



**Sudan University of Science and Technology**

**College of Graduate Studies**



**Performance Evaluation of Channel Estimation and Detection Algorithms  
for 5G Massive MIMO System**

تقييم الأداء لخوارزميات تخمين وكشف القناة لنظم الدخول والخرج المتعددة الضخمة في الجيل  
الخامس

A Research Submitted In Partial fulfillment for the Requirements of the Degree of  
M.Sc. degree in Electronics Engineering (Communication)

**Prepared By:**

Mustafa Malek Adem Alhaj.

**Supervisor:**

**Dr. Fath Elrhman Ismael Khalifa Ahmed**

**MAR 2017**

## الآية

قال تعالى :

﴿ وَقُلِ اعْمَلُوا فَسَيَرَى اللَّهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ  
وَسَتُرَدُّونَ إِلَىٰ عَالَمِ الْغَيْبِ وَالشَّهَادَةِ فَيُنبِّئُكُمْ بِمَا كُنْتُمْ  
تَعْمَلُونَ ﴾

التوبة (105)

# DEDICATION

*To*

*Endless love*

*My mother*

*To*

*Max who teach me to be max*

*Our fathers*

*To*

*Our brothers and Sisters*

*To*

*Our teachers & our colleagues*

## AKNOWLEDGMENT

First, we need to thank fully our god (Allah) that without his blessing this work will not complete.

Then I thank my supervisor **Dr. Fath Elrhman Ismael** for his patience and countless hours and valuable efforts to guide and advise to complete the work in his fair way.

Lastly, I need to thank our teachers in department of electronic engineering for their efforts in helping and support.

## ABSTRACT

This thesis introduces full filled description for fifth-generation technology, here concentrated on massive MIMO technology in detailed at channel detection. However, channel illustrated carefully, described how to estimate and detect the channel. By using both algorithms least square and minimum mean square error for channel estimation while zero forcing and minimum mean square error for channel detection by analyzing and measure their performance using BER for MIMO 2x2, also it shows these algorithms on massive MIMO but it offers high BER and latency. In addition, simple algorithms used for equalizing the channel are Gauss-Jordan Elimination, Gaussian Elimination, RQ Decomposition and LU Decomposition. In which MATLAB simulation used to analyzed and applied mathematical models. After that measured the BER, delay for each algorithm and evaluate the capacity and throughput, by way, found that the Gaussian Elimination has better delay about 49% when RQ Decomposition about 95% while LU Decomposition about 98% compared by Gauss-Jordan Elimination. In addition, show their performance at capacity and throughput for various modulation and coding rate, while the deliverables average capacity about 10 M bit and affected by the situation of the channel, LU has the best performance than other.

## المستخلص

هذه الأطروحة تقدم وصفا كاملا للجيل الخامس ولقد تم التركيز علي تقنية الهوائيات المتعددة بالتفصيل للتحقق عن القناة .كما تم توضيح ووصف كيفية تقدير وكشف القناة باستخدام كل من خوارزميات المربع البسيط والحد الأدنى لمربع الخطأ نفسه لتخمين القناة، بينما الصفر القسري والحد الأدنى لمربع الخطأ نفسه للتحقق عن القناة من خلال تحليل وقياس أدائها باستخدام هوائتين عند الاستقبال وهوائتين عند الارسال، كما بين عمل هذه الخوارزميات على عدد هائل من الهوائيات وأعطت مقدار خطأ وتأخير كبير للقناة .هنا تستخدم خوارزميات بسيطة لتخمين والتحقق من القناة وهي جاوس جوردن للحذف، والحذف لجاوس، التحليل العكسي والتعامدي للمصفوفة، والتحليل الاعلى والادنى للمصفوفة،وهنا أولا تم تحليل النموذج الرياضي ومن ثم تطبيقها باستخدام برنامج المصفوفات المكتبية للمحاكاة بعد ذلك تم قياس الخطأ والتأخر لكل من الخوارزميات وقيست كل من السعة الكلية للبيانات والسعة الحقيقية، وجدت أن خوارزمية الحذف لجاوس أفضل بحوالي 49%. بينما خوارزمية التحليل العكسي والتعامدي للمصفوفة افضل بحوالي 95%، في حين خوارزمية التحليل الاعلى والادنى للمصفوفة بحوالي 98% مقارنة بخوارزمية جاوس جوردن للحذف .ويعرض ايضا الادائية للسعة الكلية والحقيقية لعدد مختلف من التضمين والترميز، في حين ان متوسط السعة الكلية تقدر بحوالي 10 ميقات ووجدت ان خوارزمية التحليل الاعلى والادنى للمصفوفة لقد أعطت افضل ادائية مقارنة ببقية الخوارزميات حيث له اقل تاخير.

# TABLE OF CONTENTS

CHAPTERS	TITLE	PAGE
	الإستهلال	ii
	<b>DEDICATION</b>	<b>lii</b>
	<b>ACKNOWLEDGEMENT</b>	<b>iv</b>
	<b>ABSTRACT</b>	<b>v</b>
	المستخلص	vi
	<b>TABLE OF CONTENTS</b>	<b>vii</b>
	<b>LIST OF TABLES</b>	<b>x</b>
	<b>LIST OF FIGURES</b>	<b>xi</b>
	<b>LIST OF ABBREVIATION</b>	<b>xiii</b>
	<b>LIST OF SYMBOLS</b>	<b>xv</b>
<b>Chapter 1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 Preface	2
	1.2 Problem Statement	3
	1.3 Proposed Solution	3
	1.4 Methodology	3
	1.5 Aim and Objectives	4
	1.3 Proposed Solution	3
	1.4 Methodology	3
	1.5 Aim and Objectives	4
	1.6 Research Outlines	4
<b>Chapter 2</b>	<b>LITERATURE REVIEW</b>	<b>6</b>
	2.1 Background	7
	2.1.1 Wireless Communication LTE	9
	2.1.1.1 Orthogonal Frequency Division Multiplexing	10
	2.1.1.2 OFDMA/SC-FDMA	10
	2.1.1.3 Generic Frame Structure	12
	2.1.1.4 Uplink Physical Channel	13

2.2.2	Fifth Generation Technologies	16
2.2.2.1	Massive-MIMO and 5g Cellular	17
2.2.2.2	Millimeter Wave (mm-Wave)	19
2.2.2.3	Ultra-Densification	20
2.2.2.4	Conventional Cellular Network Architecture	20
2.2.2.5	Distribution Architecture of Ultra-Dense Cellular Networks	22
2.1.1.1	Device-Centric Architectures	22
2.1.1.2	Smart Devices	25
2.1.1.3	Native Support for Machine-to-Machine Communication	25
2.2	Related Works	26
<b>Chapter 3</b>	<b>FIFTH GENERATION KERNEL ALGORITHMS</b>	<b>30</b>
3.1	Introduction	31
3.2	Channel Detection and Estimation Algorithms	32
3.2.1	Channel Estimation	33
3.2.1.1	LS channel estimation	33
3.2.1.2	MMSE channel estimation	33
3.3	Channel Detection	34
3.3.1	ZF detection	34
3.3.2	MMSE detection	34
3.4	Generate channel matrix from 8x8 to 256x256	35
3.5	Algorithms For Fifth Generation Kernel Algorithms	36
3.5.1	Gauss Jordan Elimination Flow Chart	36
3.5.2	Gaussian Elimination Flow Chart	37
3.5.2.1	Pivoting and Forward Elimination	38
3.5.2.2	Back Substitution	39
3.5.3	LU Decomposition Algorithm Flow Chart	40
3.5.4	RQ Decomposition Algorithm Flow Chart	41
3.6	Processing Time for Algorithms	42

3.7	Delay and capacity	43
<b>Chapter 4</b>	<b>RESULTS AND DISCUSSION</b>	<b>45</b>
4.1	Introduction	46
4.2	Channel Estimation and Detection	47
4.2.1	Channel Estimation for 2x2 MIMO	47
4.2.2	Channel Detection for 2x2 MIMO	48
4.2.3	Channel Estimation for Massive MIMO	49
4.2.4	Channel Detection for Massive MIMO	49
4.3	Processing Delay for Algorithms and BER	50
4.4	Capacity and Throughput through AMC	52
4.4.1	BPSK Modulation and code rate 2/3	53
4.4.2	QAM Modulation and code rate 3/4	55
4.4.3	16-QAM Modulation and code rate 5/6	57
4.4.4	64-QAM Modulation and code rate 7/8	59
<b>Chapter 5</b>	<b>CONCLUSION AND RECOMMENDATIONS</b>	<b>61</b>
5.1	CONCLUSION	62
5.2	RECOMMENDATIONS	62
<b>References</b>		<b>64</b>
<b>Appendix A</b>	Mathematical Representation for algorithms	68
<b>Appendix B</b>	Simulation using MATLAB for LS MMSE at channel estimation and ZF and MMSE for channel detection	73
<b>Appendix C</b>	Simulation for delay to modified algorithm and evaluate its performance	81

## LIST OF TABLES

<b>TABLE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
4.1	Simulation for 2x2 MIMO channel	46
4.2	Parameter Used for Massive MIMO Simulation	47
4.3	Comparison between GJE and Other algorithms	51
4.4	Comparison between GJE and Other algorithms in terms of capacity and throughput (M=1 CR=2/3)	54
4.5	Comparison between GJE and Other algorithms in terms of capacity and throughput (M=2 CR=3/4)	56
4.6	Comparison between GJE and Other algorithms in terms of capacity and throughput (M=4 CR=5/6)	58
4.7	Comparison between GJE and Other algorithms in terms of capacity and throughput (M=6 CR=7/8)	60

## LIST OF FIGURES

<b>FIGURE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
2.1	Illustration of the uplink PHY Layer Blocks in LTE	11
2.2	Generic Frame Architecture Type 1 for SC-FDMA in LTE	12
2.3	Generic Frame Architecture Type 1 for SC-FDMA in LTE	13
2.4	Physical Layer Uplink between UE and eNodeB	14
2.5	Massive-MIMO Services Provided to Number of Users by Employing 2048, 4096 and 8192 Antenna Arrays	18
2.6	Distribution ultra-dense cellular networks with deployment scenario and logical structure for single gateway	21
2.7	Flow Chart Illustration the Methodology for Massive MIMO Technology	29
3.1	Flow Chart for Gauss Jordan Algorithm	36
3.2	Flow Chart for Gaussian Elimination	37
3.3	Forward Elimination in Gaussian Elimination Algorithm	38
3.4	Forward Elimination in Gaussian Elimination Algorithm	39
3.5	Flow Chart for LU Decomposition	40
3.6	Flow Chart for RQ Decomposition	41
3.7	Flow Chart of Processing Time for Each Algorithm	42

4.1	Channel Estimation for LS and MMSE	48
4.2	channel detection using ZF and MMSE Algorithms	48
4.3	Massive MIMO for channel estimation based on LS and MMSE	49
4.4	Massive MIMO channel Detection algorithms based on ZF and MMSE	49
4.5	BER for Massive MIMO versus SNR	50
4.6	Processing delay for GE, GJE, LU and RQ in Massive MIMO	51
4.7	BER for GE,GJE,RQ and LU decomposition	52
4.8	Capacity for Algorithms through using BPSK and CR=2/3 versus number of Antenna	53
4.9	Throughput for Algorithms through using BPSK and CR=2/3 versus number of Antenna	54
4.10	Capacity for Algorithms through using QAM and CR=3/4 versus number of Antenna	55
4.11	Throughput for Algorithms through using QAM and CR=3/4 versus number of Antenna	56
4.12	Capacity for Algorithms through using 16-QAM and CR=5/6 versus number of Antenna	57
4.13	Throughput for Algorithms through using 16-QAM and CR=5/6 versus number of Antenna	58
4.14	Capacity for Algorithms through using 64-QAM and CR=7/8 versus number of Antenna	59
4.15	Throughput for Algorithms through using 64-QAM and CR=7/8 versus number of Antenna	60

## LIST OF ABBREVIATIONS

2G	The Second Generation of Mobile Telecommunications Technology
3G	The Third Generation of Mobile Telecommunications Technology
3GPP	The 3rd Generation Partnership Project
3GPP2	<i>3rd Generation Partnership Project 2</i>
4G	The Fourth Generation of Mobile Telecommunications Technology
5G	The Fifth Generation of Mobile Telecommunications Technology
8-PSK	<i>eight phase shift keying</i>
16-QAM	<i>Sixteen Quadrature Amplitude Modulation</i>
64QAM	<i>Sixty Four Quadrature Amplitude Modulation</i>
AMC	<i>Adaptive Modulation and Coding</i>
BER	<i>Bit Error Rate</i>
BS	<i>Base Station</i>
CA	<i>Carrier Aggregation</i>
COMP	Cooperative communications paradigms
CP	<i>Cyclic Prefix</i>
CQI	<i>Channel Quality Indicator</i>
CR	<i>Code Rate</i>
CRC	<i>Cyclic Redundancy Check</i>
CSI	<i>Channel State Information</i>
D2D	<i>Device to Device</i>
DFT	<i>Discrete Fourier Transferee</i>
DRMS	Demodulation Reference Signal
FDD	<i>Frequency Division Duplex</i>
GE	Gaussian Elimination
GJE	Gauss Jordan Elimination

HARQ-ACK	Hybrid Automatic Repeat Request with Acknowledge
IFFT	Inverse Fast Fourier Transform
ISI	Inter Symbol Interference
LTE	Long Term Evaluation
LTE-A	Long Term Evaluation - Advanced
LU	Lower Upper
M2M	Machine to Machine
MATLAB	Mathematical Laboratory
Mbps	Megabits per Second
MIMO	Multiple Input Multiple Output
mm	Millimeter Wave
MMSE	Minimum Mean Square Error
MU-MIMO	Multi User MIMO
OFDMA	Orthogonal Frequency Division Multiple Access
PAPR	Peak to Average Power Ratio
PHY	Physical
PUCCH	Physical Uplink Control Channel
PUSCH	Physical Uplink Share Channel
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RB	Resource Block
RN	Relay Node
RQ	Reverse Quadrature
SC	Sub Carrier
SC_FDM	Single Carrier Frequency Division Multiplexing
SIMD	Single Input Multiple Data
SNR	Signal to Noise Ratio
SRS	sounding reference signal
TDD	Time Division Duplex
UE	User Equipment
ZF	Zero Forcing

## LIST of SYMBOLS

Symbol	Illustration
$H_{est}$	Estimated Channel
$H_{LS}$	Channel Estimated for Least Square
$X^H$	Received Signal
$Y$	Transmitted Signal
$H_{MMSE}$	Channel Estimated for MMSE
$R_{h,hp}$	Autocorrelation Matrix of Channel Estimation
$\sigma_w^2$	Variance for Channel
$I$	Identity Matrix
$X_{ZF}$	Received Signal From Channel For Zero Forcing
$X_{MMSE}$	Received Signal From Channel For Minimum Mean Square Error
$\sigma_n^2$	Variance for Estimated Channel
$\sigma_{nx}^2$	Variance for Transmitted Signal
$A$	Channel matrix Array
$a_{i,j}$	The element corresponding to $i^{\text{th}}$ row and the $j^{\text{th}}$ column of $A$
$N$	Number of Array elements
BW	Bandwidth
$M$	Modulation order
$C$	Code Rate
BER	Bit Error Rate

**CHAPTER ONE**  
**INTRODUCTION**

# Chapter One

## Introduction

### 1.1 Preface

With a development of Communication system toward 2G in which utilize circuit switch, it developed to 3G for offer high speed and data rate beyond advanced to 4G with fulfilled application and enabling to use multimedia on the way to 5G by developing technologies in 4G LTE-Advance.

For the mobile operator, the cost becomes increasingly important, simultaneously with the rising user greater place demand on the mobile operator's networks. Future communication technologies need to reduce power consumption, decrease latency, increase performance, and increase computability of today different standards.

The Long Term Evolution (LTE) baseband system many techniques exploits, such as synchronization, channel coding, interleaving, demodulation, channel estimation, multiple input multiple output (MIMO) detection, and so on. Many redundancies introduce like Channel estimation for a multi-antenna receiver system; these redundancies lower the channel's utilization, require additional processing power, and increase latency. The conventional method to address these problems is to add pilot signals and decrease the length of the cyclic prefix (CP). In baseband processing, control, and data correlation by selecting appropriate algorithms and then optimizing these algorithms can be minimized.

For conventional MIMO channel estimated using least square and minimum mean square error, while in detection zero forcing and minimum mean square error used and compared to each other algorithms.

The conventional channel estimation and channel detection algorithms will be compared using massive MIMO and compare with the newest algorithms that we used in this thesis especially for channel detection as we will show in next chapters.

In massive MIMO there is a huge number of the antenna element they we need to estimate large channel matrix, where, the number of element rising from  $8 \times 8$  to  $256 \times 256$ . For that also mechanisms for the inverse matrix to evaluate, receiving signal is needed, and four algorithms are proposed Gaussian, LU decomposition, RQ decomposition and Gauss-Jordan elimination Algorithms to evaluate the latency in each one and show their performance.

## **1.2 Problem Statement**

The ultra-high latency and high computation complexity of massive MIMO matrices from 16 to 256 dimensions is the vital bottleneck to realizing latency for channel estimation and MIMO detection.

## **1.3 Proposed Solution**

To reduce the problem of high computational complexity that causes huge latency, four algorithms are supposed to represent and measure their performance.

## **1.4 Aim and Objectives**

The aim of this thesis project is performance evaluation of channel estimation algorithms in 5G kernel moving from general to the specific objective, we have:

- 1- Modify algorithms used for channel estimation and detection in 2x2 MIMO to be suitable with Massive MIMO.
- 2- Design mathematical model for each of Gaussian Elimination, LU Decomposition and RQ Decomposition algorithm.
- 3- Modification the Source code for Gauss- Jordan algorithm to be suitable with the three algorithms.
- 4- Compare between Gauss- Jordan Elimination and three algorithms.
- 5- Evaluate the performance and computational complexity of four algorithms using MATLAB simulation.

## **1.5 Methodology**

The thesis makes use of Matrix Laboratory (MATLAB) simulation platform to simulate the different size of the matrix from 8x8 to 256x256 applied in MATLAB. Firstly, least square and minimum mean square error for channel estimation while zero forcing and minimum mean square error for channel detection and these algorithms give very high BER and large latency for Massive MIMO. However, the solution for channel estimation algorithms is not considered. Though, concentrated on channel detection. New four algorithms applied for channel detection are Gaussian, LU Decomposition, RQ Decomposition and Gauss-Jordan. Nevertheless, each algorithm is represented in detailed and modified the mathematical model of

gauss Jordan elimination continuously to other algorithms. By way source, the code also modified as found in Appendix A, B, C and shown the latency happen at four algorithms for chose best one. The delay effect on bandwidth and this appear directly on capacity and throughput, our main goal is to reach less delay to give better performance for channel situation.

## **1.6 Research Outlines**

Chapter One provides short Introduction; discuss Problem statement, proposed solution, and Objectives. While, Chapter Two review of the Channel Estimation, Matrix Inversion, and 5th generation technologies. In addition to Chapter Three explain the Fifth Generation Kernel Algorithms, and explain Simulating overall system Using MATLAB layout and discusses four scheme. However, Chapter Four include Results and Discussions. Finally, Chapter Five contain Conclusions, Recommendations.

**CHAPTER TWO**  
**LITERATURE REVIEW**

# **Chapter Two**

## **Literature Review**

### **2.1 Background**

At these days the communication play important role in our life and it represent is the progressive thing as we say it have annoying to solve all problem for enhancing our real life and go faster and easier, we must solve and go through these obstacles to avoid it then go further and used as good as possible.

Research on next-generation 5G wireless systems, several unprecedented it, which aims to resolve technical requirements and challenges, has attracted rising attention from both academia and industry in the past few years. More than 5 billion devices demand wireless connections that operate voice, data, and other applications in today's wireless networks [1].

For transmitting data from one node to another, there are multiple difficulties faces. Additionally how to receive data at the destination as it sends from source to destination. However, we need to receive data with low latency, BER, and high capacity. The communication system sends data from transmitter to receiver as it sent and with less delay, to deal with this problem and by what method to reduce both BER and delay time.

Multiple antenna technologies like Multiple-Input, Multiple-Output (MIMO) and beamforming will thus play an important role in defining 5G system architectures [2], in particular for millimeter wave frequencies. Multi-User MIMO (MU-MIMO) offers increased multiplexing gains and improves

spectral efficiency. Even though it has been included in the 3GPP LTE-Advanced standard, its full possible is yet to be recognized [3].

Very good communication system must be designed to transmission data from source to destination. These communication systems contain all component beginning from source coding, coding, modulation, suitable multiple access, and antenna, it will flow through the channel. While in another side at a receiver, beginning with receive antenna, multiple access, demodulation, decoding, source decoding and then to end user.

In this thesis, we focus on channel estimation and detection part. How to estimate the channel and how to equalize this channel upon estimation. Firstly we estimate the channel using two algorithms least square and maximum mean square error after that detect the received signal also using two algorithms zero forcing and maximum mean square error under channel using  $2 \times 2$  MIMO. In 4G LTE and LTE-A using MIMO from  $2 \times 2$  to  $8 \times 8$  the main goal of this thesis how to estimate channel response and equalize the channel to detect the signal. While in 5G we need to enhance the techniques in 4G and these techniques are Ultra-densification, device-centric architectures, millimeter wave (mm-Wave), Massive MIMO, smart devices, and native support for machine-to-machine (M2M) communication. In This thesis, we concentrate on Massive MIMO use large numbers of antenna starting from  $16 \times 16$  to  $256 \times 256$  [4].

Here, we estimated the channel and how to inverse it. Using four algorithms and then compare them. We assume that all communication system is complete the data is transmitted and the channel is estimated and depend on the estimated channel Matrix we calculate the inverse of channel using four algorithms Gauss- Jordan Elimination, LU Decomposition, RQ

Decomposition and Gaussian Elimination and then we evaluate the delay in the system using  $8 \times 8$  to  $256 \times 256$  [5]. In next chapter, we explain each of these algorithms and here we introduce a brief description of LTE and fifth generation technology as will show in next section.

### **2.1.1 Wireless Communication LTE**

LTE-A is a 4th generation mobile telecommunication technology. LTE-A was finalized by the third Generation Partnership Project (3GPP) in March 2011. LTE-A is not a completely new technology, rather it is an enhancement to LTE. The main objective of LTE-A is to increase the peak data rate to 1 Gbps on the downlink and 500 Mbps on the uplink, improve spectral efficiency from a maximum of 16 bps/Hz in Release8 to 30 bps/Hz in Release10, increase the number of concurrently active subscribers, and improve performance at cell edges [6]. Many technologies applied in LTE continue to be used in LTE-A, such as orthogonal frequency division multiplexing (OFDM), OFDMA, MIMO, and SC-FDMA. The aims of LTE are to ensure the continued competitiveness of 3G systems for the future and to offer high user data rates and low latency.

The main new technologies introduced in LTE-A are carrier aggregation (CA), enhanced the use of multiple antenna techniques, and relay nodes (RN). Because this thesis focuses only on physical layer transmission, the enhanced MIMO technique is the only one of these techniques considered in this thesis. Detailed information about CA and RN can be found in [7] and [8].

### **2.1.1.1 Orthogonal Frequency Division Multiplexing**

Orthogonal frequency division multiplexing (OFDM) is a well-known method of encoding digital data on multiple carrier frequencies. OFDM systems spilled the available bandwidth into many narrower sub-carriers. Data is transmitted as parallel streams over these sub-carriers. Each sub-carrier is modulated with varying levels of modulation schemes, such as Quadrature Phase Shift Keying (QPSK), Quadrature Amplitude Modulation (QAM), and 64-state QAM (64-QAM). The main merits of OFDM are low implementation complexity; good tolerance for inter-symbol interference (ISI) induced by multipath, and high spectral efficiency.

### **2.1.1.2 OFDMA/SC-FDMA**

LTE/LTE-A employs OFDMA and SC-FDMA as the multiplexing scheme for the downlink and uplink respectively. The requirements of LTE uplink and downlink differ in several ways. Since power consumption is a key consideration for User Equipment (UE), i.e., terminals. Because of OFDM's high PAPR and related loss of efficiency, an alternative to OFDM was desirable for the LTE uplink.

SC-FDMA is a suitable scheme for the LTE uplink. The basic transmitter and receiver architecture of SC-FDMA is quite similar to OFDMA, and SC-FDMA provides the same degree of multipath protection. Figure 2-1 depicts the basic SC-FDMA and OFDMA signal processing chains of the transmitter and receiver. In this figure, S/P stands for serial to parallel conversion, while P/S stands for parallel to serial conversion.

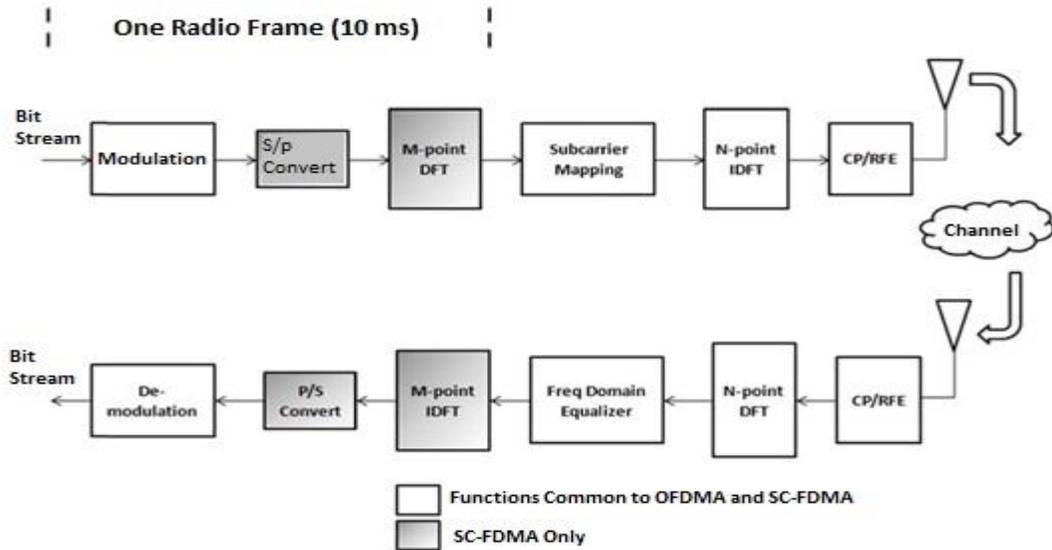


Figure 2-1: Illustration of the uplink PHY Layer Blocks in LTE [4]

As it can be seen figure 2-1, the OFDMA and SC-FDMA chain have a highly similar functional structure. In SC-FDMA. The subcarrier mapping (SC Mapping), N-point Inverse fast Fourier transform (IFFT), and cyclic prefix adding (Add CP) are the same as OFDMA. The difference is that, for the data streams, before they mapped to the subcarrier, Discrete Fourier Transform (DFT) is performed to decrease the PAPR. This DFT can also consider pre-coding. LTE-A physical layer protocols are mainly described in the following 3GPP standards:

TS 36.201 General description of Long Term Evaluation (LTE) physical layer [9].

TS 36.211 Physical channels and modulation [10].

TS 36.212 Multiplexing and channels coding [11].

TS 36.201 is the general description documentation, the rest is specific document. As this thesis only views physical (PHY) layer transmission, the relevant content of TS 36.211 is a description in the

following sub-section. Although LTE-A is an improvement of LTE, there seems to be a little improvement from LTE to LTE-A at the PHY layer. At will introduced the essential technique of LTE/LTE-A used in PHY layer, specifically OFDM, OFDMA, SC-FDM, and MIMO. The LTE PHY downlink and uplink evolved are quite different because of the various structure and capabilities of the uplink channel estimation and the MIMO detection algorithms, an overview of LTE uplink PHY layer processing flow between the UE and eNodeB will be impersonated, hence the LTE downlink flow will be ignored.

### 2.1.1.3 Generic Frame Structure

One element shared by the LTE downlink and uplink is the generic frame structure. There are two types of frame structure defined in LTE specifications (depending on the duplexing scheme). Figure 2-2 shows the generic type 1 frame structure of LTE.

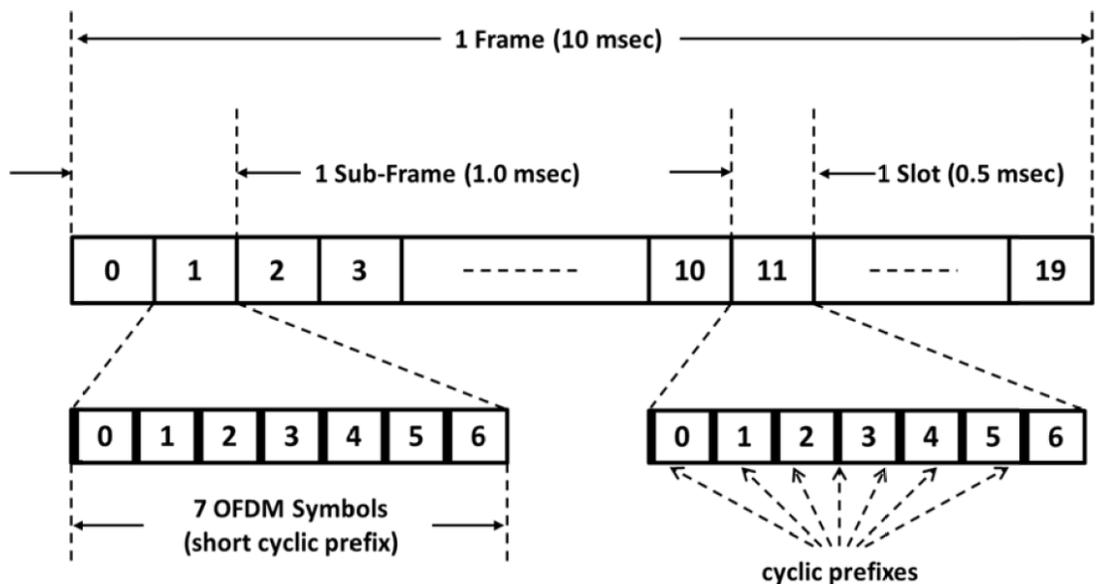


Figure 2-2: Generic Frame Architecture Type 1 for SC-FDMA in LTE [4]

The duration of one radio frame is 10 ms. there are 20 slots in a frame. These slots are numbered from 0 to 19. The duration of one slot is 0.5 ms. Sub-frames in a frame. There are 7 or 6 OFDM Symbols in each slot depending on which kind of CP (normal or extended) is used. The CP is inserted in front of every symbol.

Figure 2.3 present the frame structure type 2. Each radio frame is 10 ms duration. A frame consists of two half-frames of 5 ms each. Each half frame is comprised of five sub-frames of length 1 ms. In common with type 1, the length of a sub-frame is also 1 ms. The difference between type 1 and type 2 is that type 2 includes three different sub-frame: up-link transmission sub-frame, downlink subframe, and special sub-frame.

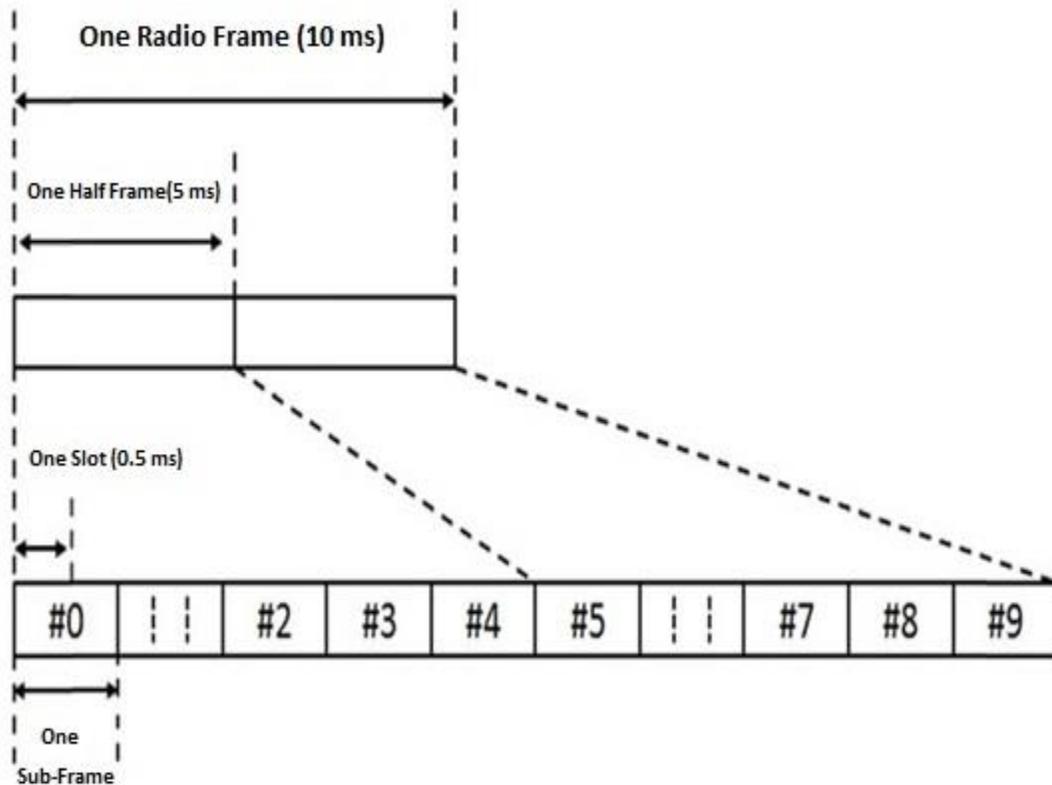


Figure 2-3: Generic Frame Type 2 for SC-FDMA in LTE [4]

### 2.1.1.4 Uplink Physical Channel

These channels used to transmit the user's data and control messages as illustrated in figure 2-4. There are two types of physical channels defined for uplink: Physical Uplink Shared Channel (PUSCH) and Physical Uplink Control Channel (PUCCH). This thesis only considers PUSCH, as a purpose of PUSCH is to transmit user data. The modulation scheme used by PUSCH are QPSK 16-state QAM (16-QAM) or 64-QAM depending on channel conditions.

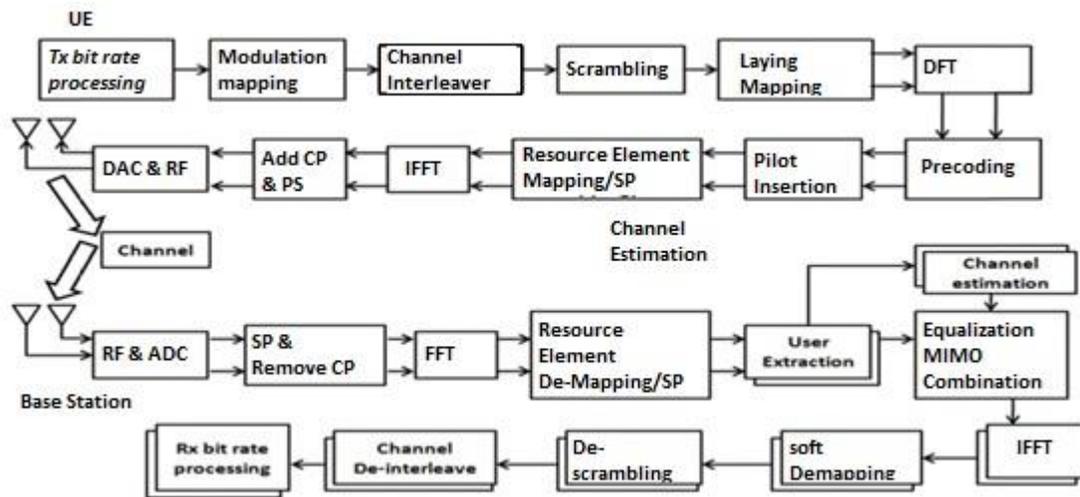


Figure 2-4: Physical Layer Uplink between UE and eNodeB [4]

#### Procedures of User Equipment UE:

- Transmitter (TX) bit rate processing his stage include transport block cyclic redundancy check (CRC) attachment.
- Scrambling a number of bits are scrambled with a UE-specific scrambling sequence prior to modulation. The main reason for scrambling is to decrease the interference from adjacent cells.
- Modulation mapper this stage maps the binary bits into complex value symbols. The modulation scheme is QPSK, 16-QAM, and 64-QAM.

- Layer mapping the complex modulation symbols for each of the code word to be transmitted are mapped onto one, two, three or four PHY layers.
- DFT performing a DFT converts the signal from the time domain to the frequency domain.
- Pre-coding maps the complex-valued modulation symbols from the layers to multiple antennas.
- Pilot Insertion pilot symbols generated and inserted into the complex values modulation symbols on each antenna port.
- Resource element mapping N-point IFFT are performed to convert the signal from the frequency domain to the time domain after resource element mapping.
- ADD CP & PS attach CP into every symbol and then perform parallel to serial conversion.
- Digital/Analog convert digital signal to analog signal and then transmit on the appropriate radio frequency.

**Procedures of eNodeB:**

- Analog /Digital the base station receives an analog RF signal and then converts this analog signal to a digital signal.
- Serial /Parallel converter & remove CP perform serial to parallel conversion and then remove CP.
- Fast Fourier Transform (FFT) N-point FFTs are performed to convert the signal from time domain to frequency domain.
- Reference signal/Data signal separation the reference signal and data signal are separated. The reference signal it used to perform channel

estimation. Every user's symbol data will be extracted from the different subcarriers according to their physical resource block configurations.

- Channel estimation Based on the pilot symbols extracted from the frame, estimated the channel matrix  $H$  during the period the channel state information (CSI) is valid
- Remove pilot remove the pilot symbol from the modulation symbol frame.
- Resource element mapping it maps the complex-valued modulated symbol frame into blocks.
- IFFT perform  $M$ -point IFFTs to convert the data from the frequency domain to the time domain.
- Soft slicer converts the received SC-FDMA symbols into soft bits according to the modulation scheme employed.
- Descrambler/Channel De-interleave the inverse stage of scrambling uses de-interleaved for rank indication bits, Hybrid Automatic Repeat Request ACK (HARQ-ACK) information bits, and PUSCH/Channel Quality Indication (CQI) multiplexing bits.
- Receiver (Rx) bit rate processing this stage is the inverse of TX bit rate processing. It involves code block de-concatenation, rate matching, turbo decoding, code block CRC removal, code block de segmentation and transport block CRC removal [3].

### **2.1.2 Fifth Generation Technologies**

To meet  $1000\times$  wireless traffic volume increment in the next decade, the fifth generation (5G) cellular network is growing a hot research topic in telecommunication industries and academics. Firstly, the massive multiple-

input multi-output (MIMO) technology was proposed to improve the spectrum efficiency of 5G mobile communication systems. Secondly, the millimeter wave communications were introduced to reach the transmission bandwidth for 5G mobile communication systems. Moreover, the small cell idea has been issued to raise the throughput and save the energy consumption in cellular situations. To satisfy the seamless coverage, a larger number of small cells have to be densely expanded for 5G cellular networks. Consequently, the ultra-dense cellular network is emerging as one of core characteristics for 5G cellular networks. Nevertheless, the study of ultra-dense cellular networks is still in an opening stage. Some basic studies, such as the network architecture and cellular densification limits require being extra investigated for future 5G cellular networks.

#### **2.1.2.1 MASSIVE-MIMO AND 5G CELLULAR**

In massive MIMO present research, challenges involve estimation of criticality of coherent channels. Propagation impairments for massive MIMO in modern context could also be hypothetically determined on an experimental basis for channel orthogonally. This could be further implemented based on deeper costs in the context of hardware power dissipation in each of the antennas.

Viewing present scenario 5G has many advantages over 4G:

- i) Non- bulky in space
- ii) Directive antennas
- iii) Coherent angle spread of the propagation

There are a limited number of antennas in MIMO. Manipulating single-user that is right for the current standard of cellular communication.

However, massive MIMO is not limited if TDD (Time Division Duplex) is combined for enabling channel characterization.

Figure 2-5 show relative scenario of massive MIMO's application which governs the multiple antennas spread in which a small town or university campus or city could be utilized.

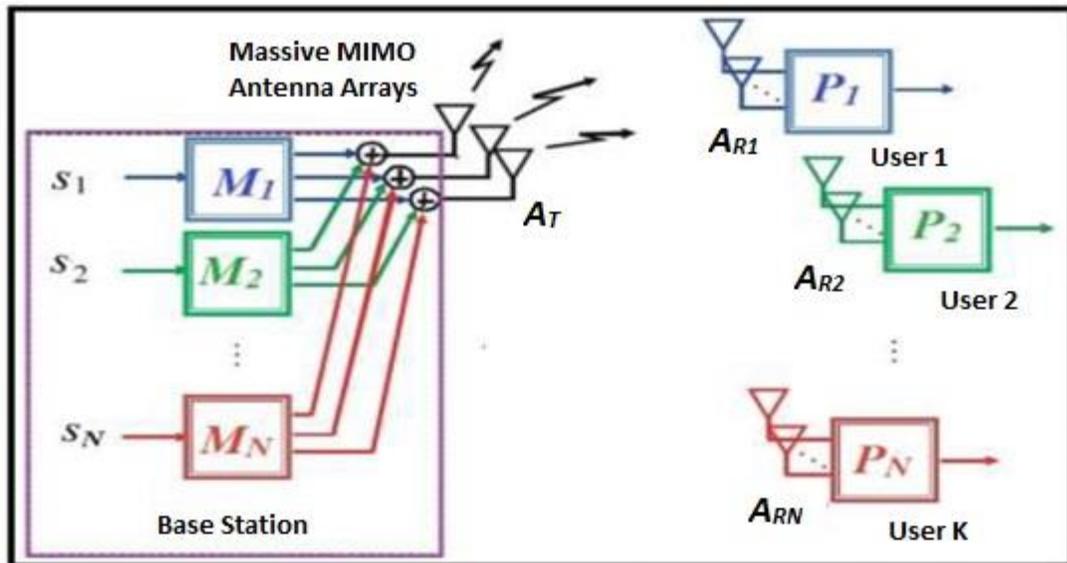


Figure 2-5: Massive-MIMO Services Provided to Number of Users by Employing 2048, 4096 and 8192 Antenna Arrays [2]

Massive MIMO proposals for this model by applying a huge number of antennas to multiplex information signals in communication systems for several machines by utilizing devices-to-devices link (D2D) on each time-frequency access schemes (TDD/FDD), focus must be on optimizing energy radiated towards the directions indicated during minimizing inter- and inter-cell interference. Figure 2-5 clearly highlights the comparison of cellular services provided in terms of data rate gain for different antenna arrays. For massive MIMO application in a 4X4 baseline for subscribers in a single cell cluster. The 8192 number of antennas is expanded by massive MIMO systems thereby enhancing the user efficiency. Services to users with 2048

antennas in simple MIMO schemes were classically adopted where both 5 and 50 percentile of full efficiency is reached. The MIMO systems with 5096 are common with optimal service provided. Thus expanding the number of an array in the antenna with advanced signal processing tools could a make huge information transmission.

### **2.1.2.2 Millimeter Wave (mm-Wave)**

The frequencies in the range of 600 MHz to 1600 MHz are currently in use for cellular. This little range can hardly be utilized for future generation wireless access systems by reframing the system. Higher spectrums in the ranges of GHz and THz could be extended by utilizing techniques in cognitive radio. The highly potential field is utilized by wavelength in millimeter range and hence the term millimeter wave is in practice. Today different cellular and wireless firms want a progressive expansion in capacity emerging aims, which has to be carried in becoming years beyond the fourth generation of wireless standards in Long Term Evolution (4GLTE).

Around 2020, the cellular networks would face a very high speech and data traffic and thereby higher capacity requirements for data rate and bandwidth. For wireless future, wireless generation of 5G mobile data rates must increase up to numerous gigabits per second (Gbps) range, which can only be processed by using the millimeter wave spectrum steerable antennas. This would help 5G cellular backhaul communications in addition to the integration of worldwide accuracy in wireless services. Since Massive MIMO is a spatial processing technique which would have orthogonal polarization and beam-forming adaptation, this smaller millimeter wavelength is proper frequencies. The highly populated geographical regions could be covered by 4G+ to 5G technologies by setting backhaul link using

massive MIMO in the case of greater bandwidth challenges. Cost per base station will significantly reduce due to innovative architectures of cooperative MIMO, thereby decreasing interference relays and servicing base stations.

The wireless operators would decrease cellular coverage area to Pico and femtocells for generating spatial reuse. Since cellular networks would meet gigantic traffic (data and speech) over next ten to twenty years, a huge challenge would be to harmonize frequency bands by ITU to GHz and THz. This will improve the low cost of service and roaming. The mobile network operators are planning to fulfill future requirements, by combining of to share spectrum for this solution that would be beneficial beyond 2020 [2].

### **2.1.2.3 Ultra-Densification**

With the development of massive MIMO antenna and millimeter wave communication technologies for 5G mobile communication systems, a large number of small cells will be expanded to form 5G ultra dense cellular networks. Therefore, the first challenge is how to design the architecture of 5G ultra-dense cellular networks. In this section, the distribution architecture of ultra-dense cellular network with single and multiply gateways is proposed for further evaluating in following Sections.

### **2.1.2.4 Conventional Cellular Network Architecture**

The conventional cellular network architecture is a type of tree network architecture, where the BS managers in the core network control every macrocell BS and all backhaul traffic is delivered to the core network by the given gateway. In order to maintain microcells deployment, e.g., femtocells, Pico-cells, and hotspots deployment, a hybrid architecture is shown for conventional cellular networks with microcells deployment. In this

hybrid network architecture, the microcell network is also configured as a kind of tree network architecture; wherever microcell BS managers in the core network and the backhaul traffic of macrocell BSs control every macrocell, BS is forwarded to the core network by the broadband Internet or fiber links. The coverage of microcells is overlapped with the coverage of macro-cells. Matched with macrocell BSs, macrocell BSs can afford the high-speed wireless transmission in indoor and hotspot scenarios. Both of the macrocell BS and the microcell BS can independently transmit the user data and the management data to associated users. Users can handover in macrocells and microcells according to their requirements. Moreover, macrocell and microcell managers in the core network control the handover process. In figure 2-6 this network architecture, the microcell network is a complement to the conventional macrocell network to perform the high-speed wireless transmission in partial regions, e.g., indoor and hotspot scenarios.

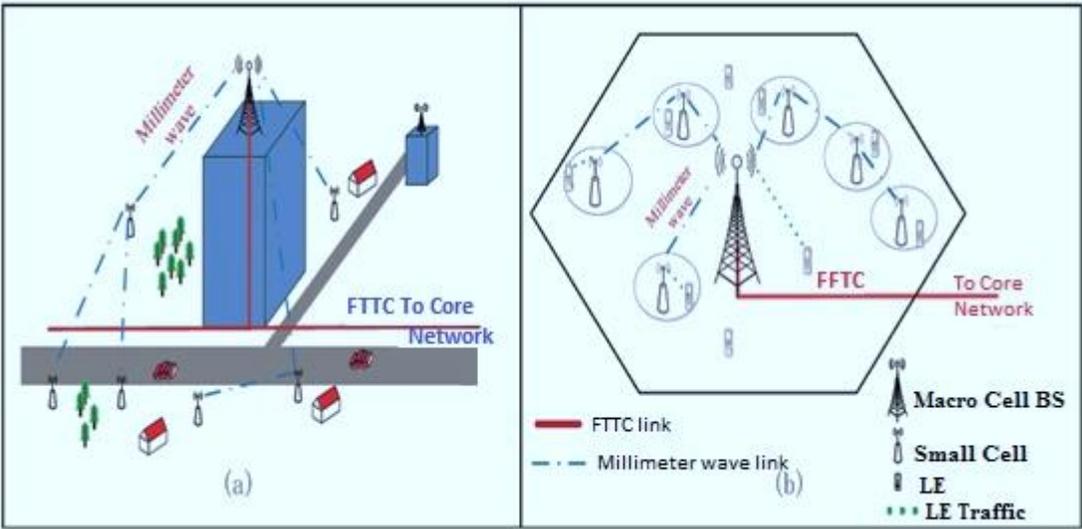


Figure 2-6: Distribution ultra-dense cellular networks with deployment scenario and logical structure for single gateway [12]

### **2.1.2.5 Distribution Architecture of Ultra-Dense Cellular Networks**

Prompted by the massive MIMO antenna and the Millimeter Wave Communication Technologies, the densification deployment of small cells is emerging into 5G cellular networks. Nevertheless, it is challenging to forward the backhaul traffic of every small cell BS by the broadband Internet or the fiber link considering the cost and geography deployment challenges in urban environments. Moreover, the small cell BS usually cannot directly transmit the wireless backhaul traffic to the given gateway since small cell BSs adopting the millimeter wave technology reduce the wireless transmission distance. In this case, the wireless backhaul traffic has to be relayed to the assigned gateway by multi-hop links. As an outcome, the distribution network architecture is a solution for 5G ultra-dense cellular networks. In 5G ultra-dense cellular scenarios, to solve the mobile user frequently handover problem in small cells, the macrocell BS is configured only to transmit the management data for controlling the user handover in small cells and the small cell BS takes charge of the user data transmission. Therefore, the small cell network is not a complement to the macrocell network. 5G ultra-dense cellular networks are jointly composed by small cells and macro cells. Based on the backhaul gateway configuration [12].

### **2.1.2.6 Device-Centric Architectures**

Cellular designs have historically relied on the obvious role of ‘cells’ as fundamental units within the radio access network. Under such a design guess, a device takes service by establishing a downlink and an uplink connection, carrying both control and data traffic, with the base station commanding the cell where the device is placed. Over the last few years, different aims have been guiding to a disruption of this cell-centric structure:

- The base-station density is increasing rapidly, driven by the rise of heterogeneous networks. Whereas heterogeneous networks were already standardized in 4G, the architecture was not natively designed to assist them. Network densification could need some major changes in 5G. The deployment of base stations with vastly different transmit powers and coverage areas, for example, calls for a decoupling of downlink and uplink in a way that allows for the corresponding information to flow through different sets of nodes.
- The need for additional spectrum will inevitably lead to the coexistence of frequency bands with radically varying propagation characteristics within the same system. In this context, [13] proposes the concept of a ‘phantom cell’ where the data and control planes are separated: high-power nodes at microwave frequencies send the control information while low-power nodes at mm-Wave frequencies convey the payload data.
- A new idea termed centralized baseband related to the concept of Cloud Radio Access Networks is developing, where virtualization drives to a decoupling between a node and the hardware allocated to manage the processing associated with this node. Hardware resources in a pool, for instance, could be dynamically allocated to various nodes depending on metrics determined by the network operator.
- Emerging service classes could want a complete redefinition of the architecture. Current works are looking at architectural designs ranging from centralization or partial centralization (e.g., via aggregators) to full distribution (e.g., via compressed sensing and/or multi-hop).
- Cooperative communications paradigms such as COMP or relaying, which despite coming short of their initial hype are nonetheless beneficial, could

need a redefinition of the functions of the different nodes. In the context of relaying, for example, recent developments in wireless network coding propose transmission principles that would allow recovering some of the losses associated with half-duplex relays. Furthermore, recent research points to the plausibility of full duplex nodes for short-range communication in a not-so-distant future.

- The use of smarter devices could affect the radio access network. In particular, both D2D and smart caching call for an architectural redefinition where the center of gravity moves from the network core to the periphery (devices, local wireless proxies, relays).

Based on these aims, our vision is that the cell-centric architecture should develop into a device-centric one: a given device (human or machine) should be able to communicate by replacing multiple information flows through several possible sets of heterogeneous nodes. In other words, the set of network nodes providing connectivity to a given device and the functions of these nodes in a particular communication session should be tailored to that particular device and session. Under this vision, the concepts of uplink/downlink and control/data channel should be rethought. While the need for a disruptive change in architectural design appears clear, major research efforts are still needed to transform the resulting vision into a coherent and realistic proposition. Since the history of innovations indicates that architectural changes are often the operators of major technological discontinuities, we conclude that the aims above might have a major impact on the development of 5G.

### **2.1.2.7 Smart Devices**

Earlier generations of cellular systems were built on the design assumption of having complete control at the infrastructure side. In this section, we discuss some of the possibilities that can be unleashed by enabling the devices to play a more active role and, consequently, how 5G's design should account for an improvement in device smartness. We concentrate on three different examples of technologies that could be incorporated into smarter devices, namely D2D, local caching, and advanced interference rejection.

### **2.1.2.8 Native Support for Machine-to-Machine Communication**

Wireless communication is becoming a commodity, just similar to electricity or water. This Commoditization, in turn, is giving growth to a large class of emerging services with new types of demands. We point to a few characteristic such requirements, each represented by a typical service:

- A massive number of connected devices. Whereas modern systems typically operate with, at most, a few hundred devices per base station, some M2M services might want over 10000 connected devices. Examples include metering, sensors, smart grid components, and other enablers of services targeting wide area coverage.
- Very high link reliability. Systems provided with critical control, safety, or production, have been ruled by wireline connectivity largely because wireless links did not offer the same degree of confidence. As these systems transition from wireline to wireless, it becomes necessary for the wireless link to be reliably operational virtually all the time.

- Low latency and real-time operation. This can be an even more stringent requirement than the ones above, as it demands that data be transferred reliably within a given time interval. A typical example is a Vehicle-to-X connectivity, whereby traffic safety can be improved through the timely delivery of critical messages (e.g., alert and control) [14].

## 2.2 Related Works

LTE is a 3.9G technology. According to the standard, the peak data rate of LTE is from 100 to 326.4 Mbps over the downlink and 50 to 86.4 Mbps over the uplink. LTE uses orthogonal frequency division multiple access (OFDMA) and single carrier frequency division multiple access (SC-FDMA) in downlink and uplink sequentially [15]. OFDM has two defects: large peak-to-average power ratio (PAPR) and high sensitivity to carrier frequency errors. [16] The main advantage of SC-FDMA is its low PAPR [17].

The potential technologies that could use in 5G are ultra-densification, device-centric architectures, millimeter wave (mm-Wave), massive MIMO, smart devices, and native support for machine-to-machine (M2M) communication [18] [19]. Single Input Multiple Data (SIMD) instruction processing is one of the newest forms of parallel processing in Flynn's taxonomy. The basic idea of SIMD is to apply the same instruction sequence simultaneously to a huge number of discrete data streams [20].

The first viewpoint is the comparison of key technologies in baseband processing. There have been many papers discussed the channel estimation and MIMO detection at LTE/LTE-A uplink. For channel estimation, many references discuss how to optimize channel estimation method to gain good performance. [21] Moreover, the used method proposed to discuss further

how to optimize it when a different number of resource blocks are allocated. Several papers evaluated the different algorithms in different channel model such as [22] investigate algorithms in flat Rayleigh fading. The author in [23] Investigated the channel estimation for LTE uplink when the traveling speed of the UE is high. For MIMO detection. The author in [24] study two low-complexity detection schemes based on MMSE for MIMO systems. [25] Evaluated the performance of different detection algorithms over Rayleigh wireless channel. Because channel estimation and MIMO detection are two advanced procedures in LTE-A Uplink. All of these references only focused on one scheme of channel estimation/MIMO detection. Meanwhile, they only research channel estimation/MIMO detection algorithm for multi-antennas  $2 \times 2$  or maximum  $4 \times 4$  MIMO system. They did not consider the future massive MIMO-system.

We have to find and a suitable algorithm for matrix inverse for these systems. This algorithm should be suitable for SIMD architecture. Following a great deal of reading of references and investigation, several conventional algorithms were selected that could be used to compute the matrix inverse for the complex matrix. The conventional methods used to perform matrix inverse are Gauss-Jordan Elimination [26], Gaussian Elimination [2], LU Decomposition [27], and QR Decomposition [28].

The research work conducted by Xin, [29] is alike to this project. Both focus on investigating and analyzing key technologies (Channel estimation and MIMO detection) in large-Scale MIMO. Our general orientation is to develop wireless system's performance in large-Scale MIMO. The chosen algorithm is Gaussian-Jordan Elimination with the sizes of matrices ranging from  $8 \times 8$  to  $256 \times 256$ . The references [30] [31] explored matrix computation

on matrices larger than  $512 \times 512$  using LU decomposition and Gauss-Jordan-Floyd-Warshall method respectively.

In [32] [33], the design and implementation of a parallel algorithm utilize multi-core task-level parallelism, another form of coarse-grained parallelism.

In order to analyze this processing are compare two conventional algorithms' performance and complexity for channel estimation and MIMO detection. The key features, which affects the algorithms' speed, it identified as the need for "massive complex matrix inversion". A parallel coding scheme it suggested to implement a matrix inversion kernel algorithm on a single instruction multiple data stream (SIMD) vector processor.

The figure 2-7 explain the methodology for last paper and as shown in blue color beginning with the physical layer in uplink for PUSCH channel and then work on massive MIMO technology and focus on the algorithm for equalizing the channel to extract transmitted signal.

The fifth generation going to increasing the frequency and trying to decrease cell area and this enable to usage huge number of antenna especially in massive MIMO can arrive at 4096 antennas. its need very high computational complexity for that we look for what method to reduce the delay happen in computation thereby compare between two methods using only Gauss-Jordan Elimination, the work with SIMD and Normal calculation founded that the SIMD has less delay than normal calculation[4].

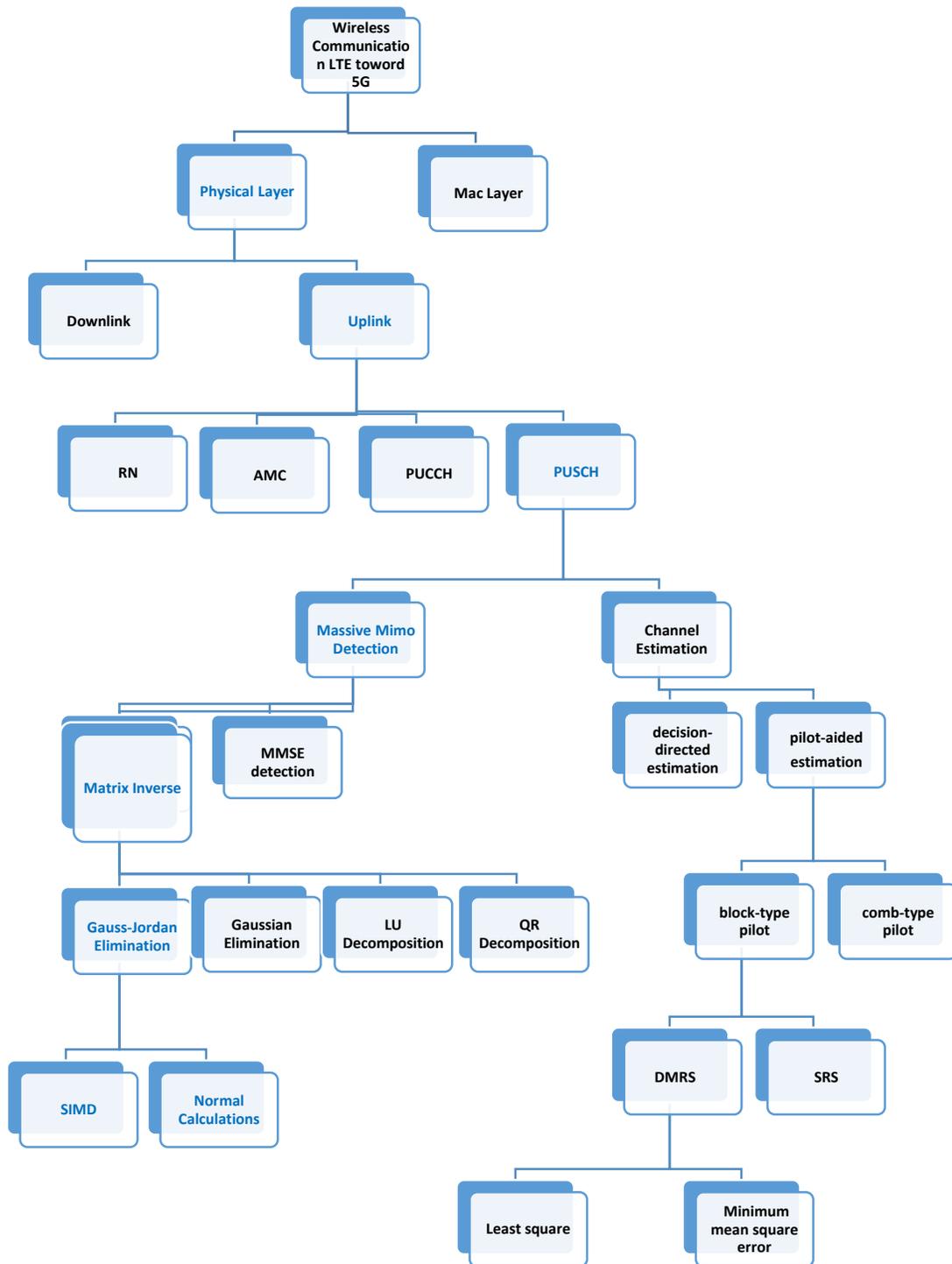


Figure 2-7: Flow Chart Illustration the Methodology for Massive MIMO Technology [4].

**CHAPTER THREE**

**FIFTH GENERATION KERNEL ALGORITHMS**

# Chapter Three

## Fifth Generation Kernel Algorithms

### 3.1 Introduction

The communication system is very sensitive to transmission delay for that we must be careful by system design. As we know, next generation going toward increasing the bandwidth and frequency while trying to decrease the area of cells.

By improving fourth generation technology toward the fifth generation such as centric device architecture, smart antenna, millimeter wave, device to device, ultra-densification and Massive MIMO, here we focus on how to develop this technology, especially in Massive MIMO.

At the beginning algorithms, for channel estimation and detection work on  $2 \times 2$  MIMO, it developed to be suitable with the Massive MIMO. For channel estimation, two algorithms are used least mean square and minimum mean square error and upgraded to be suitable with Massive MIMO and here we applied  $4 \times 4$ ,  $8 \times 8$ ,  $16 \times 16$  and  $32 \times 32$  antenna as will show in next chapter. At channel, detection two algorithms are used zero forcing and minimum mean square error, also modified to be suitable with Massive MIMO, also applied to different numbers of the antenna.

In channel, estimation when the number of antennas increases the estimated channel will be very bad and the BER will higher for that we just assume the channel randomly depend on a number of antennas.

While at channel detection the BER and latency is very high in both algorithms LS and MMSE when a huge number of the antenna are applied,

we preferred to use another algorithm for channel detection. In this thesis, four algorithms are used to detect and equalized the channel beginning by Gauss-Jordan Elimination, Gaussian Elimination, LU decomposition and RQ decomposition.

Firstly, compare the delay happen for each algorithm and show the error for each one, after that we evaluate the effect of each algorithm on bandwidth and capacity, finally show the capacity when they have different modulation and coding rate depend on channel estimation. At this stage we use three combinations of adaptive modulation and coding AMC.

### **3.2 Channel Estimation and Detection Algorithms**

In wireless communication, the signal will suffer from many phenomena such as different path losses, multipath, fading and so on. For that, we must carefully choose suitable channel estimation algorithms to equalize data correctly.

Firstly we need to estimate the channel and then equalized the data depend on estimated channel and received signal, from this we understand that they need algorithms to perform channel estimation beyond doing channel detection.

We will introduce two channel estimation algorithms and applied for simple and Massive MIMO, using both least square error (LS) and minimum mean square error (MMSE), while for channel detection zero forcing (ZF) and minimum mean square error (MMSE) used for simple and massive MIMO.

### 3.2.1 Channel Estimation

In this section, we present two typical algorithms for channel estimation that are used in LTE uplink, and we describe modified algorithms based on these two algorithms for the LTE- A uplink in Appendix B.

#### 3.2.1.1 LS channel estimation

The simplest algorithm for channel estimation. LS is characterized by low complexity

$$H_{est} = H_{LS} = (X^H X)^{-1} X^H Y = X^{-1} Y \quad 3.1$$

Where  $H_{est}$  it is the estimated channel matrix of complex numbers,  $X$  is a frequency domain transmitted pilot signal where  $Y$  is a frequency domain received pilot signal. Moreover, The LS channel estimation algorithm in the frequency domain.

The LS algorithm estimates the Channel Impulse Response CIR based on the received and transmitted symbols. The algorithm ignores the noise; thus, the performance of the LS estimator is not good.

#### 3.2.1.2 MMSE channel estimation

A better algorithm as it considers the effect of noise. This algorithm is widely used in practice. However, the major drawback of the MMSE algorithm is its high computational complexity

$$H_{est} = H_{MMSE} = R_{h,hp} (R_{hp,hp} + \sigma_w^2 I)^{-1} H_{LS} \quad 3.2$$

Where  $R_{hp,hp}$  it is the autocorrelation matrix of the channel at the pilot symbol positions;  $R_{h,hp}$  is the cross-correlation matrix between the channel at the data symbol positions and the channel at the pilot symbol position and  $I$  is the identity matrix and  $\sigma_w$  variance for effected channel noise.

### 3.3 Channel Detection

In MIMO detection, the detector calculates an estimated of the transmitted signal based on the received signal and the estimated channel matrix. After estimating and calculating the channel matrix, the system recovers the transmitted signal from the receiver signal as an output of the detector. In the following section, we described ZF and MMSE detection algorithms:

#### 3.3.1 ZF detection

Is the simplest algorithm and has the lowest computational complexity. This detector begins by multiplying the received symbol vector by the channel matrix.

$$X_{ZF} = (H^H H)^{-1} H^H Y \quad 3.3$$

A disadvantage of ZF detection is that it suffers from sudden noise enhancement; hence, the performance of ZF degrades without considering the noise.

#### 3.3.2 MMSE detection

Addresses the issues of ZF and MMSE tries to minimize the mean square error the minimum mean square error equalization matrix it represented as follows:

$$X_{MMSE} = G_{MMSE} = (H^H H + \sigma_n^2 / \sigma_{nx}^2 I)^{-1} H^H Y \quad 3.4$$

Where  $\sigma$  is the variance,  $H$  is estimated matrix channel,  $Y$  is received signal. In comparison with ZF detection, MMSE detection considers the noise variance and decreases noise enhancement, while the computational complexity of MMSE detection is greater than that of ZF detection.

### 3.4 Generate Channel Matrix from 8x8 to 256x256

For both channel estimation and detection as it mentions Massive MIMO is applied when the number of Antenna 8, 16 and 32 as shown in

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{1N} \\ a_{21} & a_{22} & a_{23} & a_{2N} \\ a_{31} & a_{32} & a_{33} & a_{3N} \\ a_{41} & a_{42} & a_{43} & a_{MN} \end{bmatrix} \quad 3.5$$

Where A is the channel matrix usually M equal N represent the number of element, N =8, 16, 32... 256.

In channel estimation, plotted estimated channel compared to ideal channel versus SNR based on LS and MMSE algorithms, for each number of antenna beginning from 2x2 until 16x16 to showing error happen in each one. Also in a channel, detection ZF and MMSE algorithms plotted BER happen from each algorithm versus SNR for each number of channel beginning from 2x2 until 16x16 and the error is very high for that if another algorithm used may give less delay.

The main goal of this research is to introduce another detection algorithm to be applied. Four algorithms are applied as illustrated in next section 3.5 and now we compute the delay happen for each algorithm using MATLAB, after that the effect of delay on bandwidth and effect on channel capacity at different Modulation and coding is tested.

MATLAB simulation used to compute delay happen for each algorithm using TIC TOC function in which compute the processing time delay for an instruction executed by algorithms.

### 3.5 Algorithms For Fifth Generation Kernel Algorithms

Four algorithms are used in this thesis, in this section will define the four algorithms with flow charts beginning with Gauss-Jordan, Gaussian Elimination, LU decomposition and RQ Decomposition.

#### 3.5.1 Gauss Jordan Elimination Flow Chart

Figure 3-1 is the Gauss-Jordan Algorithm, at the beginning we have estimated channel and received signal, by applying equalization for both we extract transmitted signal as show in below figure, more detailed found in Appendix A and C.

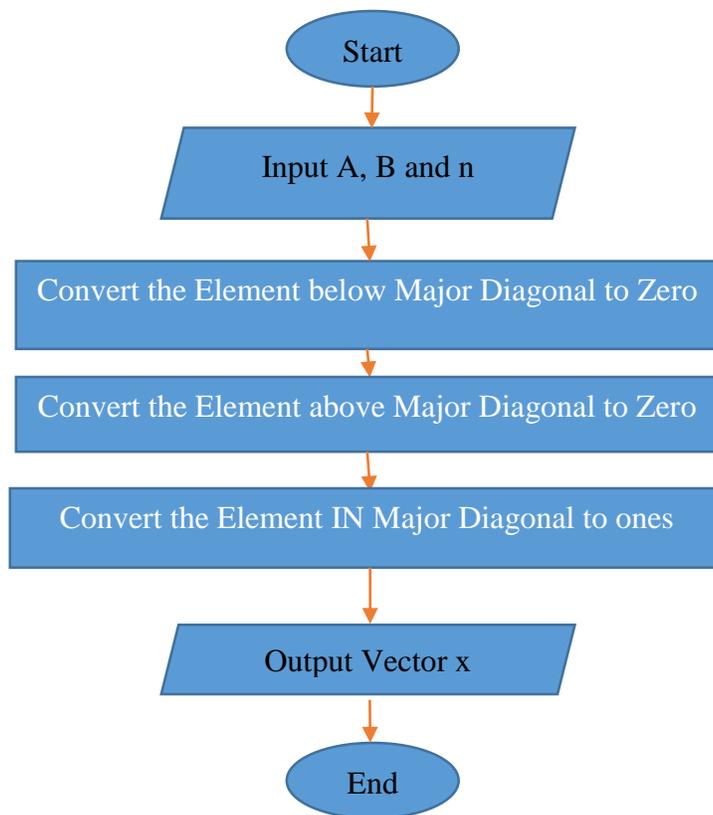


Figure 3-1: Flow Chart for Gauss Jordan Algorithm

### 3.5.2 Gaussian Elimination Flow Chart

Figure 3-2 show the Gauss Elimination, also here we have estimated channel and received signal, by applying equalization for both to extract transmitted signal as shown in below figure, more detailed are found in Appendix A and C.

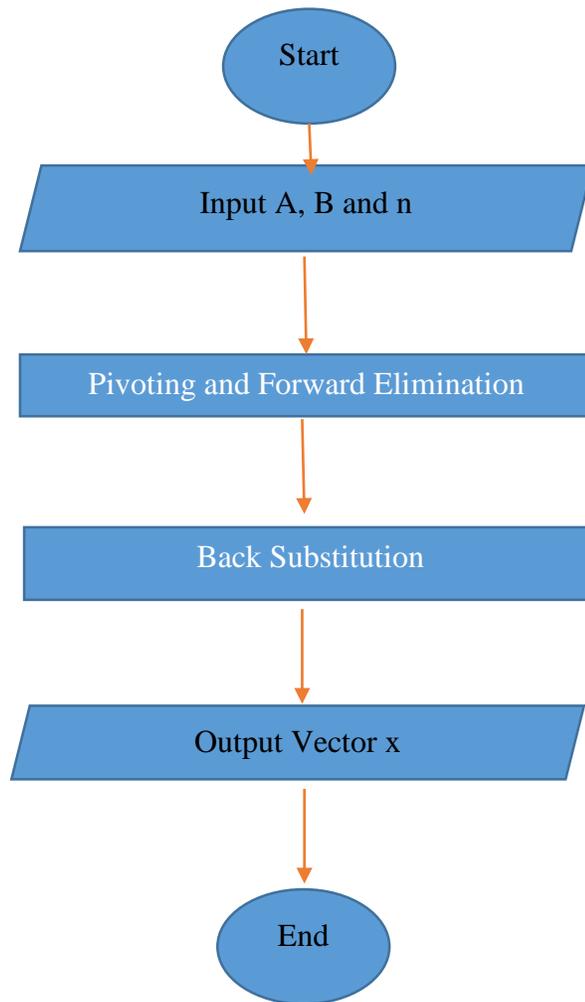


Figure 3-2: Flow Chart for Gaussian Elimination

### 3.5.2.1 Pivoting and Forward Elimination

Figure 3-3 show the forward elimination, how it will do in Gaussian elimination methods as the show they have three loops in this part, more detailed for mathematical at Appendix A, when source code at Appendix C.

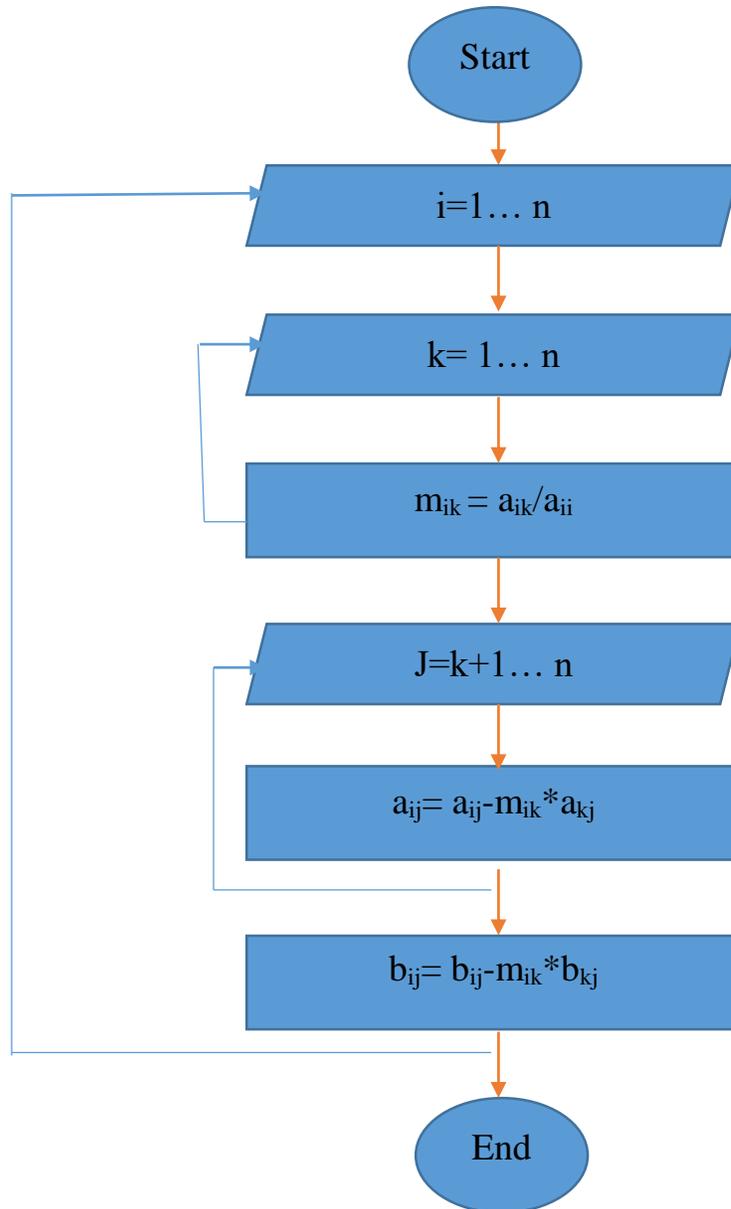


Figure 3-3: Forward Elimination in Gaussian Elimination Algorithm

### 3.5.2.2 Back Substitution

Figure 3-4 show the back substitution, how it will do in Gaussian elimination methods as a show they have two loops in this part, more detailed for mathematical at Appendix A, when source code at Appendix C.

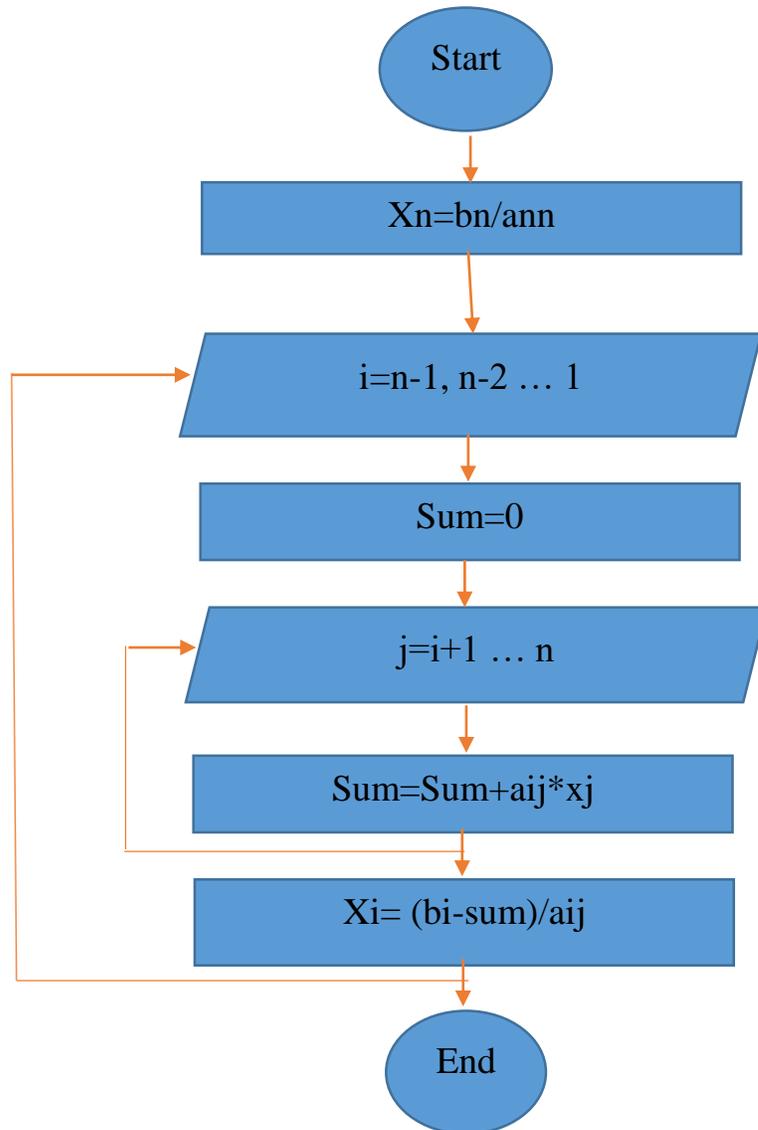


Figure 3-4: Forward Elimination in Gaussian Elimination Algorithm

### 3.5.3 LU Decomposition Algorithm Flow Chart

Figure 3-5 is flow chart for LU Decomposition, in which we estimated the channel and received signal, by applying equalization for both we extract-transmitted signal more detailed for mathematical at Appendix A, when source code at Appendix C.

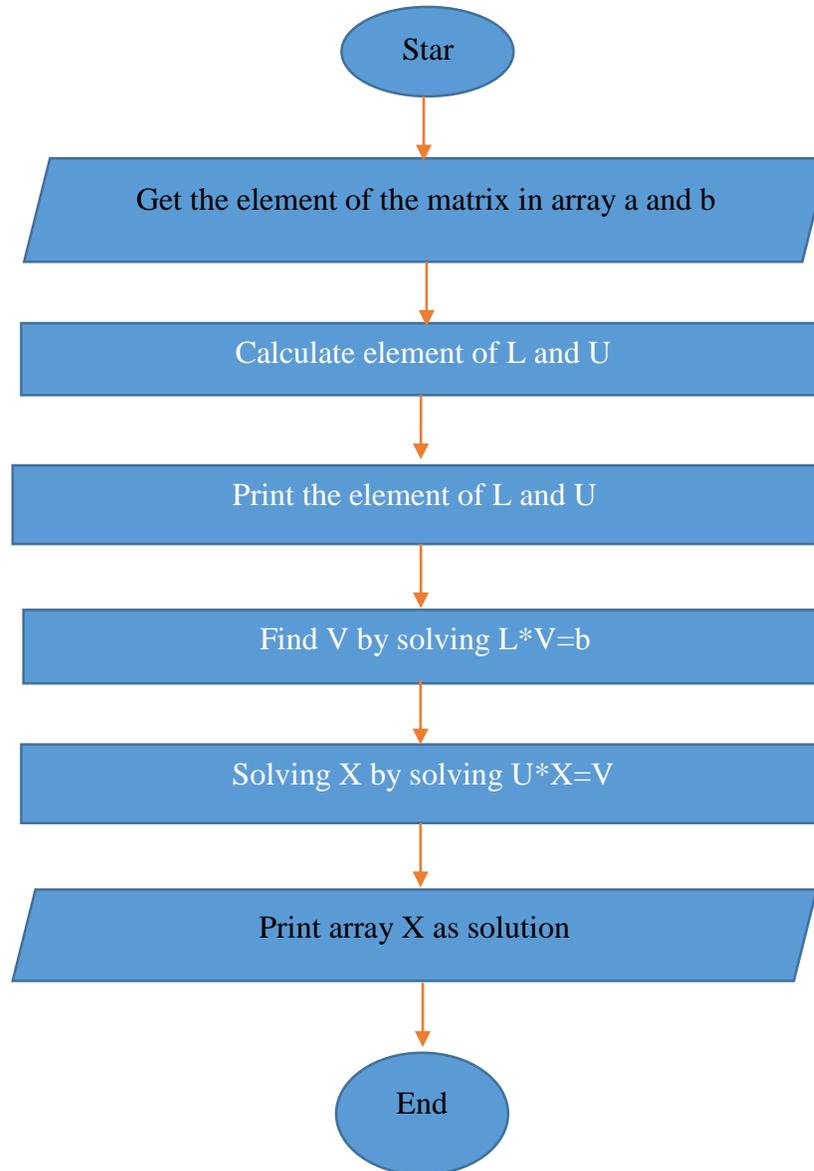


Figure 3-5: Flow Chart for LU Decomposition

### 3.5.4 RQ Decomposition Algorithm Flow Chart

Figure 3-6 illustrate the RQ Decomposition, to found estimated channel and received signal, after applying equalization for both we extract-transmitted signal as shown in below figure, more detailed for mathematical at Appendix A, when source code at Appendix C.

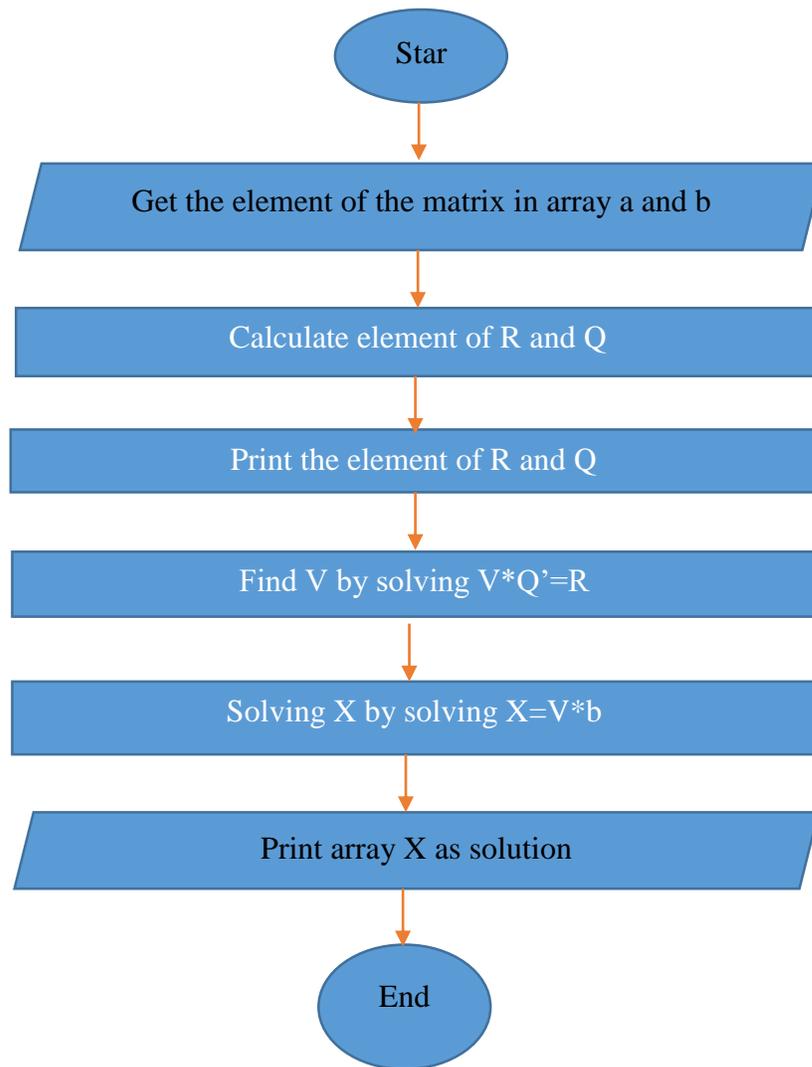


Figure 3-6: Flow Chart for RQ Decomposition

### 3.6 Processing Time for Algorithms

In this section, the delay happen for each algorithm it computed by using TIC TOC function by using MATLAB Simulation as illustrated in Figure 3-7 and Appendix C as source code .

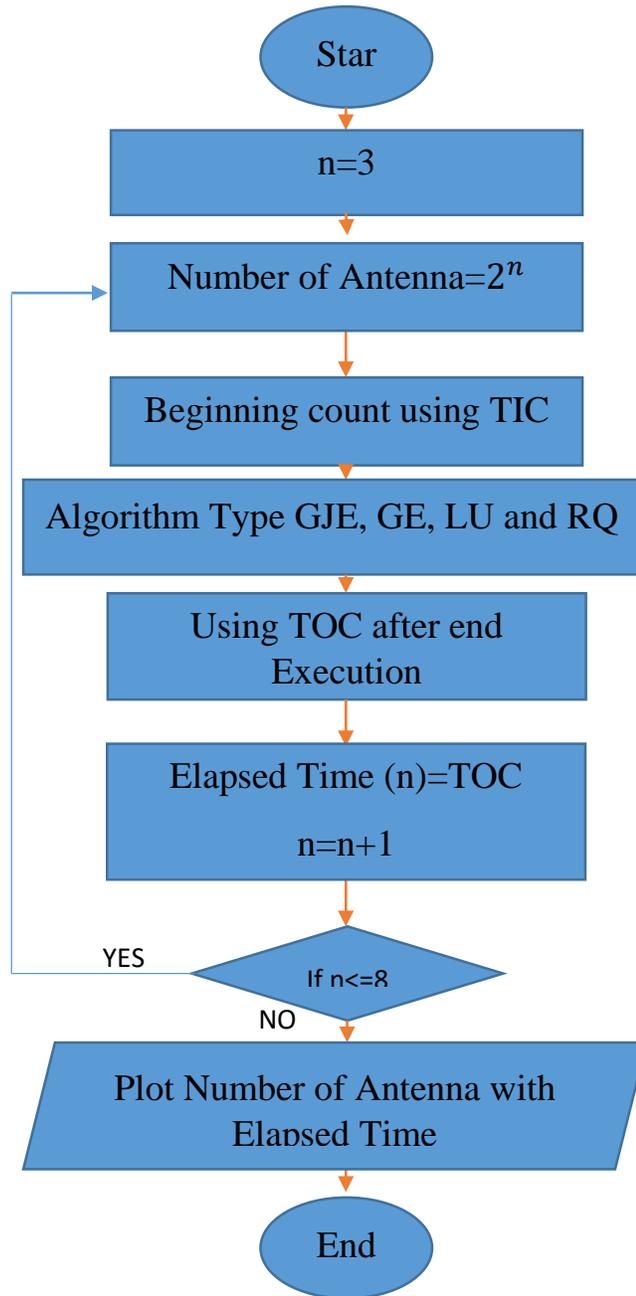


Figure 3-7: Flow Chart of Processing Time for Each Algorithm

The function TIC reset the timer, while TOC is beginning count after the reset. However, these functions used to evaluate the processing time happen in each algorithm for different number of antenna and the result it put in variable Elapsed Time.

The number of antenna with Elapsed Time it plotted as will showing in next chapter, to know the optimal algorithm. In addition, there are delay time adding from computer processor and MATLAB software it's very small compared to delay caused by massive MIMO can be negligible.

### **3.7 Delay and capacity**

The communication system is very sensitive to delay it reduce the bandwidth of whole system. In Equation (3.5) is the definition of bandwidth it is response of channel when data it applied in transmitter through communication system. When Equation (3.6,3.7) is the delay consider as vital problem that can face in our channel it will make the response very slow and going to decrease bandwidth in Equation (3.8) , while reducing the bandwidth also drive to decrease the capacity and throughput as in Equation (3.9, 3.10):

However, the bandwidth is represented as a response of system; at another way, the inverse of bandwidth give us the response of a system. In which the fading channel effect directly on the response of communication system beyond it go to minimize bandwidth.

At fifth generation when Massive MIMO applied there are a huge number of antennas, these antennas cause large latency for channel estimation and detection, in this way it will effect on the response of a system

and decrease the bandwidth resulting in giving us lowest capacity and throughput.

The latency includes channel delay, bit duration, and processing delay, this thesis focuses on processing delay for different algorithms to the comparison, measures their performance, analyzing to choose the lowest latency.

For that, appropriate algorithms are chosen in giving us lowest latency for channel detection trying to reduce the effect of algorithms on bandwidth, in order to give maximum capacity and throughput for dedicated bandwidth.

The capacity and throughput determined depend on bandwidth (BW), modulation order (M) and code rate (C). Take in our consideration the effect of algorithms delay on bandwidth. .

Capacity is the maximum data can carry through a channel. While throughput defined as valid received data from the transmitter, here calculated using a simple method by multiplying the capacity by BER to take out all invalid data and throughput represent the valid data, this evaluation shown in Appendix c.

**CHAPTER FOUR**  
**RESULTS AND DISCUSSION**

# Chapter Four

## Results and Discussion

### 4.1 Introduction

MATLAB simulation used to simulate the algorithms and equation founded in previous chapter. This chapter has four sections; the first section is an introduction and shows the initial parameter used in this simulation table 4-1 and table 4-2. While in the second section show the channel estimation and detection algorithms and their effect on massive MIMO. The third section applied four algorithms gauss Jordan elimination, Gaussian elimination, LU decomposition and RQ decomposition is shown the delay happen for each algorithm and BER in each one, finally, they show the effect of the algorithms delay on capacity and throughput for different modulation technique.

Table 4-1: Simulation for 2x2 MIMO channel

<b>Parameters</b>	<b>Value(s)</b>
Bandwidth(MHz)	20
IFFT/FFT size	2048
OFDM CP	Normal
Channel	Rayleigh fading channel
Channel estimation algorithms	LS, MMSE
Number of resource blocks	10
Number of RB per SC	12
Number of base station antennas	2
Number of UE antennas	2

These algorithms modified to be suitable with 4x4, 8x8 and 16x16 for both channel detection and estimation algorithms.

While table 4-2 represent the parameters that are used at the fifth generation for Massive MIMO kernel algorithms.

Table 4-2: Parameter Used for Massive MIMO Simulation [36]

<b>Parameter</b>	<b>Value</b>
Carrie Frequency	70GHz
Channel Delay	13us
Bandwidth	1GHz
Sample Duration Delay	0.67ns

## **4.2 Channel Estimation and Detection**

In this section, firstly show the channel estimation algorithm LS and MMSE for 2x2 MIMO by computing the error caused by estimated channel compared to an ideal channel, then show the BER caused by detection algorithms for both ZF and MMSE.

Thirdly apply massive MIMO in both Channel estimation and detection algorithm beginning from 4x4, 8x8, 16x16 and show the error happen in channel estimation beyond the BER in channel detection.

### **4.2.1 Channel Estimation for 2x2 MIMO**

As can be seen, Figure 4-1 compare between LS and MMSE and found that MMSE has less BER than LS. Because MMSE based on the variance of the channel, while in another side LS it is not considered.

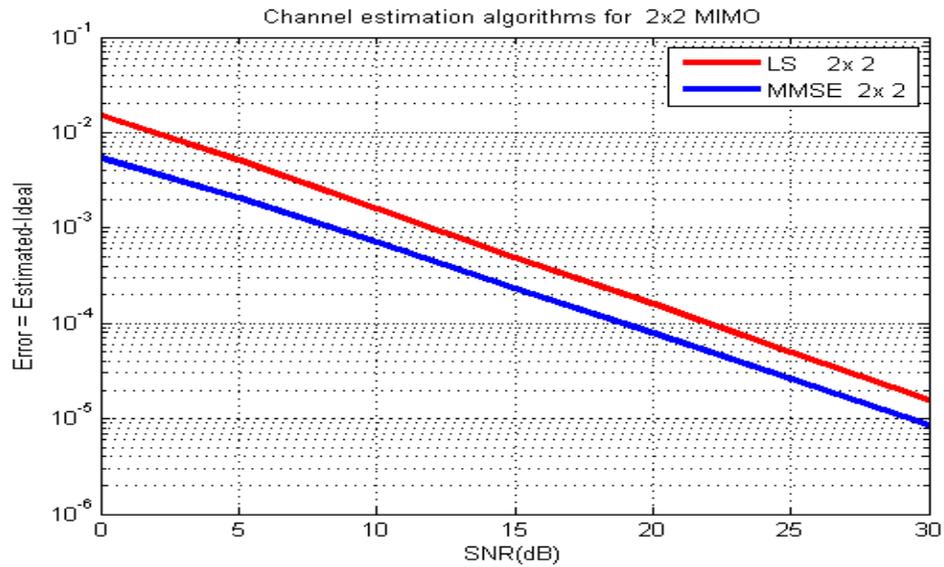


Figure 4-1: Channel Estimation for LS and MMSE

#### 4.2.2 Channel Detection for 2x2 MIMO

For Detection, two algorithms are used ZF and MMSE for comparison as show in Figure 4-2 two algorithms compared for channel detection, plotted BER for each versus SNR, and founded that MMSE has less BER than ZF result from using variance for channel.

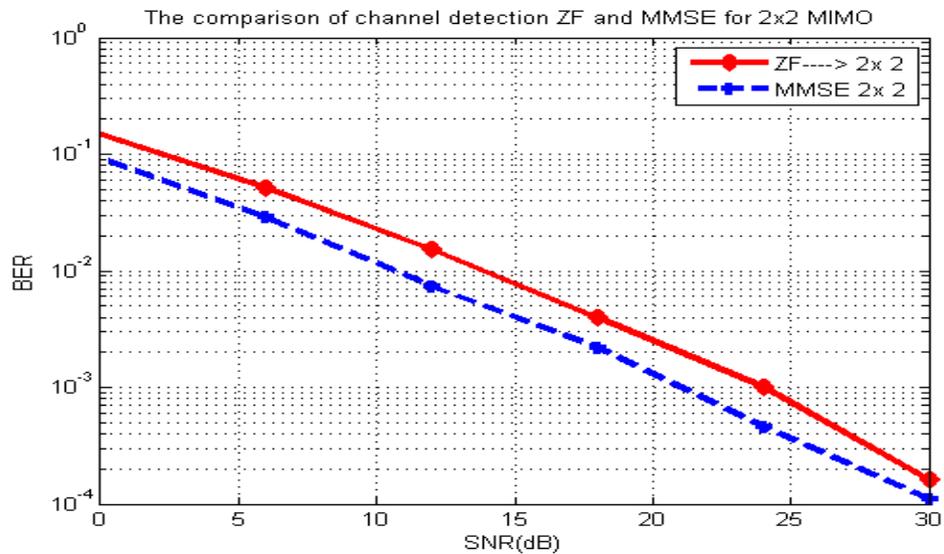


Figure 4-2: channel detection using ZF and MMSE Algorithms

### 4.2.3 Channel Estimation for Massive MIMO

The Error is very higher when massive MIMO applied for these algorithms as shown in Figure 4-3 this part is not considering in this thesis. In addition, this mean fail in estimation channel

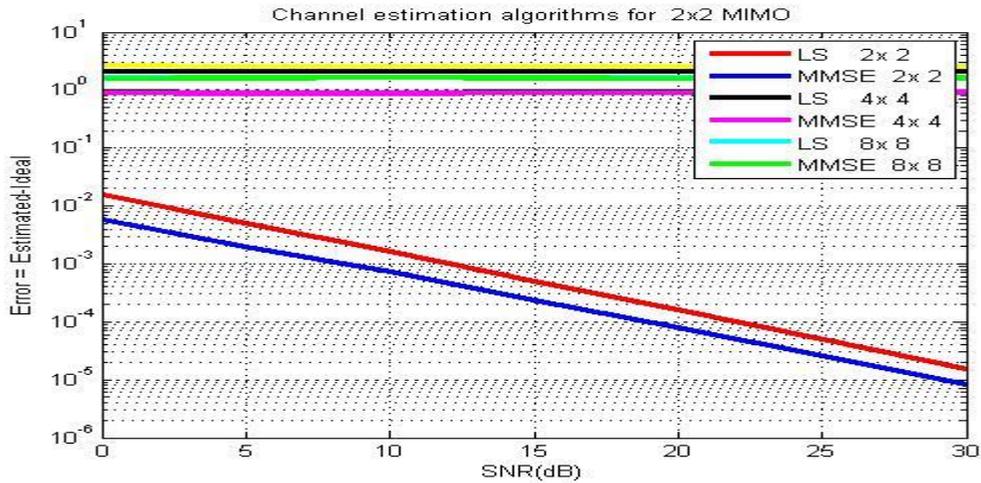


Figure 4-3: Massive MIMO for channel estimation based on LS and MMSE

### 4.2.4 Channel Detection for Massive MIMO

Same algorithms are used at channel detection for Massive MIMO here we used 2x2, 4x4, 8x8 and 16x16 as shown in figure 4-4 found that the MMSE has less BER than ZF SNR values,

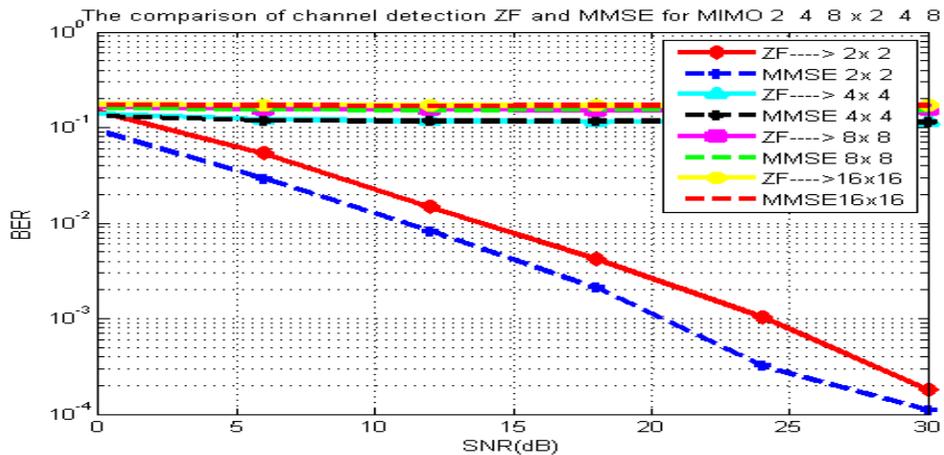


Figure 4-4: Massive MIMO channel Detection algorithms based on ZF and MMSE

They illustrated the other value of BER for massive MIMO and show the BER and found that the BER is very good when  $2 \times 2$  is used but is higher BER when used massive MIMO. In figure 4-5 in  $4 \times 4$  antenna found that when the SNR is less than 15dB the MMSE is better than ZF while after 15 it fluctuated the BER. In  $8 \times 8$  the MMSE is better when the SNR less than 15dB but after that, the result is oscillated between them. Nevertheless, in  $16 \times 16$  in all situation the MMSE better than ZF.

Finally, the BER is very high when these algorithms are applied and other algorithms suggested for massive MIMO for channel detection as will show in next section.

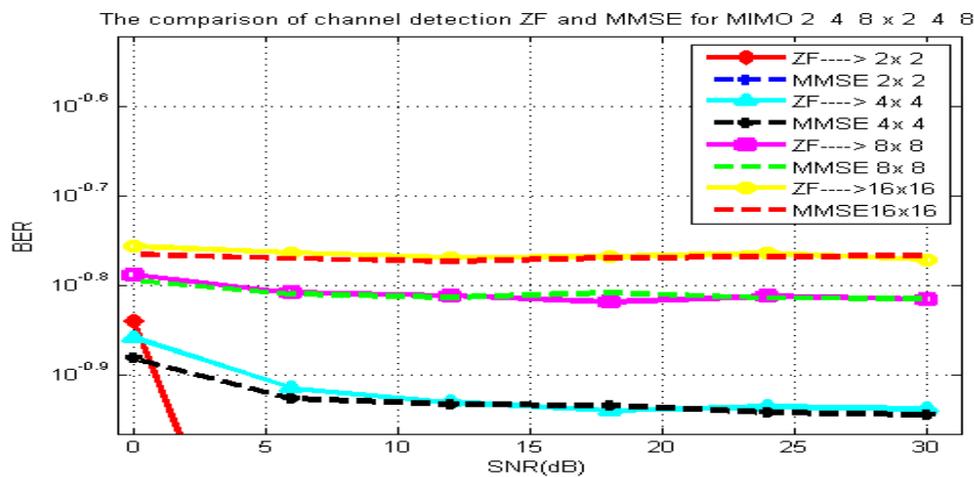


Figure 4-5: BER for Massive MIMO versus SNR

### 4.3 Processing Delay for Algorithms and BER

In this section, calculate the delay caused by algorithms plotted versus a number of antennas and calculate the error happen for each algorithm as shown in figure 4-6. also, it shows the analysis of four algorithms, in which the best one is LU decomposition that distinction by delay Timeless than 1m second and after that RQ and the third one Gaussian Elimination while Gauss-Jordan elimination had the biggest delay compare to other algorithms.

Notice that the delay increase when a number of antenna increase. This distinguish coming for algorithm method and decomposition it's very complex but minimize the loops and calculation happen in other algorithms While the percentage of enhancement illustrated in table 4-3.

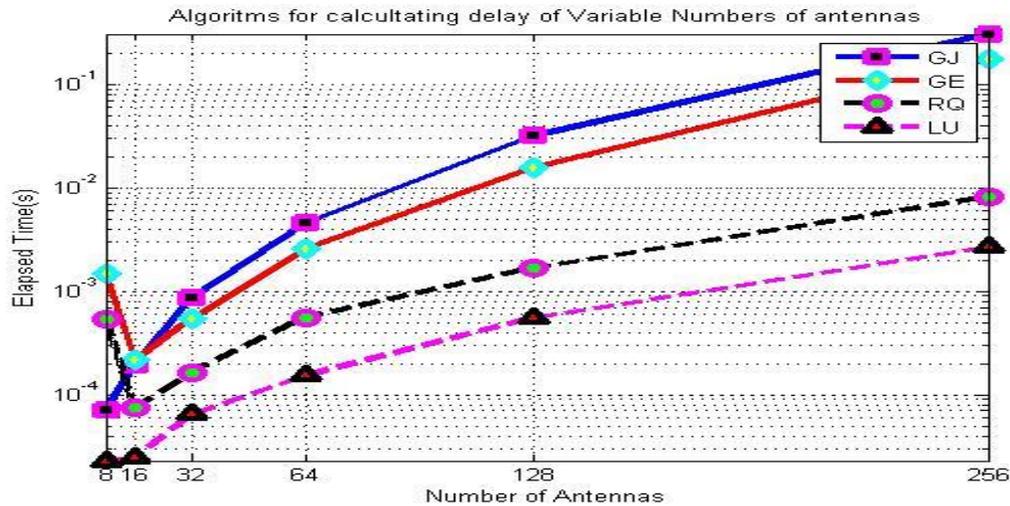


Figure 4-6: Processing delay for GE, GJE, LU and RQ in Massive MIMO

Here we compared the percentage of delay compared to Gauss-Jordan Elimination for three other algorithms Gaussian Elimination, LU and RQ decomposition. These percentages mean the value of reducing delay in each algorithm compared by Gauss-Jordan Elimination algorithm, from table 4-3 the LU had better performance than other algorithms secondly RQ and the last one is Gaussian Elimination.

Table 4-3: Comparison between GJE and Other algorithms

NO.	Name of Algorithm	Percentage compare to GJE (%)
1	GE	49
2	RQ	94
3	LU	98

In figure 4-7 show the BER happen for each algorithm and show that the Gauss-Jordan Elimination have the biggest error than Gaussian Elimination but there are no error happen in LU and RQ or can be negligible. Also, show the BER happen in each algorithm while this figure illustrates the Gaussian Elimination had less BER than GJE and the best one is LU and then RQ it has delay less than GE the decomposition method reduce the BER happen in algorithms and the value of error percentage it's very low compared to GE and GJE.

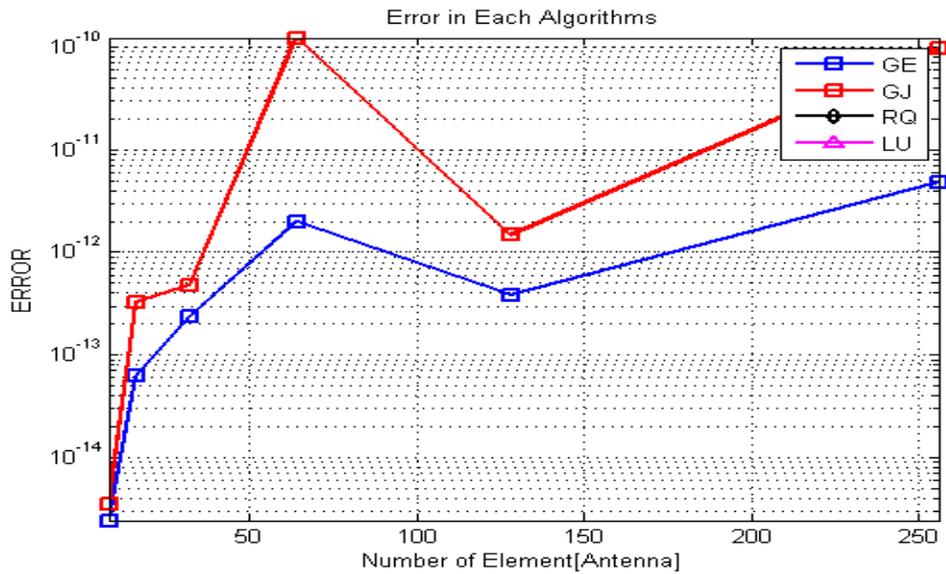


Figure 4-7: BER for GE, GJE, RQ and LU decomposition

#### 4.4 Capacity and Throughput through AMC

This section, mention the performance evaluation of these algorithms for capacity and throughput versus different modulations and coding rates. Here four combination of modulation and coding applied as shown in next sections.

#### 4.4.1 BPSK Modulation and code rate 2/3

Figure 4-8 show the capacity through different algorithms through BPSK and code rate 2/3 versus a various number of antennas. It shows that the LU algorithms had better capacity when the number of antenna increase and secondly is RQ had better capacity. While in GE and GJE the best capacity at 64 for this modulation and coding rate in each one, the delay is big to constrain in which it will effect on channel bandwidth beyond its effect in the capacity as illustrated in table 4-4.

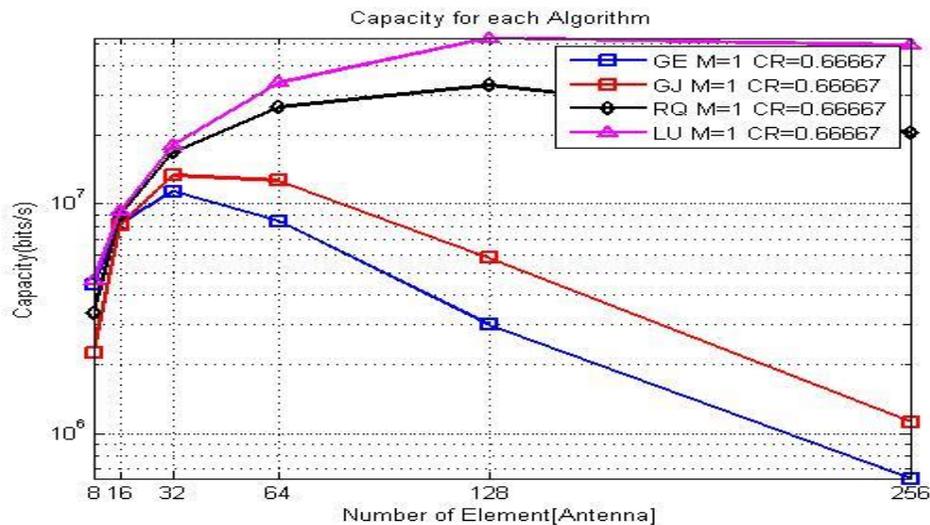


Figure 4-8: Capacity for Algorithms through using BPSK and CR=2/3 versus number of Antenna

The Figure 4-9 show the throughput for algorithms at same modulation and coding rate the bit rate plotted versus a number of antennae. It shows that the LU algorithms had better throughput when the number of antenna increase and secondly is RQ had batter throughput. While in GE and GJE the best throughput at 64 for this modulation and coding rate technique. Delay is constrained in which it will effect on channel bandwidth beyond it effect at throughput as illustrated in table 4-4.

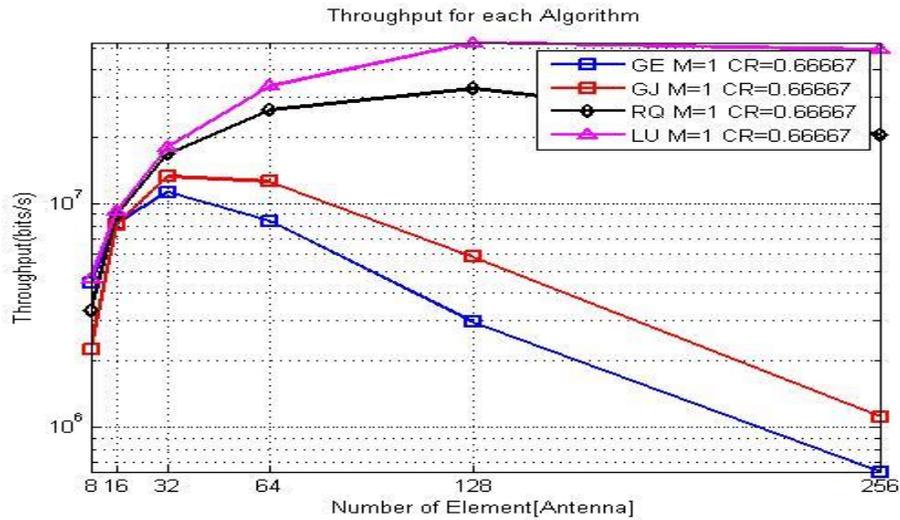


Figure 4-9: Throughput for Algorithms through using BPSK and CR=2/3 versus number of Antenna

Table 4-4 compare between four Algorithms taken Gauss-Jordan Elimination as a reference in which found that the LU had the best performance and less delay while RQ better and last one it's Gaussian Elimination, here we measured the performance of capacity and throughput.

Table 4-4: Comparison between GJE and Other algorithms in terms of capacity and throughput (M=1 CR=2/3)

NO.	Algorithms	Capacity(b/s)	Throughput(b/s)	Percentage (%)
1	GJE	6.0101e+06	5.9981e+06	0
2	GE	7.2087e+06	7.1943e+06	43.7756
3	RQ	1.8173e+07	1.8136e+07	96.7278
4	LU	2.8008e+07	2.7952e+07	98.9848

#### 4.4.2 QAM Modulation and code rate 3/4

Figure 4-10 show the capacity through different algorithms through QAM and code rate 3/4 versus a various number of antennas, also show that the LU algorithms had better capacity when the number of antenna increase and secondly is RQ had batter capacity when 128 than 256. While in GE and GJE the best capacity at 64 for this modulation and coding rate. Delay is constrained in which it will effect on channel bandwidth beyond it effect at capacity as illustrated in table 4-5.

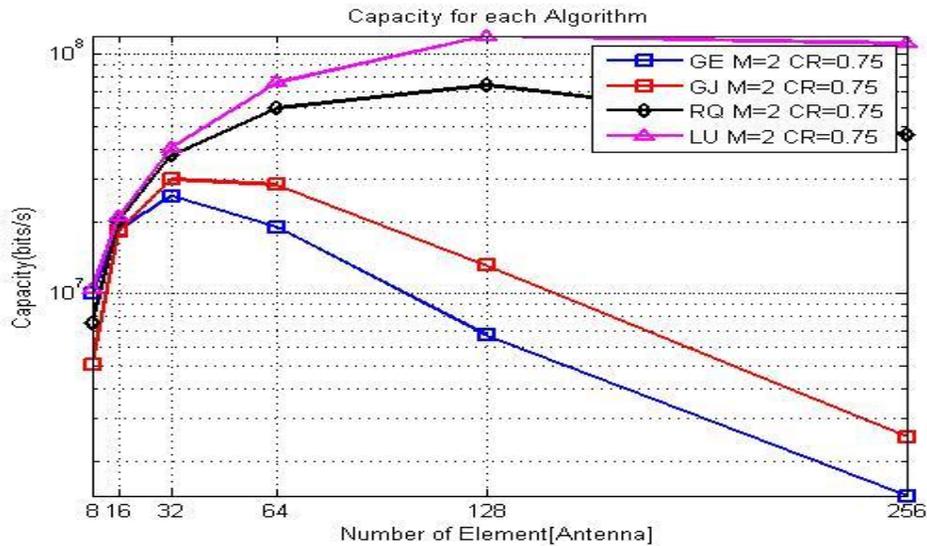


Figure 4-10: Capacity for Algorithms through using QAM and CR=3/4 versus number of Antenna

The Figure 4-11 show the throughput for algorithms at same modulation and coding rate the bit rate plotted versus a number of antennae. While it shows that the LU algorithms had better throughput when the number of antenna increase and secondly is RQ had batter throughput when 128 than 256. While in GE and GJE the best throughput at 64 for this modulation and coding rate technique. Delay is constrained in which it will effect on channel bandwidth beyond it effect at throughput as illustrated in table 4-5 and here it will take the average value for each algorithm.

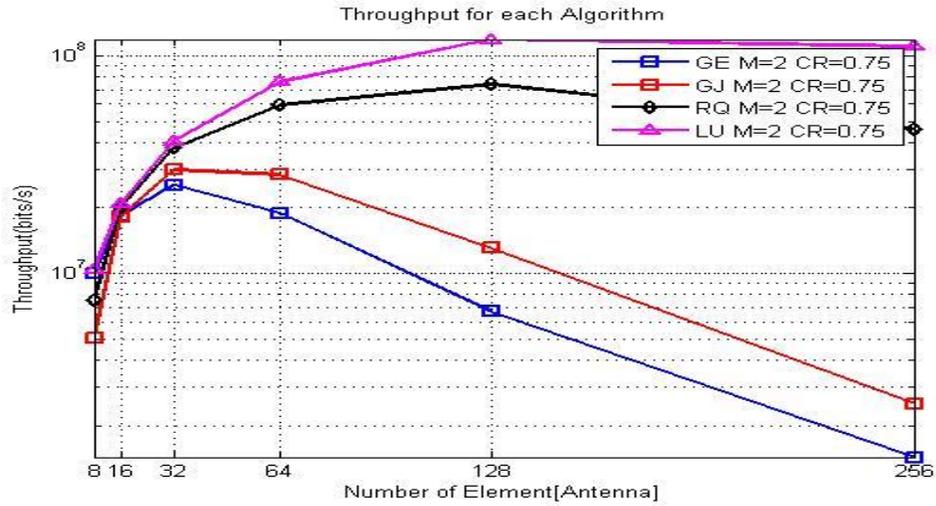


Figure 4-11: Throughput for Algorithms through using QAM and CR=3/4 versus number of Antenna

Table 4-5 compare between four Algorithms taken Gauss-Jordan Elimination as the reference in which found that the LU had the best performance and less delay while RQ better and last one it is Gaussian Elimination, here we measured the performance of capacity and throughput.

Table 4-5: Comparison between GJE and Other algorithms in terms of capacity and throughput (M=2 CR=3/4)

NO.	Algorithm	Capacity(b/s)	Throughput(b/s)	Percentage (%)
1	GJE	1.3523e+07	1.3496e+07	0
2	GE	1.6220e+07	1.6187e+07	43.7756
3	RQ	4.0889e+07	4.0807e+07	96.7278
4	LU	6.3019e+07	6.2893e+07	98.9848

### 4.4.3 16-QAM Modulation and code rate 5/6

Figure 4-12 show the capacity through different algorithms through 16-QAM and code rate 5/6 versus a various number of antennas. In addition, it shows that the LU algorithms had better capacity when the number of antenna increase and secondly RQ had better capacity at 128. While in GE and GJE the best capacity at 64 for this modulation and coding rate. Delay is constrained in which it will effect on channel bandwidth beyond it effect at capacity as illustrated in table 4-6.

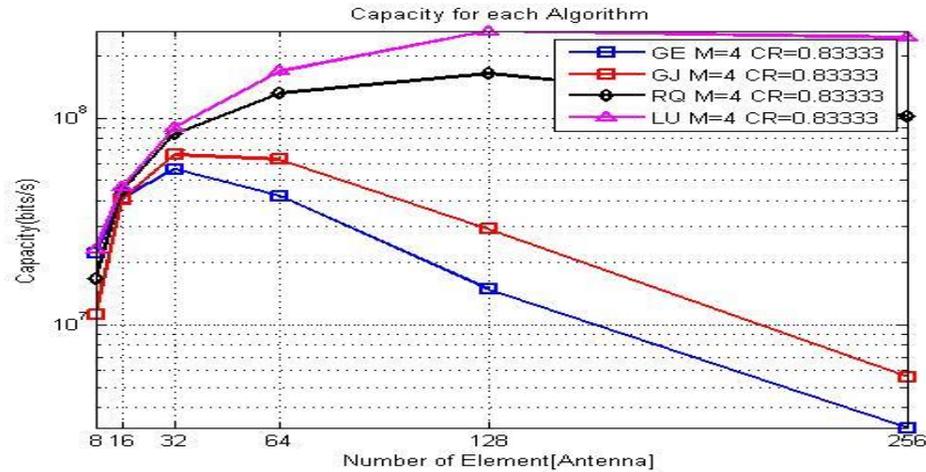


Figure 4-12: Capacity for Algorithms through using 16-QAM and CR=5/6 versus number of Antenna

The Figure 4-13 show the throughput for algorithms at same modulation and coding rate the bit rate plotted versus a number of antennae. Found that the LU algorithms had better throughput when the number of antenna increase and secondly has RQ had batter throughput at 128. While in GE and GJE the best throughput at 64 for this modulation and coding rate technique. Delay is constrained in which it will effect on channel bandwidth beyond it effect at throughput as illustrated in table 4-6, and here they take the average value for each algorithm for capacity and throughput and compared the percentage of enhancement to GJE.

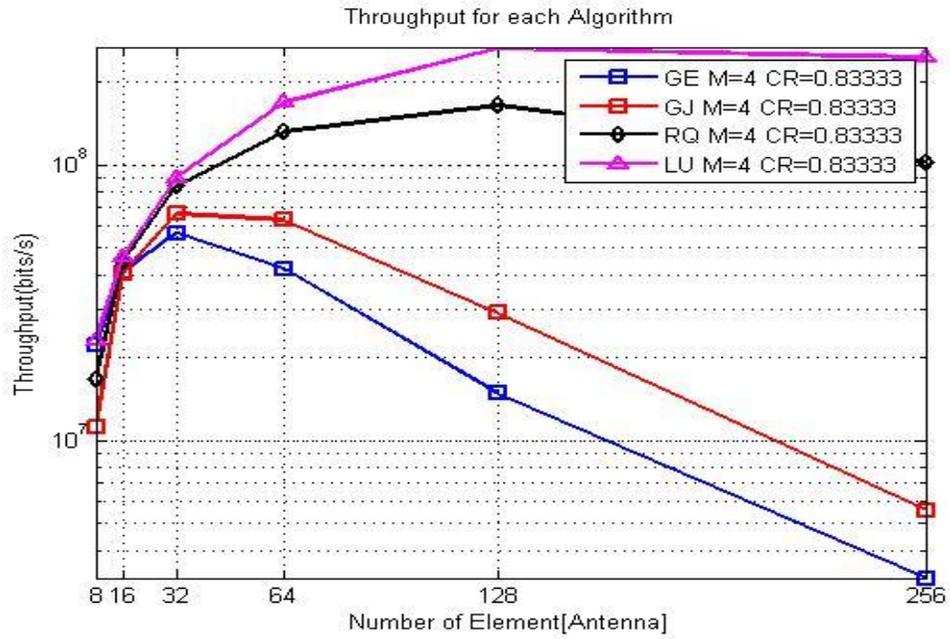


Figure 4-13: Throughput for Algorithms through using 16-QAM and CR=5/6 versus number of Antenna

Table 4-6 compare between four Algorithms taken GJE as a reference in which found that the LU had the best performance and less delay while RQ better and last one it's Gaussian Elimination, here we measured the performance of capacity and throughput.

Table 4-6: Comparison between GJE and Other algorithms in terms of capacity and throughput (M=6 CR=7/8)

NO	Algorithms	Capacity(b/s)	Throughput(b/s)	Percentage (%)
1	GJE	3.0050e+07	2.9990e+07	0
2	GE	3.6044e+07	3.5971e+07	43.7756
3	RQ	9.0864e+07	9.0682e+07	96.7278
4	LU	1.4004e+08	1.3976e+08	98.9848

#### 4.4.4 64-QAM Modulation and code rate 7/8

Figure 4-14 show the capacity through different algorithms through 64-QAM and code rate 7/8 versus a various number of antennas. However, the LU algorithms had better capacity when the number of antenna increase and secondly is RQ had batter capacity at 128. While in GE and GJE the best capacity at 64 for this modulation and coding rate. Delay is constrained in which it will effect on channel bandwidth beyond it effect at capacity as illustrated in table 4-7.

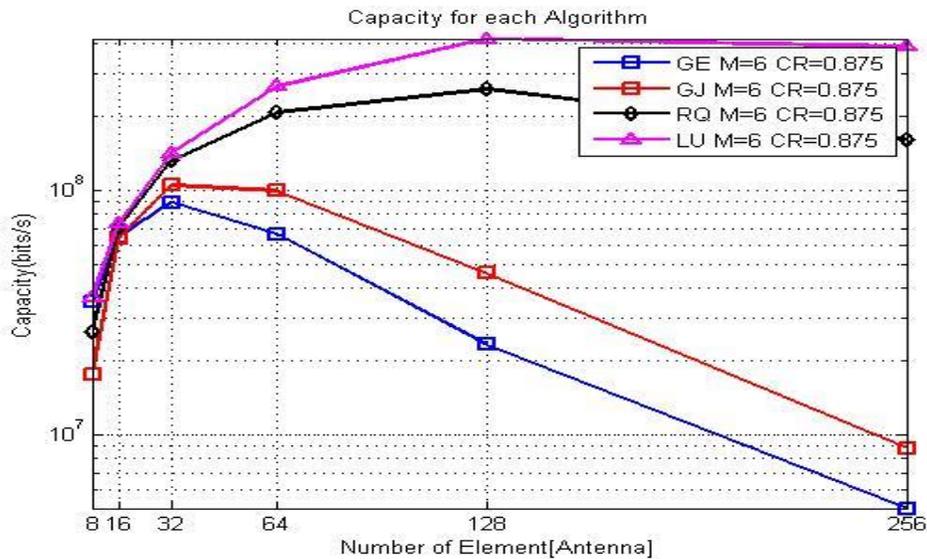


Figure 4-14: Capacity for Algorithms through using 64-QAM and CR=7/8 versus number of Antenna

The Figure 4-15 show the throughput for algorithms at same modulation and coding rate the bit rate plotted versus a number of antennae. However, the LU algorithms had better throughput when the number of antenna increase and secondly has RQ had batter throughput at 128. While in GE and GJE the best throughput at 64 for this modulation and coding rate technique. Delay is constrained in which it will effect on channel bandwidth beyond it effect at throughput as illustrated in table 4-7.

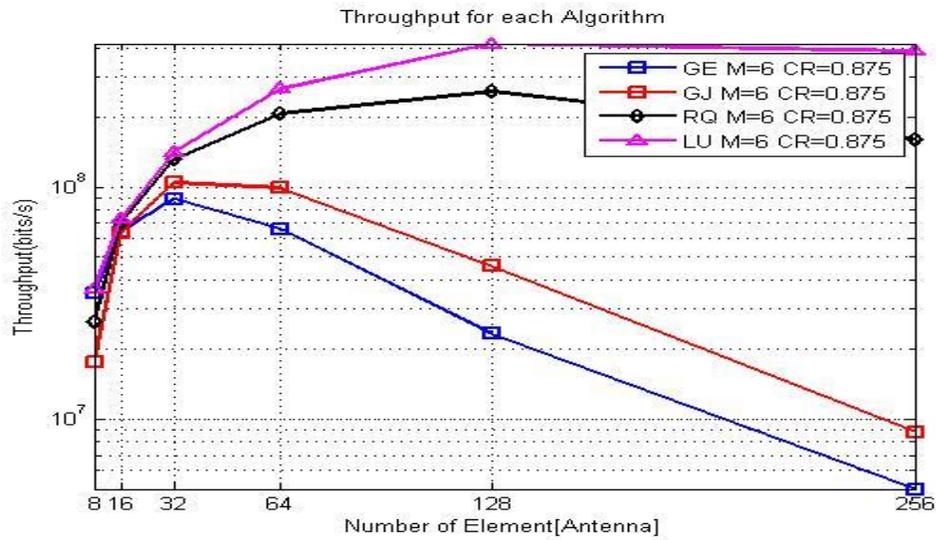


Figure 4-15: Throughput for Algorithms through using 64-QAM and CR=7/8 versus number of Antenna

Table 4-7 compare between four Algorithms taken Gauss-Jordan Elimination as a reference in which found that the LU had the best performance and less delay while RQ better and last one it's Gaussian Elimination, here we measured the performance of capacity and throughput.

Table 4-7: Comparison between GJE and Other algorithms in terms of capacity and throughput (M=6 CR=7/8)

	Algorithms	Capacity(b/s)	Throughput(b/s)	Percentage (%)
1	GJE	4.7329e+07	4.7235e+07	0
2	GE	5.6769e+07	5.6655e+07	45
3	RQ	1.4311e+08	1.4282e+08	95
4	LU	2.2057e+08	2.2013e+08	98

finally, for this section show that the capacity and throughput will increase when using a higher order of modulation and coding and the LU algorithm give higher capacity and throughput and after that RQ and then GJE and lastly have less capacity and throughput is GE and used as a reference in measuring delay.

## **CHAPTER FIVE**

### **CONCLUSION AND RECOMMENDATIONS**

# Chapter Five

## Conclusion and Recommendations

### 5.1 Conclusion

This thesis introduced a brief description of LTE physical layer. In addition, here we concentrate on channel detection and how to equalized channel. In another way, they use ZF and MMSE for massive MIMO but it will give high BER and latency for that four algorithms are suggested Gauss-Jordan Elimination, Gaussian Elimination, RQ Decomposition and LU Decomposition.

We modified the Mathematical Model of four algorithms and applied using MATLAB simulation for each algorithm.

After applying these algorithms at MATLAB we found that the LU Decomposition had the lowest delay, the percentage about (98%) compared Gauss-Jordan Elimination algorithms while RQ had (95%) and Gaussian Elimination about (45%). While the best capacity for LU Decomposition  $3 \rightarrow 25$  Mbs, RQ Decomposition  $2 \rightarrow 16$  Mbs, Gaussian Elimination  $1 \rightarrow 8$  Mbs and Gauss-Jordan Elimination  $0.7 \rightarrow 6$  Mbs and this depend on modulation and coding rate. LU Decomposition Algorithm has the best performance than another algorithm.

### 5.2 Recommendations

After finishing these research there are some other issues can be considering for future research these include:

- Work at channel rather than AWGN channel such as Rayleigh fading channel for different communication environment to be realistic.

- Complexity analysis must do it in algorithms level not software level and can use mathematical complexity analysis such as (Big O Notation).
- Work in channel estimation algorithms and how to enhance their performance to get low BER and latency.
- Using other Algorithm like SDV in channel detection to get lowest BER and latency to be suitable with the biggest number of antenna element such as 2048, 4096.
- Study Massive MIMO in Downlink rather than uplink and see its effect and performance on a channel.

## References

- [1] Tadilo Endeshaw Bogale” Massive MIMO and Millimeter Wave for 5G Wireless HetNet: Potentials and Challenges”, IEEE and Long Bao Le, IEEE 21 October 2015.
- [2] Ramya Ranjan Choudhury, ” A Network Overview of Massive MIMO for 5G Wireless Cellular: System Model and Potential”, Volume 2, Issue 4, June-July, 2014.
- [3] Hillsboro, Debabani Choudhury, “5G Wireless and Millimeter Wave Technology Evolution: An Overview”, Intel Labs, Intel Corporation, 978-1-4799-8275-2/15/\$31.00 ©2015 IEEE.
- [4] JIABING GUO “Design and implementation of LTE-A and 5G kernel algorithms on SIMD vector processor” KTH Royal Technology SE-100 44 Stockholm, Sweden Re.2015.01.30. 2015.
- [5] “Gaussian Elimination”, [Online]. Available: <http://mathworld.wolfram.com/GaussianElimination.html>. [Accessed: 5-Sep-2014, 11:30:00 PM].
- [6] “LTE-Advanced”, June 2013. [Online]. Available: <http://www.3gpp.org/technologies/keywords-acronyms/97-lte-advanced> [Accessed: March 20, 2014, 8:15:00 PM].
- [7] 3Gpp, “Evolved Universal Terrestrial Radio Access (E-UTRA): Carrier Aggregation; Base Station (BS) radio transmission and reception”, 3GPP, TR 36.808, Rel.10, 2013.
- [8] “Evolved Universal Terrestrial Radio Access (E-UTRA): Relay architectures for E-UTRA (LTE- Advanced)”, 3GPP, TR 36.806, Rel.9, 2010.

- [9] “Evolved Universal Terrestrial Radio Access (E-UTRA): LTE physical layer: General description”, 3GPP, TS 36.201, Rel.12.0.0, 2014.
- [10] “Evolved Universal Terrestrial Radio Access (E-UTRA): Physical channels and modulation”, 3GPP, TS 36.211, Re.12.3.0, 2014.
- [11] “Evolved Universal Terrestrial Radio Access (E-UTRA): Multiplexing and channel coding”, 3GPP, TS 36.212, Re.12.2.0, 2014.
- [12] Xiaohu Ge, Senior, Song Tu<sup>1</sup>, Guoqiang Mao, Cheng-Xiang Wang, Tao Han<sup>1</sup>, ” 5G Ultra-Dense Cellular Networks”, Communications Huazhong University of Science and Technology, Re.10.12.2015.
- [13] J. Han and B. Wu, “Handover in 3GPP long term evolution (LTE) systems”, Global Mobile Congress (GMC), October 2010.
- [14] Federico Boccardi, Alcatel-Lucent Robert W. Heath Jr.,” Five Disruptive Technology Directions for 5G, Bell Labs, Alcatel-Lucent Petar Popovski, Aalborg University Re.2013.
- [15] Ronit Nossenson, “Long-Term Evolution Network Architecture”, COMCAS, IEEE International Conference, Nov. 2009.
- [16] Ningning Guo, “Implementation Aspects of 3GPP TD-LTE”, Linkoping University, 2009.
- [17] Jim Zyren, “Overview of the 3GPP Long Term Evolution Physical layer”, Freescale semiconductor, White Paper, July, 2007.
- [18] Jeffrey H.Andrews, Stefano Buzzi, Wan Choi, and Stephen V. Hanly, “What will 5G be?”, IEEE Journal on selected areas in communications, VOL. 32, No.6, June 2014.
- [19] Federico Boccardi, Robert W. Heath Jr., Angel Lozano, Thomas L. Marzetta, and Petar Popovski, “Five Disruptive Technology Directions for 5G”, IEEE communications Magazine, Volume:52, Issue:2, 2014.

- [20] Michael Sung, “SIMD Parallel Processing”, MIT computer science and artificial intelligence laboratory, Feb 22, 2000.
- [21] Xing-liang Zhang and Ya-bo Li, “Optimizing the MIMO Channel Estimation for LTEAdvanced Uplink”, International Conference on Connected Vehicles and Expo, IEEE, 2012.
- [22] Edward Kasem, Roman Marsalek, and Kiri Blumenstein, “Performance of LTE Advanced uplink in a flat Rayleigh channel”, Advanced in Electrical and Electronic Engineering, Volume: 11, Number: 4, Sep, 2013.
- [23] B.Karakaya, H.Arslan, H.A.Cirpan, “Channel estimation for LTE uplink in high Doppler spread”, Proc. IEEE-WCNC, 2008.
- [24] Cheng-yu Huang and Wei-Ho Chung, “An Improved MMSE-Based MIMO detection using Low-Complexity Constellation Search”, IEEE Globecom 2010 Workshop on Broadband Wireless Access, 2010.
- [25] Amit Kumar Sahu, Sudhansu, and Sekhar Singh, “BER performance improvement using MIMO technique over Rayleigh wireless channel with different equalizers”, International Journal of Engineering and Technology (IJET), ISSN: 0975-4024, Vol.4, No.5, Oct-Nov, 2012.
- [26] “Gauss-Jordan Elimination Method”, [Online]. Available: <http://pages.pacificcoast.net/~cazelais/251/gauss-jordan.pdf>. [Accessed: 22-JAN- 2016, 07:12:00 PM].
- [27] “Module for LU factorization with pivoting”, [Online]. Available: <http://mathfaculty.fullerton.edu/mathews//n2003/LUFactorMod.html>. [Accessed: 5-FEB-2016, 10:11:00 PM].
- [28] “QR factorization”, [Online]. Available: <http://www.seas.ucla.edu/~vandenbe/103/lectures/qr.pdf>. [Accessed: 15-FEBp-2016, 08:22:00 PM].

- [29] Su Xin, Jie Zeng, Li-ping Rong, and Yu-Jun Kuang, “Investigation on Key Technologies in Large-Scale MIMO”, Journal of Computer Science and Technology, May 2013.
- [30] Joseph Jezak, Charles Berdanier, Steven Levitan, and Donald Chiarulli, “Accelerated DSP Functions on the CBE for the X-Midas Toolkit”, Proceedings: 2012 International Waveform Diversity & Design Conference, Kauai, Hawaii, and January 2012.
- [31] Nico Galoppo, Naga K. Govindaraju, Michael Henson, and Dinesh Manocha, “LUGPU: Efficient Algorithms for Solving dense Linear systems on Graphics Hardware”, Proceedings of the ACM/IEEE Supercomputing Conference, Nov 12-18, 2005.
- [32] N. Melab, E-G. Talbi, and S. Petiton, “A parallel adaptive Gauss-Jordan Algorithm”, the journal of supercomputing, Volume 17, Issue 2, pp 167-185, January 2000.
- [33] Andreas Dreyer Hysing, “Parallel Seismic Inversion for Shared Memory Systems”, Norwegian University of Science and Technology
- [34]<http://computernetworkingsimplified.com/physical-zLayer/>, [Accessed: 5-MAY-2016 , 12:19:00 PM]
- [relationship-bandwidth-data-rate-channel-capacity](#), [Accessed: 22-jun-2016, 01:15:00 PM].
- [35][https://www.researchgate.net/post/What is the relationship between throughput and BER](https://www.researchgate.net/post/What_is_the_relationship_between_throughput_and_BER), [Accessed: 8-Sep-2016].
- [36] <https://books.google.com/books/?delay='data'+5g>, [Accessed: 11-NOV-2017, 05:12:00 PM].

## Appendix A

In this part is the mathematical representation for four scheme as shown:

### 1-Gauss Elimination

$$Ax = b$$

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1j} \\ \vdots & \ddots & \vdots \\ a_{i1} & \cdots & a_{ij} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ \vdots \\ x_i \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ \vdots \\ b_i \end{bmatrix}$$

Where  $i=1,2,3, \dots, n$  and  $j=1,2,3, \dots, n$

By applying Gaussian Elimination:

$$A = \sum_{k=1}^{n-1} \sum_{i=2}^n \sum_{j=1}^n a_{ij} - a_{kj} * \frac{a_{ik}}{a_{kk}}$$

$$A = \begin{bmatrix} a_{11}' & a_{12}' & a_{13}' & \cdots & a(1-0)(j-1)' & a(1-0)j'| & b_1' \\ 0 & a_{22}' & a_{23}' & \cdots & a(2-0)(j-1)' & a(2-0)j'| & b_2' \\ & \vdots & & \ddots & & \vdots & \\ & 0 & 0 & 0 & \cdots & a(i-1)(j-2)' & a(i-1)j'| & b_i - 1' \\ & 0 & 0 & 0 & \cdots & 0 & a(i-0)j'| & b_i' \end{bmatrix}$$

After that doing back substitution

$$x_i = b_i - \sum_{k=i+1}^n a_{ik}' x_k$$

## 2- Gauss-Jordan Elimination

$$Ax = b$$

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1j} \\ \vdots & \ddots & \vdots \\ a_{i1} & \cdots & a_{ij} \end{bmatrix}, \quad x = \begin{bmatrix} x_1 \\ \vdots \\ x_i \end{bmatrix}, \quad b = \begin{bmatrix} b_1 \\ \vdots \\ b_i \end{bmatrix}$$

Where  $i=1, 2, 3 \dots n$  and  $j=1, 2, 3 \dots n$

By applying Gaussian-Jordan:

$$A = \sum_{k=1}^{n-1} \sum_{i=2}^n \sum_{j=1}^n a_{ij} - a_{kj} * \frac{a_{ik}}{a_{kk}}$$

$$A = \begin{bmatrix} a_{11}' & a_{12}' & a_{13}' & \cdots & a(1-0)(j-1)' & a(1-0)j' & b_1' \\ 0 & a_{22}' & a_{23}' & \cdots & a(2-0)(j-1)' & a(2-0)j' & b_2' \\ & \vdots & & \ddots & & \vdots & \\ & 0 & 0 & 0 & \cdots & a(i-1)(j-2)' & a(i-1)j' & b_i - 1' \\ & 0 & 0 & 0 & \cdots & 0 & a(i-0)j' & b_i' \end{bmatrix}$$

Now the upper part of matrix equal zero also they need to make the lower zero as shown below

$$A = \sum_{k=n-1}^1 \sum_{i=n}^2 \sum_{j=n}^2 a_{ij} - a_{kj} * \frac{a_{ik}}{a_{kk}}$$

$$A = \begin{bmatrix} a_{11} & 0 & \cdots & 0 & | & b_1' \\ 0 & a_{22} & \cdots & 0 & | & b_2' \\ & \vdots & \ddots & & & \vdots \\ 0 & 0 & \cdots & 0 & | & \vdots \\ 0 & 0 & \cdots & a_{ij} & | & b_i' \end{bmatrix}$$

After that divided each row by  $a_{kk}$

$$A = A/a_{kk};$$

$$A = \left[ \begin{array}{cccc|c} 1 & 0 & \dots & 0 & b1'/a11 \\ 0 & 1 & \dots & 0 & b2'/a22 \\ & \vdots & \ddots & & \vdots \\ 0 & 0 & \dots & 0 & \vdots \\ 0 & 0 & \dots & 1 & bi'/aii \end{array} \right]$$

In which

$$x = b_i/a_{ii};$$

### 3- LU Decomposition

$$A = \begin{bmatrix} a_{11} & \dots & a_{1j} \\ \vdots & \ddots & \vdots \\ a_{i1} & \dots & a_{ij} \end{bmatrix}, \quad L = \begin{bmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{i1} & l_{i2} & \dots & l_{ij} \end{bmatrix},$$

$$U = \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1j} \\ 0 & u_{22} & \dots & u_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & u_{ij} \end{bmatrix}$$

**Note**

$$a_{11} = l_{11} * u_{11}$$

$$a_{12} = l_{21} * u_{12} + l_{22} * u_{22}$$

$$a_{ij} = \sum_{k=1}^i l_{ik} * u_{kj} \quad \text{When } i > j$$

$$a_{ij} = \sum_{k=1}^j l_{ik} * u_{kj} \quad \text{When } i < j$$

$$A * x = b \quad \rightarrow L * u * x = b \quad \rightarrow L * y = b \quad \rightarrow U * x = y$$

**Solve by using**

**1-Forward substitution**

$$y_i = \frac{1}{l_{ii}} (b_i - \sum_{j=1}^{i-1} l_{ij} * y_j)$$

**2- Back Substitutions**

$$x_i = \frac{1}{u_{ii}} (y_i - \sum_{j=i+1}^n u_{ij} * x_j)$$

## 4- RQ Decomposition

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1j} \\ a_{21} & a_{22} & a_{23} & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{i1} & a_{i2} & a_{i3} & \dots & a_{ij} \end{bmatrix} = [A_1 \quad A_2 \quad A_3 \quad \dots \quad A_j]$$

Where

$$A_1 = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{i1} \end{bmatrix}, \quad A_2 = \begin{bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{i2} \end{bmatrix}, \quad A_3 = \begin{bmatrix} a_{13} \\ a_{23} \\ \vdots \\ a_{i3} \end{bmatrix}, \dots, \quad A_j = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{ij} \end{bmatrix}$$

$$R = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1j} \\ 0 & r_{22} & \dots & r_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & r_{ij} \end{bmatrix}$$

$$r_{11} = \|A_1\| = \sqrt{a_{11}^2 + a_{12}^2 + \dots + a_{i1}^2}$$

$$q_1 = \frac{1}{\|A_1\|} A_1 = \begin{bmatrix} a_{11}/\|A_1\| \\ a_{12}/\|A_1\| \\ \vdots \\ a_{ij}/\|A_1\| \end{bmatrix}$$

$$q_2 = \frac{S_2}{\|S_2\|} \rightarrow q_n = \frac{S_n}{\|S_n\|}$$

$$S_2 = (1 - q_1 * q_1^T) * A_2$$

$$S_n = (1 - q_1 * q_1^T)(1 - q_2 * q_2^T)(1 - q_3 * q_3^T) \dots (1 - q_{(n-1)} * q_{(n-1)}^T) * A_n$$

$$r_{22} = \|S_2\| \rightarrow r_{nn} = \|S_n\| \rightarrow r_{ij} = q_i^T * A_j$$

$$Q = [q_1 \quad q_2 \quad \dots \quad q_j], \quad R = \begin{bmatrix} r_{11} & r_{12} & \dots & r_{1j} \\ 0 & r_{11} & \dots & r_{2j} \\ & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & r_{ij} \end{bmatrix}$$

$$A = Q * R, \quad Q * Q^T = I, \quad A * X = B$$

## Appendix B

Code for channel detection and estimation for Massive MIMO based on LS MMSE at estimation and ZF and MMSE at channel detection.

### Channel Detection

```
clear all,clc
close all
% N = 10^5;
% Eb_N0_dB = [0:30];
% nTx = 8;
% nRx = 8
clour=['r' 'b' 'c' 'k' 'y' 'g' 'm' 'g'];
clour(15)='y';
clour(16)='r';
line=['-' '-' '-' '-' '-' '-' '-' '-'];
line(15)='-';
line(16)='-';
shape=['d' '+' '^' '*' ' ' ' ' 's' ' '];
shape(15)='o';
shape(16)=' ';
for txrx=[2 4 8 16]
    N = 10^5;
    Eb_N0_dB = [0:6:30];
    nTx = txrx;
    nRx = txrx;
    for ii = 1:length(Eb_N0_dB)
        ip = rand(1,N)>0.5;
        s = 2*ip-1;
        sMod = kron(s,ones(nRx,1));
        sMod = reshape(sMod,[nRx,nTx,N/nTx]);
        h = 1/sqrt(2)*[randn(nRx,nTx,N/nTx) + j*randn(nRx,nTx,N/nTx)];
        n = 1/sqrt(2)*[randn(nRx,N/nTx) + j*randn(nRx,N/nTx)];
        y = squeeze(sum(h.*sMod,2)) + 10^(-Eb_N0_dB(ii)/20)*n;
```

```

hCof = zeros(nRx,nTx,N/nTx) ;
for ii1=1:nRx
    for kk=1:nRx
        if(kk~=ii1)
            hCof(ii1,ii1,:) = hCof(ii1,ii1,:)+sum(h(:,kk,:).*conj(h(:,kk,:)),1);
        end
    end
    hCof(ii1,ii1,:)=hCof(ii1,ii1,:)/(nRx-1);
end
% hCof(1,1,:) = sum(h(:,2,:).*conj(h(:,2,:)),1);
% hCof(2,2,:) = sum(h(:,1,:).*conj(h(:,1,:)),1);
for ii1=1:nRx
    for kk=1:nRx
        if(kk~=ii1)
            lll=kk;
            hCof(ii1,kk,:) = hCof(ii1,kk,:)+sum(h(:,ii1,:).*conj(h(:,kk,:)),1);
        end
    end
    hCof(ii1,lll,:)=hCof(ii1,lll,:)/(nRx-1);
end
% hCof(2,1,:) = -sum(h(:,2,:).*conj(h(:,1,:)),1);
% hCof(1,2,:) = -sum(h(:,1,:).*conj(h(:,2,:)),1);
for ijk=1:N/nTx
    hDen( :,ijk) = abs(det(hCof(:,:,ijk)))/((hCof(1,1,:).*hCof(2,2,:)) - (hCof(1,2,:).*hCof(2,1,:)));
end
%hDen = ((hCof(1,1,:).*hCof(2,2,:)) - (hCof(1,2,:).*hCof(2,1,:)));
hDen = reshape(kron(reshape(hDen,1,N/nTx),ones(nRx,nTx)),nRx,nTx,N/nTx);
hInv = hCof./hDen;
hDen=[];
hMod = reshape(conj(h),nRx,N);
yMod = kron(y,ones(1,nRx));
yMod = sum(hMod.*yMod,1);
yMod = kron(reshape(yMod,nRx,N/nTx),ones(1,nRx));
yHat = sum(reshape(hInv,nRx,N).*yMod,1);

```

```

ipHat = real(yHat)>0;
nErr(ii) = size(find([ip- ipHat]),2);
end
simBer = nErr/N;
EbN0Lin = 10.^(Eb_N0_dB/10);
theoryBer_nRx1 = 0.5.*(1-1*(1+1./EbN0Lin).^(-0.5));
p = 1/2 - 1/2*(1+1./EbN0Lin).^(-1/2);
theoryBerMRC_nRx2 = p.^2.*(1+2*(1-p));
%close all
figure(1)
semilogy(Eb_N0_dB,simBer,[line(txrx-1) clour(txrx-1) shape(txrx-1)],'linewidth',3);hold on
for ii = 1:length(Eb_N0_dB)
ip = rand(1,N)>0.5;
s = 2*ip-1;
sMod = kron(s,ones(nRx,1));
sMod = reshape(sMod,[nRx,nTx,N/nTx]);
h = 1/sqrt(2)*[randn(nRx,nTx,N/nTx) + j*randn(nRx,nTx,N/nTx)];
n = 1/sqrt(2)*[randn(nRx,N/nTx) + j*randn(nRx,N/nTx)];
y = squeeze(sum(h.*sMod,2)) + 10^(-Eb_N0_dB(ii)/20)*n;
hCof = zeros(nRx,nTx,N/nTx) ;
for ii1=1:nRx
    for kk=1:nRx
        if(kk~=ii1)
            hCof(ii1,ii1,:) = hCof(ii1,ii1,.)+sum(h(:,kk,:).*conj(h(:,kk,:)),1)+10^(-Eb_N0_dB(ii)/10);
        end
    end
    hCof(ii1,ii1,:)=hCof(ii1,ii1,:)/(nRx-1);
end
% hCof(1,1,:) = sum(h(:,2,:).*conj(h(:,2,:)),1);
% hCof(2,2,:) = sum(h(:,1,:).*conj(h(:,1,:)),1);
for ii1=1:nRx
    for kk=1:nRx
        if(kk~=ii1)
            ll=kk;

```

```

        hCof(ii1,kk,:) = hCof(ii1,kk,:)+sum(h(:,ii1,:).*conj(h(:,kk,:)),1);
    end
end
hCof(ii1,lll,:)=hCof(ii1,lll,)/(nRx-1);
end
% hCof(2,1,:) = -sum(h(:,2,:).*conj(h(:,1,:)),1);
% hCof(1,2,:) = -sum(h(:,1,:).*conj(h(:,2,:)),1);
for ijk=1:N/nTx
hDen( :,ijk) = abs(det(hCof(:,ijk)));%((hCof(1,1,:).*hCof(2,2,:)) - (hCof(1,2,:).*hCof(2,1,:)));
end
%hDen = ((hCof(1,1,:).*hCof(2,2,:)) - (hCof(1,2,:).*hCof(2,1,:)));
% hCof(1,1,:) = sum(h(:,2,:).*conj(h(:,2,:)),1) + 10^(-Eb_NO_dB(ii)/10);
% hCof(2,2,:) = sum(h(:,1,:).*conj(h(:,1,:)),1) + 10^(-Eb_NO_dB(ii)/10);
% hCof(2,1,:) = -sum(h(:,2,:).*conj(h(:,1,:)),1);
% hCof(1,2,:) = -sum(h(:,1,:).*conj(h(:,2,:)),1);
% hDen = ((hCof(1,1,:).*hCof(2,2,:)) - (hCof(1,2,:).*hCof(2,1,:)));
hDen = reshape(kron(reshape(hDen,1,N/nTx),ones(nRx,nTx)),nRx,nTx,N/nTx);
hInv = hCof./hDen;
hDen=[];
hMod = reshape(conj(h),nRx,N);
yMod = kron(y,ones(1,nTx));
yMod = sum(hMod.*yMod,1);
yMod = kron(reshape(yMod,nRx,N/nTx),ones(1,nRx));
yHat = sum(reshape(hInv,nRx,N).*yMod,1);
ipHat = real(yHat)>0;
nErr(ii) = size(find([ip- ipHat]),2);
end
simBer = nErr/N;
EbNOLin = 10.^(Eb_NO_dB/10);
theoryBer_nRx1 = 0.5.*(1-1*(1+1./EbNOLin).^(-0.5));
semilogy(Eb_NO_dB,simBer,[line(txrx) '-' clour(txrx) shape(txrx)],'linewidth',3);
% axis([0 25 10^-5 0.5])
end
grid on

```

```

%hold off
legend('ZF----> 2x 2', 'MMSE 2x 2', 'ZF----> 4x 4', 'MMSE 4x 4', 'ZF----> 8x 8', 'MMSE 8x
8', 'ZF---->16x16', 'MMSE16x16');
xlabel('SNR(dB)');
ylabel('BER');
title(['The comparison of channel detection ZF and MMSE for MIMO ' num2str([2 4 8 ]) ' x '
num2str([ 2 4 8])]);

```

## Channel Estimation

```

%close all
%clf
clear all
clc
numclour=1;
c_est_count=1;
for NOA=[ 1 2 4 8 16]
TX_Num=NOA;
RX_Num =NOA;
M_RB_PUSCH=10;
N_sc_RB=12;
DFT_Size =M_RB_PUSCH*N_sc_RB;
FFT_Size = 2048;
CP_Size =160;
TX_Num=TX_Num+1;
RX_Num =RX_Num+1;
n_cs=linspace(0,6,max(TX_Num,RX_Num));
alpha=2*pi*n_cs/12;
sq2=sqrt(2);
h_pdp=[1,0.5,0.25,0.125,0.0625]*sq2;
L=length(h_pdp);
F=zeros(DFT_Size,FFT_Size);
for i=1:DFT_Size

```

```

for j=1:FFT_Size
F(i,j)=exp(-1j*2*pi*(i-1)*(j-1)/FFT_Size);
end
end
L_TWD=floor(1.2*L);
index=zeros(1,TX_Num*L_TWD);
for n=1:TX_Num
index((n-1)*L_TWD+1:n*L_TWD)=n_cs(n)*FFT_Size/12+(1:L_TWD);
end
SNR=[0:5:30];
Len_SNR=length(SNR);
pilot = Gen_cazac_for_pilot(N_sc_RB,M_RB_PUSCH);
tx_syms_map=zeros(TX_Num,FFT_Size);
for n=1:TX_Num
tx_syms_map(n,1:DFT_Size)=exp(1j*alpha(n)*[0:DFT_Size-1]).*pilot.';
end
tx_syms=zeros(TX_Num,FFT_Size);
for n=1:TX_Num
tx_syms(n,:)=ifft(tx_syms_map(n,:),FFT_Size);
end
tx_syms_ACP=[tx_syms(:,FFT_Size-CP_Size+1:end) tx_syms];%add cyclic prefix
MSE=zeros(2,Len_SNR);
Num_syms=1000;
for nsnr=1:Len_SNR
for sym_dex=1:Num_syms
h=zeros(RX_Num*TX_Num,L);
H_ideal=zeros(RX_Num,TX_Num*DFT_Size);
sig_fad=zeros(RX_Num,FFT_Size+CP_Size);
for n=1:RX_Num
for m=1:TX_Num
temp=h_pdp.*[randn(1,L)+1j*randn(1,L)]/sq2;
sig_fad(n,:)=sig_fad(n,:)+filter(temp,1,tx_syms_ACP(m,:));
h((n-1)*TX_Num+m,:)=temp;
end
end
H=fft(temp,FFT_Size);

```

```

H_ideal(n,(m-1)*DFT_Size+1:m*DFT_Size)=H(1:DFT_Size);
end
end
sig_rx=zeros(RX_Num,FFT_Size+CP_Size);
sigma=zeros(1,RX_Num);
for n=1:RX_Num
sig_rx(n,:)=awgn(sig_fad(n,:),SNR(nsnr),'measured');
sigma(n)=abs(norm(sig_rx(n,:))^2-norm(sig_fad(n,:))^2);
end
sig_rx_RCP=sig_rx(:,CP_Size+1:end);
SigRxed_Fre=fft(sig_rx_RCP,FFT_Size,2);
Y=SigRxed_Fre(:,1:DFT_Size);
H_LS=zeros(RX_Num,DFT_Size);
for m=1:RX_Num
H_LS(m,1:DFT_Size)=Y(m,:).*conj(pilot. ');
end
for alg=1:2
H_est=zeros(RX_Num,TX_Num*DFT_Size);
for m=1:RX_Num
if(alg==1)
g=pinv(F(:,index))*H_LS(m,:).';
else
g=inv(F(:,index)'*F(:,index)+sigma(m)*eye(TX_Num*L_TWD))*F(:,index)'*H_LS(m,:).';
end
for n=1:TX_Num
temp=g((n-1)*L_TWD+1:n*L_TWD).';
H=fft(temp,FFT_Size);
H_est(m,(n-1)*DFT_Size+1:n*DFT_Size)=H(1:DFT_Size);
end
end
MSE(alg,nsnr)=MSE(alg,nsnr)+sum(sum(abs(H_ideal-
H_est).^2))/(RX_Num*TX_Num*DFT_Size);
end
end

```

```

end
MSE=MSE/Num_syms;
clour=['r' 'b' 'k' 'm' 'c' 'g','y','k','b', 'k'];
line= [''''''''''''''''''''''.']
figure(1)
for i=1:2
semilogy(SNR,MSE(i,:),[line(numclour) clour(numclour)],'linewidth',3);hold on
numclour=numclour+1;
end
if (NOA==2)
channel_estimatio_est1= H_est;
channel_estimatio_Idl1= H_ideal;
elseif (NOA==4)
channel_estimatio_est2= H_est;
channel_estimatio_Idl2= H_ideal;
elseif (NOA==8)
channel_estimatio_est3= H_est;
channel_estimatio_Idl3= H_ideal;
elseif (NOA==16)
channel_estimatio_est4= H_est;
channel_estimatio_Idl4= H_ideal;
end
end
%axis tight
grid on
xlabel('SNR(dB)')
ylabel('Error = Estimated-Ideal')
legend('LS 2x 2','MMSE 2x 2','LS 4x 4','MMSE 4x 4','LS 8x 8','MMSE 8x 8','LS
16x16','MMSE 16x16','LS 32x32','MMSE 32x32')
title('Channel estimation algorithms for 2x2 MIMO');

```

## Appendix C

Code for four algorithms a shown in chapter three, in addition to capacity and throughput

```
clc,clear all,clf,close all
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%For Algorithms For Equalizing Channel
%%
Count_ALGORITHMS=1;
%%
%%
for N_ALGORITM=[8 16 32 64 128 256]
%%
%
Channel_Estimation;
%%
%
Gaussian_Elimination;
%%
%
GaussJordan;
%%
%
simulate_rq;
%%
%
LU_DECMP;
end
%%
Plot_Output;
%%
```

## Channel Estimation

```
%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%Channel Estimation  
Matrix_ALGORITHMS=100*rand (N_ALGORITM, N_ALGORITM+1);  
%%  
%In this thesis the channel asuumed as random matrix%  
%it will change randomly as show above
```

## Gaussian Elimination

```
%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%Gaussian Elimination  
%%  
tic  
Received_EChannel_RSignal_GE=Matrix_ALGORITHMS;  
%%  
%Number of Row and Coulmns  
[m,n]=size(Matrix_ALGORITHMS);  
for j=1:m-1  
%%  
%Pivoting  
    for z=2:m  
        if Matrix_ALGORITHMS(j,j)==0  
            t=Matrix_ALGORITHMS(j,:);  
            Matrix_ALGORITHMS(j,:)=Matrix_ALGORITHMS(z,:);  
            Matrix_ALGORITHMS(z,:)=t;  
        end  
    end  
end  
%%  
%Convert the Element Below Diagonal to Zero Forword Elimination  
    for i=j+1:m
```

```

        Matrix_ALGORITHMS(i,:) = Matrix_ALGORITHMS(i,:) -
Matrix_ALGORITHMS(j,:) * (Matrix_ALGORITHMS(i,j) / Matrix_ALGORITHMS
(j,j));
    end
end
%%
%Back Substitution
x=zeros(1,m);
for s=m:-1:1
    c=0;
    for k=2:m
        c=c+Matrix_ALGORITHMS(s,k)*x(k);
    end
    x(s)=(Matrix_ALGORITHMS(s,n)-c)/Matrix_ALGORITHMS(s,s);
end
Elapsed_Time_GE(Count_ALGORITHMS)=toc;
%%
NOFA_GE(Count_ALGORITHMS)          =N_ALGORITM;
Transmitted_Signal_GE
=Received_EChannel_RSignal_GE(:,1:length(Matrix_ALGORITHMS)-
1)\Received_EChannel_RSignal_GE(:,end);
Received_Signal_GE          =x';
Error_GE(Count_ALGORITHMS)      =sum(abs(Received_Signal_GE-
Transmitted_Signal_GE))/N_ALGORITM;
%%

```

## Gauss Jordan Elimination

```

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Gass-Jordan
%%
tic;
Received_EChannel_RSignal_GJ=Matrix_ALGORITHMS;
[m,n]=size(Matrix_ALGORITHMS);

```

```

for j=1:m-1
%%
%Pivoting
    for z=2:m
        if Matrix_ALGORITHMS(j,j)==0
            t=Matrix_ALGORITHMS(j,:);
            Matrix_ALGORITHMS(j,:)=Matrix_ALGORITHMS(z,:);
            Matrix_ALGORITHMS(z,:)=t;
        end
    end
end
%%
%Convert the Element below Major Diagonal to Zero
    for i=j+1:m
        Matrix_ALGORITHMS(i,:)=Matrix_ALGORITHMS(i,:)-
Matrix_ALGORITHMS(j,:)*(Matrix_ALGORITHMS(i,j)/Matrix_ALGORITHMS
(j,j));
    end
end
end
%%
%Convert the Element Above Major Diagonal to Zero
for j=m:-1:1
    for i=j-1:-1:1
        Matrix_ALGORITHMS(i,:)=Matrix_ALGORITHMS(i,:)-
Matrix_ALGORITHMS(j,:)*(Matrix_ALGORITHMS(i,j)/Matrix_ALGORITHMS
(j,j));
    end
end
end
%%
%Convert the Element IN Major Diagonal to ones
for s=1:m

Matrix_ALGORITHMS(s,:)=Matrix_ALGORITHMS(s,+)/Matrix_ALGORITHMS(
s,s);
    yy(s)=Matrix_ALGORITHMS(s,n);
end
end

```

```

Elapsed_Time_GJ(Count_ALGORITHMS)=toc;
NOFA_GJ(Count_ALGORITHMS)=N_ALGORITM;
Transmitted_Signal_GJ=Received_EChannel_RSignal_GJ(:,end);
Received_Signal_GJ=Received_EChannel_RSignal_GJ(:,1:length(Matri
x_ALGORITHMS)-1)*yy';
Error_GJ(Count_ALGORITHMS)=sum(abs(Received_Signal_GJ-
Transmitted_Signal_GJ))/N_ALGORITM;
%%

```

## RQ Decomposition

```

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%RQ Decompostion
%%
%
tic
Number_of_Antenna_RQ          =N_ALGORITM;
Cannel_Estimated_Matrix_RQ
=Matrix_ALGORITHMS(1:Number_of_Antenna_RQ,1:(end-1));%H
Received_Matrix_RQ
=Matrix_ALGORITHMS(1:Number_of_Antenna_RQ,end);%b
[Q, R]                        =qr(Cannel_Estimated_Matrix_RQ);%Q R
Transmitted_Signal_RQ        =R\Q'*Received_Matrix_RQ;%x=R/Q'*b
%%
%
Elapsed_Time_RQ(Count_ALGORITHMS)=toc;
NOFA_RQ(Count_ALGORITHMS)=N_ALGORITM;
Transmitted_Signal_RQ=Cannel_Estimated_Matrix_RQ*Transmitted_Sign
al_RQ;
Received_Signal_RQ=Received_Matrix_RQ;
Error_RQ(Count_ALGORITHMS)= sum(abs(Received_Signal_RQ-
Transmitted_Signal_RQ))/N_ALGORITM;
%%

```

## LU Decompositions

```
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
%LU Decompostion
%%
%
tic
Number_of_Antenna_LU          =N_ALGORITM;
Channel_Estimated_Matrix_LU  =
Matrix_ALGORITHMS(1:Number_of_Antenna_LU,1:(end-1));
Received_Matrix_LU
=Matrix_ALGORITHMS(1:Number_of_Antenna_LU,end);
[L, U]                        =lu(Channel_Estimated_Matrix_LU);
Y                              =L\Received_Matrix_LU;
Transmitted_Signal_LU        =U\Y;
%%
%channel_estimated_matrix*Transmitted_Signal
%Received_matrix
Elapsed_Time_LU(Count_ALGORITHMS)=toc;
NOFA_LU(Count_ALGORITHMS)=N_ALGORITM;
Received_Signal_LU=Channel_Estimated_Matrix_LU*Transmitted_Signal_LU;
Transmitted_Signals_LU=Received_Matrix_LU;
Error_LU(Count_ALGORITHMS)= sum(abs(Transmitted_Signals_LU-
Received_Signal_LU))/N_ALGORITM;
Count_ALGORITHMS              =Count_ALGORITHMS+1;
%%
```

## Capacity, through put and plotting

```
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%Plot the Result
%%
%Plot Elapsed Delay in Each Algorithms
figure(6)
semilogy(NOFA_GJ,Elapsed_Time_GJ,'-bs','linewidth',3,
'MarkerSize',10,'MarkerEdgeColor','m','MarkerFaceColor','k');hold
on
semilogy(NOFA_GE,Elapsed_Time_GE,'-
rd','linewidth',3,'MarkerSize',10,'MarkerEdgeColor','c','MarkerF
aceColor','y');
semilogy(NOFA_RQ,Elapsed_Time_RQ,'ko--
','linewidth',3,'markersize',10,'MarkerEdgeColor','m','MarkerFac
eColor','g');
semilogy(NOFA_LU,Elapsed_Time_LU,'m^--
','linewidth',3,'markersize',10,'MarkerEdgeColor','k','MarkerFac
eColor','r');
xlabel('Number of Antennas');
ylabel('Elapsed Time');
title('Algoritms for calculatating delay of Variable Numbers of
antennas');
legend('GJ','GE','RQ','LU')
grid;
axis tight
hold off
%%
%Plot Error in Each Algorithms
figure(2)
semilogy(NOFA_GJ,Error_GJ,'-Bs','linewidth',2);hold on
semilogy(NOFA_GE,Error_GE,'-rs','linewidth',2)
semilogy(NOFA_RQ,Error_RQ,'-kd','linewidth',2)
```

```
semilogy(NOFA_LU,Error_LU,'-m^','linewidth',2)
xlabel('Number of Element[Antenna]');
ylabel('ERROR');
title('Error in Each Algorithms');
legend('GE','GJ','RQ','LU')
grid
axis tight
hold off
sum(Elapsed_Time_GJ);
sum(Elapsed_Time_GE);
sum(Elapsed_Time_LU);
sum(Elapsed_Time_RQ);
PER_GE=(sum(Elapsed_Time_GJ) -
sum(Elapsed_Time_GE))/sum(Elapsed_Time_GJ)*100
PER_LU=(sum(Elapsed_Time_GJ) -
sum(Elapsed_Time_LU))/sum(Elapsed_Time_GJ)*100
PER_RQ=(sum(Elapsed_Time_GJ) -
sum(Elapsed_Time_RQ))/sum(Elapsed_Time_GJ)*100
%%
```