-\bar{v_j}}{\Delta})^2 \sum_{m=1}^{M}(\sin(f_R(m-1)\pi\frac{\Delta_{\sin}}{2}))^2}{(\frac{\bar{v_i}-\bar{v_j}}{\Delta})^2 \sum_{m=1}^{M}(\sin((m-1)\pi\frac{\Delta_{\sin}}{2}))^2}$$
$$= \frac{\sum_{m=1}^{M}(\sin(f_R(m-1)\pi\frac{\Delta_{\sin}}{2}))^2}{\sum_{m=1}^{M}(\sin((m-1)\pi\frac{\Delta_{\sin}}{2}))^2}, \quad \text{(A.19)}$$

where

$$\Delta_{\sin} = \sin\left(\frac{\bar{v_i}+\bar{v_j}}{2}+\Delta\right) - \sin\left(\frac{\bar{v_i}+\bar{v_j}}{2}-\Delta\right)$$
$$= 2\cos\left(\frac{\bar{v_i}+\bar{v_j}}{2}\right)\sin(\Delta). \quad \text{(A.20)}$$

Let $C := \cos\left(\frac{\bar{v_i}+\bar{v_j}}{2}\right)$ and $b := C\sin(\Delta)$. Then, Δ_{sin} can be written as

$$\Delta_{\sin} = \sin\left(\frac{\bar{v_i}+\bar{v_j}}{2}+\Delta\right) - \sin\left(\frac{\bar{v_i}+\bar{v_j}}{2}-\Delta\right)$$
$$= 2\cos\left(\frac{\bar{v_i}+\bar{v_j}}{2}\right)\sin(\Delta) = 2b. \quad \text{(A.21)}$$

Thus, using equations (A.19), (A.20) and (A.21), one can write

$$\frac{\|\mathbf{p}_{DL}{}^i - \mathbf{p}_{DL}{}^j\|^2}{\|\mathbf{p}_{UL}{}^i - \mathbf{p}_{UL}{}^j\|^2} \approx \frac{\sum_{m=1}^{M}(\sin(f_R(m-1)\pi b))^2}{\sum_{m=1}^{M}(\sin((m-1)\pi b))^2}. \quad \text{(A.22)}$$

Let us denote the sine ratio as $R_{\sin} := \frac{\sum_{m=1}^{M}(\sin(f_R(m-1)\pi b))^2}{\sum_{m=1}^{M}(\sin((m-1)\pi b))^2}$. The K constant introduced in Lemma 1 can then be defined as the maximum value that $R_{\sin}$ can take. □

B Proof of Theorem 1

The norm of the difference between an arbitrary test point in the DL CCM dataset and its estimate obtained by the mapping of its UL counterpart via the interpolation function $f(.)$ can be bounded as the following:

$$\begin{aligned}
&\left\| f({\mathbf{r}_{UL}}^{test}) - {\mathbf{r}_{DL}}^{test} \right\| = \\
&\qquad \left\| f({\mathbf{r}_{UL}}^{test}) - \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right. \\
&\qquad\qquad \left. + \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) - {\mathbf{r}_{DL}}^{test} \right\| \\
&\leq \left\| f({\mathbf{r}_{UL}}^{test}) - \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right\| \\
&\qquad + \left\| {\mathbf{r}_{DL}}^{test} - \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right\| \\
&\qquad = \left\| f({\mathbf{r}_{UL}}^{test}) - \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right\| \\
&\qquad\quad + \left\| {\mathbf{r}_{DL}}^{test} - \frac{1}{|A^{UL}|} \sum_{i:{\mathbf{r}_{UL}}^{i} \in A^{UL}} {\mathbf{r}_{DL}}^{i} \right. \\
&\qquad\qquad \left. + \frac{1}{|A^{UL}|} \sum_{i:{\mathbf{r}_{UL}}^{i} \in A^{UL}} {\mathbf{r}_{DL}}^{i} - \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right\| \\
&\leq \left\| f({\mathbf{r}_{UL}}^{test}) - \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right\| \\
&\qquad + \left\| {\mathbf{r}_{DL}}^{test} - \frac{1}{|A^{UL}|} \sum_{i:{\mathbf{r}_{UL}}^{i} \in A^{UL}} {\mathbf{r}_{DL}}^{i} \right\| \\
&\qquad\quad + \left\| \frac{1}{|A^{UL}|} \sum_{i:{\mathbf{r}_{UL}}^{i} \in A^{UL}} {\mathbf{r}_{DL}}^{i} - \frac{1}{|A^{UL}|} \sum_{{\mathbf{r}_{UL}}^{i} \in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right\|
\end{aligned}$$

$$\leq \left\| f(\mathbf{r}_{UL}{}^{test}) - \frac{1}{|A^{UL}|} \sum_{\mathbf{r}_{UL}{}^{i} \in A^{UL}} f(\mathbf{r}_{UL}{}^{i}) \right\|$$
$$+ \left\| \mathbf{r}_{DL}{}^{test} - \frac{1}{|A^{UL}|} \sum_{i:\mathbf{r}_{UL}{}^{i} \in A^{UL}} \mathbf{r}_{DL}{}^{i} \right\|$$
$$+ \frac{1}{|A^{UL}|} \sum_{i:\mathbf{r}_{UL}{}^{i} \in A^{UL}} \left\| \mathbf{r}_{DL}{}^{i} - f(\mathbf{r}_{UL}{}^{i}) \right\|$$

Let us name

$$\left\| f(\mathbf{r}_{UL}{}^{test}) - \frac{1}{|A^{UL}|} \sum_{\mathbf{r}_{UL}{}^{i} \in A^{UL}} f(\mathbf{r}_{UL}{}^{i}) \right\|$$

as (UB-1),

$$\left\| \mathbf{r}_{DL}{}^{test} - \frac{1}{|A^{UL}|} \sum_{i:\mathbf{r}_{UL}{}^{i} \in A^{UL}} \mathbf{r}_{DL}{}^{i} \right\|$$

as (UB-2) and

$$\frac{1}{|A^{UL}|} \sum_{i:\mathbf{r}_{UL}{}^{i} \in A^{UL}} \left\| \mathbf{r}_{DL}{}^{i} - f(\mathbf{r}_{UL}{}^{i}) \right\|$$

as (UB-3).

(UB-1) can be upper bounded by using Lemma 2, which is the adaptation of Lemma 1 in [41] to our study. The proof of Lemma 2 is presented in Appendix C.

Lemma 2. *Let the training sample set contain at least N training samples $\{\boldsymbol{r}_{UL}{}^{i}\}_{i=1}^{N}$ with $\boldsymbol{r}_{UL}{}^{i} \sim \upsilon$. Assume that the interpolation function $f : \mathbb{R}^{1\times 2M-1} \rightarrow \mathbb{R}^{1\times 2M-1}$ is Lipschitz continuous with constant L.*

Let $\boldsymbol{r}_{UL}{}^{test}$ be a test sample drawn from υ independently of the training samples. Let A^{UL} be defined as in (3.5)*.*

Then, for any $\epsilon > 0$, for some $\frac{1}{N\eta_\delta} \leq a < 1$ and $\delta > 0$, with probability at least

$$\left(1 - \exp\left(-2N((1-a)\,\eta_\delta)^2\right)\right)\left(1 - 2\sqrt{2M-1}\exp\left(-\frac{aN\eta_\delta\epsilon^2}{2L^2\delta^2}\right)\right),$$

the set A^{UL} contains at least $aN\eta_\delta$ samples and the distance between the embedding of $\boldsymbol{r}_{UL}^{test}$ and the sample mean of the embeddings of its neighboring training samples is

bounded as

$$\left\| f(\boldsymbol{r}_{UL}{}^{test}) - \frac{1}{|A^{UL}|} \sum_{\boldsymbol{r}_{UL}{}^{i} \in A^{UL}} f(\boldsymbol{r}_{UL}{}^{i}) \right\| \quad \leq \quad L\delta \quad + \quad \sqrt{2M-1}\epsilon. \quad \text{(B.1)}$$

Next, (UB-2) can be bounded by using Lemma 1 as follows:

$$\begin{aligned} &\left\| \mathbf{r}_{DL}{}^{test} - \frac{1}{|A^{UL}|} \sum_{\mathbf{r}_{UL}{}^{i} \in A^{UL}} \mathbf{r}_{DL}{}^{i} \right\| \\ &\qquad = \left\| \frac{1}{|A^{UL}|} \sum_{i:\mathbf{r}_{UL}{}^{i} \in A^{UL}} (\mathbf{r}_{DL}{}^{test} - \mathbf{r}_{DL}{}^{i}) \right\| \\ &\qquad \leq \frac{1}{|A^{UL}|} \sum_{i:\mathbf{r}_{UL}{}^{i} \in A^{UL}} \left\| \mathbf{r}_{DL}{}^{test} - \mathbf{r}_{DL}{}^{i} \right\| \\ &\qquad \leq \frac{1}{|A^{UL}|} \sum_{\mathbf{r}_{UL}{}^{i} \in A^{UL}} K \left\| \mathbf{r}_{UL}{}^{test} - \mathbf{r}_{UL}{}^{i} \right\| \\ &\qquad \leq \frac{1}{|A^{UL}|} |A^{UL}| K\delta = K\delta, \quad \text{(B.2)} \end{aligned}$$

for some constant $K > 0$.

Finally, (UB-3) is the average training error of the points in A^{UL}. Thus, by finding upper bounds on (UB-1) and (UB-2) as in (B.1) and (B.2) respectively, the difference between the test error of any point and the average training error of its neighboring training points can be upper bounded as given in Theorem 1. □

C Proof of Lemma 2

A training sample $\mathbf{r}_{UL}{}^{i}$ drawn independently from $\mathbf{r}_{UL}{}^{test}$ lies in a δ-neighborhood of $\mathbf{r}_{UL}{}^{test}$ with probability

$$P\left(\mathbf{r}_{UL}{}^{i} \in B_{\delta}\left(\mathbf{r}_{UL}{}^{test}\right)\right) = \upsilon\left(B_{\delta}\left(\mathbf{r}_{UL}{}^{test}\right)\right) \geq \eta_{\delta}.$$

From [41] and the references therein, one can show that

$$P\left(\left|A^{UL}\right| \geq Q\right) \geq 1 - \exp\left(-\frac{2\left(N\eta_{\delta} - Q\right)^{2}}{N}\right),$$

for $1 \leq Q < N\eta_\delta$. Assuming that $\left|A^{UL}\right| \geq Q$, from [41] and the references therein, one can show that, with probability at least

$$1 \;-\; 2\sqrt{2M-1}\exp\left(-\frac{\left|A^{UL}\right|\epsilon^2}{2L^2\delta^2}\right) \quad \geq \quad 1 \;-\; 2\sqrt{2M-1}\exp\left(-\frac{Q\epsilon^2}{2L^2\delta^2}\right),$$

the distance between the embedding of ${\mathbf{r}_{UL}}^{test}$ and the sample average of the embeddings of training samples lying inside the δ-neighborhood of ${\mathbf{r}_{UL}}^{test}$ is bounded as

$$\left\| f({\mathbf{r}_{UL}}^{test}) - \frac{1}{|A^{UL}|}\sum_{{\mathbf{r}_{UL}}^{i}\in A^{UL}} f({\mathbf{r}_{UL}}^{i}) \right\| \qquad \leq \qquad L\delta \quad + \quad \sqrt{2M-1}\epsilon. \quad \text{(C.1)}$$

Let B_1 be the event that the inequality in (C.1) holds. Combining the probability expressions above,

$$\begin{aligned} P\left(\left(\left|A^{UL}\right| \geq Q\right) \cap B_1\right) &= P\left(\left|A^{UL}\right| \geq Q\right)\; P\left(B_1 \;\middle|\; \left(\left|A^{UL}\right| \geq Q\right)\right) \\ &\geq \left(1-\exp\left(-\frac{2(N\eta_\delta - Q)^2}{N}\right)\right)\left(1-2\sqrt{2M-1}\exp\left(-\frac{Q\epsilon^2}{2L^2\delta^2}\right)\right). \quad \text{(C.2)} \end{aligned}$$

Thus, one can see that with probability at least

$$\left(1-\exp\left(-\frac{2(N\eta_\delta - Q)^2}{N}\right)\right)\left(1-2\sqrt{2M-1}\exp\left(-\frac{Q\epsilon^2}{2L^2\delta^2}\right)\right),$$

$\left|A^{UL}\right| \geq Q$ and B_1 occurs. Setting $Q = aN\eta_\delta$ for $0 < a < 1$, one can reach the statement given in Lemma 2. □

D Numerical Analysis About the Constant K:

Since $-1 \leq C \leq 1$, we have $-\sin(\Delta) \leq b \leq \sin(\Delta)$. Since $\sin^2(\cdot)$ is an even function, it is enough to examine only the positive side of the interval, i.e., $0 \leq b \leq \sin(\Delta)$.

We evaluate the K constant for different Δ values (hence, different maximum values of b) for a range of base station antenna numbers, M.

Table D.1 reports the values that K takes for $2 \leq M \leq 1000$ and for different Δ values, where $f_R = 1.0974$ is taken as in our communication scenario.

Table D.1: K Values for $f_R = 1.0974$ and for Different Δ Values

$\Delta(^\circ)$	Corresponding K Value
5	1.0974
10	1.0974
15	1.0974
35	1.0974
45	1.1317
60	1.1893