DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS
CHENNAI – 600036

Enhancing Joint Activity Detection and Channel Estimation in Massive Random Access Systems through Learning-Based Sparse Recovery Techniques: A Comparative Performance Evaluation

A Thesis

Submitted by

NARENDRA SAI KUMAR CHEJARLA

For the award of the degree

Of

MASTER OF TECHNOLOGY

May 2023

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS
CHENNAI – 600036

Enhancing Joint Activity Detection and Channel Estimation in Massive Random Access Systems through Learning-Based Sparse Recovery Techniques: A Comparative Performance Evaluation

A Thesis

Submitted by

NARENDRA SAI KUMAR CHEJARLA

For the award of the degree

Of

MASTER OF TECHNOLOGY

May 2023

THESIS CERTIFICATE

This is to undertake that the Thesis titled **ENHANCING JOINT ACTIVITY DETECTION AND CHANNEL ESTIMATION IN MASSIVE RANDOM ACCESS SYSTEMS THROUGH LEARNING-BASED SPARSE RECOVERY TECHNIQUES: A COMPARATIVE PERFORMANCE EVALUATION**, submitted by me to the Indian Institute of Technology Madras, for the award of **Master of Technology**, is a bona fide record of the research work done by me under the supervision of **Dr. Srikrishna Bhashyam**. The contents of this Thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Chennai 600036 **Narendra Sai Kumar chejarla**

Date: May 2023

Dr. Srikrishna Bhashyam
Research advisor
Professor
Department of Electrical Enginerring
IIT Madras

ACKNOWLEDGEMENTS

I would like to begin by expressing my sincere gratitude to my guide, Dr. Prof. Srikrishna Bhashyam, from the Department of Electrical Engineering at IIT Madras. I am truly thankful for the opportunity to work under his guidance. His patience, valuable feedback, suggestions, and motivations have been instrumental in shaping the success of my project. I am indebted to him for his unwavering support and expertise.

I would also like to extend my heartfelt appreciation to all my friends who have been there for me throughout this journey. Their help and support have been invaluable in overcoming challenges and completing my project successfully. I am grateful for their encouragement, insights, and willingness to lend a helping hand whenever needed.

ABSTRACT

KEYWORDS Massive Random Access; Sparse Recovery; Joint Activity Detection; Model Based Network;Pathloss

Massive random access has emerged as a critical requirement in modern communication systems, including 5G, Internet of Things (IoT), and machine-type communications. This study focuses on addressing the challenges associated with jointly detecting user activity and estimating channels in scenarios where users experience different pathloss levels.

By considering the pathloss model, we modify the conventional MMSE shrinkage function to better adapt to varying pathloss levels. Furthermore, we introduce a model-based neural network to learn the modified MMSE shrinkage function, thereby enhancing the accuracy of user activity detection and channel estimation.

The modified MMSE-MMV-TISTA approach, with varying number of nodes in the neural network, is designed to handle increased input complexity, specifically a higher number of covariance matrices. To evaluate its performance, comprehensive performance evaluations are conducted comparing the modified approach with the existing MMSE-MMV-TISTA method in equal pathloss scenarios. In addition to comparing the two methods, the performance of the learned versions of both approaches is also assessed.

Lastly, we estimate the pathloss using the ML estimator, we generate a comparison plot between the modified MMSE-MMV-TISTA method, which utilizes the estimated pathloss values, and the true pathloss.

CONTENTS

LIST OF TABLES

Table	Caption	Page

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 MASSIVE RANDOM ACCESS

In Massive random access a base station needs to connect with a large number of devices that have irregular patterns of activity. These devices are designed to conserve energy by staying inactive most of the time and only becoming active when needed. As a result, only a small fraction of devices are active at any given moment, making it challenging to establish reliable and efficient communication.

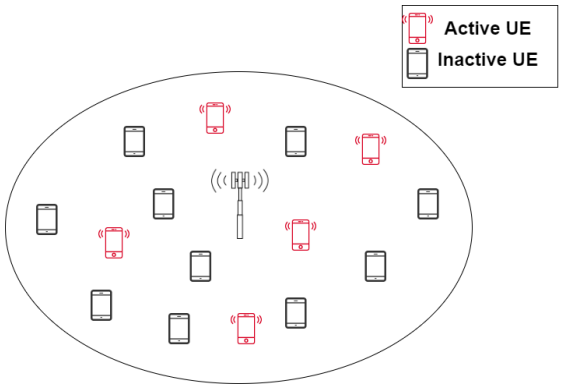

Figure 1.1: Massive Random Access.

In the context of Internet-of-Things (IoT) and machine-type communications (MTC), the base station (BS) encounters the challenge of dynamically identifying active users among a large number of devices. As these devices exhibit sporadic patterns of activity and transmit short data packets, a random access procedure is utilized for data transmission. This procedure involves users transmitting pilot sequences followed by their actual data.

The focus of this study is on addressing the joint problem of detecting device activity and estimating the channel in the context of massive random access. To tackle this problem, sparse recovery algorithms have emerged as effective tools. These algorithms leverage

the sparsity of user activity and exploit channel characteristics to accurately detect active users and estimate their channels.

The proposed methods will enable the BS to effectively identify active users and estimate their channels, thus enhancing the reliability and efficiency of communication in IoT and MTC systems Liu and Yu (2018).

1.2 PATH LOSS

In real-world scenarios, different users often experience different pathloss due to their random locations and varying distances from the base station. As the signal travels from the devices, it undergoes attenuation, causing the power level to decrease. Additionally, the signals from different users may arrive at the base station with different power levels.

To capture the effects of pathloss, there are several ways to model it. One common approach is to use pathloss models that describe the relationship between the distance between the user and the base station and the received signal power. Examples of popular pathloss models include the Free Space Path Loss (FSPL) model, the Two-Ray Ground Reflection model, and the Log-Distance Path Loss model.

These pathloss models consider factors such as the transmission frequency, the environment (e.g., open space or urban area), and the presence of obstacles. By incorporating these models into the analysis and design of communication systems, it is possible to account for the variations in pathloss experienced by different users and optimize system performance accordingly.

1.3 SPARSE RECOVERY AND ACTIVITY DETECTION

In order to estimate the channels and decode the transmitted messages from the active devices, the base station (BS) is required to identify which devices are currently active. This process of activity detection and channel estimation relies on the utilization of pilot

sequences, which are transmitted by the devices before sending their actual data within each coherence time interval.

Given that the activity pattern of the devices is sporadic, the joint problem of activity detection and channel estimation can be framed as a sparse recovery problem. The goal is to accurately recover the sparse activity pattern and estimate the corresponding channel coefficients based on the received pilot sequences. By solving this sparse recovery problem, the BS can effectively identify the active devices and estimate their channels, facilitating reliable communication and data decoding. The theory of CS demonstrates that if the signal vector x, which is to be estimated, possesses sparsity (i.e., a small number of non-zero elements), it is possible to recover the signal using significantly fewer measurements than the size of the original signal. In other words, the measurement vector y can be much smaller than the size of the signal vector x. This fundamental principle of CS has paved the way for efficient and low-dimensional signal recovery techniques, enabling the reconstruction of sparse signals from limited measurements(Zhang *et al.* (2015))

1.3.1 Multiple Measurement Vector (MMV) Problem

The problem of joint activity detection and channel estimation in the presence of a multi-antenna base station (BS) can be mathematically formulated as a multiple-measurement-vector (MMV) problem or a joint-sparse recovery problem. In this context, MMV refers to the scenario where multiple measurement vectors are obtained from different antennas, and the objective is to simultaneously recover the sparse activity patterns and estimate the channels associated with the active users. By exploiting the sparsity of the user activity and the correlation across antennas, the joint-sparse recovery framework allows for more accurate and efficient estimation of the user activity and channel parameters. Various algorithms and techniques have been developed to address the MMV or joint-sparse recovery problem in the context of joint activity detection and channel estimation in wireless communication systems. Figure 1.2 illustrates the case

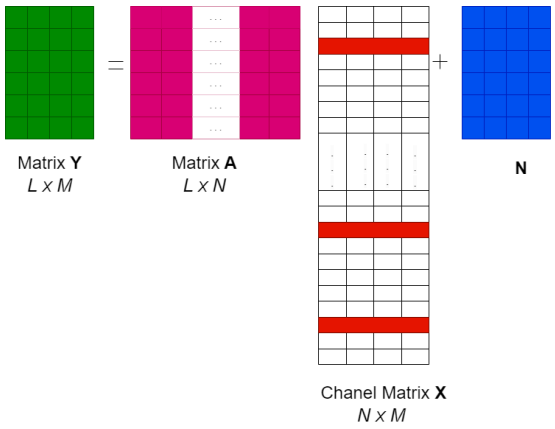

Figure 1.2: Multiple Measurement Vector problem

where the base station (BS) has M antennas and the pilot sequence length is L. In this scenario, the observation Y is represented as a matrix that contains data from all M antennas. Similarly, the unknown X is also a matrix, where each row corresponds to a user in the network. If a user is active, the corresponding row in X represents the channel vector from that user to the M-antenna BS. On the other hand, if a user is inactive, the row in X is filled with zeros. Therefore, the matrix X to be estimated is a row-sparse matrix, where only a subset of rows corresponding to active users contains non-zero entries. This formulation indicates that activity detection and channel estimation can be viewed as a multiple measurement vector problem or a joint sparse recovery problem. In our work, we specifically consider this multi-antenna setting to address the joint activity detection and channel estimation problem.

1.4 CONTRIBUTIONS

In this study, we investigated the performance of an existing learning-based sparse recovery method for joint activity detection and channel estimation in massive random access systems. Specifically, we focused on scenarios where different users experience varying pathloss.

Our findings revealed that the learning-based approach outperforms the conventional

TISTA method in such scenarios. We achieved this by modifying the existing Minimum Mean Squared Error (MMSE) shrinkage function to account for the effects of different pathloss. Additionally, we employed a Maximum Likelihood (ML) estimator to estimate the pathloss values.

The results demonstrated that our modified learning-based method, which incorporates pathloss considerations, significantly improves the accuracy of joint activity detection and channel estimation in massive random access systems.

1.5 OUTLINE OF THE THESIS

This chapter served as an introduction to the concept of massive random access and the different types of sparse recovery algorithms used in this context. We discussed the main objectives of the thesis and highlighted our contributions in addressing the challenges of joint activity detection and channel estimation in massive random access systems. The remaining chapters of this thesis are organized as follows: Chapter 2 presents an overview of the system model used in this thesis, focusing on the massive random access system and the problem of joint activity detection and channel estimation. Additionally, we introduce a modified version of the MMSE-MMV-TISTA method called L-MMSE-MMV-TISTA, which incorporates a learning-based approach.Chapter 3 of the thesis presents the simulation results obtained from the proposed methods. Chapter 4 concludes the thesis by providing a brief summary of the main findings and contributions discussed in the previous chapters

CHAPTER 2

SYSTEM MODEL AND MODIFICATIONS TO EXISTING ALGORITHM

2.1 SYSTEM MODEL

We consider a massive random access system with a single-antenna IoT devices and a M-antenna base station (BS). The devices become active independently with a probability p, resulting in a set of active devices denoted by $\mathcal{S} = \{1, 2, \ldots, S\}$ .Each active user transmits an L-length pilot sequence followed by data. In our study, we assume that the training sequences, $\mathbf{a}_n \in \mathbb{C}^{L\times 1}$, used by the users are designed as i.i.d. complex Gaussian entries with zero mean and unit variance for simplicity.

The complex baseband channel vector between user n and the BS is denoted as $\mathbf{h}_n^T \in \mathbb{C}^{1\times M}$. The channels could be correlated across antennas, and we model the channel vector as $\mathbf{h}_n^T \sim \mathcal{CN}(0, \alpha_n \mathbf{C}_h)$, where $\mathbf{C}_h$ represents the covariance matrix of the channel, and α_n denotes the path-loss and shadowing. The path-loss and shadowing components, which depend on the user location, are assumed to be known at the BS.

The received training signal at the BS is denoted as $\mathbf{Y} \in \mathbb{C}^{L\times M}$ and is modeled as.

$$\mathbf{Y} = \sum_{i\in\mathcal{S}} \mathbf{a}_i \mathbf{h}_i^T + \mathbf{N} \tag{2.1}$$

, where the noise $\mathbf{N}$ has i.i.d. $\mathcal{CN}(0, \sigma^2)$ entries. For each user n, let $\lambda_n = 0$ if the user is inactive and $\lambda_n = 1$ if the user is active. Then, we can rewrite the above as

$$\mathbf{Y} = \sum_{n=1}^{N} \lambda_n \mathbf{a}_n \mathbf{h}_n^T + \mathbf{N} = \mathbf{A}\mathbf{X} + \mathbf{N} \tag{2.2}$$

where $\mathbf{X} = [\lambda_1 \mathbf{h}_1, \lambda_2 \mathbf{h}_2, \ldots, \lambda_N \mathbf{h}_N]^T \in \mathbb{C}^{N\times M}$ and $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_N]$.Considering that the number of active users, denoted by $\mathcal{S}$, is significantly smaller than the total

number of users N, the matrix $\mathbf{X}$ becomes row sparse. Consequently, the receiver is faced with a joint sparse recovery problem, where the objective is to estimate a row sparse matrix $\mathbf{X}$ based on the observations of $\mathbf{Y}$.

2.2 TRAINABLE ISTA (TISTA)

We provide a brief overview of TISTA for the single measurement vector case ($M = 1$) as presented in the work by Ito et al. (2019). Each iteration (or layer) of TISTA is described by the following equations. Here, t represents the iteration number, and the estimate of $\mathbf{X}$ (which is a vector when $M = 1$) at iteration t is denoted as S_t.

$$\begin{aligned} \boldsymbol{r}_t &= \boldsymbol{s}_t + \gamma_t \boldsymbol{W} \left(\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{s}_t \right) \\ \boldsymbol{s}_{t+1} &= \eta_{\mathrm{MMSE}} \left(\boldsymbol{r}_t ; \tau_t^2 \right), \\ v_t^2 &= \max \left\{ \frac{\| \boldsymbol{Y} - \boldsymbol{A}\boldsymbol{s}_t \|_2^2 - L\sigma^2}{\mathrm{tr}\left(\boldsymbol{A}^T \boldsymbol{A} \right)}, \epsilon \right\} \\ \tau_t^2 &= \frac{v_t^2}{N} \left(N + \left(\gamma_t^2 - 2\gamma_t \right) L \right) + \frac{\gamma_t^2 \sigma^2}{N} \mathrm{tr}\left(\boldsymbol{W}\boldsymbol{W}^T \right), \end{aligned} \tag{2.3}$$

In the TISTA algorithm, the matrix $\mathbf{W}$ represents the pseudo-inverse of the detection matrix $\mathbf{A}$, and ϵ is a small constant. The initial condition $\mathbf{s}_0$ can be initialized as zero, and the final estimate of $\mathbf{X}$ after T iterations is denoted as $\mathbf{s}_T$. The scalar variables γ_t for $t = 0, 1, \ldots, T-1$ can be learned and adjusted during the learning process. The function $\eta_{\mathrm{MMSE}}(\cdot;\cdot)$ corresponds to the MMSE contraction function derived based on the aforementioned assumptions about $\mathbf{X}$.

In the study by Ito *et al.* (2019), $\mathbf{X}$ is considered as a vector of i.i.d. Bernoulli-Gaussian random variables. Each entry of $\mathbf{X}$ follows a Gaussian distribution with mean zero and variance α^2 with probability p, and it is zero with probability $1 - p$. In each iteration, $\mathbf{r}_t$ is modeled as $\mathbf{r}_t = \mathbf{X} + \mathbf{Z}$, where $\mathbf{Z}$ follows a Gaussian distribution with mean zero and variance $\tau^2\mathbf{I}$. For this particular model, the MMSE noise canceller can be implemented

to denoise the input.

$$\eta\left(y;\tau^2\right) = \frac{y\alpha^2}{\alpha^2+\tau^2} \cdot \frac{pF\left(y;\alpha^2+\tau^2\right)}{(1-p)F\left(y;\tau^2\right)+pF\left(y;\alpha^2+\tau^2\right)} \tag{2.4}$$

where

$$F(z;v) = \frac{1}{\sqrt{2\pi v}} \exp\left(\frac{-z^2}{2v}\right) \tag{2.5}$$

Please note that $\{\gamma_t\}$, p, and α^2 are learned from the training data in TISTA, while σ^2 is assumed to be known. τ^2 is estimated in each iteration. The sparse recovery problem with complex vectors and matrices in equation (2.1) can be converted to an equivalent real-valued problem, as done in several other works (Borgerding *et al.* (2017); Li *et al.* (2019);Sabulal and Bhashyam (2020)) that employ deep learning-based optimizers proposed by Kingma and Ba (2017), which operate with real-valued parameters. Hence, all the algorithms presented in this thesis are designed for the real-valued case.

2.3 MODIFIED MMSE-MMV-TISTA

To handle the MMV model in equation (2.1) with $M > 1$ antennas, one approach is to apply TISTA using equations (2.2)-(2.5) by vectorizing $\mathbf{Y}$ and treating it as a single measurement vector (SMV) model. However, this method treats each entry in the vectorized $\mathbf{X}$ independently and does not take advantage of the row-sparsity structure of $\mathbf{X}$, the correlation structure of each $\mathbf{h}_n$,and the path loss of individual user α_n . To exploit these structures, the MMSE shrinkage step in equation (2.3) needs to be modified. Before discussing this modification, let us briefly explain how the other two steps of TISTA are adapted to our problem. .The linear estimation step in (2.2) is modified to

$$\mathbf{R}_t = \mathbf{S}_t + \gamma_t \mathbf{W}(\mathbf{Y} - \mathbf{A}\mathbf{S}_t) \tag{2.6}$$

where $\mathbf{R}_t$ and $\mathbf{S}_t$ are now $N \times M$ matrices. The error variance estimation in (2.4) can be used either with the vectorized model or simply with the observation from any one of the antennas.

The MMSE shrinkage function can be derived as follows. Let $\mathbf{x}_i^T$ represent the i-th row of matrix $\mathbf{X}$, which corresponds to the received signal vector from the i-th user at the M receive antennas. If user i is active, $\mathbf{x}_i$ is distributed as complex Gaussian with mean zero and covariance matrix $\alpha_i \mathbf{C}_h$, and if user i is not active, $\mathbf{x}_i$ is zero. It is assumed that the correlation matrix $\mathbf{C}_h$ is different for all users and arises due to the mutual coupling between receive antennas. The channel vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ are assumed to be independent for $i \neq j$. Therefore, the MMSE denoiser, denoted by η_{MMSE}, is given by

$$\mathbf{S}_{t+1} = \eta_{\text{MMSE}}(\mathbf{R}_t; \tau_t^2) \tag{2.7}$$

is used to estimate the signal matrix $\mathbf{S}_{t+1}$ from the received signal matrix $\mathbf{R}_t$ and the noise variance τ_t^2 at iteration t.

Let $\mathbf{y}_i^T$ represent the ith row of $\mathbf{R}_t$, and $\hat{\mathbf{x}}_i^T$ represent the ith row of $\mathbf{S}_{t+1}$. In order to obtain $\hat{\mathbf{x}}_i$, the MMSE estimate of $\mathbf{x}_i$, we model $\mathbf{y}_i = \mathbf{x}_i + \mathbf{z}_i$, where $\mathbf{z}_i \sim \mathcal{N}(0, \tau_t^2 \mathbf{I})$ and $\mathbf{x}_i$ is a Bernoulli-Gaussian random vector as described above. we can obtain the row-wise **MMSE** estimator using approach in Neumann *et al.* (2018) and Shiv *et al.* (2022).

In our scenario, we have a total of T observations denoted as $\mathbf{Y}$, where each observation $\mathbf{y}_t$ is given by $\mathbf{y}_t = \mathbf{x}_t + \mathbf{z}_t$. The noise term $\mathbf{z}_t$ follows a complex Gaussian distribution $\mathbf{z}_t \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$, and the channel vector $\mathbf{h}_t$ is conditionally Gaussian with a parameter δ, i.e., $\mathbf{h}_t \sim \mathcal{N}(0, \mathbf{C}_\delta)$. The observations at different intervals t are independent. In our specific case, $T = 1$, meaning we have a single observation. Each observation $\mathbf{y}_i$ is given by $\mathbf{y}_i = \mathbf{x}_i + \mathbf{z}_i$, where $\mathbf{z}_i$ follows a Gaussian distribution $\mathbf{z}_i \sim \mathcal{N}(0, \tau_t^2 \mathbf{I})$, and $\mathbf{x}_i$ is distributed as a complex Gaussian $\mathbf{x}_i \sim C\mathcal{N}(0, \alpha_i \mathbf{C}_h)$ with probability p, and is zero with probability $1 - p$.

$$\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_T\}$$

$$\mathbf{y}_t = \mathbf{h}_t + \mathbf{z}_t, \quad \mathbf{z}_t \sim \mathcal{N}(0, \sigma^2\mathbf{I}), \quad \mathbf{h}_t \sim \mathcal{N}(0, \mathbf{C}_\delta)$$

$$\mathbf{y}_i = \mathbf{x}_i + \mathbf{z}_i, \quad \mathbf{z}_i \sim \mathcal{N}(0, \tau_t^2\mathbf{I}), \quad \mathbf{x}_i \sim C\mathcal{N}(0, \alpha_\mathbf{i}\mathbf{C}_h),$$

where $\mathbf{C}_\delta$ and $\mathbf{C}_h$ represent the covariance matrices associated with the parameter δ and the channel vector $\mathbf{h}$, respectively.

The MMSE estimate of the channel vector $\mathbf{x}_i$ conditioned on δ is given by

$$E[\mathbf{x}_i|\mathbf{Y}, \delta] = \mathbf{W}_\delta \mathbf{y}_i \tag{2.8}$$

where

$$\mathbf{W}_\delta = \mathbf{C}_\delta(\mathbf{C}_\delta + \sigma^2\mathbf{I})^{-1} \tag{2.9}$$

This estimate of $\mathbf{x}_i$ depends only on $\mathbf{y}_i$ if the parameters δ and the covariance matrix $\mathbf{C}_\delta$ are available. Now, let us consider the case where δ is an unknown random variable following the distribution $p(\delta)$. In this case, the MMSE estimator, which is the conditional mean of $\mathbf{x}_i$ given $\mathbf{Y}$, can be written as:

$$E[\mathbf{x}_i|\mathbf{Y}] = E[E[\mathbf{x}_i|\mathbf{Y}, \delta]|\mathbf{Y}] = E[\mathbf{W}_\delta \mathbf{y}_i|\mathbf{Y}] = E[\mathbf{W}_\delta|\mathbf{Y}]\mathbf{y}_i = \mathbf{W}^*(\mathbf{Y})\mathbf{y}_i.$$

Using Bayes' theorem, the posterior distribution of δ can be expressed as

$$p(\delta|\mathbf{Y}) = \frac{p(\mathbf{Y}|\delta)p(\delta)}{\int p(\mathbf{Y}|\delta)p(\delta)d\delta} \tag{2.10}$$

Now, we can write the MMSE filter $\mathbf{W}^*(\mathbf{Y})$ as

$$\hat{\mathbf{W}}^*(\mathbf{Y}) = \int p(\delta|\mathbf{Y})\mathbf{W}_\delta d\delta = \frac{\int p(\mathbf{Y}|\delta)\mathbf{W}_\delta p(\delta)d\delta}{\int p(\mathbf{Y}|\delta)p(\delta)d\delta} \tag{2.11}$$

In Neumann *et al.* (2018), the equation for the MMSE filter $\mathbf{W}^*$ in (2.11) is simplified for the case where the noise covariance matrix is $\Sigma = \sigma^2\mathbf{I}$ as follows:

$$\mathbf{W}^*(\hat{\mathbf{C}}) = \frac{\int \exp\left\{\text{tr}(\mathbf{W}_\delta\hat{\mathbf{C}}) + T\log|\mathbf{I} - \mathbf{W}_\delta|\right\}\mathbf{W}_\delta p(\delta)d\delta}{\int \exp\left\{\text{tr}(\mathbf{W}_\delta\hat{\mathbf{C}}) + T\log|\mathbf{I} - \mathbf{W}_\delta|\right\}p(\delta)d\delta} \tag{2.12}$$

where $\mathbf{W}_\delta$ is given by (2.9) and $\hat{\mathbf{C}}$ is given as:

$$\hat{\mathbf{C}} = \frac{1}{\sigma^2}\sum_{t=1}^{T}\mathbf{y}_t\mathbf{y}_t^H \tag{2.13}$$

Note that the MMSE estimator depends on $\mathbf{Y}$ only through the scaled sample covariance matrix. It should be noted here that $\mathbf{C}_\delta$ is considered to be known while deriving the equation. We can then estimate $\mathbf{x}_t$ as:

$$\hat{\mathbf{x}}_t = \mathbf{W}^*(\hat{\mathbf{C}})\mathbf{y}_t \tag{2.14}$$

The prior $p(\delta)$ in our system model is discrete, $\{\delta_i : i = 1, 2\}$. For discrete and uniform $p(\delta)$ with $p(\delta_i) = \frac{1}{N}$, for all $i = 1, \ldots, N$, the MMSE estimator can be calculated as (Neumann *et al.* (2018)):

$$\widehat{\boldsymbol{W}}(\widehat{\boldsymbol{C}}) = \frac{\frac{1}{N}\sum_{i=1}^{N}\exp\left(\text{tr}\left(\boldsymbol{W}_{\boldsymbol{\delta}_i}\widehat{\boldsymbol{C}}\right) + b_i\right)\boldsymbol{W}_{\boldsymbol{\delta}_i}}{\frac{1}{N}\sum_{i=1}^{N}\exp\left(\text{tr}\left(\boldsymbol{W}_{\boldsymbol{\delta}_i}\widehat{\boldsymbol{C}}\right) + b_i\right)} \tag{2.15}$$

where $\mathbf{W}_{\delta_i}$ can be obtained by evaluating (2.9) for $\delta = \delta_i$.

In our setting, the probabilities for the $\delta_\mathbf{i}$'s are unequal. In this case, we can derive the estimator to be:

$$\hat{\boldsymbol{x}}_i = \boldsymbol{W}(\hat{\boldsymbol{C}})\boldsymbol{y}_i, \text{with}\, \hat{\boldsymbol{C}} = \frac{1}{\tau_t^2}\boldsymbol{y}_i\boldsymbol{y}_i^T \tag{2.16}$$

$$\widehat{\boldsymbol{W}}(\widehat{\boldsymbol{C}}) = \frac{\sum_{i=1}^{N} p_i \exp\left(\text{tr}\left(\boldsymbol{W}_{\delta_i}\widehat{\boldsymbol{C}}\right) + b_i\right)\boldsymbol{W}_{\delta_i}}{\sum_{i=1}^{N} p_i \exp\left(\text{tr}\left(\boldsymbol{W}_{\delta_i}\widehat{\boldsymbol{C}}\right) + b_i\right)} \tag{2.17}$$

where

$$\boldsymbol{W}_{\delta_i} = \alpha_i \boldsymbol{C}_h \left(\alpha_i \boldsymbol{C}_h + \tau_t^2 \boldsymbol{I}\right)^{-1} \tag{2.18}$$

$$b_i = \log|\mathbf{I} - \boldsymbol{W}_{\delta_i}| \tag{2.19}$$

The MMSE denoiser was derived under the assumption of prior knowledge of the pathloss.

2.4 L-MMSE-MMV-TISTA

Both the proposed "learnt MMSE-MMV-TISTA" version by Shiv *et al.* (2022) and the modified method demonstrate no differences. The learnt version closely adheres to the original approach, indicating that no modifications or alterations have been introduced during the learning process. This observation affirms the consistency between the learnt version and the modified method.

The learned version, referred to as L-MMSE-MMV-TISTA, offers several advantages. Firstly, it eliminates the need for prior knowledge of $\mathbf{C_h}$ and p, as these parameters are

learned implicitly during the training phase. Secondly, it does not require individual user's pathloss information, making it more flexible and applicable in scenarios where such information may be unavailable. Lastly, the inaccuracies arising from the model assumptions in the linear estimation step, specifically the estimation $\mathbf{R_t} = \mathbf{X} + \mathbf{Z}$, can potentially be mitigated through the training process, enhancing the overall performance and robustness of the algorithm.

Let vec($\mathbf{X}$) denote the column vector obtained by stacking the columns of $\mathbf{X}$. Simplifying (2.16), we get:

$$\text{vec}(\mathbf{W}(\hat{\mathbf{C}})) = \frac{\mathbf{A_W} \exp(\mathbf{A_W^T} \text{vec}(\hat{\mathbf{C}}) + \mathbf{b})}{1^T \exp(\mathbf{A_W^T} \text{vec}(\hat{\mathbf{C}}) + \mathbf{b})} \tag{2.20}$$

$\mathbf{A_w} = [\text{vec}(\mathbf{W}_{\delta_1}), \ldots, \text{vec}(\mathbf{W}_{\delta_F})] \in \mathbb{C}^{M^2 \times F}$ and $\mathbf{b} = [b_1, \ldots, b_F]$ where $b_i = \log \left|\mathbf{I} - \mathbf{W}_{\delta_i}\right| + \log p_i$ and F represents numbe of nodes in the hidden layer. In our scenario, the rows of $\mathbf{X}$ are generated from a Gaussian distribution with many potential covariance matrices: zero with a probability of $1 - p$, and $\alpha_i \mathbf{C}_h$ with a probability of p_i(sum of $p_i = p$).

In our scenario, the exact probabilities associated with the different covariance matrices are unknown due to the variability in pathloss and the generation of α_i with specific probabilities. To address this, we aim to approximate the continuous probability function using discrete samples. This involves increasing the dimensions of the weight matrix, allowing it to incorporate the covariance matrix that corresponds to the most probable pathloss. The equation (2.18) is derived by incorporating unequal probabilities for the different possible covariance matrices. This is achieved by adding the term $\log p_i$ to b_i, where p_i represents the probability of covariance matrix i.We do not have exact knowledge of the probabilities involved, as they depend on the distribution of pathloss and the generation of $\alpha_\mathbf{i}$.

The structure described in (2.18) resembles that of a feed-forward neural network consisting of two linear layers connected by a softmax activation function. This denoiser

structure, depicted inside a box in Figure 2.1, can be simplified to (2.18) when we set $\mathbf{A_1} = \mathbf{A_W^T}$, $\mathbf{b_1} = \mathbf{b}$, and $\mathbf{A_1} = \mathbf{A_W}$. In this network, the input is $\text{vec}(\hat{\mathbf{C}})$ and the output is $\text{vec}(\boldsymbol{W}(\hat{\mathbf{C}}))$.

Figure 2.1 illustrates the computations involved in a single iteration of the learnt MMSE-MMVTISTA method(Shiv *et al.* (2022)). The trainable parameters for each iteration, including γ_t, $\mathbf{A_1}$, $\mathbf{b_1}$, and $\mathbf{A_2}$, are highlighted in red. It should be noted that the same network is utilized to denoise each row of $\mathbf{R_t}$, with $\hat{\mathbf{C}}$ computed as described in (2.13) for each row. Consequently, the total number of trainable parameters in one iteration is $4M^2 + 3$, which is expected due to the unknown covariance matrix $\mathbf{C}_h$ being an $M \times M$ matrix. However, the number of parameters can be reduced through the following approaches: (i) By leveraging the knowledge that one of the covariances is zero (when the user is not active), the structure of $\mathbf{A_W}$ in (2.18) can be modified accordingly; (ii) If the covariance matrix exhibits a Toeplitz structure (common in many practical scenarios), the number of parameters can be further reduced.

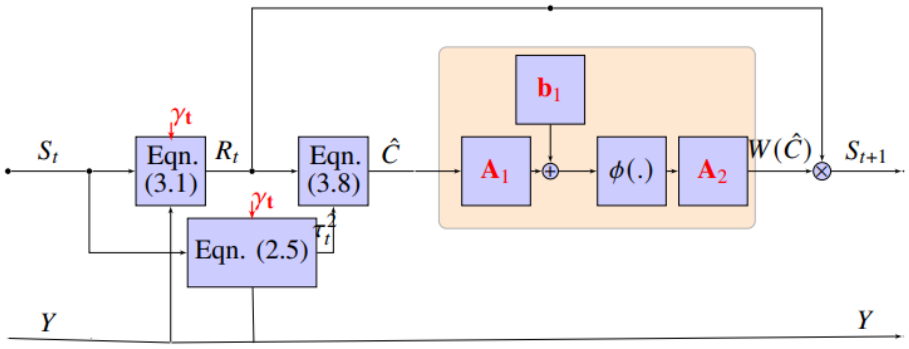

Figure 2.1: One Iteration of L-MMSE-MMV-TISTA [8]

2.5 SNR DERIVATION

2.5.1 Signal Power

(l,m) element in **AX** matrix is

$$\mathbf{AX}_{l,m} = \sum_{j=1}^{N} a_{lj} x_{jm} \alpha_j$$

$$E[|\mathbf{AX}_{l,m}|_F^2] = E[\sum_{j=1}^{N} |a_{lj} x_{jm} \alpha_j|^2]$$

a_{lj} is independent of x_{jm} and α_j

$$E[|\mathbf{AX}_{l,m}|_F^2] = \sum_{j=1}^{N} E[|a_{lj}|^2] E[|x_{jm}|^2] E[|\alpha_j|^2]$$

a_{lj} is normally distributed with mean and standard deviation $1/M$:

$$a_{lj} \sim \mathcal{CN}(\ 0,\ \ 1/M)$$

$$E[|a_{lj}|^2] = 1/M$$

x_{jm} is Bernoulli-Gaussian distribution

$$E[|x_{jm}|^2] = p$$

$$E[|\mathbf{AX}_{l,m}|_F^2] = \frac{p}{M} \sum_{j=1}^{N} E[|\alpha_j|^2]$$

$$E[|\mathbf{AX}|_F^2] = \sum_{l=1}^{L} \sum_{m=1}^{M} E[|\mathbf{AX}_{l,m}|_F^2] = \frac{p}{M} \sum_{l=1}^{L} \sum_{m=1}^{M} \sum_{j=1}^{N} E[|\alpha_j|^2]$$

$$E[|\mathbf{AX}|_F^2] = pL \sum_{j=1}^{N} E[|\alpha_j|^2]$$

2.5.2 Noise Power

n_{ij} is normally distributed with mean and standard deviation σ^2:

$$n_{lj} \sim \mathcal{CN}(\ 0,\ \sigma^2)$$

$$E[|\mathbf{N}|^2] = \sum_{i=1}^{L} \sum_{k=1}^{M} E[|n_{ij}|^2]$$

$$E[|\mathbf{N}|^2] = LM\sigma^2$$

2.5.3 SNR

$$SNR = \frac{p \sum_{j=1}^{N} E[|\alpha_j|^2]}{M\sigma^2} \tag{2.21}$$

2.6 ESTIMATOR FOR PATHLOSS

1. Assume that the received signal R follows the distribution:

$$R = \sqrt{\alpha}X + Z \tag{2.22}$$

Where $X \sim \mathcal{N}(0, C_h)$represents the signal component and, $Z \sim \mathcal{N}(0, \tau^2 I)$ represents the noise component.

2. The likelihood function L() is given by the probability density function (PDF) of the received signal R, which can be expressed as:

$$L(\alpha) = \frac{1}{\sqrt{(2\pi)^n |\alpha C_h + \tau^2 I|}} \exp\left(-\frac{1}{2} R^T (\alpha C_h + \tau^2 I)^{-1} R\right) \tag{2.23}$$

where n is the dimension of the received signal.

3. To find the MLE for , we maximize the likelihood function L() with respect to . Taking the logarithm of L() to simplify the calculations, we have:

$$\log L(\alpha) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\alpha C_h + \tau^2 I| - \frac{1}{2}R^T(\alpha C_h + \tau^2 I)^{-1}R \tag{2.24}$$

4. Using eigenvalue decomposition, we can express the matrices Ch and XXT as:

$$C_h = V\Lambda V^T \tag{2.25}$$

where V is unitary matrix containing the eigenvectors, and is the diagonal matrix containing the eigenvalues.

5.Substituting the eigendecomposition into the equation, we get:

$$\log L(\alpha) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\alpha\Lambda + \tau^2 I| - \frac{1}{2}R'^T(\alpha\Lambda + \tau^2 I)^{-1}R' \tag{2.26}$$

where R'=VR

6. To find the minimum, we take the derivative of the logarithm of the likelihood function with respect to and set it to zero:

$$\frac{\partial}{\partial\alpha}\log L(\alpha) = \frac{\partial}{\partial\alpha}\left(\sum_{j=1}^{n}\log(\alpha\Lambda_j + \tau^2) - R_j'^T(\alpha\Lambda_j + \tau^2)^{-1}R_j'\right) \tag{2.27}$$

$$= \sum_{j=1}^{n}\frac{\partial}{\partial\alpha}\log(\alpha\Lambda_j + \tau^2) - \frac{\partial}{\partial\alpha}\left(R_j'^T(\alpha\Lambda_j + \tau^2)^{-1}R_j'\right) \tag{2.28}$$

Now, let's focus on the two terms separately:

1. For the first term:

$$\frac{\partial}{\partial \alpha} \log(\alpha \Lambda_j + \tau^2) = \frac{1}{\alpha \Lambda_j + \tau^2} \cdot \Lambda_j \tag{2.29}$$

$$= \frac{\Lambda_j}{\alpha \Lambda_j + \tau^2} \tag{2.30}$$

2. For the second term:

$$\frac{\partial}{\partial \alpha} \left({R'_j}^T (\alpha \Lambda_j + \tau^2)^{-1} R'_j \right) = -{R'_j}^T (\alpha \Lambda_j + \tau^2)^{-2} \cdot \Lambda_j \cdot R'_j \tag{2.31}$$

$$= -\frac{\Lambda_j}{(\alpha \Lambda_j + \tau^2)^2} \cdot {R'_j}^T R'_j \tag{2.32}$$

Now, substituting these derivatives back into the expression, we have:

$$\frac{\partial}{\partial \alpha} \log L(\alpha) = \sum_{j=1}^{n} \frac{\Lambda_j}{\alpha \Lambda_j + \tau^2} - \frac{\Lambda_j}{(\alpha \Lambda_j + \tau^2)^2} \cdot {R'_j}^T R'_j \tag{2.33}$$

$$= \sum_{j=1}^{n} \frac{\Lambda_j (\alpha \Lambda_j + \tau^2) - \Lambda_j {R'_j}^T R'_j}{(\alpha \Lambda_j + \tau^2)^2} \tag{2.34}$$

Setting this derivative to zero, we can solve for the maximum likelihood estimator of .

2.7 TRAINING

There is no difference in the training process compared to L-MMSE-MMV-TISTA proposed by Shiv *et al.* (2022). However, there is a distinction in the signal generation stage. In our approach, we initially generate the pathloss values for individual users using a prior model. We then take the square root of the generated pathloss values. The channel matrix is generated using a Bernoulli-Gaussian process. Finally, we multiply the pathloss values and the channel matrix to obtain new channel matrix with the pathloss.

The recursive step of L-MMSE-MMV-TISTA can be unfolded to reveal a signal-flow graph that resembles a multi-layer feed-forward neural network. Figure 2.2 illustrates a unit of the signal-flow graph representing the t-th iteration of L-MMSE-MMV-TISTA.

By concatenating these units together, the entire network can be constructed.

This configuration can be identified as a feed-forward neural network (NN) featuring two linear layers connected by a non-linear activation function. In this specific network, the activation function employed is the softmax function: $\varphi(x) = \frac{\exp(x)}{1^T \exp(x)}$ and specific parameter choices: $A^{(1)} = A_{\mathrm{W}}^T$, $A^{(2)} = A_{\mathrm{W}}$, $b^{(1)} = b$, and $b^{(2)} = 0$.

To mathematically formulate the learning problem, we define the set

$$f(\cdot) : \mathbb{C}^{M^2} \to \mathbb{C}^{M^2}, \quad \boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{A}_2 \boldsymbol{\Phi}(\boldsymbol{A}_1 x + \mathbf{b}_1) + \mathbf{b}_2,$$

where $\boldsymbol{A}_1 \in \mathbb{C}^{N \times M^2}$, $\boldsymbol{A}_2 \in \mathbb{C}^{M^2 \times N}$, $\boldsymbol{b}_1 \in \mathbb{C}^N$, and $\boldsymbol{b}_2 \in \mathbb{C}^{M^2}$.

Here, N and M represent the dimensions of the matrices.

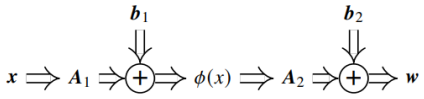

Figure 2.2: One Iteration of L-MMSE-MMV-TISTA [4].

To realize a system with good recovery performance, the trainable variables $[\gamma_t]_{t=T-1}^{t=0}$, $\boldsymbol{A}_1$, $\boldsymbol{b}_1$, and $\boldsymbol{A}_2$ need to be properly trained. We divide our training data, i.e., randomly generated pairs (X, Y), into mini-batches and use the Adam optimizer (Kingma and Ba, 2014). Based on the experiments in Ito et al (2019),, it has been observed that incremental training of each layer provides superior performance while learning parameters as it can reduce the effect of the vanishing gradient. The loss function is the mean squared error between the output from each layer $\mathbf{X}_{k+1}$ and the true $\mathbf{X}$. The neural network system derives gradients and updates weights from this loss function.

The L-MMSE-MMV-TISTA estimator performs the following optimization:

$$\mathrm{vec}(W(\cdot)) = \arg \min_{\mathrm{vec}(\hat{W}(\cdot)) \in W_{\mathrm{NN}}} \epsilon(\hat{W}(\cdot)) \tag{2.35}$$

Notably, this optimization formula does not rely on covariance matrices or any other analytical data, enabling the network to be trained using a large dataset of channel realizations and corresponding observations. Our training data consists of samples $\mathbf{X}$ following a prior distributions of $\mathbf{P}_{\mathbf{X}}(x)$,$\mathbf{P}_{\alpha}(\alpha)$, and the observed data $\mathbf{Y}$ is given by $\mathbf{Y} = \mathbf{AX} + \mathbf{N}$, where the observation noise $\mathbf{N}$ is independently and identically distributed (i.i.d.) Gaussian. It is assumed that the network has prior knowledge of the matrix $\mathbf{A}$.

During incremental training, the network is optimized layer-by-layer. The mean squared error (MSE) is minimized by tuning all the variables using an optimizer. In each generation, denoted by the index t, the network is trained with D mini-batches. After completing the training of the t-th layer, a new layer $t + 1$ is added to the network, and the network is trained again with D mini-batches. The variable values obtained after optimization in one generation serve as the initial values for the variables in the next generation's optimization process. Each layer of the network is trained using a batch size of 200, and a total of 200 batches are used. Gradient updates per batch are performed using the ADAM optimizer Kingma and Ba (2017) with a learning rate of 0.04. During the testing phase, the normalized mean squared error (NMSE) is computed using 7500 samples. All training procedures are implemented using the PyTorch framework.

CHAPTER 3

SIMULATION RESULTS AND DISCUSSION

In this chapter, we evaluate the performance of two modified methods called MMSEMMV-TISTA and L-MMSE-MMV-TISTA in a massive random access setting. These methods are compared with the original MMSEMMV-TISTA and L-MMSE-MMV-TISTA algorithms, but with different pathloss values for individual users. Additionally, the performance of these modified algorithms is also compared with the original algorithms using equal pathloss.

In a study conducted by Shiv *et al.* (2022), it was observed that MMSEMMV-TISTA and L-MMSE-MMV-TISTA with equal pathloss outperformed the Trainable ISTA method and reduced the preamble length by 30-40%. Therefore, it is expected that the modified algorithms would also yield a similar reduction in the preamble length by 30-40%, even in scenarios where different pathloss values are considered.

By evaluating the performance of these modified algorithms with different pathloss values and comparing them to the original algorithms with equal pathloss, we aim to assess the impact of pathloss variations on the system's performance and preamble length. This analysis will provide insights into the effectiveness of the modified algorithms in handling different pathloss conditions and their potential for reducing the preamble length in real-world scenarios.

Overall, this chapter focuses on evaluating and comparing the performance of modified algorithms with varying pathloss values, providing valuable information for improving massive random access systems and reducing preamble length in wireless communication networks.

In the conducted simulations, several parameters were set as follows: the number of users N was set to 500, the number of antennas M was 4, and the number of iterations (or layers after unfolding) of each algorithm was set to 12. These choices were consistent with previous studies such as those by Shiv *et al.* (2022)).

The signal-to-noise ratio (SNR) was defined as the ratio of the expected value of the squared norm of the transmitted signal $\|\mathbf{AX}\|^2$ to the expected value of the squared norm of the noise $\|\mathbf{N}\|^2$. This definition provides a measure of the signal strength relative to the background noise in the system.

The covariance matrix $\alpha_i \mathbf{C}_h$, which characterizes the channel conditions, had entries given by $\alpha_i \rho^{|i-j|}$, where i and j represent the indices of the entries. This choice of covariance matrix is based on previous works such as Clerckx *et al.* (2008) .

The L-MMSE-MMV-TISTA network was trained layer-by-layer in a supervised manner using the ADAM optimizer (Kingma and Ba (2017)). This approach involves training each layer of the network sequentially while utilizing the minimum mean square error (MMSE) criterion and multiple measurement vectors (MMV) framework.

To evaluate the performance of the L-MMSE-MMV-TISTA network, the normalized mean square error (NMSE) was employed. NMSE is calculated as the expected value of the squared norm of the difference between the estimated signal $\hat{\mathbf{X}}$ and the true signal $\mathbf{X}$, divided by the expected value of the squared norm of $\mathbf{X}$. This metric provides a measure of the accuracy of the network's estimates relative to the true signal.

Overall, these simulation settings and evaluation metrics were chosen to analyze the performance of the L-MMSE-MMV-TISTA network in a realistic wireless communication scenario, considering the number of users, antennas, iterations, channel covariance, and training methodology.

3.0.1 Path Loss Model

In the considered scenario, let d_n denote the distance between user n and the base station (BS) for all n. It is assumed that the distances d_n are randomly distributed within the range of 0.5 km to 1 km.

The path loss model characterizing the wireless channel for user n is given by $\alpha_n = -128.1 - 36.7 \log_{10}(d_n)$ in decibels (dB) for all n(Liu and Yu (2018)). This model captures the attenuation or loss of signal power as it propagates through the wireless medium, taking into account the distance between the user and the base station. The logarithmic term reflects the logarithmic relationship between the distance and path loss, and the constant coefficients (-128.1 and -36.7) determine the specific characteristics of the path loss model.

3.1 PHASE TRANSITION

The phase transition in a multiple access system characterizes the minimum required ratio of preamble length to the number of users $\frac{L}{N}$ to achieve a target probability of error for a given activity probability p.

The activity probability, denoted by p, represents the probability that a user is active and transmits a signal during a given time slot. It quantifies the level of user activity in the system.

The phase transition phenomenon occurs when there is a critical threshold value of $\frac{L}{N}$ below which it is impossible to achieve the target probability of error, regardless of the activity probability. This critical threshold marks a phase transition point in the system's behavior. The phase transition analysis helps in determining the minimum value of $\frac{L}{N}$ required to ensure reliable signal recovery and estimation performance in the presence of user activity.

By studying the phase transition behavior, we can gain insights into the fundamental trade-offs between preamble length, number of users, activity probability, and the achievable estimation performance in multiple access systems.

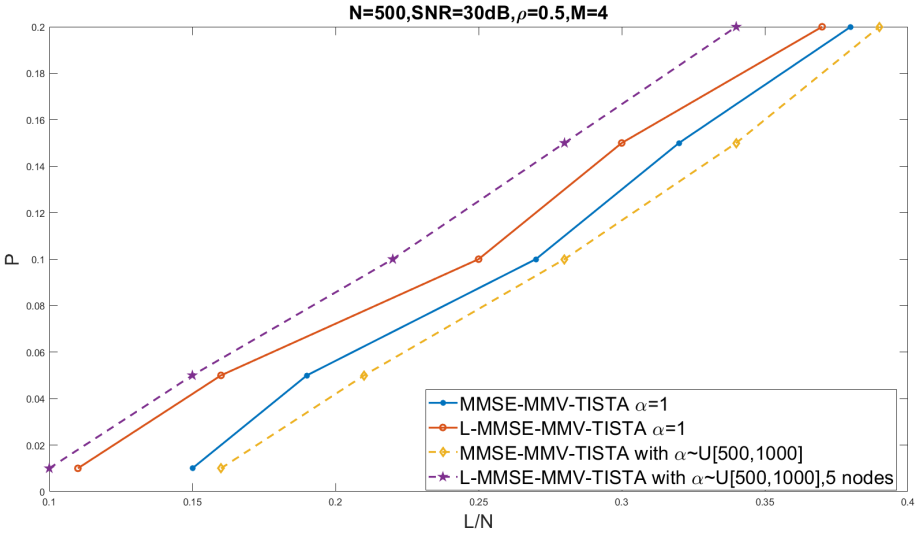

Figure 3.1: $N = 500$, $M = 4$, SNR $= 30$dB, $\rho = 0.5$. Users are uniformly located between 500m and 1000m.

Figure 3.1 showcases the outcomes of an experimental setup with specific parameters. The signal-to-noise ratio (SNR) is fixed at 30 dB, and $\rho = 0.5$. The networks are trained and tested using different activity values of p within the range of 0.01 to 0.2.

During the training phase, the loss function employed is the normalized mean square error (NMSE). However, during the evaluation or testing phase, the success of network recovery is determined based on the probability of error falling below 0.05. In this context, the probability of error refers to the combined sum of the probability of false alarm and the probability of missed detection.

The probability of false alarm represents the likelihood of incorrectly detecting an inactive user's signal as active. It quantifies the probability of false positives, where the network wrongly identifies a non-transmitting user as transmitting.

The probability of missed detection, on the other hand, denotes the probability of failing to detect an active user's signal correctly. It captures the likelihood of false negatives, where the network fails to identify and recover the signal of an active user.

To achieve successful recovery, the network aims to minimize the combined probability of false alarm and missed detection, ensuring that it falls below the threshold of 0.05. This criterion serves as a measure of the network's effectiveness in accurately detecting and recovering the transmitted signals, while minimizing both types of errors.

In Figure 3.1, a comparison is made between the MMSE-MMV-TISTA and L-MMSE-MMV-TISTA network architectures in two different scenarios: one with uniformly distributed users between 500m and 1000m, and the other with all users located at 500m from the base station.

The results clearly indicate that the modified L-MMSE-MMV-TISTA with 5 nodes in the hidden layer performs well even in cases with various path losses for each user. However, it is observed that the MMSE-MMV-TISTA algorithm requires a longer preamble length compared to the scenario where all users are at the same fixed location. This is expected because as the distance between the users and the base station increases, a longer preamble is necessary to account for weaker user signals.

Previous research by Shiv *et al.* (2022) has demonstrated that L-MMSE-MMV-TISTA can reduce the preamble length by 30-40% compared to the TISTA algorithm. Therefore, it can be inferred that the modified version of L-MMSE-MMV-TISTA also provides a similar reduction in the preamble length, leading to more efficient transmission in the system.

Figure 3.2 showcases the outcomes of an experimental setup with specific parameters. The signal-to-noise ratio (SNR) is fixed at 20 dB, and $\rho = 0.5$. The networks are trained and tested using different activity values of p within the range of 0.01 to 0.2

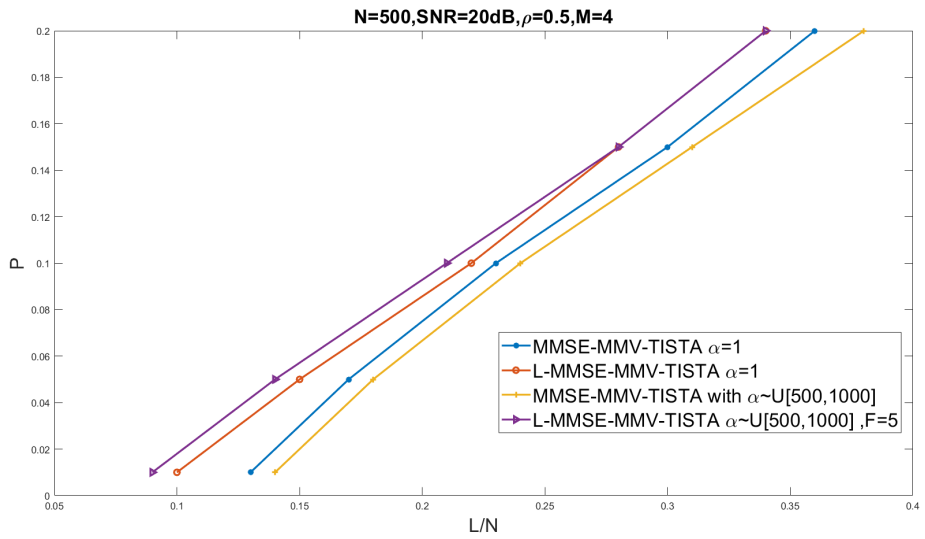

Figure 3.2: $N = 500$, $M = 4$, SNR = 20dB, $\rho = 0.5$. Users are uniformly located between 500m and 1000m.

We can observe a similar behavior as in the case of SNR=30dB, but with L-MMSE-MMV-TISTA, at higher active probabilities, the performance matches that of the scenario where all users are at the same location.

Overall, this behavior highlights the advantages of the modified L-MMSE-MMV-TISTA algorithm in handling different user distributions and achieving robust signal recovery in varying wireless communication scenarios.

Furthermore, the L-MMSE-MMV-TISTA algorithm outperforms the MMSE-MMV-TISTA algorithm. This can be attributed to the fact that the model $\mathbf{R_t} = \mathbf{X} + \mathbf{Z}$, which is used to derive the minimum mean square error (MMSE) denoiser expression in MMSE-MMV-TISTA, may not accurately capture the true signal and noise components.

Similarly, in TISTA, assumptions are made regarding the residual error at each stage during the error variance estimation step. These assumptions can introduce inaccuracies in the estimation process.

In contrast, the neural network-based approach of L-MMSE-MMV-TISTA is able to

compensate for some of these inaccuracies by learning from the training data. The network learns the underlying patterns and relationships between the received signals and the desired signals, allowing it to provide more accurate estimates and improve the overall performance of the algorithm.

By leveraging the power of deep learning and training on a large dataset, the L-MMSE-MMV-TISTA algorithm is able to overcome the limitations of the traditional MMSE-MMV-TISTA approach and achieve better performance in recovering the transmitted signals.

3.1.1 Users at Fixed Locations

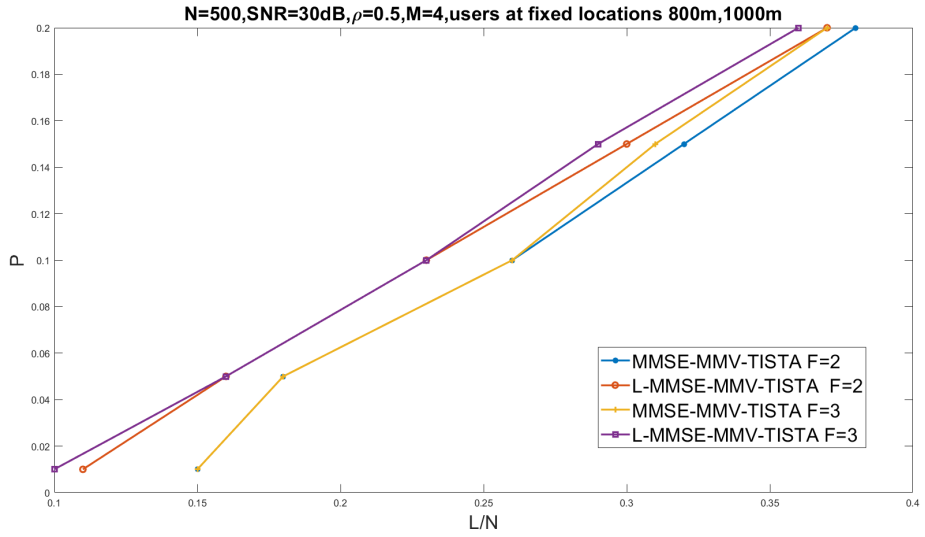

Figure 3.3: $N = 500$, $M = 4$, SNR = 30dB, $\rho = 0.5$. Users are uniformly located at 800m and 1000m.

Figure 3.3 showcases the outcomes of an experimental setup with specific parameters. The signal-to-noise ratio (SNR) is fixed at 30 dB, and $\rho = 0.5$ users at fixed location 800m,1000m. The networks are trained and tested using different activity values of p within the range of 0.01 to 0.2.

In this particular case, it is evident that there are only three possible covariance matrices . To account for this, the network architecture has been modified to accommodate

these three covariance matrices. Additionally, the received signal with three possible covariance matrices is applied to the existing MMSE-MMV-TISTA method.

Despite the limited number of covariance matrices, the L-MMSE-MMV-TISTA algorithm continues to outperform the MMSE-MMV-TISTA algorithm. This suggests that the modified L-MMSE-MMV-TISTA algorithm, which takes into account the three covariance matrices, achieves better performance in terms of signal recovery compared to the MMSE-MMV-TISTA algorithm.

Moreover, the modified algorithms that incorporate the three covariance matrices exhibit superior performance compared to using only two covariance matrices. This highlights the importance of considering additional covariance matrices in the network architecture, as it leads to improved accuracy in estimating and recovering the transmitted signals.

3.1.2 Determining the Optimal Number of Neurons in the Hidden Layer

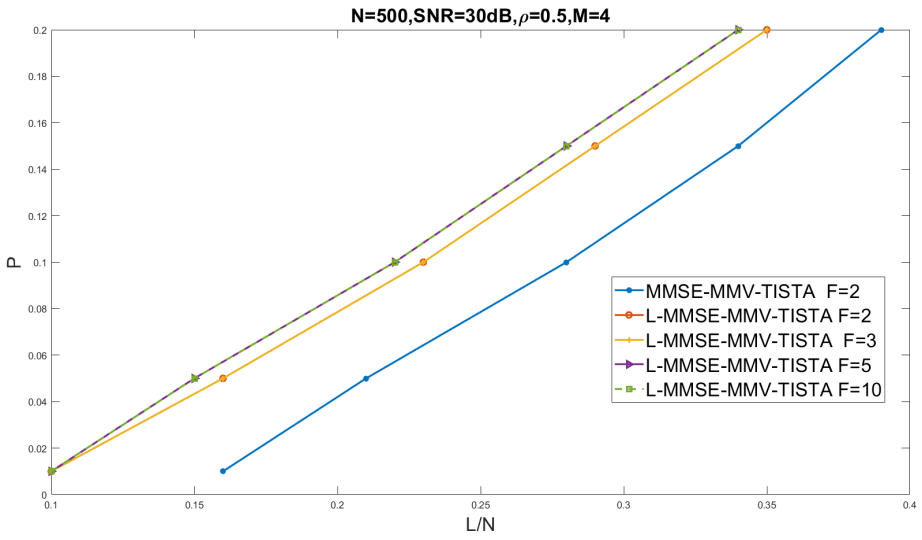

Figure 3.4: Activity Probability vs Number of Nodes, SNR=30dB, Users are uniformly located between 500m and 1000m

Figure 3.4 ,3.5 illustrates the results obtained from an experimental setup with specific parameters. The signal-to-noise ratio (SNR) is set at 30 dB,20dB. The networks are trained and tested using various activity values of p ranging from 0.01 to 0.2. The

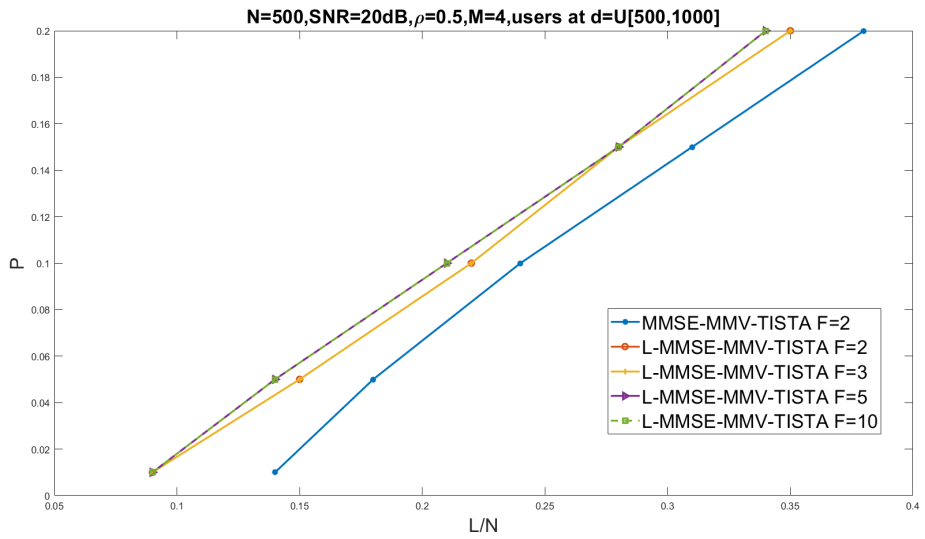

Figure 3.5: Activity Probability vs Number of layers, SNR=20dB, Users are uniformly located between 500m and 1000m

number of nodes in hidden layer in the network architecture is varied, including options such as 2, 3, 5, and 10.

From Figures 3.4 and 3.5, it is evident that increasing the number of layers in the network leads to an improvement in performance. However, it is interesting to note that for 10 nodes in hidden layer, the performance remains similar to that of 5 nodes in hidden layer. This observation suggests that there is a saturation point beyond which adding more nodes in hidden layer does not significantly enhance the performance.

Increasing the number of nodes in hidden layer in the network can be seen as an attempt to approximate all possible covariance matrices by using 2, 3, 5, or 10 covariance matrices. This approach allows the network to capture more complex relationships and patterns in the data, leading to improved performance in signal recovery.

3.1.3 NMSE Performance

Figure 3.6 and 3.7 present the results of an experimental setup with specific parameters. The signal-to-noise ratio (SNR) is held constant at 30 dB for Figure 3.6 and at 20 dB for

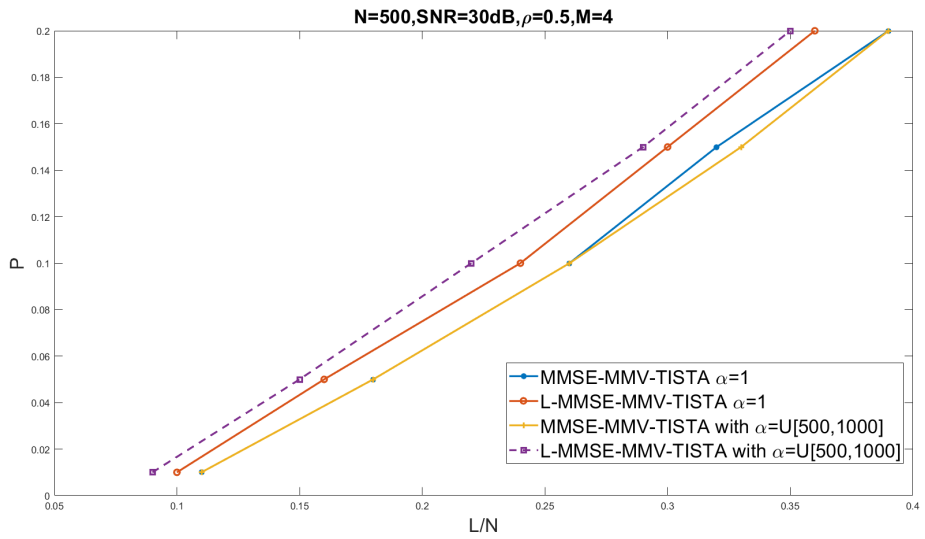

Figure 3.6: $N = 500$, $M = 4$, SNR = 30dB, $\rho = 0.5$. Users are uniformly located between 500m and 1000m.

Figure 3.7, while ρ is set to 0.5 in both cases. The networks are trained and tested using various activity values of p ranging from 0.01 to 0.2.

During the training phase, the normalized mean square error (NMSE) is utilized as the loss function to train the networks. However, during the evaluation or testing phase, the success of the network recovery is determined based on specific thresholds. For the SNR=30dB case, the network is considered successful if the NMSE falls below 20 dB. Similarly, for the SNR=20dB case, the NMSE threshold is set to 15 dB. These thresholds serve as benchmarks to evaluate the accuracy and effectiveness of the network in signal recovery.

The obtained results provide clear evidence that the modified L-MMSE-MMV-TISTA algorithms demonstrate strong performance even in scenarios where there are varying path losses for each user. This indicates that the network is able to effectively adapt to different channel conditions and accurately estimate the transmitted signals.

Although the MMSE-MMV-TISTA algorithm may require the same preamble length

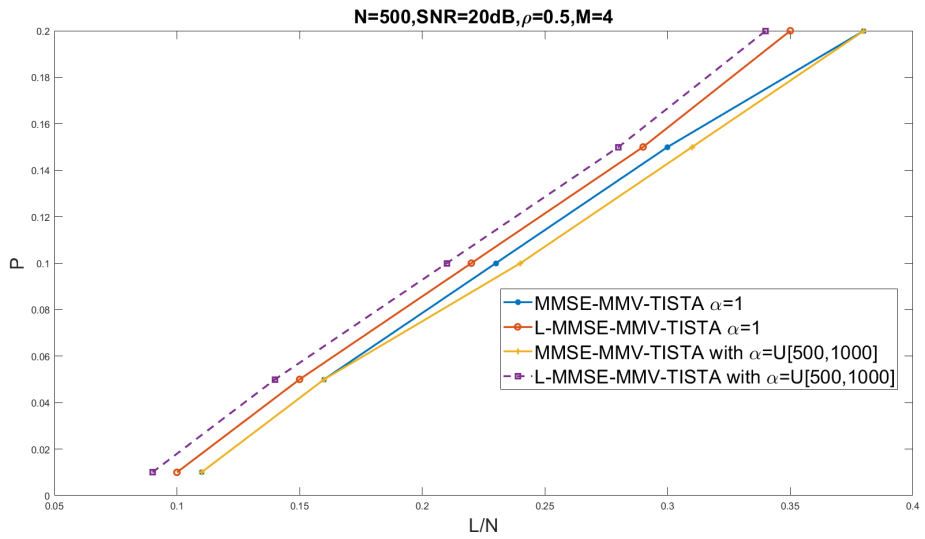

Figure 3.7: $N = 500$, $M = 4$, SNR $= 20$dB, $\rho = 0.5$. Users are uniformly located between 500m and 1000m.

for both scenarios (uniformly distributed users and users at a fixed position) in the NMSE sense, the probability of error can still vary between the two cases. In the NMSE (Normalized Mean Squared Error), the existing MMSE-MMV-TISTA method fails to capture any distinction between equal path loss and unequal path loss scenarios.

3.1.4 Comparing Estimated Error Variance with True Error Variance

Figure 3.8 showcases the estimated error variance $\bar{\tau}^2$ using equation (3.1) and the empirically estimated values of the true error variance τ^2 using equation (3.2).

$$\bar{\tau}^2 = \frac{v_t^2}{N}\left(N + \left(\gamma_t^2 - 2\gamma_t\right)L\right) + \frac{\gamma_t^2\sigma^2}{N}\,\mathrm{tr}\left(\mathbf{W}\mathbf{W}^T\right) \tag{3.1}$$

$$\tau^2 = \frac{E\left[(r_t - x)^2\right]}{N} \tag{3.2}$$

The estimator $\bar{\tau}^2$ accurately captures the error variance, providing validation for the effectiveness of equations (3.1) and supporting our assumptions regarding the residual errors. It is important to note that the error variance does not exhibit a strictly monotonic decrease. This non-monotonic behavior can be attributed to the influence of the trainable parameters $\{\gamma_t\}_{t=0}^{T-1}$. The zigzag pattern observed in the parameters γ_t may impact

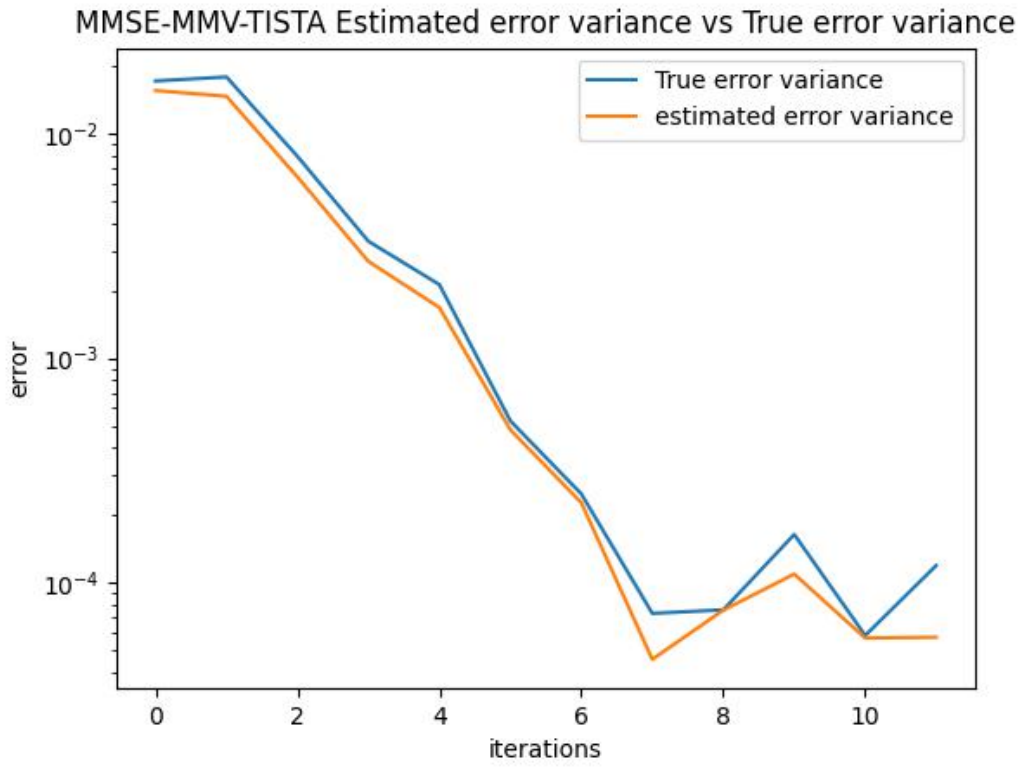

Figure 3.8: estimate $\bar{\tau}^2$ and the true error variance τ^2, given the parameters $N = 500$, $M = 4$, and SNR = 30 dB

the shapes of τ^2 and $\bar{\tau}^2$. However, despite this nontrivial tendency, the residual error demonstrates a rapid decrease, indicating successful signal recovery.

Similarly, for the L-MMSE-MMV-TISTA, we plotted the estimated error variance $\bar{\tau}^2$ using equation (3.1) and the empirically estimated values of the true error variance τ^2 using equation (3.2). We observed a similar trend as in the MMSE-MMV-TISTA case, where the estimator $\bar{\tau}^2$ accurately captures the error variance, validating the effectiveness of equations (3.1) and supporting our assumptions regarding the residual errors. The non-monotonic behavior of the error variance, influenced by the trainable parameters $\{\gamma_t\}_{t=0}^{T-1}, W_h, p$, is also observed in the L-MMSE-MMV-TISTA. Despite this nontrivial tendency, the residual error demonstrates a rapid decrease, indicating successful signal recovery.

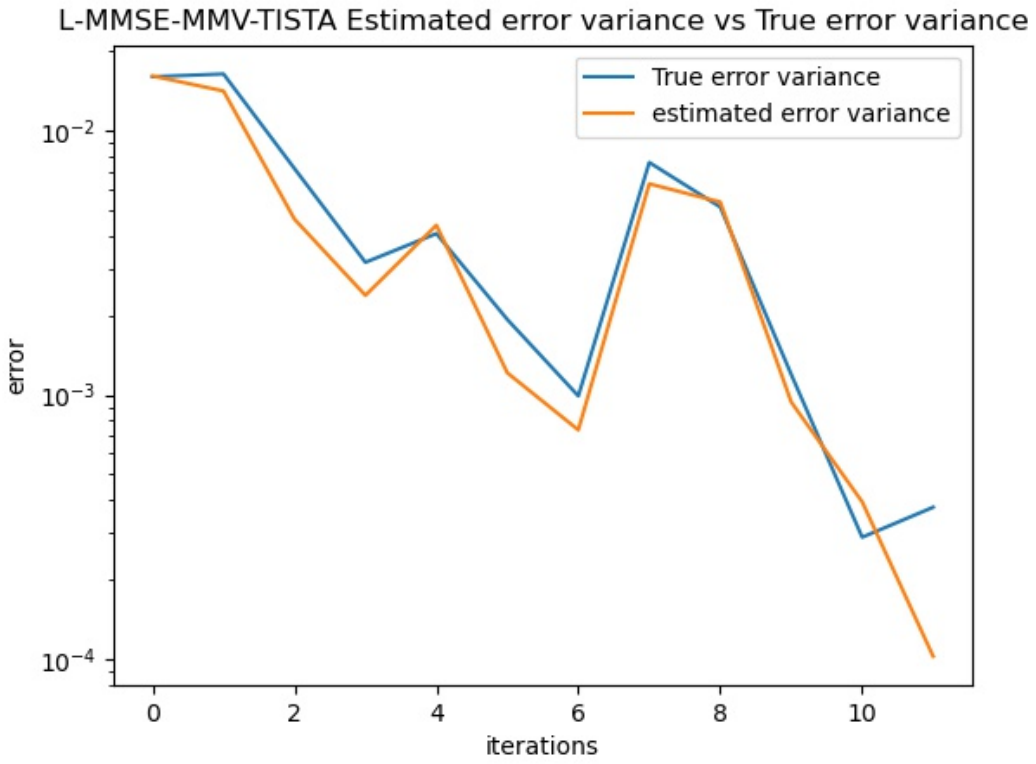

Figure 3.9: estimate $\bar{\tau}^2$ and the true error variance τ^2, given the parameters $N = 500$, $M = 4$, and SNR = 30 dB

3.1.5 Comparing MMSE-MMV-TISTA with Estimated vs. True Path Loss

Figure 3.10,3.11 depicts the estimated path loss versus the true path loss. The signal-to-noise ratio (SNR) is fixed at 30 dB,20dB, and the parameter ρ is set to 0.5. The experimental setup involves training and testing different networks using various activity values of p within the range of 0.01 to 0.2. During the training phase, the loss function employed is the normalized mean square error (NMSE). However, during the evaluation or testing phase, the success of network recovery is determined based on the probability of error falling below 0.05 for 30dB,0.1 for 20dB.

In the plot, it is evident that the MMSE-MMV-TISTA algorithm with estimated path loss performs better compared to the true path loss. This implies that the estimated path loss values approximate the actual path loss more closely to 1, resulting in improved

performance for the MMSE-MMV-TISTA algorithm.

On the other hand, the L-MMSE-MMV-TISTA algorithm performs even better than the MMSE-MMV-TISTA algorithm with estimated path loss. This suggests that the L-MMSE-MMV-TISTA algorithm has an advantage over the MMSE-MMV-TISTA algorithm, particularly when it comes to handling estimated path loss values.

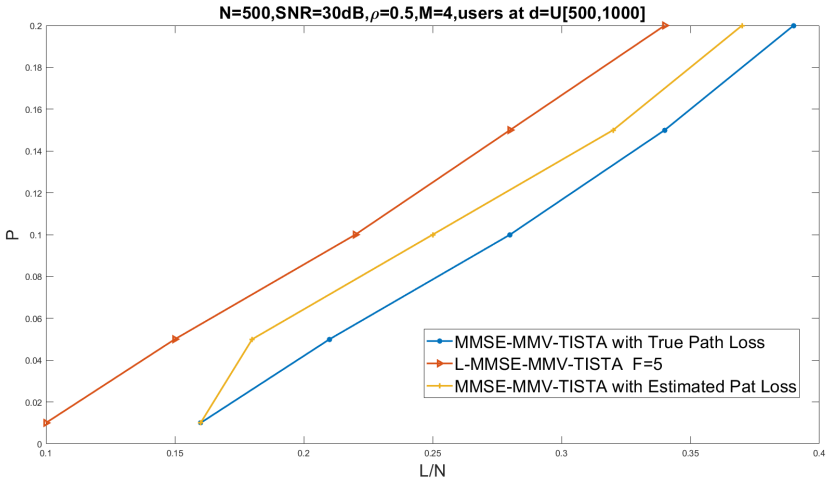

Figure 3.10: Comparing MMSE-MMV-TISTA with Estimated Path Loss and True Path Loss (Fig. 3.1)

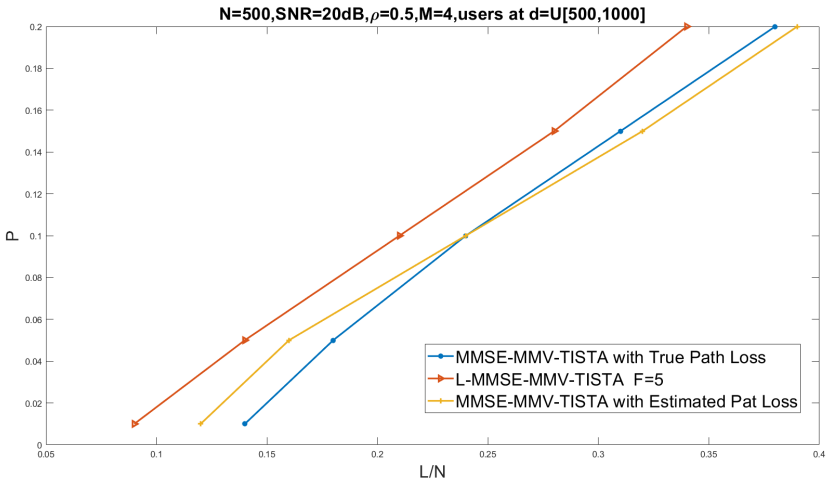

Figure 3.11: Comparing MMSE-MMV-TISTA with Estimated Path Loss and True Path Loss (Fig. 3.2)

CHAPTER 4

CONCLUSIONS

The Trainable ISTA (ISTA) algorithm, introduced by Ito et al. (2019), is known for its strong sparse recovery capabilities, efficient convergence, and fast training process. In Shiv *et al.* (2022) developed an extension of ISTA called L-MMSE-MMV-TISTA, which replaces the MMSE shrinkage function with a model-based neural network. This neural network can learn an appropriate shrinkage function. However, their approach considered the shrinkage function to be the same for all users.

In our modification, we addressed the case of different path losses by adapting the existing algorithm. We varied the number of nodes in hidden layer in the neural network to better approximate the path loss model. Through our observations, we found that the modified algorithm works effectively for various path losses. We also noticed that L-MMSE-MMV-TISTA outperforms MMSE-MMV-TISTA, and it doesn't require prior knowledge of the covariance matrix, activity probability, or path loss.

Based on Shiv *et al.* (2022)'s findings, it is evident that L-MMSE-MMV-TISTA can reduce the preamble length by 30-40%. In our modified version, incorporating 5 nodes in hidden layer in the neural network, we achieved even better performance compared to the existing L-MMSE-MMV-TISTA. Consequently, the modified algorithm also reduces the preamble length by 30-40%.

REFERENCES

1. **Borgerding, M.**, **P. Schniter**, and **S. Rangan** (2017). Amp-inspired deep networks for sparse linear inverse problems. *IEEE Transactions on Signal Processing*, **65**(16), 4293–4308.

2. **Clerckx, B.**, **G. Kim**, and **S. Kim**, Correlated fading in broadcast mimo channels: Curse or blessing? *In IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*. 2008.

3. **Ito, D.**, **S. Takabe**, and **T. Wadayama** (2019). Trainable ista for sparse signal recovery. *IEEE Transactions on Signal Processing*, **67**(12), 3113–3125.

4. **Kingma, D. P.** and **J. Ba** (2017). Adam: A method for stochastic optimization.

5. **Li, S.**, **W. Zhang**, **Y. Cui**, **H. V. Cheng**, and **W. Yu** (2019). Joint design of measurement matrix and sparse support recovery method via deep auto-encoder. *IEEE Signal Processing Letters*, **26**(12), 1778–1782.

6. **Liu, L.** and **W. Yu** (2018). Massive connectivity with massive MIMO—part i: Device activity detection and channel estimation. *IEEE Transactions on Signal Processing*, **66**(11), 2933–2946. URL

7. **Neumann, D.**, **T. Wiese**, and **W. Utschick** (2018). Learning the MMSE channel estimator. *IEEE Transactions on Signal Processing*, **66**(11), 2905–2917. URL

8. **Sabulal, A. P.** and **S. Bhashyam**, Joint sparse recovery using deep unfolding with application to massive random access. *In ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020.

9. **Shiv, U. K. S.**, **S. Bhashyam**, **C. R. Srivatsa**, and **C. R. Murthy** (2022). Learning-based sparse recovery for joint activity detection and channel estimation in massive random access systems. *IEEE Wireless Communications Letters*, **11**(11), 2295–2299.

10. **Zhang, Z.**, **Y. Xu**, **J. Yang**, **X. Li**, and **D. Zhang** (2015). A survey of sparse representation: Algorithms and applications. *IEEE Access*, **3**, 490–530.