ABSTRACT

Title of dissertation: PRECODER DETECTION FOR COOPERATIVE DECODE-AND-FORWARD RELAYING IN OFDMA SYSTEMS

Abhijit Kiran Valluri, Master of Science, 2016

Thesis directed by: Professor Richard La
Department of Electrical and Computer Engineering

We consider an LTE network where a secondary user acts as a relay, transmitting data to the primary user using a decode-and-forward mechanism, transparent to the base-station (eNodeB). Clearly, the relay can decode symbols more reliably if the employed precoder matrix indicators (PMIs) are known. However, for closed loop spatial multiplexing (CLSM) transmit mode, this information is not always embedded in the downlink signal, leading to a need for effective methods to determine the PMI. In this thesis, we consider 2x2 MIMO and 4x4 MIMO downlink channels corresponding to CLSM and formulate two techniques to estimate the PMI at the relay using a hypothesis testing framework. We evaluate their performance via simulations for various ITU channel models over a range of SNR and for different channel quality indicators (CQIs). We compare them to the case when the true PMI is known at the relay and show that the performance of the proposed schemes are within 2 dB at 10% block error rate (BLER) in almost all scenarios. Furthermore, the techniques add minimal computational overhead over existent receiver structure.

Finally, we also identify scenarios when using the proposed precoder detection algorithms in conjunction with the cooperative decode-and-forward relaying mechanism benefits the PUE and improves the BLER performance for the PUE. Therefore, we conclude from this that the proposed algorithms as well as the cooperative relaying mechanism at the CMR can be gainfully employed in a variety of real-life scenarios in LTE networks.

PRECODER DETECTION FOR COOPERATIVE DECODE-AND-FORWARD RELAYING IN OFDMA SYSTEMS

by

Abhijit Kiran Valluri

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Master of Science
2016

Advisory Committee:
Professor Richard La, Chair/Advisor
Professor Mark Shayman
Professor Sennur Ulukus

Acknowledgments

I owe my gratitude to all the people who have made this thesis possible and who have made my experience at graduate school worthwhile.

First and foremost, I would like to thank my advisor, Professor Richard La for giving me invaluable guidance and teaching me the art and spirit of academic research. Richard has always made himself available for help and advice, and throughout the course of my graduate studies he has taught me the true meaning and purpose of research. It has been a pleasure to work with and learn from such an experienced individual.

Next, I would like to thank Professor Mark Shayman, who has been a vital part of the research project that this work stemmed out of. He has provided valuable feedback throughout this research, during numerous weekly meetings and provided a safe space to discuss new ideas. I would also like to thank Professor Bobby Bhattacharjee, whose interactions during this project were highly illuminating and useful. Furthermore, I would like to thank Professor Mark Shayman once again, and Professor Sennur Ulukus for agreeing to serve on my thesis committee and for sparing their time in reviewing this manuscript.

My colleagues and classmates have been an integral part of this journey, who enriched my graduate student life in many ways and deserve a special mention. I have had numerous useful discussions with Vaibhav Singh, who graduated with his PhD recently. He was a really good soundboard to bounce off ideas for the project I was working on and to also get general advice on graduate student life. My

interactions with Siddharth Pal, Ginnah Lee, Matthew Lentz and Zhihao Li have also been very fruitful. There are many more people whom I am grateful for knowing and interacting with, but it would be impossible to list them all individually.

I would like to acknowledge financial support from the Laboratory for Telecommunication Sciences (LTS) and the department of Electrical and Computer Engineering for all the projects discussed herein. Furthermore, I would like to thank the close involvement of two researchers from LTS, namely Jay Weingart and Michael Jordan, for their expertise and guidance on the research project for the duration of the funding from LTS.

I would also like to acknowledge help and support from the amazing staff members in the ECE department. My thanks goes to Melanie Prange, Bill Churma, Heather Stewart, Carrie Hilmer and Kristen Little for helping me with both the academic processes and for my duties when I was the ECEGSA president.

Last but not least, I owe my deepest gratitude to my family – my mom, dad and brother – who have always stood by me and guided me throughout my graduate student life, patiently listened to my concerns and always provided me with much needed moral support. I could not have done this without them! Thanks, amma!

Table of Contents

List of Figures

List of Abbreviations

3GPP	3rd Generation Partnership Project
ARQ	Automatic Repeat Request
AWGN	Additive White Gaussian Noise
BLER	Block Error Rate
CFO	Carrier Frequency Offset
CLSM	Closed Loop Spatial Multiplexing
CMR	Cooperative Mobile Relay
CQI	Channel Quality Index
CRC	Cyclic Redundancy Check
CRS	Cell-specific Reference Signal
DCI	Downlink Control Information
DL-SCH	Downlink Shared Channel
E-UTRA	Evolved UMTS Terrestrial Radio Access
eNodeB	E-UTRAN Node B, is the evolved base transceiver station. It is the base station in an LTE network.
EPA	Extended Pedestrian A, an ITU channel model
ETU	Extended Typical Urban, an ITU channel model
EVA	Extended Vehicular A, an ITU channel model
FRN	Fixed Relay Node
HARQ	Hybrid ARQ
ITU	International Telecommunication Union
LS	Least Squares
LTE	Long Term Evolution
LTE-A	LTE-Advanced
MCS	Modulation and Coding Scheme
MEX	A binary executable format for a MATLAB® function
MIMO	Multi-Input Multi-Output
ML	Maximum Likelihood
MMSE	Minimum Mean Squared Error
$O(\cdot)$	Big O notation, used for denoting the limiting behavior of

	computational complexity
OFDMA	Orthogonal Frequency Division Multiple Access
PCI	Precoding Control Information
PDP	Power Delay Profile
PMI	Precoder Matrix Index
PUE	Primary User Equipment
QAM	Quadrature Amplitude Modulation
RB	Resource Block
RI	Rank Index
SNR	Signal to Noise Ratio
SSD	Soft Sphere Decoder
SU-MIMO	Single User MIMO
UE	User Equipment
UMTS	Universal Mobile Telecommunications System

Chapter 1: Introduction

Relaying in wireless networks can be used for several purposes, such as increasing the throughput, improving cellular coverage, as temporary network deployment and to satisfy increases in the demand from users of the wireless network. In LTE-A, 3GPP Release 10, relaying is one of the features adopted to improve and extend cellular coverage. Support for fixed cellular relays was introduced in [1,2]. The fixed relay node (FRN) is wirelessly connected to the radio-access network via a *donor cell*, namely the cell with which the relay is associated. Incorporating such relays into the LTE network requires a sizeable investment from the cellular operator. These relays can either be *transparent* or *non-transparent* to the user equipment (UE). If the relay is transparent, the UE is unaware of whether or not it is communicating with a relay, whereas it is aware of the relay's presence in the non-transparent case.

Researchers are actively investigating various aspects of LTE-A relay architecture, including mobile relays [3,4]. Furthermore, there is an abundance of literature on cooperative relay techniques for wireless networks, such as [5–9].

In this thesis, we shall consider only transparent relays. More specifically, we will explore the use of cooperative relaying techniques amongst mobile users in LTE systems, rather than FRNs that need to be deployed at fixed locations. This

has the advantage of needing minimal network infrastructure upgrades, and only relying on advances in the user equipment (UE), making it easier to deploy. A similar system was previously studied in [10], focusing on the system architecture. Here, we consider a cooperative mobile relay (CMR) operating "transparently" to the eNodeB (i.e. base station), by suitably communicating with the primary user equipment (PUE), and focus on the decode-and-forward relay mechanism, previously discussed in several studies [5–7]. In order to ensure that the CMR works properly, we need certain information at the CMR. The focus of this thesis is to solve the problems that arise when such information is unavailable at the relay.

The CMR could use various technologies to communicate with the PUE for the purpose of transmitting the decoded data to the PUE, such as ad-hoc WiFi networks or LTE. We do not address this aspect of the relay in this work and assume that a suitable approach is implemented to transmit the data to the PUE. We assume that the PUE communicates with the eNodeB using Closed Loop Spatial Multiplexing (CLSM) transmit mode defined in [11], since studies show that spatial multiplexing outperforms transmit diversity schemes in modern MIMO systems [12]. In this case, the CMR may need to estimate an unknown precoder used in the downlink transmission from eNodeB to the PUE, as explained in section 2. Given that the CMR operates transparently to the PUE, it cannot obtain this information directly from the PUE. Hence, we focus on the problem of precoder detection at the CMR.

We propose two precoder detection algorithms. Their performance is evaluated using two ITU channel models defined in [13, 14], over a wide range of SNR and for different Channel Quality Indicators (CQIs). We compare them to the

case when precoders are known at the CMR. We demonstrate that the performance is within 2 dB at a block error rate (BLER) of 10% for almost all the cases we consider, with minimal computational overhead to the existent receiver structure. Additionally, we also conduct experiments to evaluate the overall benefit of using the proposed precoder detection algorithms at the CMR in conjunction with the cooperative decode-and-forward relaying mechanism to improve the performance of the PUE. To this end, we identify specific scenarios where using the proposed techniques helps improve the performance at the PUE. Through these experiments, we can conclude that the proposed cooperative relaying mechanism at the CMR using our precoder detection algorithms can be profitably employed in LTE networks.

The rest of this thesis is organized as follows. We outline the problem in chapter 2. In chapter 4, we briefly describe the simulator used for performance evaluation and the modifications we made for our experiments. Chapter 5 explains our hypothesis testing formulation for precoder detection along with the proposed algorithms and a brief computational complexity analysis. Our simulation setup and results are shown in chapter 6. We conclude in chapter 7 and discuss future work in chapter 8.

Chapter 2: Problem Formulation

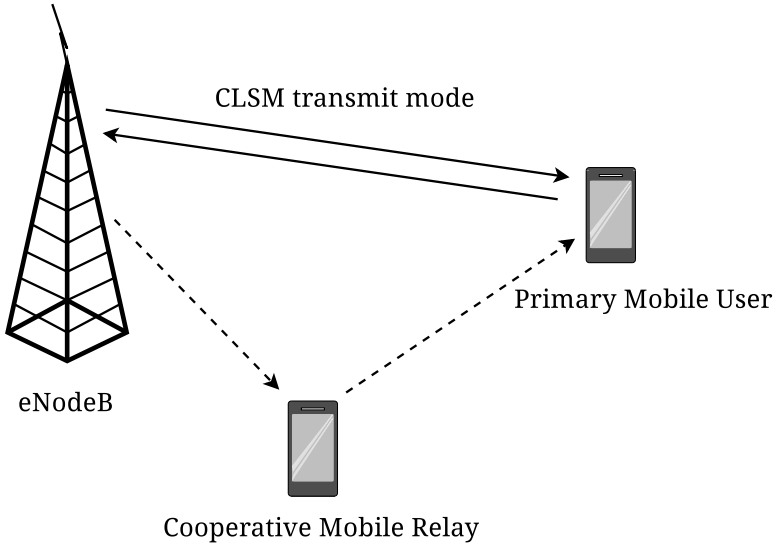

Figure 2.1: Decode-and-forward relay scenario considered for precoder detection.

2.1 Description

In this thesis, we consider the scenario shown in Fig. 2.1. We assume that the CMR is operating transparently to the eNodeB, as that will ease the implementation of such a mobile relay node in an LTE system, avoiding the need to modify the implementation of LTE. We assume that the CMR can pick up the transmission intended for the PUE and decode and forward the data to the PUE when needed, such as when the PUE suffers from poor coverage. However, in order to perform the

decode-and-forward relaying, the CMR needs to have access to some information that could potentially be unavailable as explained in the next section, in which case it may not be able to decode the transmitted symbols correctly with high probability. We cannot directly retrieve this information from the eNodeB due to our assumption that the CMR is transparent to the base station. Hence, we resort to estimating the missing information. This provides motivation to our study.

2.2 Motivation

In CLSM, as the name suggests, there is feedback between the base station and the UE. The UE generates feedback that consists of the Precoder Matrix Index (PMI), Rank Index (RI), and CQI. This feedback is used by the eNodeB to adapt the transmission to the UE, such as changing the modulation and coding scheme (MCS), and changing the precoder used as suggested by the PMI. Unlike in open loop spatial multiplexing, where no feedback is used and a fixed precoder is employed, the precoder may change in CLSM.

The downlink transmission in CLSM relies on antenna ports {0, 1, 2, 3} depending on antenna configuration, i.e. whether we use 2x2 MIMO or 4x4 MIMO. These antenna ports only use Cell-specific Reference Signals (CRSs) [15, section 6.10], which are added to the transmit signal *after* precoding is performed [15, section 6.3.4.2]. One CRS is added per antenna port. The UE can estimate the radio channel using the received CRSs, since the CRS is a known signal to the receiver. However, because the CRSs are not multiplied with the precoder, the UE additionally needs to

know the precoder used by the eNodeB in order to demodulate the received signal and reconstruct the data. Unfortunately, the eNodeB need not explicitly specify the PMI in the Downlink Control Information (DCI) [16, section 5.3.3.1.5].

For this reason, the CMR may not be aware of the employed PMIs and needs to determine them based on the received signals in order to be able to decode the transmitted data. We need to design a computationally efficient precoder detector for the CMR, to accurately decode the user data and forward it to the PUE. By directly estimating the precoder, we can reduce the signaling overhead needed between the PUE and the relay, as well as make sure that the relay operates truly transparently to the eNodeB.

2.3 Problem statement

We can now state our problem succinctly. Once again, we consider the scenario presented in Fig. 2.1, wherein the PUE is communicating with the eNodeB using CLSM and the CMR operates transparent to the eNodeB and the PUE. The CMR decodes-and-forwards the data to the PUE, so that in case the PUE has poor channel conditions with the eNodeB, it can still decode the data due to the transmission from the CMR. Our goal is to design a computationally efficient algorithm at the CMR such that it can detect the precoder matrix used by the eNodeB by observing the received data. It then uses the precoder matrix along with the channel estimate to decode the data and then subsequently forward the decoded data to the PUE. We stress that we do not consider the transmission between the CMR and PUE in this

work. We assume that the CMR can transmit the data to the PUE using a suitable approach, such as ad-hoc WiFi, and that this transmission is relatively error-free, because typically the relay would have good channel conditions for the user.

In chapter 5, we propose two hypothesis testing based approaches to design such a precoder detection algorithm.

Chapter 3: Related Work

There is extensive research available currently that deals with the problem of blind source separation (BSS), which is a concept that is related to the problem posed in this thesis. In essence, BSS is the task of separating a set of source signals from a set of mixed signals, when there is little to no information about the source signals or the mixing process. A standard example to illustrate this problem is the so-called "cocktail party problem", where we want to recover the speech signals of multiple speakers who are simultaneously talking in a room.

The problem that we focus on, of detecting the unknown precoder matrices, is similar to BSS. The precoder matrices correspond to the unknown mixing process that combines the signals from the various antennas. Given these mixed signals at the receiver, we need to identify the correct precoder matrix and hence recover the original signals. Hence, we shall explore some of the existent literature on BSS to understand the available solutions and the issues faced in extending these approaches for our problem.

3.1 Blind source separation

The work done by Hyvärinen et al. [17] discusses Independent Component Analysis (ICA) and motivates the study by considering the cocktail party problem. ICA is one approach to tackle BSS. An important assumption made in ICA is that the subcomponents in the multivariate, mixed signals are non-Gaussian and statistically independent.

3.1.1 Constant Modulus-based algorithms

In [18], Godard introduces a type of blind, "self-recovering" equalizer using the framework of BSS, which recovers the transmitted signal from the received signal by minimizing a nonconvex cost function that characterizes the inter symbol interference (ISI) in the received signal. The paper considers PAM and QAM modulations for the source signals. Using a steepest decent algorithm, the proposed equalizer updates its weights till the gradient of the cost function is minimized. A special case of the algorithm discussed in [18] is Constant Modulus Algorithm (CMA) [19]. In [19], the author discusses the use of CMA in connection to phase modulated signals such as FM signals and QPSK signals. However, these algorithms introduce a phase ambiguity in the recovered signals.

As noted by several researchers, the CMA technique doesn't work very well for non-constant modulus signals, such as QAM. Hence, in [20], the authors discuss a multistage equalizer that performs CMA and AMA (Alphabet Matched Algorithm) in separate stages to detect wireless signals (denoted to as CMA/AMA). The paper

also discusses the local convergence property of AMA. This algorithm does significantly better than just CMA, as the second stage of AMA explicitly uses the actual constellation used for the signal, such as QAM. This allows the algorithm to minimize the distance between the equalized symbols and the original symbols in the constellation, hence improving the likelihood of correct detection. Likewise, in [21], the authors consider a CMA+AMA equalizer which minimizes a linear combination of the CMA and the AMA cost function. In our implementation of these two algorithms, we found both provide similar performance.

There are several drawbacks to using CMA type of approaches to solve the problem we proposed, such as an inherent phase ambiguity in the recovered signals, slow convergence time, high sensitivity to gradient descent parameters used in the algorithm, as well as poor performance in the case of frequency selective channels. Some modifications have been proposed in the literature for this algorithm, such as the modified CMA (MCMA) [22], where the cost function is separated out in terms of the real and imaginary parts of the complex signals, allowing for simultaneous blind equalization and carrier phase recovery. Furthermore, in [23], the authors add variable step size to MCMA for improved convergence time.

We initially tested the efficacy of such a MCMA/AMA algorithm, where we used MCMA and AMA in two stages. While the algorithm worked reasonably for frequency flat channels, it failed completely for frequency selective scenarios. Still, there was much to be desired with regard to BLER performance. Additionally, the algorithm had an inherent phase ambiguity for which additional pilot signals were needed, which would modify the LTE specification.

3.1.2 Other blind equalization methods

Furthermore, we explored the use of blind and "semi-blind" MIMO channel estimation techniques to estimate the channel and the precoder matrices combined such as [24–26], and then recover the signals. Likewise, we also explored a semi-BSS approach developed for a MIMO-OFDM communication system [27]. While the BLER performance was better compared to the CMA-based approach, it was still insufficient to improve upon the performance of the primary user, and many of the same issues remained as with CMA-based approach, such as the phase ambiguity.

Furthermore, we note that due to the presence of reference signals embedded in the LTE downlink signal, it is unnecessary to resort to blind channel equalization as one can utilize the known reference signals to obtain the channel matrix. Hence, in the following chapters, we will detail a technique that utilizes all the pertinent information present in the downlink signal and detect the unknown precoders using a hypothesis testing approach.

Chapter 4: System Model

Our LTE simulator is based on the LTE Downlink link-level simulator (v1.7r1089) [28] from the Vienna University of Technology, written in MATLAB®. The MATLAB® based simulator simulates the base station, a mobile user, the channel conditions between them as well as the scheduling behavior of the base station. The simulator [28] can create a variety of scenarios, including multi-user scenarios, multi-cell scenarios and also create interference. For the purpose of our research, we focus on the single-user, single cell scenario and neglect inter-cell interference.

Some of the time-intensive operations in the simulator have been implemented in C via MEX functions [29], including bit interleaving, convolutional encoding/decoding, rate matching, symbol demapping, as well as our proposed precoder detection algorithms (see chapter 5). Here, we briefly describe the setup of the simulator along with our modifications to enable our experiments. For a more detailed description of the simulator, refer to [28, 30].

4.1 Setup

We consider a Single-User MIMO (SU-MIMO) scenario, with a single eNodeB and a PUE. A secondary mobile user acts as a CMR for the PUE, using a decode-

and-forward relay mechanism. The eNodeB transmits data using CLSM, thereby adapting the MCS, the precoder matrix and the number of layers used for transmission according to the CQI, PMI and RI feedback from the PUE. In our setup, we consider both 2x2 and 4x4 MIMO modes supported by CLSM. For simulating the mobile relay, we consider various channel models between the eNodeB and the CMR, which have different power delay profiles (PDP), along with the speed and SNR at the CMR as explained in section 6.1.

The simulator simulates the time-varying multipath fading corresponding to different channel models using a sum-of-sinusoids statistical simulation model described in [31]. The simulator can take in various parameters including the transmit and receive antenna correlations and the speed of the user along with the PDP for the channel model to generate the multipath fading characteristics for the channel considered.

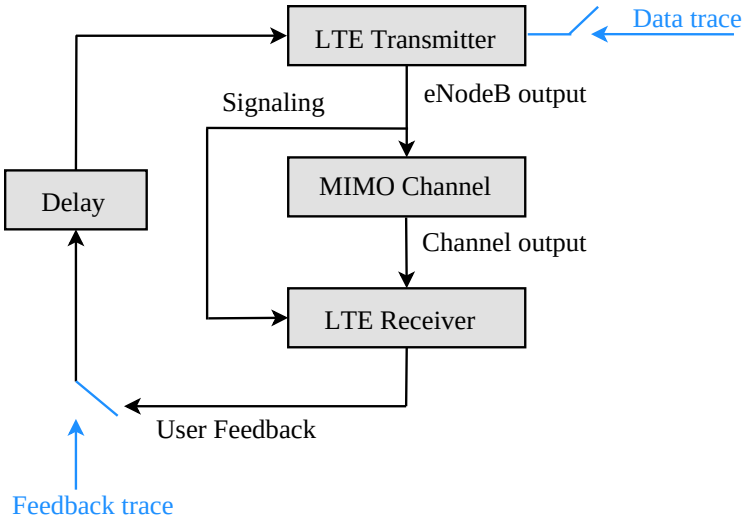

Figure 4.1: Overall structure of the LTE simulator, with our modifications in blue.

We use the SU-MIMO simulation setup to test our proposed precoder detection

algorithms, while neglecting the effect of interference and the impact of scheduling, as a starting point. We leave the more elaborate setup for our future work.

4.1.1 Overall simulator structure

The main portions of the simulator, as used for the SU-MIMO scenario, are shown in Fig. 4.1, with our modifications shown in blue. The simulator simulates the transmission of the Downlink Shared Channel (DL-SCH), which consists of the transmitted symbols along with the reference signals (CRSs), as well the control information. For the feedback, the simulator simulates an error-free uplink feedback channel with an adjustable delay [28], rather than a realistic feedback channel. However, given that the feedback channel is protected by error correction mechanisms the feedback is less susceptible to errors than the transmitted data. Moreover, because we are primarily interested in analyzing the performance of our precoder detection algorithms, the idealistic implementation of the feedback channel does not impact this performance analysis.

4.1.2 LTE transmitter

The structure of the LTE transmitter is described in Fig. 4.2. As mentioned in [28, table 1], the user feedback has not been completely incorporated into the CLSM transmit mode of the simulator, as per v1.7r1089. We explain the changes we made to the simulator to incorporate such feedback in section 4.2. The simulator uses the convolution encoder from [32] for the turbo encoder at the transmitter to

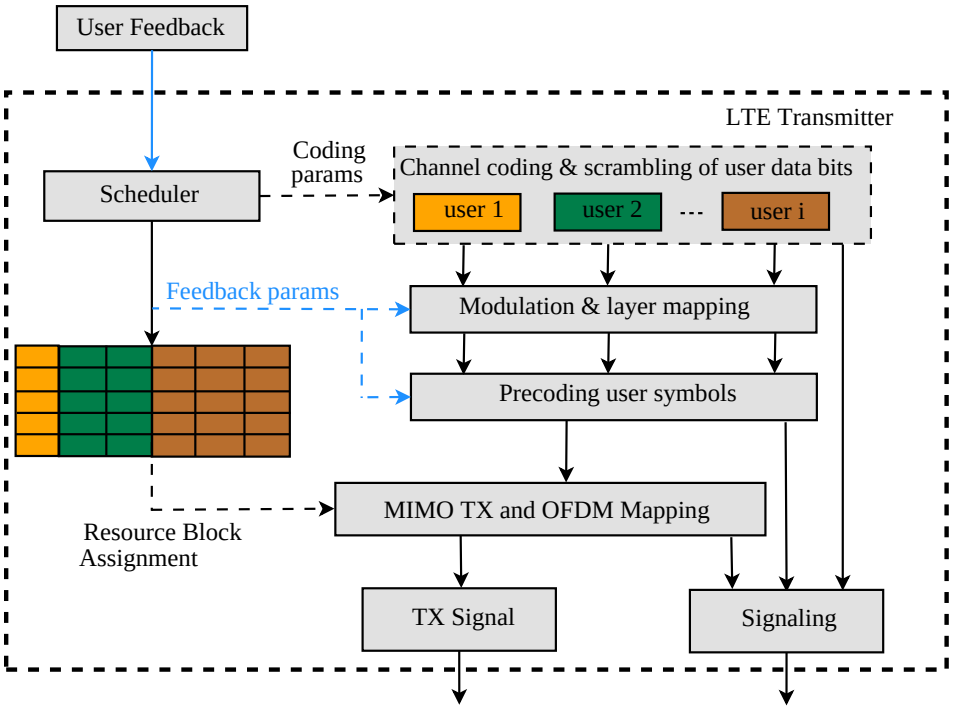

Figure 4.2: Structure of the LTE transmitter, with our modified user feedback implementation in blue.

perform the channel coding.

For each user, after the channel coding and scrambling of the data bits, the transmitter modulates the data, using an MCS corresponding to the CQI given by the feedback, in CLSM mode. It then maps the data to the correct number of layers, as specified by the RI feedback, and applies a precoder matrix based on the rank indicator and Precoding Control Information (PCI) feedback from the UE. Finally, the individual precoded symbols to be transmitted on each antenna are mapped to the appropriate elements of the resource blocks (RBs) scheduled for the user. The LTE simulator (v1.7r1089) supports either static or round robin scheduling. Since we are considering a SU-MIMO scenario, we are not concerned with scheduling.

4.1.3 LTE receiver

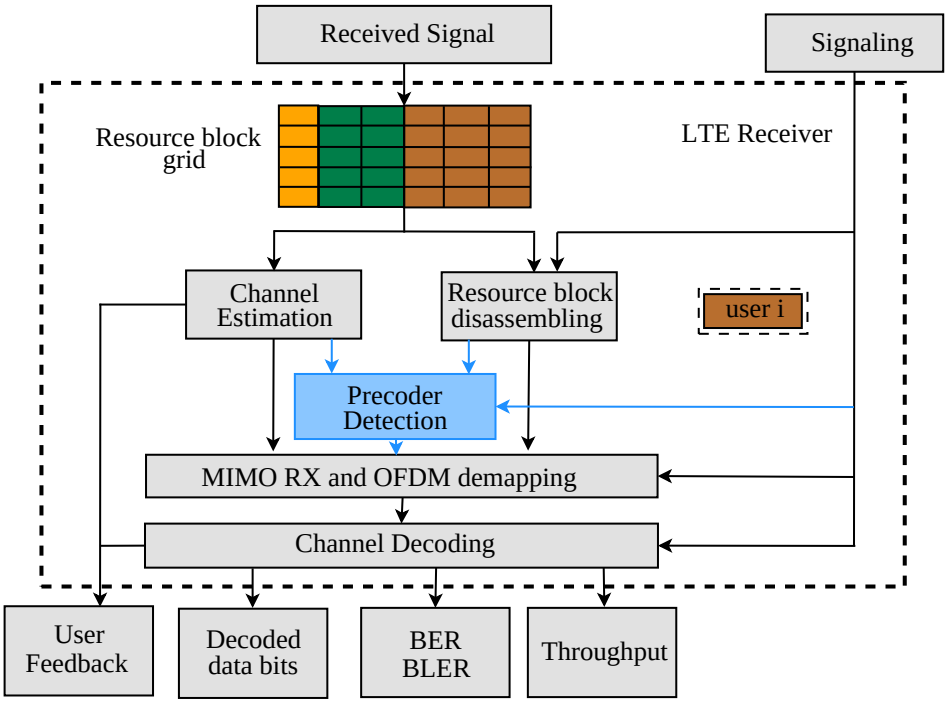

Figure 4.3: Modified structure of the LTE receiver, with our precoder detector in blue.

The structure of the LTE receiver is described in Fig. 4.3, with the precoder detection block, in blue, only applicable to the CMR. First, the receiver determines the RBs assigned for the user to retrieve the data sent to the user, and then calculates the received SNR by estimating the signal power and noise variance using regularly transmitted pilot signals. Next, it compensates for the carrier frequency offset (CFO) and timing offset (if they are introduced in the simulation). The channel is estimated using the CRSs present in the received signal, along with the estimated noise variance. The channel estimate is used for both user feedback calculation and subsequent demodulation and soft-demapping of the transmitted data symbols. These tasks are performed at both the PUE and CMR.

In addition to the above steps that are common to both the PUE and CMR, the CMR also employs our proposed precoder detection algorithms, in the block shown in blue in Fig. 4.3. It then uses the detected precoder matrix for soft-demapping of the transmitted symbols and the channel decoding.

In MIMO mode, the simulator uses a C implementation of a soft-output sphere decoder algorithm with a single tree search [33] for the soft-demapping. However, we observed that this implementation suffers from high computational requirements in case of 4x4 MIMO using 64 QAM. Hence, in order to avoid unreasonably long computation times for the simulations, we consider only lower modulation schemes for 4x4 MIMO, though our results can be extrapolated for 64 QAM as well.

4.2 Modifications

In this section, we explain the modifications we made to the simulator (v1.7r1089) to account for the user feedback in CLSM (shown in blue in Fig. 4.2) and to capture the feedback from the PUE and the data transmitted to it (shown in blue in Fig. 4.1), which is then played back when we simulate the CMR. We explain the precoder detection shown in Fig. 4.3 in chapter 5.

4.2.1 Transmitter

We modify the "LTE_TX.m" MATLAB® function to use the feedback provided by the user as follows. For each simulation run, we fix the MCS (corresponding to a particular CQI) used by the transmitter, along with the number of layers

used for each antenna configuration, i.e., 2x2 and 4x4 MIMO. Specifically, we send 2 codewords to the user, using 2 layers in 2x2 MIMO and 4 layers in 4x4 MIMO. Hence we consider full rank transmission in all simulations. The PMI is varied dynamically using the PCI feedback from the user, and the corresponding matrix is chosen from the codebook [15, section 6.3.4.2.3]. With this setup, for a given antenna configuration (i.e. 2x2 or 4x4 MIMO) and CQI, we evaluate the BLER of the proposed precoder detection methods for a range of SNR and compare them to the case where the precoder is known at the CMR.

4.2.2 Overall simulator structure

In order to simulate a secondary receiver that acts as a CMR, we first run a simulation between the eNodeB and the PUE and capture the user feedback generated by the "LTE_RX.m" MATLAB® function and the data transmitted to the PUE as a trace. In this case, the PMI is considered known at the PUE. Then, in the second run, we playback the trace to simulate the CMR and evaluate our precoder detection methods, as shown in Fig. 4.1, where we consider the PMI as unknown to the CMR. In this case, we also change the channel realization in the second run in order to simulate different channel conditions between the eNodeB and the CMR, as compared to that between the eNodeB and the PUE.

Chapter 5: Precoder Detection

As depicted in Fig. 4.3, we consider the problem of precoder detection after certain processing of the received signal is done, such as the CFO and timing offset compensation, and channel estimation. For a particular LTE subframe, we work with the received OFDM data symbols in frequency domain, which can be concisely represented as

$$\mathbf{y}_i = \mathbf{H}_i \mathbf{P} \mathbf{x}_i + \mathbf{n}_i, \tag{5.1}$$

where i indexes the symbols in the subframe, $\mathbf{y}_i$ is the i^{th} received data symbol vector, $\mathbf{P}$ is the precoder matrix used by the transmitter for the subframe, $\mathbf{H}_i$ is the frequency domain channel gain matrix observed by the i^{th} transmitted data symbol vector, $\mathbf{x}_i$, and $\mathbf{n}_i$ is the observed noise vector assumed to be Additive White Gaussian Noise (AWGN). The elements of $\mathbf{x}_i$ come from at most two different complex constellations, for the two different codewords sent by the transmitter.

The dimension of $\mathbf{y}_i$ is equal to the number of receive antennas, r, and that of $\mathbf{x}_i$ is equal to the number of layers used by the transmitter, l. The dimension of the channel gain matrix, $\mathbf{H}_i$, is $r \times t$ where t is the number of transmit antennas. The dimension of the precoder matrix, $\mathbf{P}$, is $t \times l$, with the constraint $l \leq t$.

In our setting, the channel gain matrix, $\mathbf{H}_i$, and noise variance, σ^2, are es-

timated, but the precoder matrix, $\mathbf{P}$, is unknown. Our goal is to estimate $\mathbf{P}$ based on the observation of M received data symbol vectors in the subframe, $\mathbf{y} := (\mathbf{y}_1, \ldots, \mathbf{y}_M)$. To this end, we propose two methods – simplified Maximum Likelihood (ML) detection and cluster variance algorithm – under a hypothesis testing framework.

5.1 Hypothesis testing framework

We formulate the problem under the framework of hypothesis testing, by defining the following hypotheses:

$$\mathcal{H}_j := \{\text{Precoder } \mathbf{P}_j \text{ is used}\}, \forall j \in \{1, \ldots, N\}, \tag{5.2}$$

where N is the size of the precoder codebook, i.e., the number of available precoder matrices, which depends on t and l. Since the prior distribution over the codebook may be time-varying, it is unlikely to be known in advance. For this reason, we assume that the adopted precoder is uniformly distributed over the codebook. Under this assumption, the optimal Bayesian hypothesis test that minimizes the error probability is the ML detector.

We wish to develop an ML hypothesis test to detect the precoder matrix, $\mathbf{P}$, based on the observation $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_M)$. To realize this ML detector, we would require the knowledge of a prior distribution on $\mathbf{x}_i$. Since the coded bits are scrambled at the transmitter to uniformly distribute them, it is reasonable to assume that the transmitted symbols are chosen according to a discrete uniform distribution

over constellations. Thus, we express the likelihood function for hypothesis $\mathcal{H}_j$ as

$$\begin{aligned} \mathcal{L}\left(\mathbf{P}_j|\,\mathbf{H}_i, \mathbf{y}_i\right) &= \frac{1}{|\mathcal{X}|} \sum_{\mathbf{x}_i \in \mathcal{X}} f\left(\mathbf{y}_i|\mathbf{H}_i, \mathbf{P}_j, \mathbf{x}_i\right), \\ \mathcal{L}\left(\mathbf{P}_j|\mathbf{H}, \mathbf{y}\right) &= \prod_{i=1}^{M} \mathcal{L}\left(\mathbf{P}_j|\mathbf{H}_i, \mathbf{y}_i\right), \end{aligned} \tag{5.3}$$

where $\mathbf{H} = (\mathbf{H}_1, \ldots, \mathbf{H}_M)$, $\mathcal{X}$ is the set of all possible transmit symbol vectors, and $f\left(\mathbf{y}_i|\mathbf{H}_i, \mathbf{P}_j, \mathbf{x}_i\right)$ is the conditional probability density of received symbol vector $\mathbf{y}_i$ given channel gain matrix $\mathbf{H}_i$, precoder matrix $\mathbf{P}_j$, and transmitted symbols $\mathbf{x}_i$. This conditional distribution is Gaussian with mean $\mathbf{H}_i\mathbf{P}_j\mathbf{x}_i$ and variance σ^2.

Conceptually, we can substitute the estimates $\hat{\mathbf{H}}_i$ and $\hat{\sigma}^2$ of the channel gain matrix and noise variance, respectively, in (5.3) to compute the likelihood function. Then, we can estimate $\mathbf{P}$ by maximizing (5.3) with respect to $\mathbf{P}_j$. However, this approach is computationally impractical due to the summation over all possible transmit symbol vectors.

To skirt this issue, we propose a modified ML detector that first decodes $\mathbf{x}_i$, for an observed $\mathbf{y}_i$, by assuming that the true precoder matrix is $\mathbf{P}_j$, for each $\mathbf{P}_j$. Then, we treat the decoded symbols as the actual transmitted symbols and formulate a simplified likelihood function for each hypothesis, $\mathcal{H}_j$, thereby avoiding the summation in (5.3). We elaborate on this technique, called as simplified ML detection, in the following section.

5.1.1 Simplified ML detection algorithm

First, we construct an MMSE filter, $\mathbf{G}_{MMSE,i}$, using the estimated channel gain matrix $\hat{\mathbf{H}}_i$ and the estimated noise variance $\hat{\sigma}^2$ to equalize the channel as

follows:

$$\begin{aligned} \mathbf{G}_{MMSE,i} &:= \left(\hat{\mathbf{H}}_i^H \hat{\mathbf{H}}_i + \hat{\sigma}^2 I\right)^{-1} \hat{\mathbf{H}}_i^H, \\ \tilde{\mathbf{y}}_i &:= \mathbf{G}_{MMSE,i} \mathbf{y}_i, \end{aligned} \tag{5.4}$$

where $(\cdot)^H$ denotes the Hermitian transpose, and I is the $t \times t$ identity matrix. Next, in order to compute the likelihood function for hypothesis $\mathcal{H}_j$, we obtain a hard decision, $\hat{\mathbf{x}}_i^{(j)}$, of the transmitted symbol vector assuming that the actual precoder is $\mathbf{P}_j$ under hypothesis $\mathcal{H}_j$, i.e.,

$$\hat{\mathbf{x}}_i^{(j)} := \underset{\mathbf{x} \in \mathcal{X}}{\operatorname{argmin}} \left|\left| \mathbf{P}_j^+ \tilde{\mathbf{y}}_i - \mathbf{x} \right|\right|, \tag{5.5}$$

where $\mathbf{P}_j^+$ is the pseudoinverse of $\mathbf{P}_j$. Therefore, this means that if $\mathbf{P}_j$ is indeed the actual precoder matrix used in (5.1), then $\hat{\mathbf{x}}_i^{(j)}$ would be the best estimate of the transmitted symbol vector. In fact it is the ML estimate or equivalently the minimum distance estimate of the transmitted symbol vector. We then formulate the following likelihood functions:

$$\begin{aligned} \mathcal{L}\left(\mathbf{P}_j | \hat{\mathbf{H}}_i, \mathbf{y}_i, \hat{\mathbf{x}}_i^{(j)}\right) &= f\left(\mathbf{y}_i | \hat{\mathbf{H}}_i, \mathbf{P}_j, \hat{\mathbf{x}}_i^{(j)}\right), \\ \mathcal{L}\left(\mathbf{P}_j | \hat{\mathbf{H}}, \mathbf{y}, \hat{\mathbf{x}}^{(j)}\right) &= \prod_{i=1}^{M} \mathcal{L}\left(\mathbf{P}_j | \hat{\mathbf{H}}_i, \mathbf{y}_i, \hat{\mathbf{x}}_i^{(j)}\right), \end{aligned} \tag{5.6}$$

where $\hat{\mathbf{x}}^{(j)} = \left(\hat{\mathbf{x}}_1^{(j)}, \ldots, \hat{\mathbf{x}}_M^{(j)}\right)$, and $\hat{\mathbf{H}} = \left(\hat{\mathbf{H}}_1, \ldots, \hat{\mathbf{H}}\mathbf{H}_M\right)$. Note that the difference between (5.3) and (5.6) is that we take the hard decisions, $\hat{\mathbf{x}}_i^{(j)}$, to be the correct transmitted symbols. Using (5.6), we obtain the log-likelihood function as

$$\log \mathcal{L}\left(\mathbf{P}_j | \hat{\mathbf{H}}, \mathbf{y}, \hat{\mathbf{x}}^{(j)}\right) = -\sum_{i=1}^{M} \frac{1}{2\hat{\sigma}^2} \left|\left| \mathbf{y}_i - \hat{\mathbf{H}}_i \mathbf{P}_j \hat{\mathbf{x}}_i^{(j)} \right|\right|^2 + K, \tag{5.7}$$

where K is a constant independent of j. Since logarithm is a strictly increasing function, we can maximize (5.7) to obtain the detected precoder as

$$\mathbf{P} := \underset{\mathbf{P}_j}{\operatorname{argmax}} \log \mathcal{L}\left(\mathbf{P}_j | \hat{\mathbf{H}}, \mathbf{y}, \hat{\mathbf{x}}^{(j)}\right). \tag{5.8}$$

The above analysis provides us with the simple ML algorithm, with (5.8) showing us what the algorithm essentially does. Algorithm 1 describes the steps performed in this approach. We implemented this in MATLAB, using C to improve the efficiency.

Algorithm 1: The overall simple ML algorithm to detect the precoder matrix at the CMR.

algorithm Simplified_ML ($\hat{\mathbf{H}}, \mathbf{y}, \hat{\sigma}^2$):
Input : The channel estimates, $\hat{\mathbf{H}}$, the received symbols, $\mathbf{y}$, and the noise variance estimate, $\hat{\sigma}^2$, in addition to the precoder matrix codebook
Output: The detected precoder matrix, $\mathbf{P}$
Compute the MMSE filter, $\mathbf{G}_{MMSE,i}$, and equalized received symbols, $\tilde{\mathbf{y}}_i$ for $i = 1, \ldots, M$ using (5.4);
foreach *Precoder matrix in the Precoder codebook* **do**
    Evaluate the hard decision, $\hat{\mathbf{x}}_i^{(j)}$, using (5.5);
    Compute the likelihood function and log-likelihood function using (5.6) and (5.7);
end
Evaluate the detected precoder by maximizing the log-likelihood function using (5.8);

5.1.2 Cluster variance algorithm

A variation of the simplified ML detector can be obtained as follows: For each $\mathbf{P}_j$, $j = 1, \ldots, N$, define

$$l\left(\mathbf{P}_j; \hat{\mathbf{H}}, \mathbf{y}, \hat{\mathbf{x}}^{(j)}\right) := \sum_{i=1}^{M} \left|\left|\hat{\mathbf{x}}_i^{(j)} - \mathbf{P}_j^{+} \tilde{\mathbf{y}}_i\right|\right|^2. \tag{5.9}$$

The cluster variance scheme chooses the precoder given by

$$\mathbf{P} := \underset{\mathbf{P}_j}{\operatorname{argmin}} \; l\left(\mathbf{P}_j; \hat{\mathbf{H}}, \mathbf{y}, \hat{\mathbf{x}^{(j)}}\right). \tag{5.10}$$

Notice the similarity of the cluster variance approach to the minimum distance decision rule. Intuitively, for the true precoder $\mathbf{P}$, $\mathbf{P}^{+}\tilde{\mathbf{y}}_i$ should lie near the constellation point, $\mathbf{x}_i$, as $\mathbf{G}_{MMSE,i}$ is designed to nullify the effect of the channel, $\mathbf{H}_i$. Thus, we expect that (5.9) is minimized for the true precoder matrix, which is indeed the case for the minimum distance decision rule.

Here, however, the decision rule defined in (5.10) is not the minimum distance rule, because (5.9) is not a sufficient statistic for the given problem. This is because the modified noise term in the hard decision, $\hat{\mathbf{x}}_i^{(j)}$, obtained after applying (5.4) and (5.5), namely $\mathbf{P}_j^{+}\mathbf{G}_{MMSE,i}\mathbf{n}_i$, does not consist of i.i.d. random variables. For this reason, the performance of this algorithm is slightly poorer than that of the first scheme. On the other hand, as we will demonstrate below, this scheme enjoys lower computational requirements, and hence can be more practical in some scenarios, especially when the performance degradation compared to the simplified ML algorithm is not significant.

We describe the steps taken in the cluster variance algorithm in Algorithm 2. This algorithm is very similar to Algorithm 1, except for line 5 and 7.

5.2 Complexity

The computational complexity for both algorithms is $O(MNC)$, where C is the size of the symbol constellation (i.e., 4-, 16- or 64-QAM), because (5.7) and (5.9)

Algorithm 2: The overall cluster variance algorithm to detect the precoder matrix at the CMR.

algorithm Cluster_Variance ($\hat{\mathbf{H}}, \mathbf{y}, \hat{\sigma}^2$):

Input : The channel estimates, $\hat{\mathbf{H}}$, the received symbols, $\mathbf{y}$, and the noise variance estimate, $\hat{\sigma}^2$, in addition to the precoder matrix codebook

Output: The detected precoder matrix, $\mathbf{P}$

Compute the MMSE filter, $\mathbf{G}_{MMSE,i}$, and equalized received symbols, $\tilde{\mathbf{y}}_i$ for $i = 1, \ldots, M$ using (5.4);

foreach *Precoder matrix in the Precoder codebook* **do**

    Evaluate the hard decision, $\hat{\mathbf{x}}_i^{(j)}$, using (5.5);

    Compute the cluster variance metric, $l\left(\mathbf{P}_j; \hat{\mathbf{H}}, \mathbf{y}, \hat{\mathbf{x}}^{(j)}\right)$, using (5.9);

end

Evaluate the detected precoder by minimizing the cluster variance metric using (5.10);

are performed once for each of the N precoders, and C comparisons are needed for the hard decision $\hat{\mathbf{x}}_i^{(j)}$, $i = 1, \ldots, M$, $j = 1, \ldots, N$. To better discern the difference in computational requirements, we need to consider the number of floating point operations needed in these two equations, assuming all operations consume the same number of processor cycles. We find that (5.7) needs $M\left(2t\left(l + r\right) + 2r - t\right)$ floating point operations, while (5.9) needs only $3Ml - 1$ operations. Thus, for 2x2 MIMO, (5.7) needs $(4l+10)M$ floating point operations, while it needs $(8l+36)M$ operations for 4x4 MIMO. Hence, the first scheme is computationally more demanding than the second scheme. Note that both the algorithms have lower complexity than a Soft Sphere Decoding (SSD) receiver, which is $O\left(M^3\right)$ [34] (note: $N, C \ll M$). Hence, these algorithms are computationally feasible.

5.3 Ambiguity in precoders in 4x4 MIMO with 4 layers

In the case of the 4x4 MIMO, when using 4 layers, there are 16 different 4×4 precoder matrices [15, table 6.3.4.2.3-2]. These 16 precoders can be grouped into 3 sets of 4 precoders, and 2 sets of 2 precoders, with the property that each set of precoders are "permutations" of each other, i.e., if $\mathbf{P}_i$ and $\mathbf{P}_j$ belong to the same set, then $\mathbf{P}_i = \mathbf{A}\mathbf{P}_j$, where matrix $\mathbf{A}$ rearranges the rows of $\mathbf{P}_j$ with possible sign changes. Due to this structure of the precoder matrices, an algorithm that solely relies on received symbols, including ours, will be unable to resolve the precoder matrix any further than these sets. The reason for this can be intuitively observed through the symmetry of the transmit symbol vectors set, $\mathcal{X}$, and the symmetry in the above mentioned precoder sets. As a consequence of this symmetry, (5.7) and (5.9) evaluate to the same quantity for each precoder, $\mathbf{P}_j$, within a precoder set.

For this reason, our proposed schemes are augmented with an additional step, in the 4x4 MIMO case, to utilize the channel coding, built into the LTE communication scheme, to resolve this ambiguity as follows. Once our algorithm identifies a set of precoders, $\mathcal{P}$, which solve (5.8) (or (5.10)), we pass each of the precoders, $\mathbf{P}_j \in \mathcal{P}$, to the subsequent stages of the receive chain at the CMR. The CMR tries to decode the transmitted symbols using each of the precoders, $\mathbf{P}_j \in \mathcal{P}$, in parallel receive chains, applies the turbo decoder and verifies the Cyclic Redundancy Check (CRC) bits. It chooses the precoder that passes the CRC verification as the detected precoder $\mathbf{P}$.

Chapter 6: Simulation Results

In this chapter, we will discuss two sets of experiments that we have conducted using the MATLAB® based LTE simulator described in chapter 4. In the first set of experiments, we investigate the performance of the two proposed precoder detection algorithms detailed in chapter 5. In this experiment, using the data and feedback traces captured from a simulation between the eNodeB and the PUE, we simulate a scenario between the eNodeB and CMR for two different channel models and compare the performance at the CMR when it knows the PMI versus estimates it using our algorithms. This allows us to compute the performance penalty due to the precoder detection algorithms.

Next, we aim to evaluate the overall benefit of using the precoder detection algorithms in conjunction with the cooperative decode-and-forward relaying mechanism at the CMR for improving the performance of the PUE. We identify specific real world channel conditions at the PUE and CMR when it is beneficial for the PUE to make use of the cooperative relaying mechanism with the CMR employing our proposed algorithms. This helps us demonstrate real scenarios where our solution is valuable.

6.1 Simulation setup

The simulator operates at a carrier frequency of 2110 MHz with a system bandwidth of 1.4 MHz, which is the lowest system bandwidth in the LTE specifications. For this bandwidth, there are 12 RBs in each subframe of the LTE frame. The transmitter schedules all 12 RBs to the PUE via static scheduling as we are only considering a single user scenario, and 1000 subframes are simulated in each run. The maximum HARQ (Hybrid Automatic Repeat reQuest) retransmissions are set to 0 in the presence of the CMR. The CMR uses an MMSE channel estimator and an SSD receiver, as they are well known to outperform other methods such as Least Squares (LS) channel estimator [35] and linear equalizers [36], respectively. The proposed precoder detection methods use $M = 500$ in (5.7) and (5.9) and perform the detection for every subframe. In addition, we also considered a system bandwidth of 10 MHz and observed similar results to those for 1.4 MHz, and hence we omit these results for brevity.

6.1.1 Channel models

We model the channel PDP using the ITU channel models described in [13,14]. Specifically, we consider the following channel models in our experiments, namely Extended Pedestrian A (EPA), Extended Typical Urban (ETU), Extended Vehicular A (EVA), Vehicular A (VA) and Vehicular B (VB), which incorporate three degrees of spatial correlation between the antennas for MIMO conformance testing.

6.2 Experiment 1

In the first set of experiments, as explained at the beginning of this chapter, we investigate the performance penalty observed because of using our proposed precoder detection algorithms to identify the PMI at the CMR. We consider low antenna correlation level as defined in [14, table B.2.3.2-1], which is part of the minimum performance test conditions specified in [14] for multi-layer CLSM. This condition can be met at eNodeB in practice, while it can be justified at the CMR by assuming cross polarized antennas [37]. We simulate both 2x2 and 4x4 MIMO modes and consider two channel conditions – (1) EPA at user speed of 3km/h, and (2) ETU at user speed of 60km/h. These two speeds correspond to a maximum Doppler frequency of 5 Hz (low), and 120 Hz (high), respectively. For each scenario, we evaluate the BLER with the precoder detectors and compare them to the case where the precoder is known at the CMR. For each CQI and MIMO mode, we generate a trace of the PMI feedback and transmitted data of the PUE with the same channel PDP as the CMR, but with a different channel realization by changing the random seed, and playback this trace in our simulations with the CMR.

In Fig. 6.1 and Fig. 6.2, we observe that the simplified ML detection algorithm experiences a performance degradation of at most 2 dB at a BLER of 10% for both EPA and ETU channel models, for all CQIs we consider. Similarly, in Fig. 6.3 and Fig. 6.4, we again note that there is at most a 2 dB degradation in BLER performance for the proposed simplified ML algorithm compared to the known PMI case. Moreover, the performance for high CQIs is very close to the "known PMI"

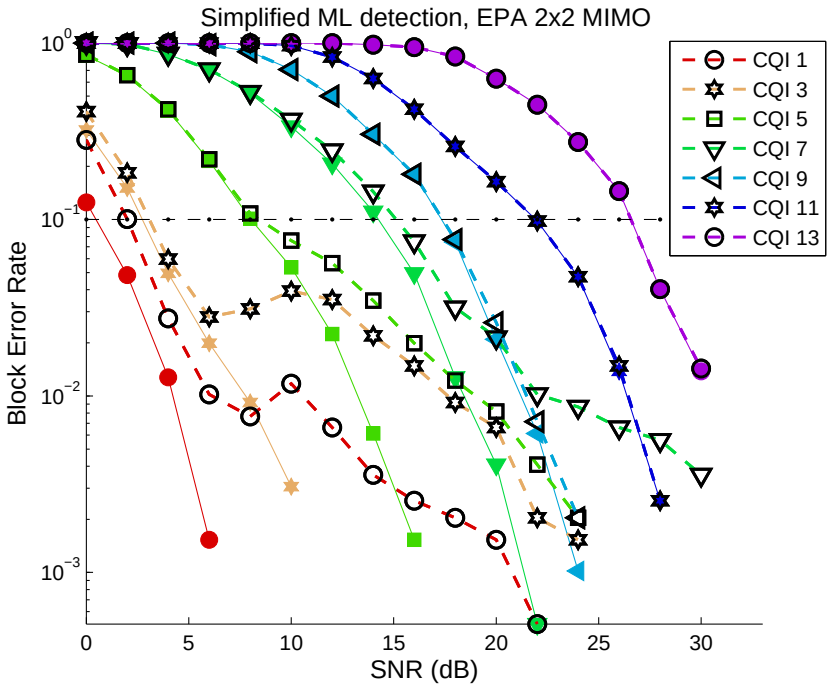

Figure 6.1: BLER performance for EPA at 3km/h, 2x2 MIMO. Solid lines – "known PMI"; dashed lines – "simplified ML detection".

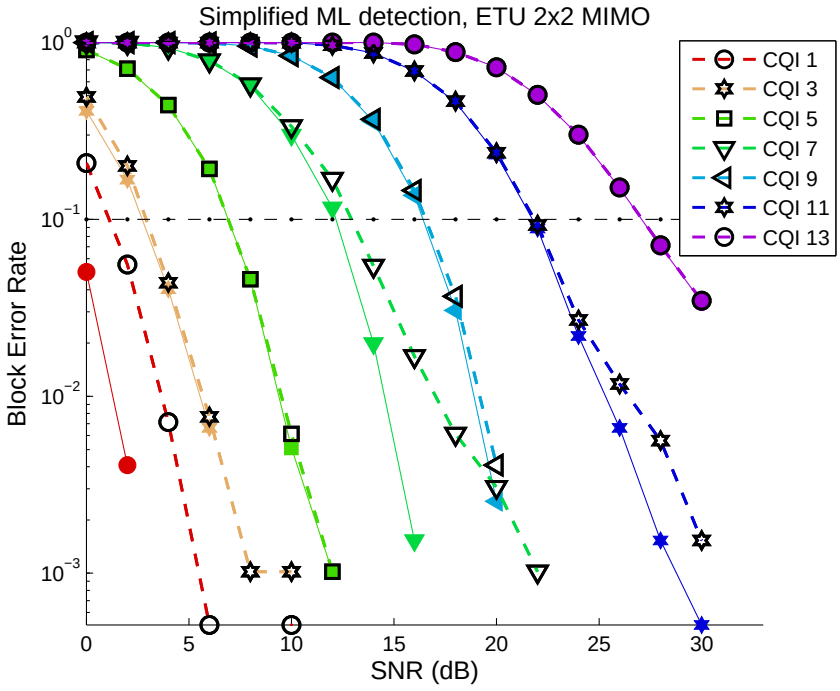

Figure 6.2: BLER performance for ETU at 60km/h, 2x2 MIMO. Solid lines – "known PMI"; dashed lines – "simplified ML detection".

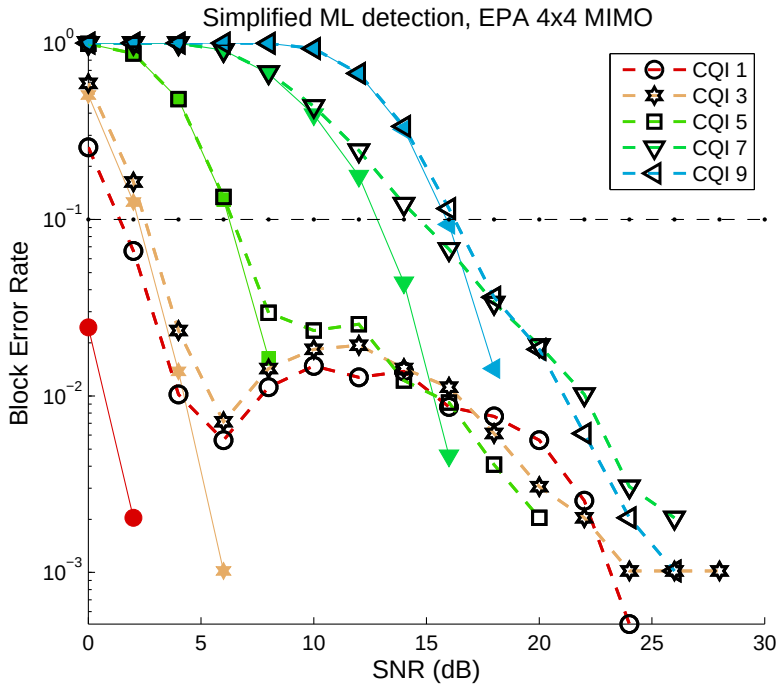

Figure 6.3: BLER performance for EPA at 3km/h, 4x4 MIMO. Solid lines – "known PMI"; dashed lines – "simplified ML detection".

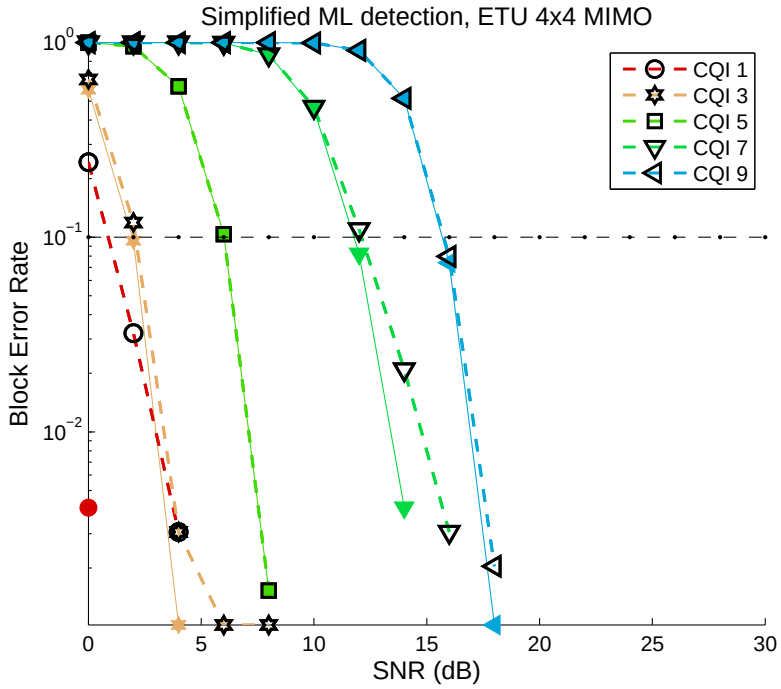

Figure 6.4: BLER performance for ETU at 60km/h, 4x4 MIMO. Solid lines – "known PMI"; dashed lines – "simplified ML detection".

case, especially for CQI 9 and above in 2x2 MIMO. Recall that we do not consider CQIs 10 through 15 (corresponding to 64 QAM) in 4x4 MIMO for the reason stated at the end of section 4.1.

While we did compute the confidence intervals for all the figures shown in this section, we do not show them in the above figures to keep them readable. We ran each simulation 5 times, and we used the data from these 5 runs to compute the confidence intervals. We limited ourselves to no more than 5 runs due to the prohibitively long simulation times. We calculated the sample mean, $\bar{x}$, for the BLER data in each simulation, and then computed the sample standard deviation, $\bar{\sigma}$. Then, we computed the margin of error using the formula

$$\text{Margin of error} = z^* \frac{\bar{\sigma}}{\sqrt{n}}, \tag{6.1}$$

where the critical value, $z^* = 1.96$, for a 95% confidence interval. Here, $n = 5$, for our 5 runs. Plugging in the values for sample mean and sample standard deviation for each scenario, we obtained a margin of error, for a 95% confidence interval, of about 1.5% of BLER for the ETU channel model scenarios in both 2x2 and 4x4 MIMO, for both the simplified ML and cluster variance algorithms, seen in Figs. 6.2 and 6.4, as well as Figs. 6.6 and 6.8, discussed later on.

However, for the EPA channel model, in Figs. 6.1, 6.3, 6.5 and 6.7, the margin of error for a 95% confidence interval was higher, nearly 4% of BLER for the lower CQI, from CQI 1 to 7. At higher CQI, beyond CQI 7, the margin of error for a 95% confidence interval is about 1% of BLER. While we are not certain as to the reason for the higher margin of error in the EPA scenarios, it appears to be due to

the non-monotonicity issue at the lower CQI, discussed later, which mainly occurs with the EPA channel model. Nevertheless, our primary claim that the performance degradation is at most 2 dB at a BLER of 10% remains valid, as the high margin of error in EPA scenario occurs only at below 10% BLER.

We find it interesting that the performance degradation due to unknown PMIs tends to diminish with increasing SNR and CQI. We suspect that this is because when the CMR enjoys high SNR it can obtain more accurate *channel estimates* using the CRSs. This enables the precoder detectors to determine the employed precoders more reliably, thereby leading to a smaller performance gap.

However, there are some discrepancies in the figures above. As seen from the simulations with the EPA channel model in Figs. 6.1 and 6.3, specifically for the lower CQI, the performance of the simplified ML algorithm is not monotonous as the SNR increases. The performance deteriorates at moderate SNR before improving at higher SNR. This non-monotonicity is reduced as we averaged more simulation runs; the figures were obtained by averaging the data from 5 separate simulation runs of 1000 subframes each. We couldn't consider more runs due to the prohibitively long simulation times. Additionally, at lower CQI the BLER is limited by errors caused by the simplified ML algorithm for the EPA channel model, rather than errors due to transmission. These errors are not eliminated by increasing the SNR and hence, the performance of the algorithm levels out at higher SNR.

Furthermore, by comparing Figs. 6.1–6.4 to those in Figs. 6.5–6.8 that show the analogous results for the cluster variance algorithm, we observe that this algorithm performs poorer than simplified ML detection for low CQI, especially for 2x2

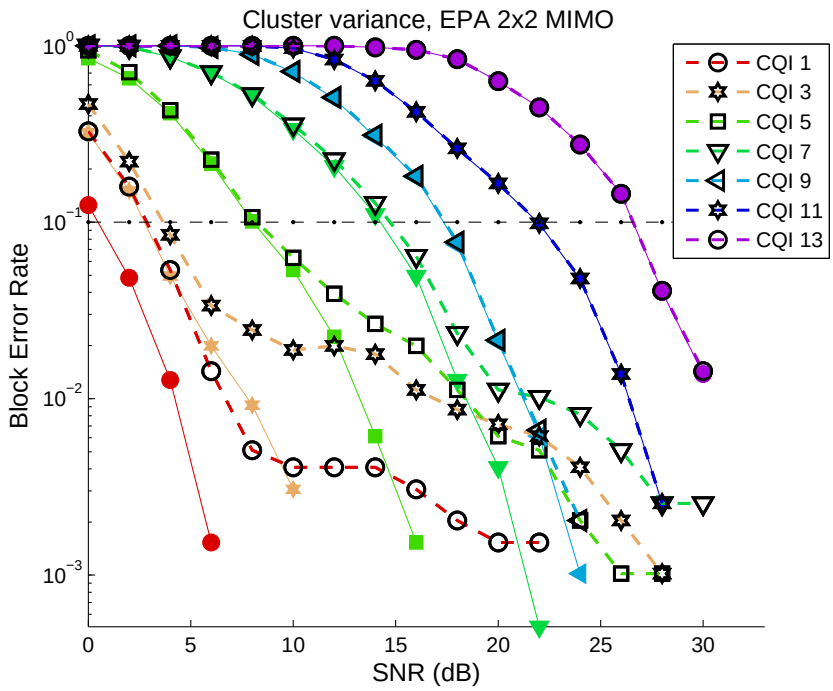

Figure 6.5: BLER performance for EPA at 3km/h, 2x2 MIMO. Solid lines – "known PMI"; dashed lines – "cluster variance".

MIMO, while its performance is typically closer to that of the simplified ML detection at higher CQIs. Moreover, the performance of both schemes is nearly identical for 4x4 MIMO for all CQIs considered. We expect such behavior since (5.9) ignores the effect of the modified noise term and the effect of this approximation is lessened at higher SNRs. Also, in case of 4x4 MIMO, each codeword is transmitted on 2 antennas, slightly increasing diversity and hence diminishing the performance gap.

The confidence interval for these scenarios is as explained earlier. Furthermore, we again note, as before, that for the low CQI scenarios depicted in Figs. 6.5 and 6.7, the BLER performance is non-monotonic as SNR is increased. Again, this issue occurs only for the EPA channel model. We offer a similar explanation as before, and note that for EPA channel model the errors caused due to the algorithm outweigh

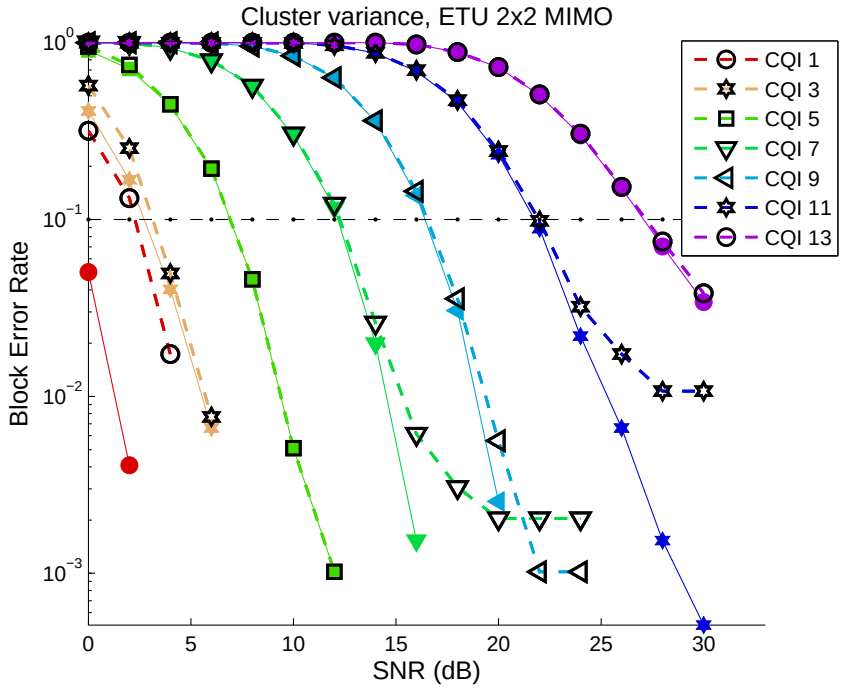

Figure 6.6: BLER performance for ETU at 60km/h, 2x2 MIMO. Solid lines – "known PMI"; dashed lines – "cluster variance".

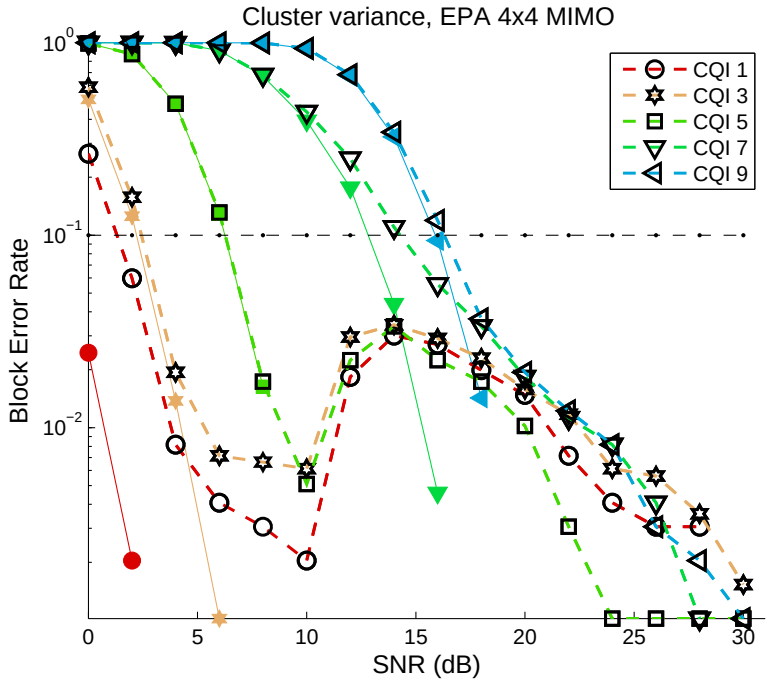

Figure 6.7: BLER performance for EPA at 3km/h, 4x4 MIMO. Solid lines – "known PMI"; dashed lines – "cluster variance".

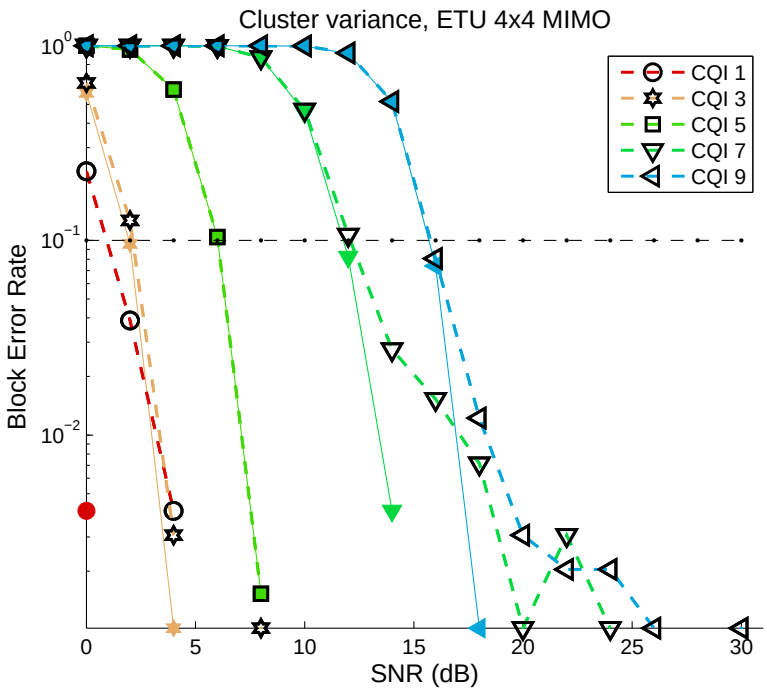

Figure 6.8: BLER performance for ETU at 60km/h, 4x4 MIMO. Solid lines – "known PMI"; dashed lines – "cluster variance".

that due to transmission and hence the performance of the algorithm plateaus at higher SNR, for the lower CQI scenarios.

6.3 Experiment 2

Going forward, we are keen to find out if the proposed cooperative relaying technique at the CMR, using the precoder detection algorithms, is valuable in real-life scenarios. We already found out that the precoder detection algorithms themselves perform quite well in detecting unknown PMI. We know aim to identify specific real world channel conditions at the PUE and CMR where the CMR can obtain a better BLER performance compared to the PUE. Then, in such circumstances, it would be beneficial for the PUE to utilize the mobile relay to improve its

performance.

Note that in order to fully evaluate the performance of the relay mechanism, we should also consider the transmission between the CMR and PUE, after the CMR decodes the data. However, we do not study this aspect, due to the limitation of our simulation setup. We assume that the transmission between the eNodeB and CMR is the bottleneck and hence focus on this aspect.

We consider a 2x2 MIMO channel setup in all our simulations. We use a similar setup to the previous section and consider a low antenna correlation level at both the PUE and CMR. We simulate a 2x2 MIMO channel between the eNodeB and PUE as well as between the eNodeB and CMR. We consider several channel models at both the PUE and CMR, namely the EPA, ETU, EVA, VA and VB channel models. We consider a pedestrian speed of 3km/h for the user experiencing the EPA channel model and a vehicular speed of 60km/h for the user experiencing the other channel models. All our simulations consider the simplified ML algorithm at the CMR for detecting the precoders due to its better performance. However, we expect similar results with the cluster variance algorithm as well.

6.3.1 Channel model interpretation

The above channel models can be interpreted as specific real world scenarios. Given the setup considered when developing the channel models in [13, 14], we can consider the EPA channel to represent an indoor environment, characterized by small cells and low transmit power. The ETU channel model can be treated as

an urban environment, with a high density of tall buildings, and a relatively large delay spread, where the user is in a moving vehicle. The large delay spread causes poor performance, so the user is likely at the edge of the cell. The EVA channel model is again a vehicular scenario with large cells and medium delay spread, usually urban or sub-urban. The VA and VB models are similar to EVA, except that VB is more extreme with a much larger delay spread (maximum excess delay of $20,000$ns). Therefore, VB depicts a user who is at the edge of the cell.

With this interpretation in mind, we note that each simulation setup corresponds to a real world scenario, albeit approximately, which we shall explain in the text.

6.3.2 Simulation setups and results

For each scenario, we first simulate the transmission between the eNodeB and the PUE and capture the feedback and data traces. Then, we simulate the transmission between the eNodeB and the CMR by replaying the captured traces. We vary the channel models at the PUE and CMR to create different scenarios and then compute the BLER performance for both of them for various SNR and CQI.

Figs. 6.9–6.11 depict scenarios where the CMR performs better than the PUE. Firstly, we note that we do not show the confidence intervals to keep the figures readable. As before, we computed the 95% confidence intervals by taking the results from 5 simulation runs. We plot the figures by averaging the results from these 5 simulation runs. The margin of error is about 3% of BLER, in Fig. 6.9, for

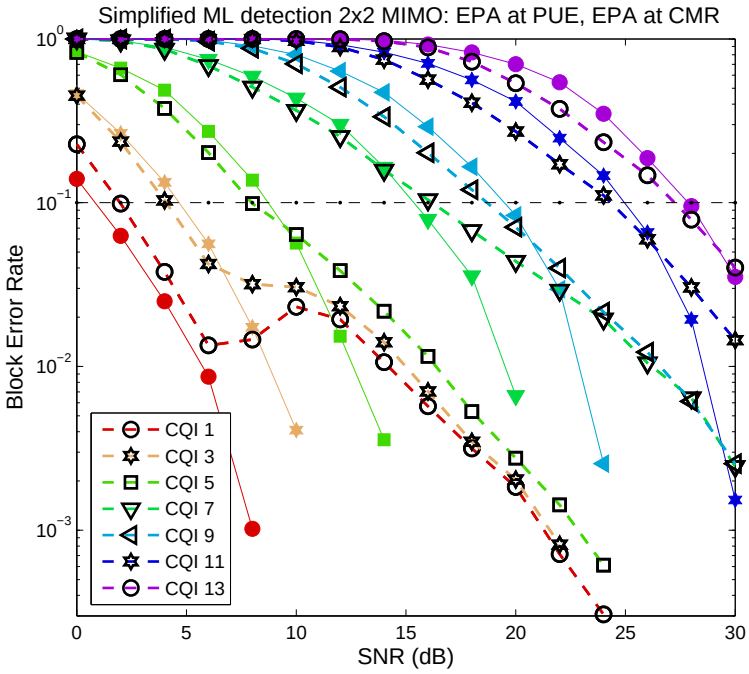

Figure 6.9: BLER performance: both PUE and CMR with EPA channel model, 2x2 MIMO. Solid lines – PUE; dashed lines – CMR using simplified ML detection.

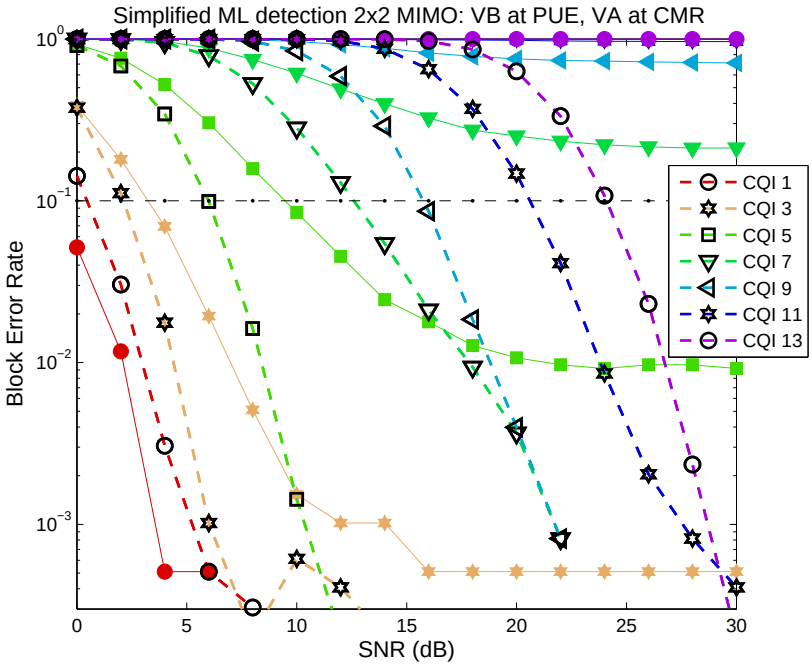

Figure 6.10: BLER performance: PUE with VB channel model, CMR with VA channel model, 2x2 MIMO. Solid lines – PUE; dashed lines – CMR.

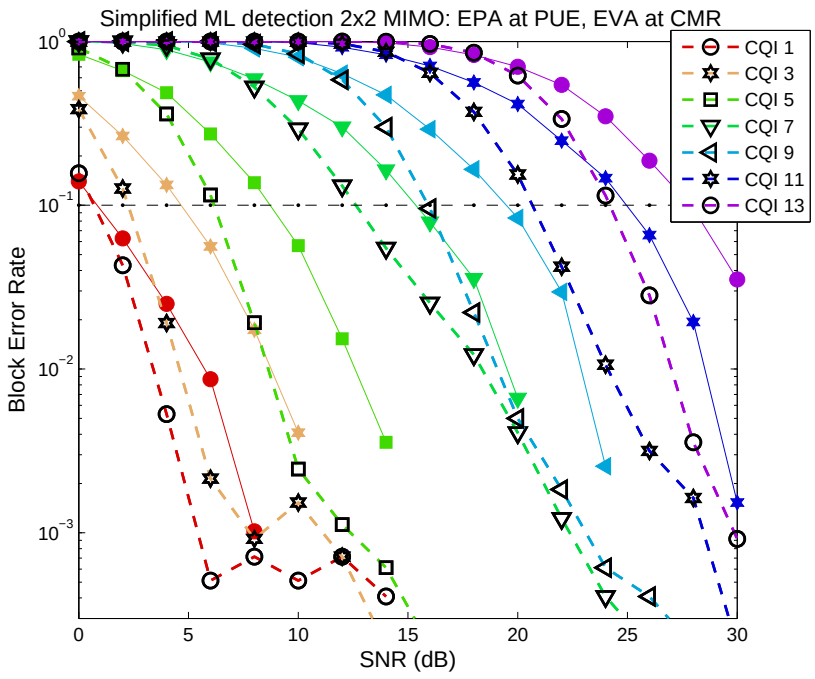

Figure 6.11: BLER performance: PUE with EPA channel model, CMR with EVA channel model, 2x2 MIMO. Solid lines – PUE; dashed lines – CMR.

lower CQI below CQI 7, whereas it is about 1% of BLER at CQI above 7. On the other hand, the margin of error is in the range of $1-1.5\%$ for the other scenarios in this section. Once again, the higher margin of error primarily occurs with the EPA channel model at the CMR.

In Fig. 6.9, we see that when both the PUE (shown in solid lines) and the CMR (shown in dashed lines) experience the EPA channel and move at 3km/h, the CMR observes better BLER performance than the PUE. Specifically, for most CQI, the CMR sees about 1 dB improvement in error performance at 10% BLER compared to the PUE. Similarly, in Fig. 6.10, we see that the CMR performs better when it observes a VA channel and the PUE observes a VB channel, both moving at 60km/h, especially at higher CQI where it sees several dB of improvement at

10% BLER. In fact, from our simulations we noted that the VB channel model is one of the poorest performing scenarios. Finally, in Fig. 6.11, we again see the CMR performing better when it observes an EVA channel moving at 60km/h, with about 2-4 dB improvement in performance at 10% BLER compared to the PUE that observes an EPA channel moving at 3km/h.

Once again, the BLER performance for the EPA channel, shown in Fig. 6.9, is not monotonous at lower CQI, as discussed in the previous experiment. On the other hand, in Fig. 6.10, the performance of the VB channel model is very poor. The reason for this is the extremely high maximum excess delay of the power delay profile for this model, which is $20,000$ns. In contrast, the maximum excess delay for other channel models is around $500 - 5,000$ns. One can consider this channel to represent a user in a moving vehicle at the edge of a cell of the mobile network. Consequently, the performance is quite poor.

On the other hand, the scenario depicted in Fig. 6.9, with the EPA channel, can be considered as one where PUE is a pedestrian inside a building with poor reception, and the CMR is also a pedestrian near the building with better reception. Furthermore, the scenario in Fig. 6.11 is one where the PUE is again inside a building and the CMR is in a moving vehicle nearby. As demonstrated by the results, we would expect better performance from the CMR for these two scenarios.

On the other hand, in Figs. 6.12–6.14, the performance of the CMR is not as clear-cut as earlier, compared to the PUE. Specifically, in Fig. 6.12, where the PUE observes an EVA channel and the CMR observes an ETU channel, both moving at 60km/h, the CMR performs poorer than the PUE and sees about 1 dB deterioration

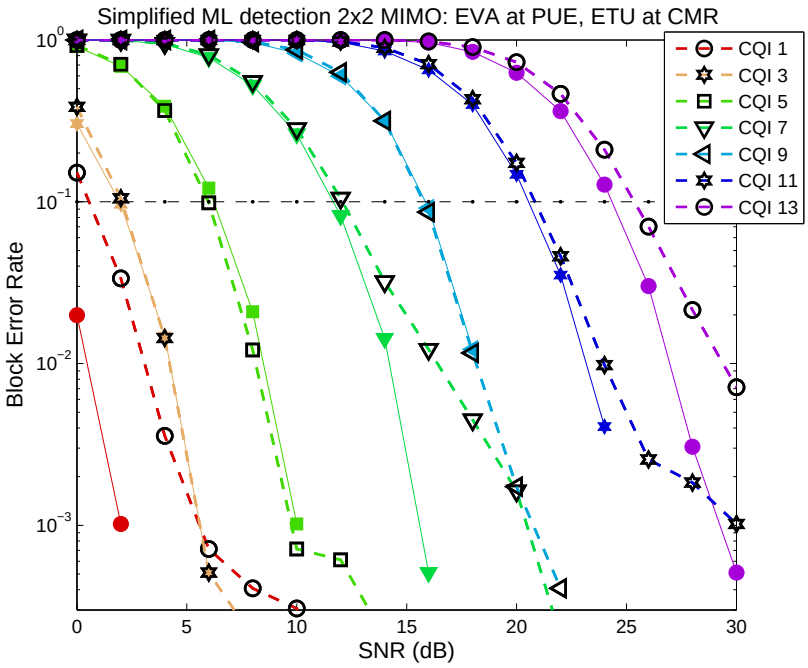

Figure 6.12: BLER performance: PUE with EVA channel model, CMR with ETU channel model, 2x2 MIMO. Solid lines – PUE; dashed lines – CMR.

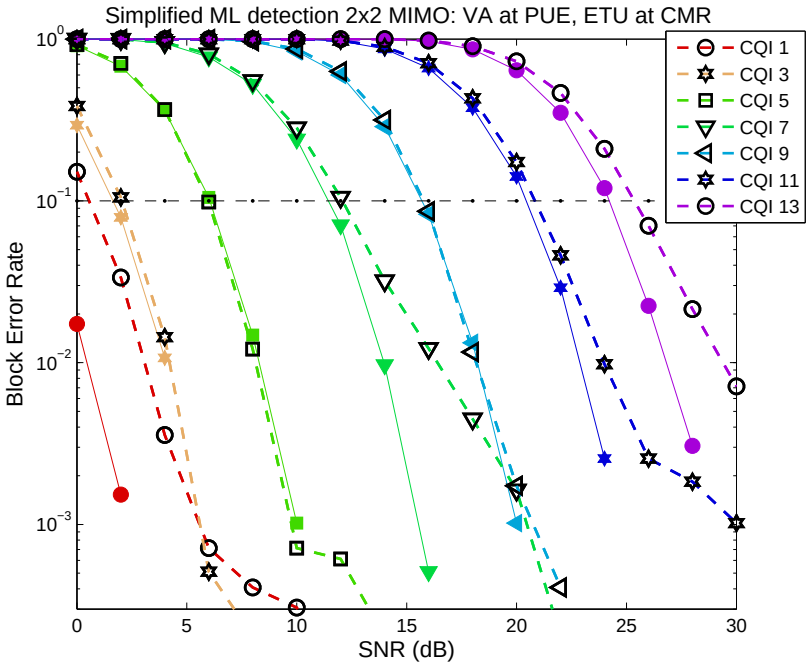

Figure 6.13: BLER performance: PUE with VA channel model, CMR with ETU channel model, 2x2 MIMO. Solid lines – PUE; dashed lines – CMR.

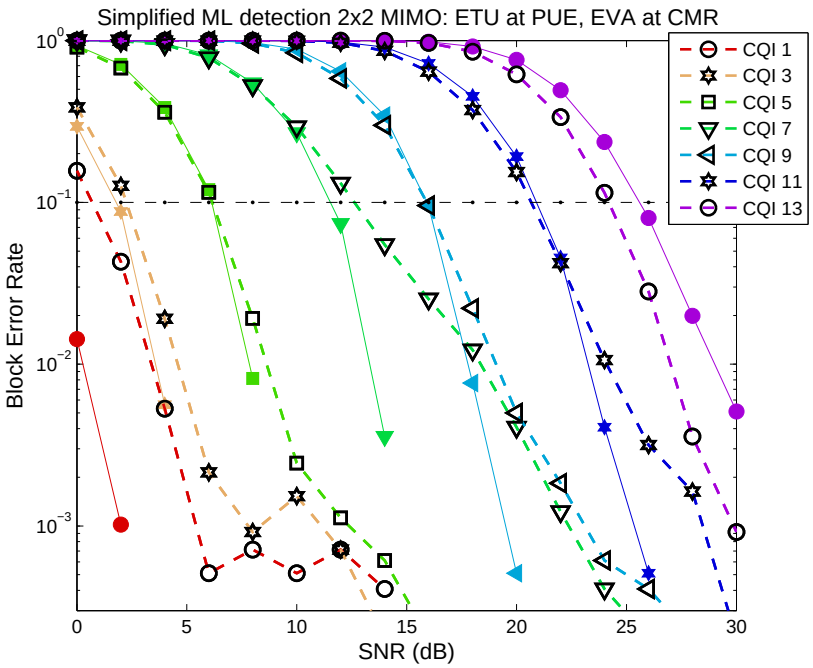

Figure 6.14: BLER performance: PUE with ETU channel model, CMR with EVA channel model, 2x2 MIMO. Solid lines – PUE; dashed lines – CMR.

in performance at 10% BLER compared to the PUE, especially at higher CQI. Similarly, in Fig. 6.13, where the PUE observes a VA channel and the CMR sees an ETU channel, both moving at 60km/h, the CMR sees a 1 dB deterioration in performance compared to the PUE at 10% BLER at higher CQI, while it performs comparably at lower CQI. Furthermore, in Fig. 6.14, where the PUE observes an ETU channel and the CMR observes an EVA channel, both moving at 60km/h, the CMR performs poorer than the PUE for low CQI, but performs about 1 dB better at higher CQI.

Once again, if we interpret these scenarios, we note that Figs. 6.12 and 6.13 depict scenarios where the PUE and CMR are moving in a vehicle in an urban area, but the CMR experiences poorer performance, presumably at the edge of the cell.

Furthermore, Fig. 6.14 is similar to the previous case but the PUE is the one that experiences the poorer channel (ETU channel) at the cell edge. Again, we find that the results match with the interpretation quite well.

We conducted similar experiments for the 4x4 MIMO scenario and observed similar results.

From these results, we can identify quite a few real world scenarios that we have simulated, albeit approximately due to the limitations of the simulator, where the CMR does improve the performance of the PUE using the cooperative relaying technique with our proposed algorithms. Indeed, the results seem to follow our intuition based on the interpretation of each setup - such as the cell edge user being outperformed by one with a better channel. Hence, from these simulation results, we conclude that our proposed cooperative decode-and-forward relaying mechanism using a CMR, in conjunction with the precoder detection algorithms, can be profitably employed in LTE networks.

Chapter 7: Conclusions

We considered the problem of precoder detection at the CMR under the scenario where the PUE communicates with the eNodeB using CLSM. We formulated the problem in a hypothesis testing framework and developed two algorithms - *simplified ML detection* and *cluster variance*. We described our system setup using the LTE Downlink link-level simulator [28] and evaluated the performance of detecting the PMI using the proposed algorithms in comparison to the scenario where it is known at the CMR. Our results show that the simplified ML detection algorithm suffers at most a 2 dB deterioration in performance for all CQIs, compared to the case where the PMI is known at the CMR. Furthermore, the performance improves for higher CQI and at higher SNR. Moreover, the cluster variance algorithm performs poorly at lower CQI for 2x2 MIMO, but performs almost identically to the simplified ML detection algorithm for higher CQI and for 4x4 MIMO scenarios, while being computationally simpler. Furthermore, we note that both algorithms are computationally feasible and the complexity is linear in M, N, and C.

Additionally, we also performed experiments for both the 2x2 MIMO and 4x4 MIMO scenarios to identify the channel conditions when using the CMR benefits the PUE and improves its BLER performance. Our aim was to evaluate the overall

benefit of using the proposed precoder detection algorithms at the CMR in conjunction with the cooperative decode-and-forward relaying mechanism for improving the performance at the PUE. We identified several different channel conditions when the CMR does perform better than the PUE, in some cases observing a 1-2 dB improvement in performance and even observing 4 dB and higher amount of improvement in performance depending on the scenario. While we also observed a few instances when the CMR performs poorly than the PUE, nevertheless we conclude that there exist a large variety of physical scenarios where using the precoder detection algorithms at the CMR and performing the cooperative decode-and-forward relaying mechanism can help improve the BLER performance at the PUE. Hence, our proposed algorithms and the cooperative relaying mechanism at the CMR can be gainfully employed in LTE networks.

Chapter 8: Future Work

For our future work, we wish to investigate other techniques to resolve the ambiguity mentioned in section 5.3. Furthermore, we would like to expand our simulation setup to include interference from neighboring eNodeBs at the relay and further study the effect of dynamic scheduling of the primary user's transmit signal in a multi-user MIMO scenario. Additionally, we would like to perform simulations where we transmit meaningful data to the PUE, rather than the random data that we have used in the simulations presented in this thesis. Doing so would allow us to additionally verify whether the data is successfully decoded at the PUE and CMR, and observe any negative effects of errors in the decoding of the data. Furthermore, we must also address the transmission from the CMR to the PUE in order to analyze the end-to-end performance for the PUE when using the mobile relay, which we ignored in this thesis. Finally, we wish to evaluate the performance of the proposed schemes on a physical LTE channel using a physical testbed.

Bibliography

[1] *3GPP TR 36.814, Evolved Universal Terrestrial Radio Access (E-UTRA); Further advancements for E-UTRA physical layer aspects*, Release 9, v9.0.0, March 2010.

[2] *3GPP TR 36.912, Feasibility Study for Further Advancements for E-UTRA (LTE-Advanced)*, Release 10, v10.0.0, March 2011.

[3] W. Li, C. Zhang, X. Duan, S. Jia, Y. Liu, and L. Zhang, "Performance Evaluation and Analysis on Group Mobility of Mobile Relay for LTE Advanced System," in *Vehicular Technology Conference (VTC Fall), 2012 IEEE*, Sept 2012, pp. 1–5.

[4] A. Krendzel, "LTE-A Mobile Relay Handling: Architecture Aspects," in *Wireless Conference (EW), Proceedings of the 2013 19th European*, April 2013, pp. 1–6.

[5] J. Laneman, D. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *Information Theory, IEEE Transactions on*, vol. 50, no. 12, pp. 3062–3080, Dec 2004.

[6] T. Wang, A. Cano, G. Giannakis, and J. Laneman, "High-Performance Cooperative Demodulation With Decode-and-Forward Relays," *Communications, IEEE Transactions on*, vol. 55, no. 7, pp. 1427–1438, July 2007.

[7] S. Ikki and M. Ahmed, "Performance analysis of adaptive decode-and-forward cooperative diversity networks with best-relay selection," *Communications, IEEE Transactions on*, vol. 58, no. 1, pp. 68–72, January 2010.

[8] O. Simeone, Y. Bar-Ness, and U. Spagnolini, "Stable Throughput of Cognitive Radios With and Without Relaying Capability," *Communications, IEEE Transactions on*, vol. 55, no. 12, pp. 2351–2360, Dec 2007.

[9] Q. Zhang, J. Jia, and J. Zhang, "Cooperative relay to improve diversity in cognitive radio networks," *Communications Magazine, IEEE*, vol. 47, no. 2, pp. 111–117, February 2009.

[10] B. Raghothaman, G. Sternberg, S. Kaur, R. Pragada, T. Deng, and K. Vanganuru, "System architecture for a cellular network with cooperative mobile relay," in *Vehicular Technology Conference (VTC Fall), 2011 IEEE*, Sept 2011, pp. 1–5.

[11] *3GPP TS 36.213, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures*, Release 8, v8.8.0, September 2009.

[12] A. Lozano and N. Jindal, "Transmit diversity vs. spatial multiplexing in modern MIMO systems," *Wireless Communications, IEEE Transactions on*, vol. 9, no. 1, pp. 186–197, January 2010.

[13] T. Sorensen, P. Mogensen, and F. Frederiksen, "Extension of the ITU channel models for wideband (OFDM) systems," in *Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd*, vol. 1, Sept 2005, pp. 392–396.

[14] *3GPP TS 36.101, Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) radio transmission and reception*, Release 8, v8.8.0, December 2009.

[15] *3GPP TS 36.211, Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation*, Release 8, v8.8.0, September 2009.

[16] *3GPP TS 36.212, Evolved Universal Terrestrial Radio Access (E-UTRA); Multiplexing and channel coding*, Release 8, v8.8.0, December 2009.

[17] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, no. 4-5, pp. 411–430, 2000.

[18] D. Godard, "Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems," *IEEE Transactions on Communications*, vol. 28, no. 11, pp. 1867–1875, Nov 1980.

[19] J. Treichler and B. Agee, "A new approach to multipath correction of constant modulus signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 31, no. 2, pp. 459–472, Apr 1983.

[20] S. Barbarossa and A. Scaglione, "Blind equalization using cost function matched to the signal constellation," in *Signals, Systems amp; Computers, 1997. Conference Record of the Thirty-First Asilomar Conference on*, vol. 1, Nov 1997, pp. 550–554 vol.1.

[21] F. Moazzami and A. Cole-Rhodes, "An adaptive blind equalizer with signal separation for a MIMO system transmitting QAM signals," in *Military Communications Conference, 2008. MILCOM 2008. IEEE*, Nov 2008, pp. 1–5.

[22] K. N. Oh and Y. O. Chin, "Modified constant modulus algorithm: blind equalization and carrier phase recovery algorithm," in *Communications, 1995. ICC '95 Seattle, 'Gateway to Globalization', 1995 IEEE International Conference on*, vol. 1, Jun 1995, pp. 498–502 vol.1.

[23] J.-C. Lin and L.-S. Lee, "A modified blind equalization technique based on a constant modulus algorithm," in *Communications, 1998. ICC 98. Conference Record. 1998 IEEE International Conference on*, vol. 1, Jun 1998, pp. 344–348 vol.1.

[24] H. Bölcskei, *Space-time wireless systems: from array processing to MIMO communications*. Cambridge University Press, 2006.

[25] Y. Zeng and T.-S. Ng, "A semi-blind channel estimation method for multiuser multiantenna OFDM systems," *IEEE Transactions on Signal Processing*, vol. 52, no. 5, pp. 1419–1429, May 2004.

[26] F. Wan, W. P. Zhu, and M. N. S. Swamy, "A Semiblind Channel Estimation Approach for MIMO-OFDM Systems," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 2821–2834, July 2008.

[27] D. Obradovic, N. Madhu, A. Szabo, and C. S. Wong, "Independent component analysis for semi-blind signal separation in MIMO mobile frequency selective communication channels," in *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, vol. 1, July 2004, p. 58.

[28] C. Mehlführer, M. Wrulich, J. C. Ikuno, D. Bosanska, and M. Rupp, "Simulating the Long Term Evolution Physical Layer," in *Proc. of the 17th European Signal Processing Conference (EUSIPCO 2009)*, Glasgow, Scotland, Aug. 2009. [Online]. Available: http://publik.tuwien.ac.at/files/PubDat_175708.pdf

[29] "MEX-files Guide." [Online]. Available: http://www.mathworks.com/help/matlab/call-mex-files-1.html

[30] "TU Wien's Institute of Telecommunications LTE simulator page." [Online]. Available: http://www.nt.tuwien.ac.at/ltesimulator

[31] Y. Zheng and C. Xiao, "Simulation models with correct statistical properties for Rayleigh fading channels," *Communications, IEEE Transactions on*, vol. 51, no. 6, pp. 920–928, June 2003.

[32] "Iterative Solutions Coded Modulation Library (ISCML)." [Online]. Available: http://www.iterativesolutions.com/

[33] C. Studer, M. Wenk, A. Burg, and H. Bolcskei, "Soft-Output Sphere Decoding: Performance and Implementation Aspects," in *Signals, Systems and Computers, 2006. ACSSC '06. Fortieth Asilomar Conference on*, Oct 2006, pp. 2071–2076.

[34] R. Wang and G. Giannakis, "Approaching MIMO channel capacity with reduced-complexity soft sphere decoding," in *Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE*, vol. 3, March 2004, pp. 1620–1625 Vol.3.

[35] J.-J. van de Beek, O. Edfors, M. Sandell, S. Wilson, and P. Ola Borjesson, "On channel estimation in OFDM systems," in *Vehicular Technology Conference, 1995 IEEE 45th*, vol. 2, Jul 1995, pp. 815–819 vol.2.

[36] J.-S. Kim, S.-H. Moon, and I. Lee, "A new reduced complexity ML detection scheme for MIMO systems," *Communications, IEEE Transactions on*, vol. 58, no. 4, pp. 1302–1310, April 2010.

[37] S. Schwarz, M. Wrulich, and M. Rupp, "Mutual information based calculation of the Precoding Matrix Indicator for 3GPP UMTS/LTE," in *Smart Antennas (WSA), 2010 International ITG Workshop on*, Feb 2010, pp. 52–58.