Sudan University of Science and Technology
College of Engineering
Electronics Engineering Department

Radio Resource Management In LTE

إدارة مصادر الراديو في شبكات الجيل الرابع

A Research Submitted In Partial fulfillment for the Requirements of the Degree of B.Sc. (Honors) in Electronics Engineering

Prepared By:

IBRAHIM BASHIR MOHAMED
ABD ELRAHEEN IDRIS FADLALLAH
SAMI OTHMAN ABD ALHAMEED

Supervised By:

Dr. IBRAHIM KIDDER ALTAHIR

October 2015

بسم الله الرحمن الرحيم

(تَبَارَكَ الَّذِي بِيَدِهِ الْمُلْكُ وَهُوَ عَلَى كُلِّ شَيْءٍ قَدِيرٌ { 1} الَّذِي خَلَقَ الْمَوْتَ وَالْحَيَاةَ
لِيَبْلُوَكُمْ أَيُّكُمْ أَحْسَنُ عَمَلًا وَهُوَ الْعَزِيزُ الْغَفُورُ { 2} الَّذِي خَلَقَ سَبْعَ سَمَاوَاتٍ
طِبَاقًا مَّا تَرَى فِي خَلْقِ الرَّحْمَنِ مِن تَفَاوُتٍ فَارْجِعِ الْبَصَرَ هَلْ تَرَى مِن فُطُورٍ {
3})

صدق الله العظيم

سورة الملك الآية (1-3)

الإهداء

إلهي لايطيب الليل إلا بشكرك ولايطيب النهار إلا بطاعتك .. ولاتطيب اللحظات إلا بذكرك ..
ولا تطيب الآخرة إلا بعفوك .. ولا تطيب الجنة إلابرؤيتكالله

لى من بلغ الرسالة وأدىالأمانة .. ونص جل جلاله .. إلى نبي الرحمة ونور العالمين

إلى من كلله الله بالهيبةوالوقار .. إلى من علمني ا محمد رسول الله انتظار .. إلى من أحمل أسمه بكل افتخار
..أرجو من الله أن يمد في عمرك لترى ثماراً قد حان قطافها بعد طول انتظار وستبقىكلماتك

Acknowledgement

In preparing this thesis, I was in contact with many people, researchers, Academicians and practitioners. They have contributed towards my understanding and thoughts. In particular, I wish to express my sincere appreciation to my main thesis supervisor, Dr. Ibrahim kidder, for encouragement, guidance, critics and friendship. I am also very thankful to my co-supervisors Dr. Fath Elrahman Ismael Without their continued support and interest, this thesis would not have been the same as presented here.

My fellow postgraduate students should also be recognized for their support. My sincere appreciation also extends to all my colleagues and others who have provided assistance at various occasions. Their views and tips are useful indeed. Unfortunately, it is not possible to list all of them in this limited space. I am grateful to all my family members.

Abstract

This thesis is based on the study of scheduling algorithms in LTE (Long Term Evolution). LTE is an evolution of the UMTS (Universal Mobile Telecommunications System) standardized by the 3GPP (3rd Generation Partnership Project) in its Rel. 8 for the development of wireless broadband networks with very high data rates. It enables mobile devices such as smart phones, laptops, tablets to access internet at a very high speed data along with lots of multimedia services. The future of LTE lies in being implemented in various electronic devices to exchange data wirelessly at very high speeds.

Technically, the Long Term Evolution provides a high data rate and can operate in different bandwidths ranging from 1.4MHz up to 20MHz. In terms of features the latest Release of LTE (Rel. 10 – LTE-Advanced) aims to deliver enhanced peak data rates to support advanced services and applications (100 Mbit/s for high and 1 Gbit/s for low mobility, low latency (10ms round-trip delay), improves system capacity and coverage, supports multi-antenna and reduces operating costs by introducing concepts like SON and allowing seamless integration with existing mobile network systems.

Scheduling is basically the process of making decisions by a scheduler regarding the distribution of resources (time and frequency) in a telecommunications system among its users. The Proportional Fair and the Round Robin scheduling algorithms have been considered and discussed in this dissertation. The analysis of these scheduling algorithms has been done through simulations executed on a MATLAB

The system simulation steps are programmed each in a separate mfile. These files contain a group of communication toolbox functions are used to represent different channels including LTE channels and to simulate transmitting and receiving processes.

.

المستخلص

في هذا البحث تمت دراسه خوارزميات توزيع المصادر في شبكات الجيل الرابع وتم اختيار خوارزميتين لتوزيع المصادر ودراسه كيفيه قيام اي من الخوارزميتين بتوزيع المصادر .

وتم تمثيل تلك الخوارزميات باستخدام برنامج محاكاه وقورنت الخوارزميتين من حيث سرعه توزيع المصادروكذلك الانصاف في توزيعها بين المستخدمين

TABLE OF CONTENTS

List of Tables

List of Figures

Abbreviations

3GPP	3rd Generation Partnership Project
LTE	Long Term Evolution
HSPA	High Speed Packet Access
3G	Third Generation of Cellular Wireless Standards
GSM	Global System for Mobile Communication
UMTS	Universal Mobile Telecommunications System
UTRA	UMTS terrestrial radio access
E-UTRA	Evolved UMTS terrestrial radio access
UTRAN	UMTS Terrestrial Radio Access Network
E-UTRAN	Evolved UMTS Terrestrial Radio Access Network
MIMO	Multiple Input Multiple Output
FDD	Frequency Division Duplex
TDD	Time Division Duplex
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
SC-FDMA	Single Carrier Frequency Division Multiple Access
FDMA	Frequency Division Multiple Access
BS	Base Station
eNodeB	Base Station
MS	Mobile Station
UE	User Equipment
RB	Resource Block
RE	Resource Element
SNIR	Signal to Noise-Interference Ratio
RR	Round Robin
PF	Proportional Fair
CQI	Channel Quality Indicator
DL	Downlink
UL	Uplink
HSDPA	High Speed Downlink Packet Access
EUL	Enhanced Uplink
SC	Single Carrier
SISO	Single Input Single Output
MME	Mobility Management Entity
CP	Cyclic Prefix
HARQ	Hybrid Automatic Retransmission Request
RS	Reference Signal

CIR	channel impulse response
PRN	pseudorandom number
WCDMA	Wideband Code Division Multiple Access
PHY	Physical layer
MAC	Medium Access Control
RLC	Radio Link Control
RRC	Radio Resource Control
IEEE	Institute of Electrical and Electronics Engineers
4G	Fourth Generation of Cellular Wireless Standards
SNR	Signal to Noise Ratio
PS	Packet Scheduler
TTI	Transmission Time Interval
MCS	Modulation and Coding Scheme
QoS	Quality of Service
AMC	Adaptive modulation and coding
PRBs	Physical Resource Blocks
RRM	Radio Resource Management
CCI	co-channel interference
VoIP	Voice over Internet Protocol
QPSK	Quadrature Phase-Shift Keying
QAM	Quadrature Amplitude Modulation

Chapter One
Introduction

Chapter One

Introduction

1.1 Preface

Long Term Evolution (LTE) is the name given to a 3GPP project to evolve UTRAN to meet the needs of future broadband cellular communications. This project can also be considered as a milestone towards 4G standardization. Different organizations and individuals are involved in this project to specify requirements of LTE which satisfies both operators and consumers. Till the time of writing, it was in the standardization phase and many of its specifications have been standardized and many companies like Ericsson and Nortel have developed a prototype of LTE just to demonstrate the effectiveness of Long Term Evolution.

Following are a few of the requirements on this newly evolving cellular technology ,Long Term Evolution (LTE) One of the main requirements is the transition of circuit-switched (CS) and packet-switched (PS) networks into an all-IP network which can support different types of services with different QoS and which also provide the easy integration with the other communication networks. This will ultimately reduce the integration cost and provide with the users the seamless integration with other services.

Support of scalable bandwidth i.e. 1.25, 2.5, 5, 10, and 20 MHz The subscribers can be assigned bandwidth as low as 1.25 MHz or as high as 20 MHz and it may also be aggregate assigned the bandwidth of above bands.

Peak downlink data rates :Users may attain the instantaneous downlink data rate as high as 100 Mbps while provided 20 MHz bandwidth and uplink data rate of 50 Mbps while provided with 20 MHz bandwidth. Latency of 50-100 msec for Control-plane and less than 10 msec for User-plane.

Optimized mobility for speed of less than 15 km/hr, high performance mobility for speed up to 120 km/hr and mobility support for speed up to 350 km/hr. Coverage with full performance up to 5km and with slight degradation in performance for coverage up to 30km and support of coverage of up to 100 km. Control Plane Capacity At least 200 users per cell should be supported in active state for allocation of 5MHz spectrum.

Multi-antenna configuration the multi-antenna configuration will significantly improve the system performance and service capability and it would be used to achieve the transmit diversity, multi-stream transmission, and beam forming. Radio resource management attracts great attention while utilizing available resources to provide users with enhanced system throughput. Radio resources management include transmission power management, mobility management, and scheduling of radio resources. An intelligent radio resource management is at the heart of LTE to make it a robust technology to meet the broadband mobility needs of upcoming years. This will schedule the available resource in a best way and provide to the users with the enough transmission capability to achieve the decided QoS even while they move freely and also will make sure that these assigned resources would not interfere with already assigned resources. This will also be of interest that the transmitted signal will reach the receiver in a good health while utilizing the power efficiently available at the transmitter.

1.2 Problem statement

Select efficient scheduling algorithm that provide high data rate, throughput and spectral efficiency.

1.3 Thesis goals

The main purpose of this thesis is to verify and compare selected scheduling algorithm in Matlab. The simulation part of the thesis enables us to understand the scheduling algorithms for the LTE networks in much more detail and gain experience in modeling and simulation of such networks in detail.

1.4 Methodology

Round robin scheduling and Proportional fair scheduling have been simulated in a MATLAB. The system simulation steps are programmed each in a separate mfile. These files contain a group of communication toolbox functions are used to represent different channels including LTE channels and to simulate transmitting and receiving processes. The performance of these scheduling algorithms in terms of throughput, data rate and spectral efficiency is analyzed.

1.4 Research Outlines

Chapter 1 includes introduction to the Radio Resource manage in LTE.
Chapter 2 Discuss Technical background of Long Term Evolution (LTE) and review several scheduling algorithms.
Chapter 3 contains the methodology of comparing scheduling algorithm.
Chapter 4 includes simulation scenario and discuss the simulation results.
Chapter 5 contains conclusions and future work.

Chapter Two

Background

&

Literature Review

Chapter Two

Background & Literature Review

2.1 Introduction

Long Term Evolution (LTE)has evolved from an earlier 3GPP system known as the Universal Mobile Telecommunication System. LTE is a standard for wireless communication of high-speed data for mobile phones and data terminals. LTE is having very high speed data rate with the help of which user can access the internet through their mobiles and results in high data throughput. LTE is self organized network . LTE aims to provide higher data rate, packet optimized radio access, low latency and flexible bandwidth deployment. LTE uses Orthogonal Frequency Division Multiplexing (OFDM) for the downlink that is, from base station to the terminal to transmit the data over many narrow band carriers each of 180 KHz. Orthogonal frequency-division multiplexing (OFDM), is a frequency-division multiplexing (FDM) scheme used as a digital multi-carrier modulation method . OFDMA can support flexible radio resource management. LTE uses a pre-coded version of OFDM called Single Carrier Frequency Division Multiple Access (SC-FDMA) in the uplink [2].

2.2 Multiple Access Techniques

3GPP LTE have selected different transmission schemes in uplink and downlink due to certain characteristics. OFDMA has been selected for downlink i.e. from eNodeB to UE and SC-FDMA has been selected for uplink i.e. for transmission from UE to eNodeB.

2.2.1 Uplink – SC-FDMA

Single Carrier – Frequency Division Multiple Access(SC-FDMA) has been selected as 3GPP LTE uplink transmission technique (MS to eNodeB). It is a modified form of OFDMA and has similar throughput performance and essentially the same overall Complexity as OFDMA. Like OFDM, SC-FDMA also consists on subcarriers but it Transmits on subcarriers in sequence not in parallel which is the case in OFDM , which Prevents power fluctuations in SC-FDMA signals , SC-FDMA signals might cause inter symbol interference when they reach at the base station. The base station uses the adaptive frequency domain equalization to cancel the inter symbol interference [4].

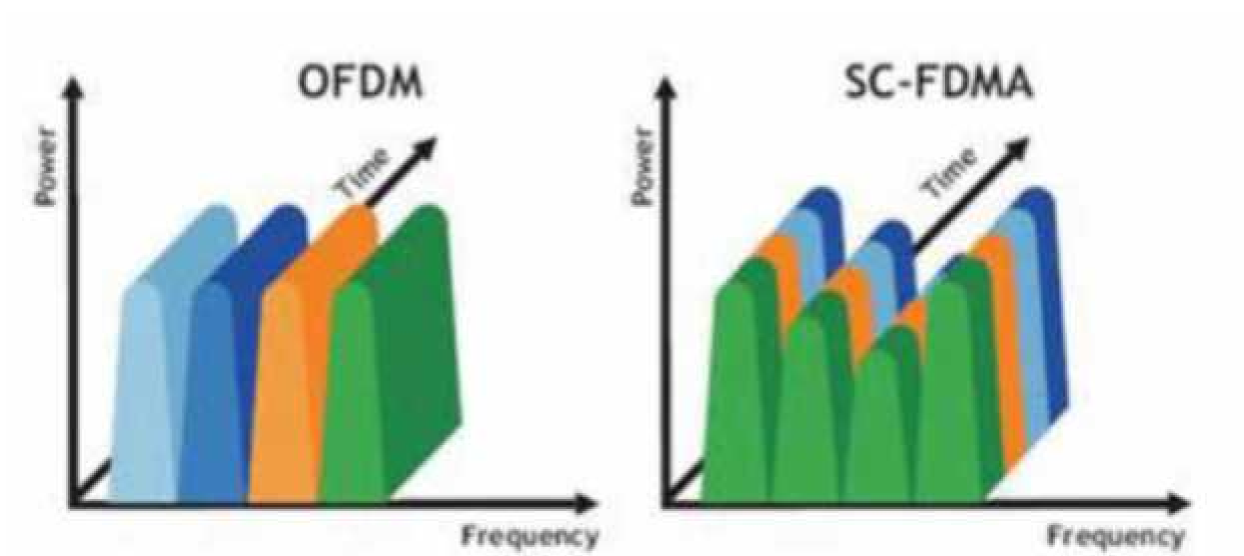


Figure 2.1 OFDM and SC-FDMA [5]

2.2.2 Downlink_OFDMA

OFDMA is an extension of the OFDM transmission scheme by allowing multiple number of users. In OFDM the user data is transmitted in parallel across multiple orthogonal narrow subcarriers. Each subcarrier transports a small part of the whole transmission. The orthogonal subcarriers are generated with Inverse Fast Fourier Transform (IFFT) processing .The number of subcarriers depends only on the available bandwidth. OFDM meets the LTE requirement for spectrum flexibility and enables cost-efficient solutions for very wide carriers with high peak rates. The basic LTE downlink physical resource can be seen as a time-frequency grid. Basically To overcome the effect of multiple path fading problems available in UMTS, LTE uses OFDM for the downlink - that is, from the base station to the terminal to transmit the data over many narrow band careers of 180 KHz each instead of spreading one signal over the complete 5MHz career bandwidth. OFDM uses a large number of narrow sub-carriers for multi-carrier transmission to carry data. [2].

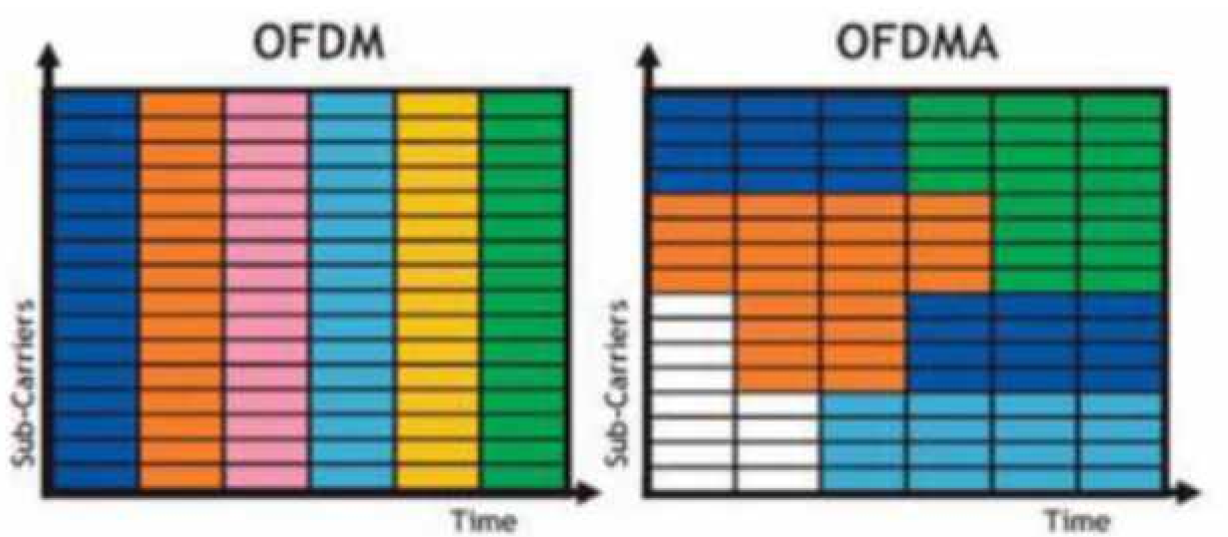


Figure 2.2 OFDM and OFDMA [5]

2.3 Physical Layer Frame Structure

The radio resource block can be seen as a frequency-time grid. Frequency domain is divided into sub-carriers where each sub-carrier spans 15 KHz. One sub-band is comprised of 12 sub-carriers. Time domain can be divided into slots which has duration of 0.5ms. One sub-frame consists on 2 time slots and has duration of 1ms and one frame is consisted on 10 sub-frame and thus it spans for 10ms (10 * 2 * 0.5ms). Minimized radio resource block that can be allocated on both uplink and downlink is called sub-band and contains 12 sub-carriers transmitted in one time slot (0.5ms). Thus, minimum allowable spectrum is 180 KHz [3].

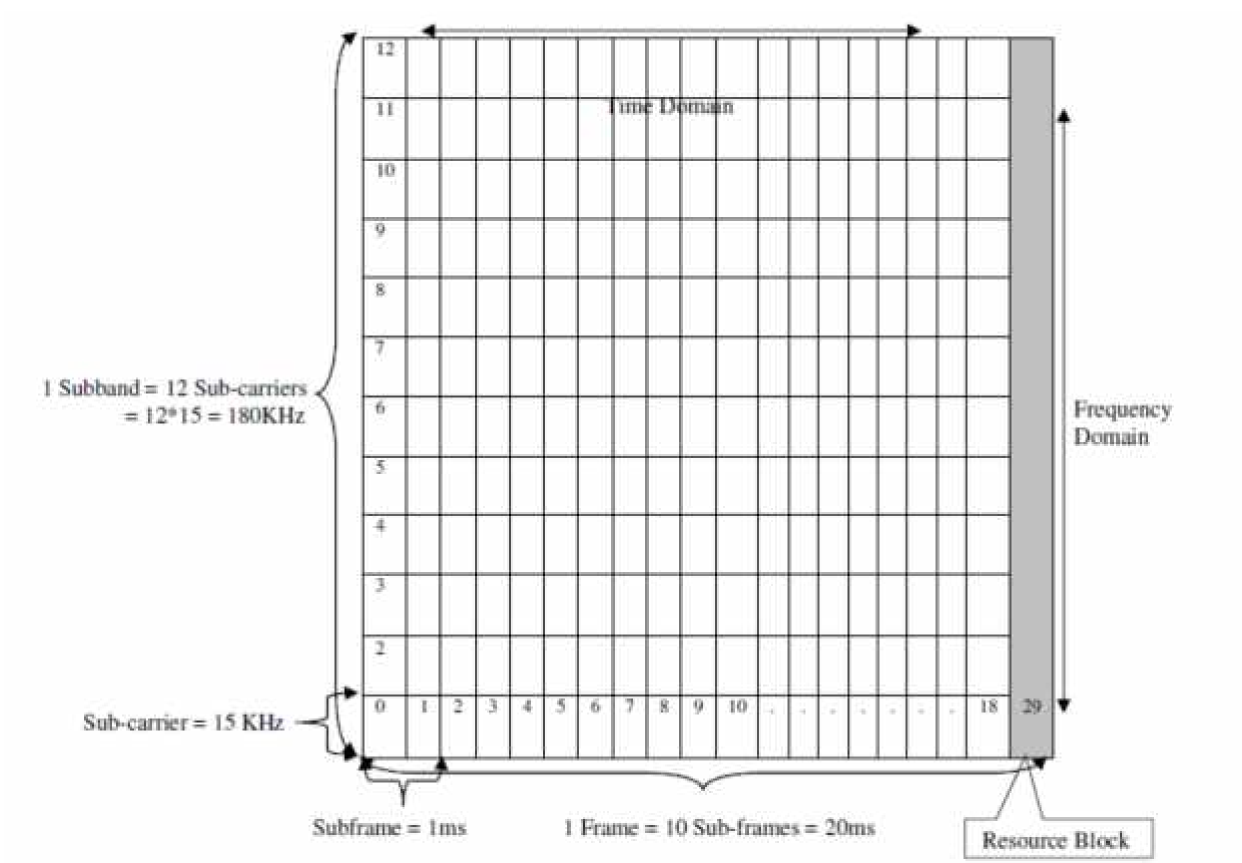


Figure 2.3 LTE frame structure [3]

2.4 Radio Resource Scheduling in LTE

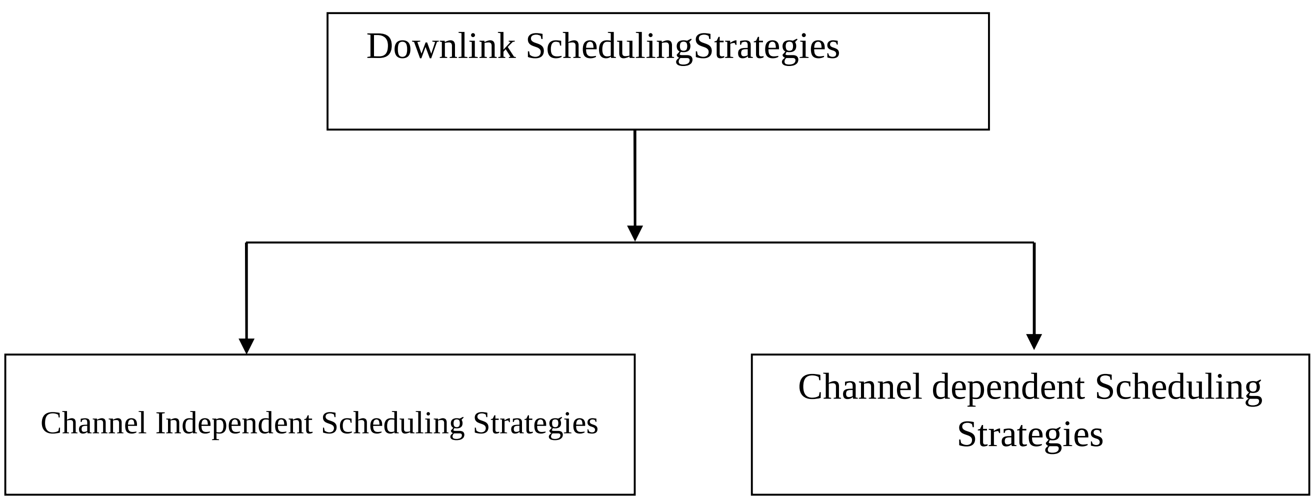
Scheduling plays a great role in the resource allocation. Resources are allocated to the user according to their need and priority. LTE is simple and easy to use which is having higher privacy and security. It improved the speed and data rate. OFDM is used by LTE for downlink transmission where OFDM divide the bandwidth into multiple narrower sub-carries and data is transmitted on these carries in parallel streams. In OFDM the subcarrier is modulated with different modulation method like QPSK, QAM, 64QAM and the use of higher order modulation such like 16QAM and 64QAM provides the higher bandwidth utilization and high data rate, within a particular bandwidth an OFDM symbol is obtained by adding different modulated subcarrier signals In downlink of an OFDMA the resources are divided in the frequency and time domains. In frequency domain the resources are divided into N traffic channels which are a cluster of OFDM subcarriers. Whereas in the time domain the resources are divided into slots called frames and super frames. OFDM is used in other of systems like WLAN, WIMAX to broadcast technologies [2].

2.5 Scheduling Algorithms

Generally, scheduling can be divided into two classes: channel-independent scheduling and channel-dependent scheduling. The performance of channel-independent scheduling can never be optimal in wireless networks due to varying nature of instantaneous channel conditions. On the contrary, channel-dependent scheduling can achieve better performance by allocating resources based on channel conditions with optimal algorithms. [1].

2.5.1 Down link scheduling algorithms

In this sub-section, gived an art's state of the well-known scheduling algorithms families for the LTE downlinks side.



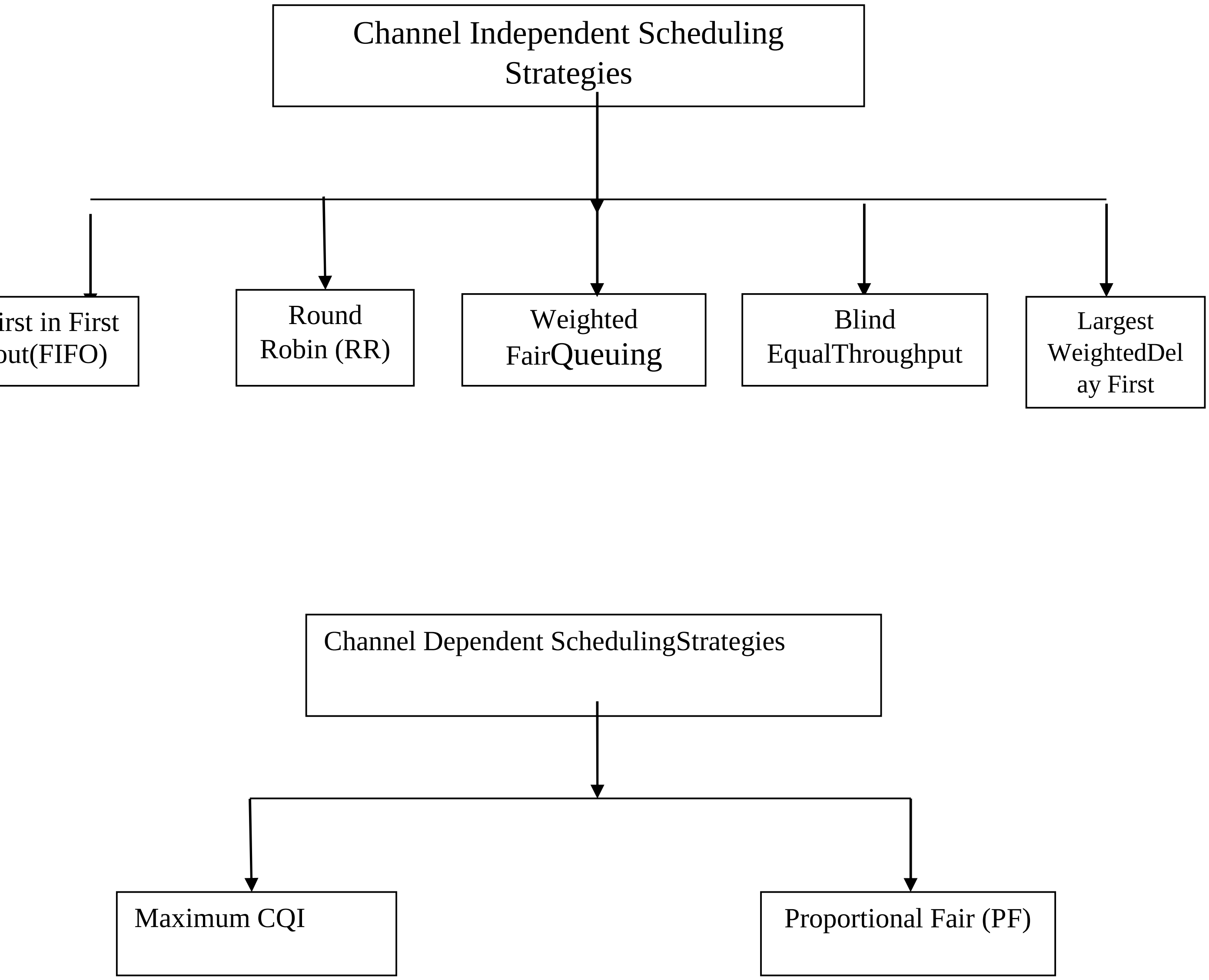


Figure 2.4 Downlink scheduling algorithm

A- Channel Independent Scheduling Strategies Channel independent strategies were firstly introduced in wired networks and are based on the assumption of time invariant and error-free transmission media. Being unrealistic for LTE networks, they are typically used in conjugation with channel-dependent strategies to improve system performance.

2.5.2First in First out (FIFO)

Though FIFO is the simplest of all possible scheduling disciplines but it is inefficient and unfair. This scheduler serves the packets in the queue in order of arrival and when the queue is full, it drops the packets that are just arriving. The major setback is that it cannot differentiate among connections; therefore all packets experience the same delay,and packet loss irrespective of which packet it is[1].

2.5.3Round Robin (RR) Scheduling Algorithms

Round Robin is the simplest scheduling algorithm used by the CPU during execution of the process. Resources are allocated to each user without using channel condition . Each user can use the resources in proper time interval. First user can use the resource for the given time interval after the completion of time then these resources is assigned to another user. The new user has placed at the end of waiting queue. The implementation of RR is simplest, easy and good in fairness but its throughput is not good[1].

2.5.4 Weighted Fair Queuing

In Weighted Fair scheduling the packets are grouped into various queues. a weight is assigned to each queue which determines the fraction of the total bandwidth available to the queue . To assure that flows with larger packets are not allocated more bandwidth than flows with smaller packets, it also supports variable-length packets. The Weighted Fair scheduling assigns the bandwidth for each service based on the weight assigned to each queue and not based on the number of packets[1].

2.5.5 Blind Equal Throughput

The Blind equal throughput (BET) is a channel unaware strategy that aims at providing throughput fairness among all the users. To counteract the unfair sharing of the channel capacity, the BET scheduler uses a priority metric which considers past average user throughput. BET scheduler prioritizes users with lower average throughput in the past.

This implies that users with bad channel conditions are allocated more resources compared to the users with good channel conditions. Thus throughput fairnessis achieved but at the cost of spectral efficiency[1].

2.5.6 Largest Weighted Delay First

To avoid packet drops, it is required that each packet has to be received within a certain delay deadline in Guaranteed delay services. It incorporates the information about the specific packet timing.

Similar to Round Robin, neglecting channel conditions leads to poor throughput in LWDF[1].

2.6 B-Channel Dependent Scheduling Strategies

2.6.1 Maximum CQI Scheduling Algorithm

The highest value of CQI (channel quality indicator) means that the channel quality is good. It provides good throughput but not good in fairness. In this resources are assigned to the user according to the link quality. During scheduling the terminals which are located far away from the base station are not scheduled and nearby terminals are scheduled by sending CQI to the base station[1].

2.6.2 Proportional Fair (PF) Scheduling Algorithm

The most commonly used scheduling algorithm is PF. PF results in good fairness and high throughput. Channel condition is calculated in this case and then resources are allocated to user which is having the highest priority and further the allocation is done to next priority user. This allocation is continuing until all the resources are allocate to the user. The scheduler can exercise Proportional Fair (PF) scheduling allocating more resources to a user with relatively better channel quality. This offers high cell throughput as well as fairness satisfactorily [1].

2.7 Uplink scheduling algorithms

In this sub-section, we will give an art's state of the well known scheduling algorithms families for the LTE uplink side.

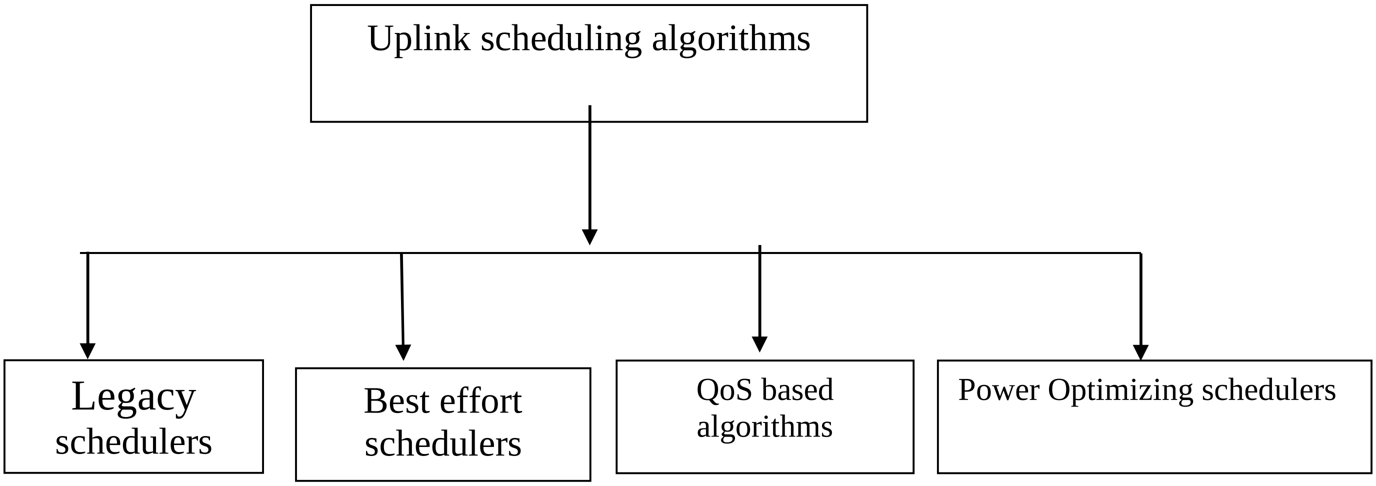


Figure 2.5 Uplink scheduling algorithms

2.7.1 Legacy schedulers

This family contains the famous classical algorithm, the Round Robin algorithm; it is also called the base scheduler's family, the RR algorithm has been used in many old systems. The Round Robin algorithm principle is to divide the available RBs into groups of RBs .Then, distribute the formed groups among available UEs[1].

2.7.2Best effort schedulers

The main objective of this category is to maximize the utilization of the radio resource and the equity in the system. It doesn't mean that this category treat only best effort flows, best effort schedulers means it is a greedy algorithm that try to do the better that it can. As we have already said, each algorithm has an objective function to optimize, this type of algorithm uses the PF metric. Several algorithms have been proposed in this family, we noted that the greedy algorithms are very suitable for this kind of traffic. The principle of greedy algorithm is that the RBs are grouped into RCs, with each RC containing a set of contiguous RBs. After that each RC gets allocated to the UE having the highest metric in the matrix, the RC and UE will be removed from the available RC list and UE schedulable list. The algorithm aims to maximize the fairness in resource allocation among UEs[1].

2.7.3QoS based algorithms

Two important elements must be taken into account by this scheduler's family, the maximum delay and the throughput. Also the algorithm must offer the required QoS parameters for each user regarding to the already served users. The Proportional Fair with Guaranteed Bit Rate (PFGBR) is a QoS based algorithm, From its name, we identify two metrics, PF and GBR, the PF metric is used to schedule the UEs with non GBR flows and for those with GBR flows , the algorithm changes the metric in order to differentiate the EU (giving priorities for UEs handling GBR streams). This algorithm has two objectives, maximizing the fairness of non GBR flows and preserves the QoS of GBR[1].

2.7.4 Power-Optimizing schedulers

The main purpose of this class of algorithms is to minimize the transmitted signal power trying to extend the duration activity of UE. In fact, it coincides with the objective of using SC-FDMA method. Schedulers of this family usually have some QoS treatments, so they perform some decisions to reduce the transmitted power till maintaining the minimal QoS requirements [4].

Chapter Three
Methodology

Chapter Three

Methodology

3.1Methodology

In general terms scheduling is basically the process of making decisions by a scheduler regarding the assignment of various resources (time and frequency) in a telecommunications system between users. In LTE the scheduling is carried out at eNodeB by dynamic packet scheduler (PS) which decides upon allotment of resources to various users under its coverage as well as transmission parameters including modulation and coding scheme (MCS).LTE defines 1 mssubframes as the Transmission Time Interval (TTI) resulting in the scheduling of resources every 1 ms . It means after every 1 ms the assignment of resources could change depending upon various factors including CQI (Channel Quality Indicator) sent as a feedback by the user to the eNodeB. In LTE networks, the role of resource scheduling is very important because great performance gain can be achieved by properly observing the amount of radio resources assigned to each user. As the 3GPP hasn't standardized any scheduling algorithm, we are free to choose and implement any algorithm that would meet the expected our QoS. While choosing or designing a scheduling algorithm many factors such as expected QoS level, the behavior of data sources, and the channel status have to be kept in mind. The problem becomes more complex in the presence of users with different requirements in term of bandwidth, tolerance to delay, and reliability.

3.2 Selected Scheduling Algorithms

There are various scheduling methods that have been developed over time to enhance the process of scheduling. But in this thesis, we shall be concentrating on particularly two algorithms which have been implemented in the software environment provided for testing by IS-Wireless. Among them are:

- Round Robin,
- Proportional Fair.

3.2.1 Round Robin Scheduling

This scheduling method is based on the idea of being fair in the long term by assigning equal no. of Physical Resource Blocks (PRBs) to all active UEs. It operates by assigning the PRBs to UEs in turn i.e one after another without taking into account their CQI. Hence the users are equally scheduled. For e.g. If we have 4 users U1, U2, U3, U4 and PRBs, this algorithm will assign the resources in the following manner: U1, U2, U3, U4, U1, and U2. It can be illustrated by the following flow chart:

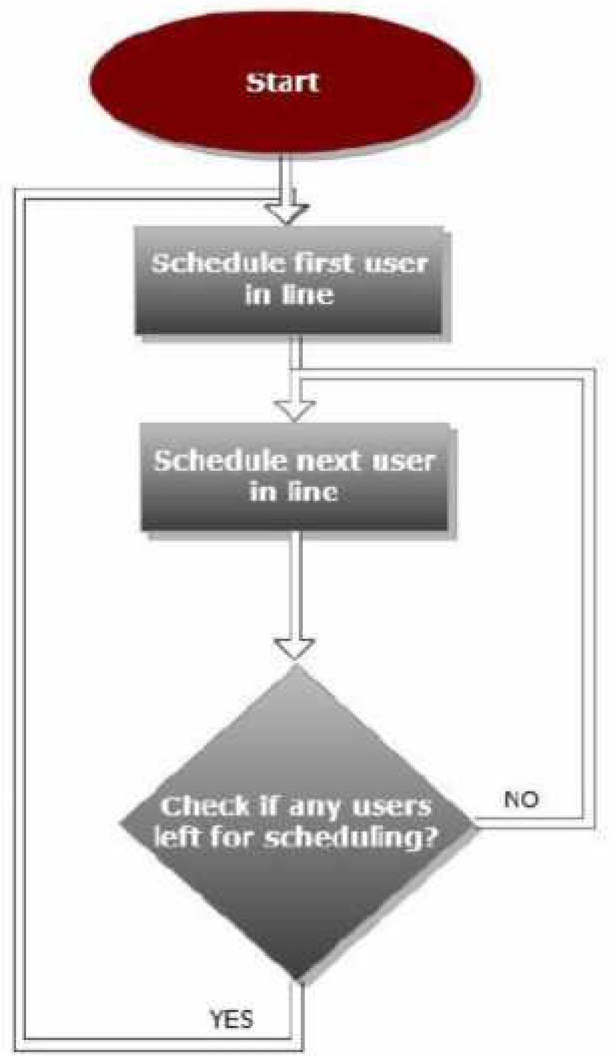


Figure 3.1 Flow Chart for Round Robin Algorithm

3.2.2 Proportional Fair (PF) Scheduling Algorithm

This algorithm assigns the PRBs to the UE with the best relative channel quality i.e. a combination of CQI & level of fairness desired. There are various versions of PF algorithm based on values it takes into account. Main goal of this algorithm is to achieve a balance between Maximizing the cell throughput and fairness, by letting all users to achieve a minimum QoS (Quality of Service).

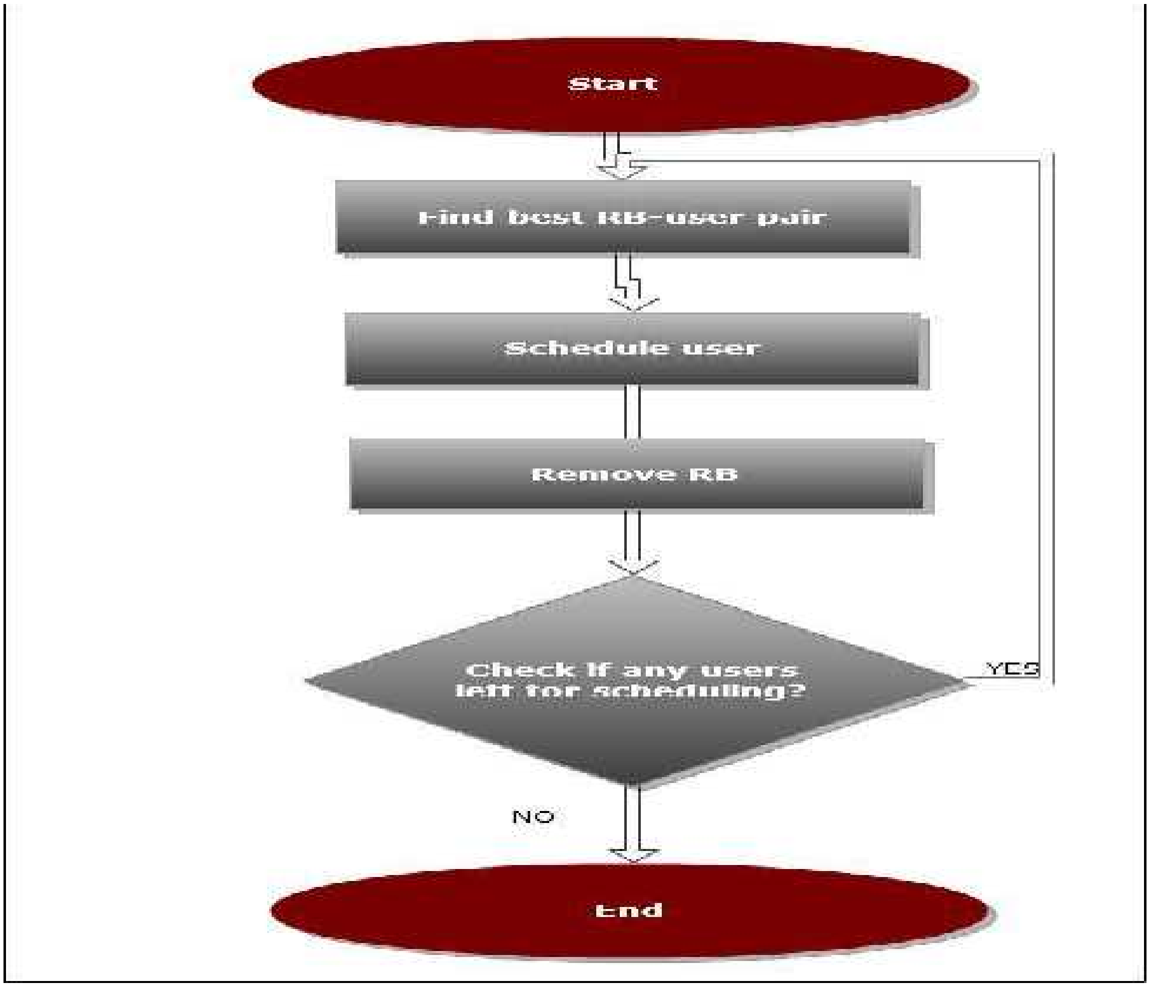


Figure .3.2 Flow chart for Proportional Fair Algorithm

The above Fig. 5 depicts one of the possible methods of implementing proportional fair algorithm. Such an algorithm is designed to be better in terms of average user throughput as well as being fair to most of the users and meeting the minimum QoS requirements during the scheduling process.

Chapter Four

Results and Discussion

Chapter Four

Results and Discussion

4.1 Simulations and Discussion

This chapter is meant for the analysis of the scheduling algorithms we have discussed in the earlier section by means of simulations and practical experiments. The analysis has been carried out by comparing the throughput, data rate and spectral efficiency for different scenarios (different scheduling schemes, different number of RB and different number of users, different SNR system). The system simulation steps are programmed each in a separate mfile. These files contain a group of communication toolbox functions are used to represent different channels including LTE channels and to simulate transmitting and receiving processes. Then graphical representations of the performance of these scheduling algorithms in terms of throughputs, data rate and spectral efficiency are plotted.

4.2 Simulation Scenarios

In order to verify and compare the scheduling algorithms with help of matlab, we have selected no. of scenarios. These scenarios are meant to help us understand the working of the scheduling algorithms. We investigate the performance of the scheduling algorithms in terms of data rate and throughput and spectral efficiency for different scenarios (different number of RB, different SNR and different number of users).The scenarios have been selected to analyze the impact of the scheduling algorithms in different conditions, hence understand their functioning in much more detail.

4.3 Simulation Results and Analysis

Simulation results are presented along with their analysis. During the simulations we will be set some basic default parameters which are depicted below:

Table 4.1 summary of simulation parameters used for all the testing scenarios

Parameters	**Value**
Number of Equipment (UEs)	3
Number of base station	1
Bandwidth	1.4 MHz
Channel type	AWGN
Scheduling algorithms	Round Robin and Proportional Fair
Simulation length	1 frame
Transmission Scheme	MIMO 1*2

Case 1: SimulateDifferent No. Of Users LTE Systems

In this first case we simulate RR and PF and we show the data rate and throughput and spectral efficiency for different number of users.

In Figure 4.1 different number of user's case, the data rate of RR increase linearly. on the other hands PF remain at constant value 7Mbps

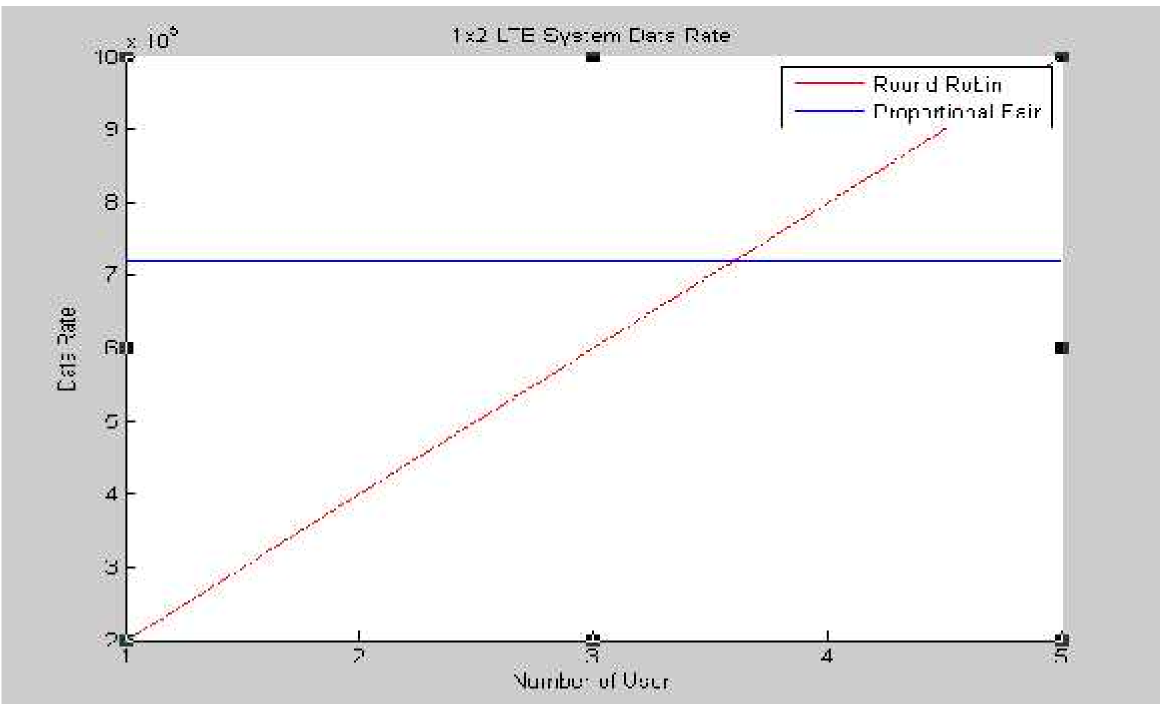


Figure4.1 comparison of RR and PF in term of data rate based on varies NO. of users

In Figure 4.2 different number of user's case, the throughput of RR increases linearly. on the other hands PF throughput increase by increasing number of users .after three users the throughput becomes constant.

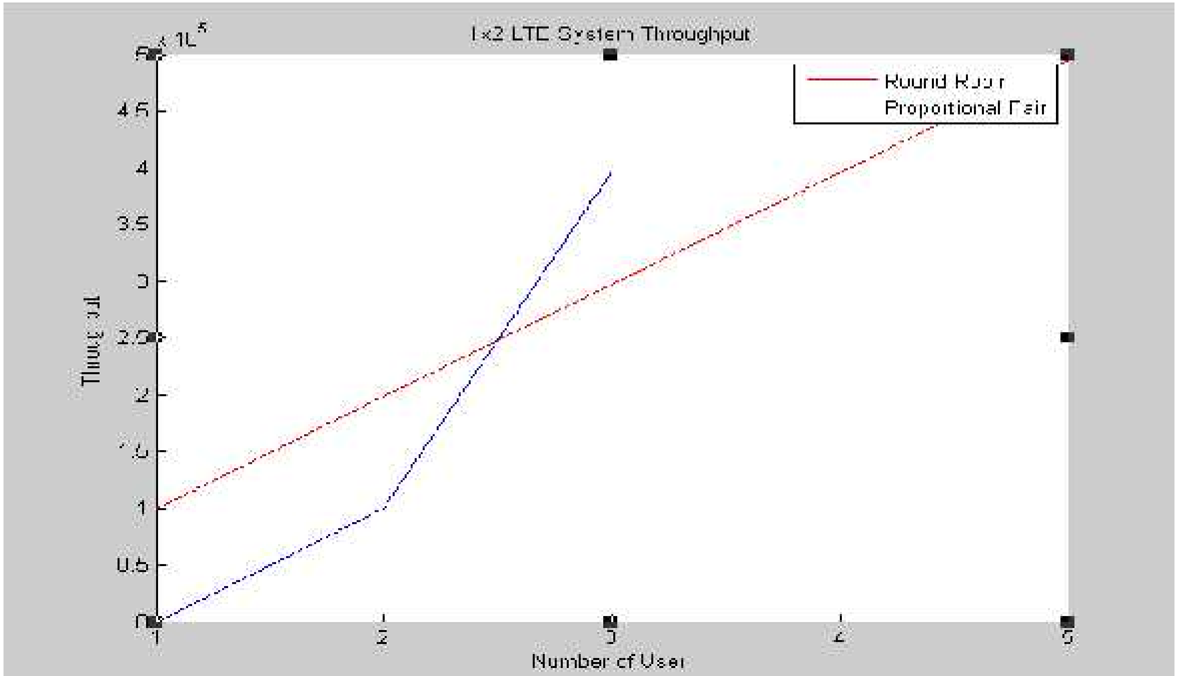


Figure 4.2 comparison of RR and PF in term of Throughput based on varies NO. of users

In Figure 4.3 different number of user's case, the spectral efficiency of RR increase linearly. on the other hands PF remain at constant value.

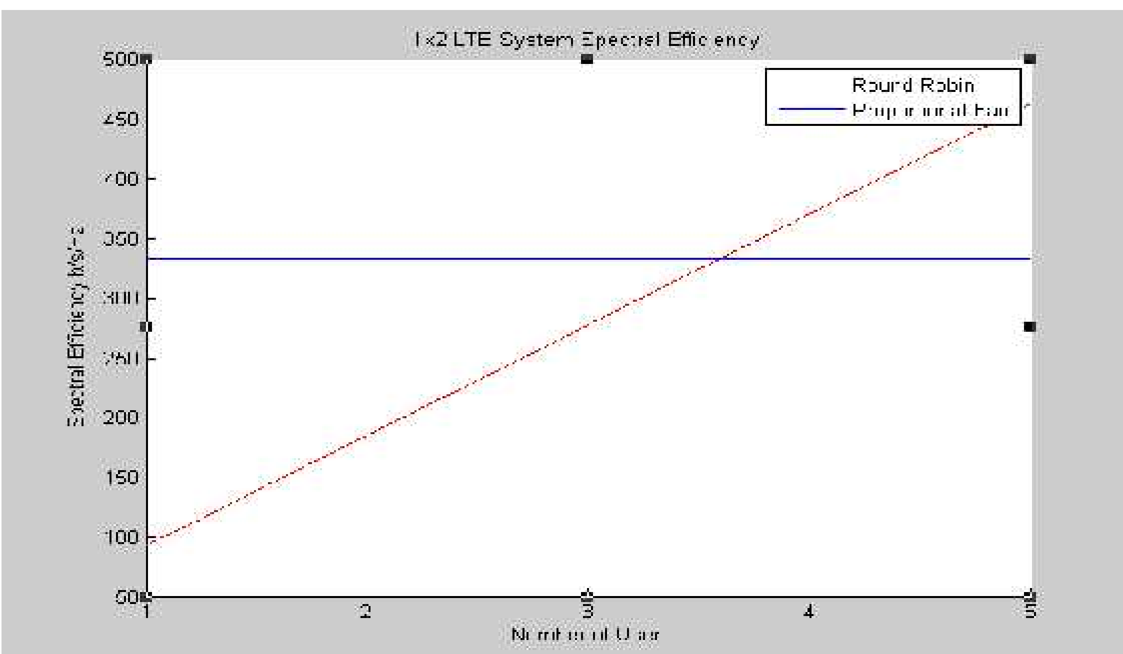


Figure4.3 comparison of RR and PF in term of spectral efficiency based on varies NO. of users

Case 2: Simulate Different BW LTE Systems

In this case we simulate RR and PF and we show the data rate and throughput and spectral efficiency for different number of RB bandwidth.

In Figure 4.4 different system bandwidth or different NO of resource blocks. In Both RR and PF the data rate increase linearly and the performance of RR is near to PF.

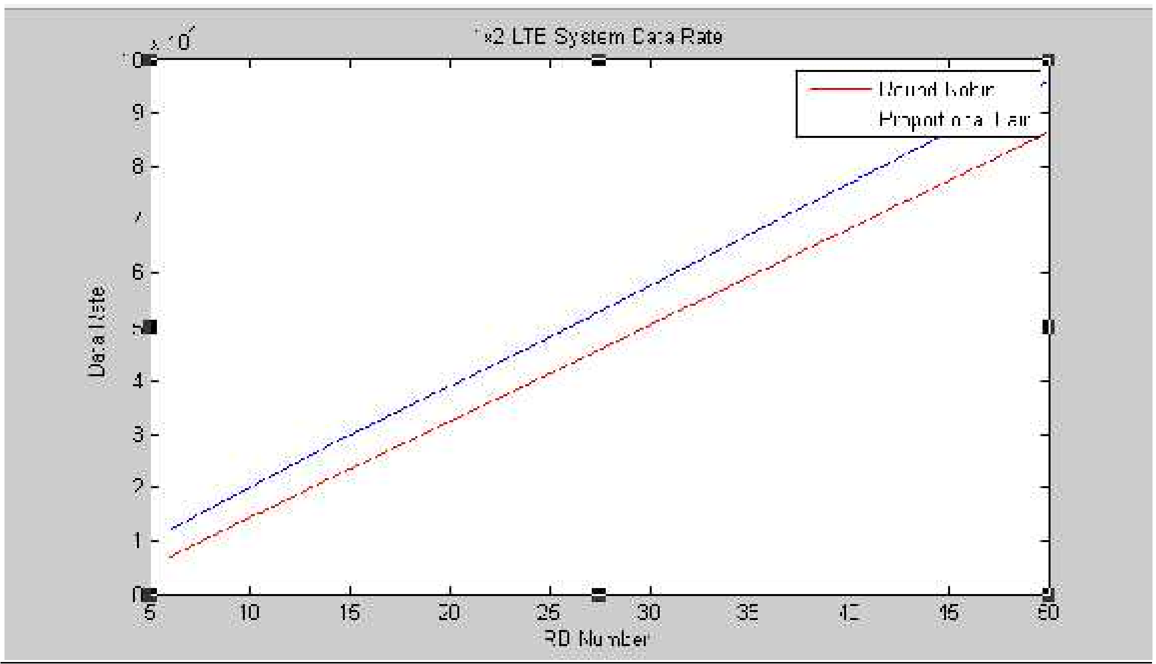


Figure 4.4comparison of RR and PF in term of data rate based on varies RB number

In Figure 4.5 different system bandwidth or different NO of resource blocks. In Both RR and PF the throughput increase linearly and the performance of RR is near to PF.

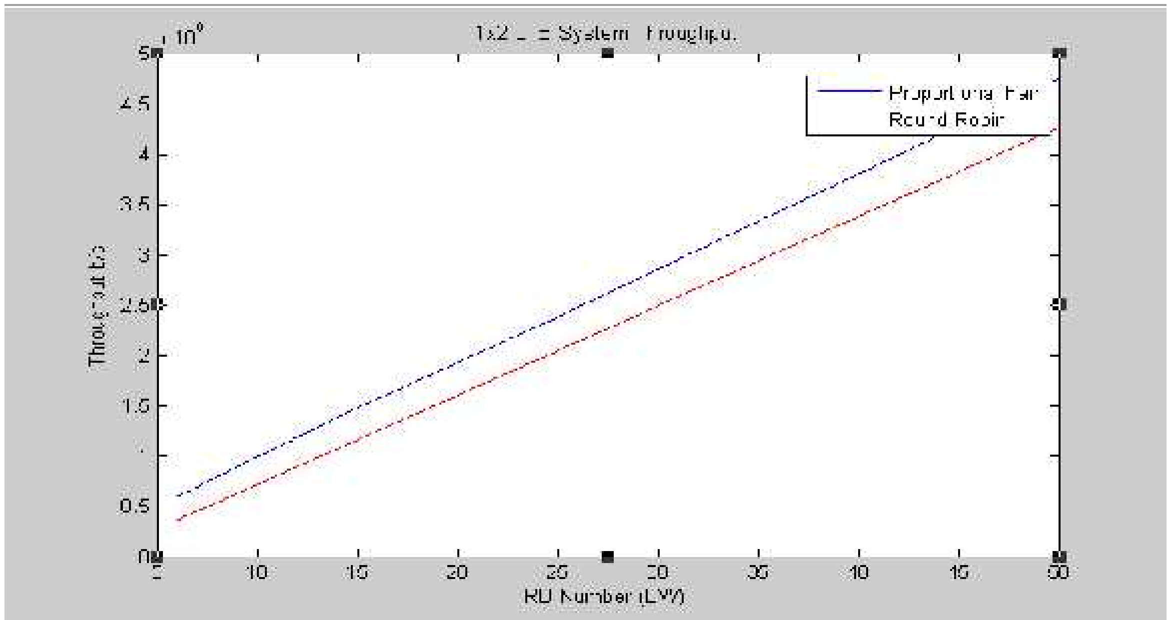


Figure4.5 comparison of RR and PF in term of throughput based on varies RB number

In Figure 4.6 different number of resource blocks. RR spectral efficiency increase by increase NO of RB .but PF decrease by increase NO of RB .

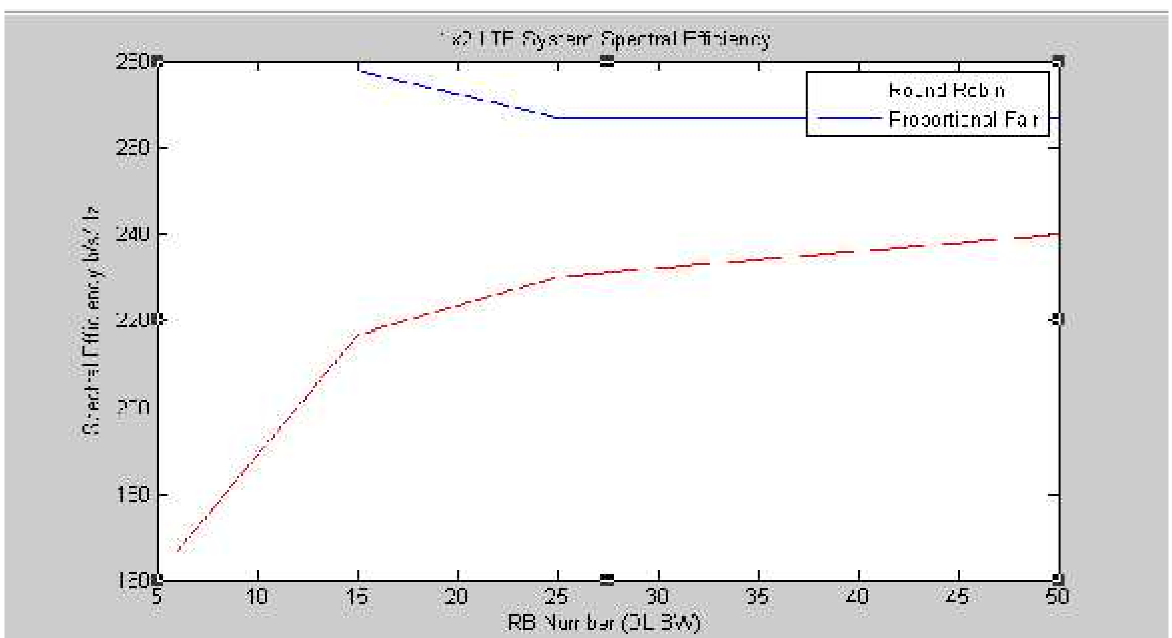


Figure4.6 comparison of RR and PF in term of spectral efficiency based on varies RB number

Case 3 : Simulate Different SNR LTE Systems

In this case we simulate RR and PF and we show the data rate and throughput and spectral efficiency for different SNR LTE systems.

In Figure 4.7 different system SNR in Both RR and PF the data rate increase by increasing SNR and the performance of RR very far from PF.

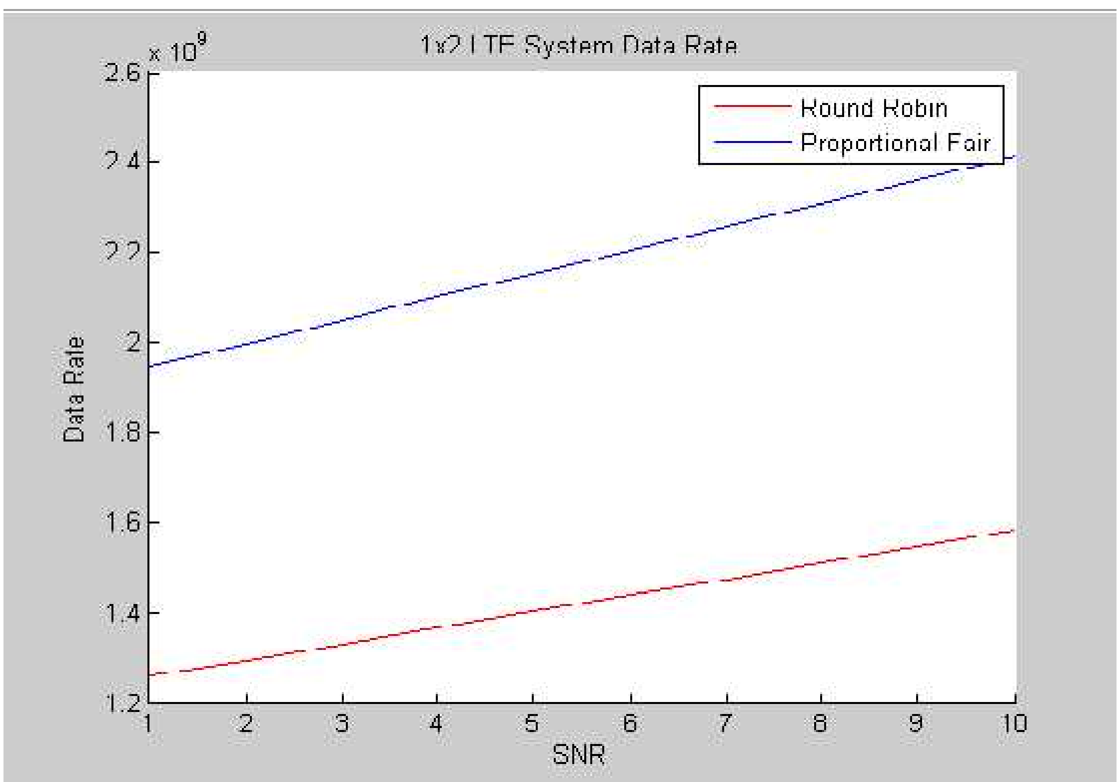


Figure4.7comparison of RR and PF in term of data rate based on varies SNR

In Figure 4.8 different system SNR in Both RR and PF the throughput increase by increasing SNR and the performance of RR very far from PF.

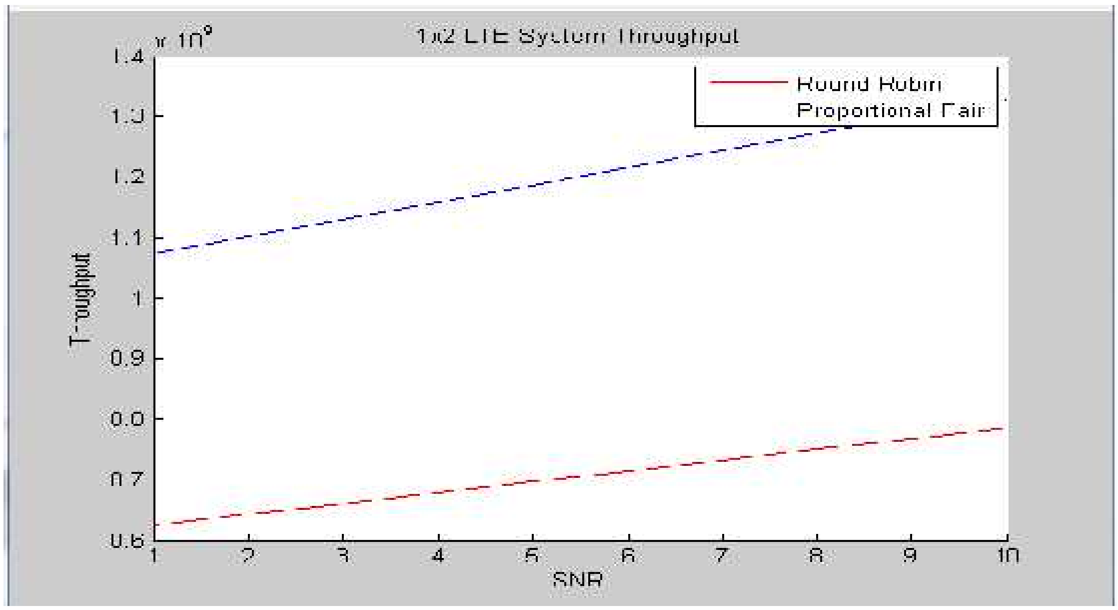


Figure4.8comparison of RR and PF in term of Throughput based on varies SNR

In Figure 4.9 different system SNR in Both RR and PF the spectral efficiency increase by increasing SNR and the performance of RR very far from PF.

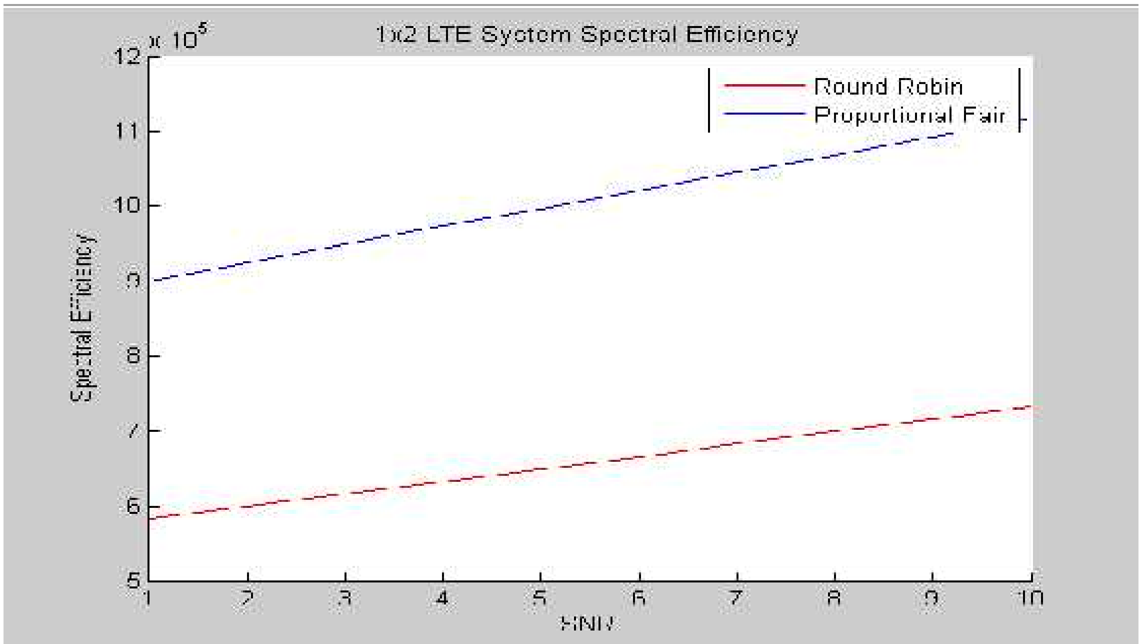


Figure4.9comparison of RR and PF in term of spectral efficiency based on varies SNR

From the results It is clear that the Round Robin scheduling performs worst since it results in lower data rate, throughput and spectral efficiency in different cases although its spectral efficiency increases with the increment of the users. From another side, it offers higher resource allocation fairness among different users.

Chapter Five

Conclusion

&

Recommendation

Chapter Five

Conclusion & Recommendation

5.1 Conclusion

In this thesis I have done a detailed study of the scheduling algorithms. These algorithms include Round Robin Scheduling and Proportional Fair Scheduling. The study was followed by the modeling and simulation of these scheduling algorithms using MATLAB The system simulation steps are programmed each in a separate mfile. These files contain a group of communication toolbox functions are used to represent different channels including LTE channels and to simulate transmitting and receiving processes. From the results of simulation It is clear that the Round Robin scheduling performs worst since it results in lower data rate, throughput and spectral efficiency in different cases although its spectral efficiency increases with the increment of the users. From another side, it offers higher resource allocation fairness among different users.

5.2 Recommendations

As 3GPP LTE proposes that the radio resources should be scheduled every 1ms which is also called TTI in scheduling, this will place a lot of processing on the eNodeBs. The ways to speed up this scheduling process must be found. 3GPP LTE also offers the use of higher order modulation schemes like 64QAM which will enhance the system throughput to a greater extent but it will also place a lot of processing on both the eNodeB as well as the UE. Investigating its impact should also be beneficial. Utilization of higher order modulation techniques may also suffer from the noise created by the processing in both ends of the transmission. So, this noise behavior should also be investigated.

References

[1] davinder and preeti Radio Resource Scheduling in 3GPP LTE: A Review,June 2013

[2] Shampy and others Scheduling algorithms of Resource allocation in LTE system, June 2014

[3] Sajid Hussain, Dynamic Radio Resource Management in 3GPP LTE, Blekinge Institute of Technology, January 2009

[4] fayssal bendaoud and others Survey on Scheduling And Radio Resource Allocation in LTE,March 2014

[5] Dinesh Mannani ,Modeling and Simulation of Scheduling Algorithms in LTE Networks, Bachelor of Science Thesis,2011/2012

Appendix

```
%initialize system parameters
init_parm;
snr=5;
bitNum=[];
berNum=[];
slots=1:20;
Frames=1:5;
SNR_vec=1:10;
RB=[6 15 25 50];% 75 100];
%%%%%%%%%%%%%%
   %%%%%%%%%%%%%
   %%%% round robin


userPlotDRate=zeros(1,length(RB));
plotDRateRR=zeros(1,length(RB));
plotDataRateRR=zeros(1,length(RB));
TPRR=zeros(1,length(RB));
UsedBitPerSecPerH=zeros(1,length(RB));
plotSpecEffRR=zeros(1,length(RB));
%loop among users
for NumRB=1:4
    Nrb=RB(NumRB);
    berNum(NumRB)=0;
    bitNum(NumRB)=0;
    TPRR(NumRB)=0;
    %%% initialize users
    user_CQI=randperm(6,NUE);
    for i=1:NUE
   % [max_CQI,max_idx]=max(user_CQI);
    UEs(i).index=i;
    UEs(i).CQI=user_CQI(i);
    UEs(i).avg_CQI=user_CQI(i);
    UEs(i).dataRate=0;
    UEs(i).dataBits=0;
    UEs(i).TP=0;
    end
    BitPerSecPerH=0;
    for nS=1:20   %20 slots in each frame (2 per subframe)
    unAssRB=Nrb;

   % while unAssRB>0
    while unAssRB>=NUE
    for user=1:NUE
        if unAssRB>0
            unAssRB=unAssRB-1;        %decreasing RB number
            [ber,bit]=transceiverSteps(snr);
            bitNum(NumRB)=bitNum(NumRB)+bit;
            berNum(NumRB)=berNum(NumRB)+ber;
            UEs(user).dataBits=UEs(user).dataBits+bit;
            UEs(i).TP=UEs(i).TP+(bit-(bit*ber));
            %%% sysetm evaluation
            if unAssRB == 1
               if NUE-user>0
```

```
                   unServedUser=NUE-user;
               end
            end
        end
     % end
    end
    end
 BitPerSecPerH=BitPerSecPerH+(Nrb-unAssRB);%*(12*7*6)/(180*1e3);
 end
 UsedBitPerSecPerH(NumRB)=(BitPerSecPerH)/((20*Nrb));%*12*7*6)/(180* 1e3));
%20-2slots for scheduling
usersDataBits=0;
systemTP=0;
    for i=1:NUE
        usersDataBits=usersDataBits+ UEs(i).dataBits;
        systemTP=systemTP+UEs(i).TP;
    end
userPlotDRate(NumRB)=usersDataBits/(NUE*1e4);   %(/10) per second(2
slots=1ms)
TPRR(NumRB)=systemTP*1e3/10;%1e4;
plotDataRateRR(NumRB)=bitNum(NumRB)/(Nrb*12*7*2*6*10);
plotSpecEffRR(NumRB)=bitNum(NumRB)*1e3/(20*(Nrb*2*180)); %(Nrb*2*1e-
3*180*1e3 H/s)
plotDRateRR(NumRB)=bitNum(NumRB);
end




   %%%%%%%%%%%%%%
   %%%%%%%%%%%%%
   %%%% proportional fair
userPlotDRate=zeros(1,length(RB));
plotDRatePF=zeros(1,length(RB));
plotDataRatePF=zeros(1,length(RB));
TPPF=zeros(1,length(RB));
UsedBitPerSecPerH=zeros(1,length(RB));
plotSpecEffPF=zeros(1,length(RB));
 for NumRB=1:4
     Nrb=RB(NumRB);
    berNum(NumRB)=0;
    bitNum(NumRB)=0;
    TPPF(NumRB)=0;
    user_CQI=randperm(6,NUE);
    %%% initialize users
    for i=1:NUE
   % [max_CQI,max_idx]=max(user_CQI);
    UEs(i).index=i;
    UEs(i).CQI=user_CQI(i);
    UEs(i).avg_CQI=user_CQI(i);
    UEs(i).dataRate=0;
    UEs(i).dataBits=0;
    UEs(i).TP=0;
```

```
    end

    for users=1:NUE
    UEs(users).avg_CQI=(UEs(users).CQI+user_CQI(users))/NumRB;
    avarges_CQI(users)=user_CQI(users)/UEs(users).avg_CQI;
    UEs(users).dataBits=0;
    end
    BitPerSecPerH=0;
    for nS=3:20   %20 slots in each frame (2 per subframe)
        unAssRB=Nrb;

        while  unAssRB>1
        restCQI=avarges_CQI;
        currentIndex=1;
            while length(restCQI)>=1
             [max_CQI,max_idx]=max(restCQI);
             UEs(max_idx).index=currentIndex;
            if currentIndex==1
               for rbc=1:3
                unAssRB=unAssRB-1;
               if unAssRB>1
                 [ber,bit]=transceiverSteps(snr-2);         %decrease snr
to simulate higher cqi
                 bitNum(NumRB)=bitNum(NumRB)+bit;
                 berNum(NumRB)=berNum(NumRB)+ber;
                 UEs(max_idx).dataBits=(UEs(max_idx).dataBits+bit);
                 UEs(max_idx).TP=UEs(max_idx).TP+(bit-(bit*ber));
               end
               end
            elseif currentIndex==2
                for rbc=1:2
                unAssRB=unAssRB-1;
               if unAssRB>1
                 [ber,bit]=transceiverSteps(snr-1);         %decrease snr
to simulate higher cqi
                 bitNum(NumRB)=bitNum(NumRB)+bit;
                 berNum(NumRB)=berNum(NumRB)+ber;
                 UEs(max_idx).dataBits=(UEs(max_idx).dataBits+bit);
                 UEs(max_idx).TP=UEs(max_idx).TP+(bit-(bit*ber));
               end
               end
            else
             unAssRB=unAssRB-1;
             if unAssRB>1
             [ber,bit]=transceiverSteps(snr);
             bitNum(NumRB)=bitNum(NumRB)+bit;
             %berNum(sn)=berNum(sn)+ber;
             UEs(max_idx).dataBits=(UEs(max_idx).dataBits+bit);
              UEs(max_idx).TP=UEs(max_idx).TP+(bit-(bit*ber));
             end
            end

          restCQI=[restCQI(1:max_idx-1),restCQI(max_idx+1:end)];
          currentIndex=currentIndex+1;
```

```
             end
         end
      BitPerSecPerH=BitPerSecPerH+(Nrb-unAssRB);%*(12*7*6)/(180* 1e3);
    end
    UsedBitPerSecPerH(NumRB)=BitPerSecPerH/(20*Nrb); %10e-3*20;        %20-
2slots for scheduling
    usersDataBits=0;
    systemTP=0;
    for i=1:NUE
        if UEs(i).index==1
           UEs(i).dataRate= UEs(i).dataBits;%/bitNum(NumRBr);
           plotDRate(NumRB)=UEs(i).dataBits;

        end
        usersDataBits=usersDataBits+ UEs(i).dataBits;
        systemTP=systemTP+UEs(i).TP;
    end
    userPlotDRate(NumRB)=usersDataBits/(NUE*9*1e3); %only 18 slots (9ms)
    TPPF(NumRB)=systemTP*1e3/(9);%*1e3);
    plotDataRatePF(NumRB)=bitNum(NumRB)/(Nrb*12*7*2*6*10);     %(4 --> 4QAM
4bit per symbol)   (6 --> QPSKM 4bit per symbol)
    plotSpecEffPF(NumRB)=bitNum(NumRB)*1e3/(20*(Nrb*2*180)); %(Nrb*2*1e-
3*180*1e3 H/s)
    plotDRatePF(NumRB)=bitNum(NumRB);
 end

figure;
 hold on;
xlabel('Number of RB');
ylabel('Data Rate');
title('1x2 LTE System Data Rate');
%plot(RB,plotDataRateRR, 'r');
%plot(RB,plotDataRatePF, 'b');
plot(RB,plotDRateRR, 'r');
plot(RB,plotDRatePF, 'b');
legend(['Round Robin ' , ' Proportional Fair']);
hold off

figure;
hold on
xlabel('Number of RB');
ylabel('Throughput');
title('1x2 LTE System Throughput');
plot(RB,TPRR, 'r');
plot(RB,TPPF, 'b');
legend(['Round Robin ' , ' Proportional Fair']);
hold off

figure;
hold on
xlabel('Number of RB');
ylabel('Spectral Efficiency');
title('1x2 LTE System Spectral Efficiency');
plot(RB,plotSpecEffRR, 'r');
```

```
%plot(Frames,UsedBitPerSecPerH, 'r');
plot(RB,plotSpecEffPF, 'b');
%plot(Frames,UsedBitPerSecPerH, 'b');

legend(['Round Robin ' , ' Proportional Fair']);
hold off
```

```
%initialize system parameters
init_parm_per_user;
snr=5;
Frames=1:5;
bitNum=zeros(1,length(SNR_vec));
berNum=zeros(1,length(SNR_vec));

user_CQI=zeros(1,NUE);
avarges_CQI=zeros(1,NUE);
restCQI=zeros(1,NUE);
slots=1:20;

SNR_vec=1:10;

%%%%%%%%%%%%%
   %%%%%%%%%%%%%
   %%%% round robin


%loop among frames
for snr=1:10
   % NUE=sn;
    %%% initialize users
    user_CQI=randperm(6,NUE);
    for i=1:NUE
   % [max_CQI,max_idx]=max(user_CQI);
    UEs(i).index=i;
    UEs(i).rateHistory=1;
    UEs(i).CQI=user_CQI(i);
    UEs(i).avg_CQI=user_CQI(i);
    UEs(i).dataRate=zeros(1,length(SNR_vec));
    UEs(i).dataBits=zeros(1,length(SNR_vec));
    UEs(i).TP=zeros(1,length(SNR_vec));
    end
    BitPerSecPerH=0;
    for nS=1:20   %20 slots in each frame (2 per subframe)
    unAssRB=Nrb;

   % while unAssRB>0
   % while unAssRB>NUE
    for user=1:NUE
        if unAssRB>0
            unAssRB=unAssRB-1;        %decreasing RB number
            [ber,bit]=transceiverSteps(snr);
            bitNum(snr)=bitNum(snr)+bit;
            berNum(snr)=berNum(snr)+ber;
            UEs(user).dataBits(snr)=UEs(user).dataBits(snr)+bit;
            UEs(user).TP(snr)=UEs(user).TP(snr)+(bit-(bit*ber));
            %%% sysetm evaluation
            if unAssRB == 1
               if NUE-user>0
                   unServedUser=NUE-user;
               end
            end
```

```
        end
       % end
    end
    %end

    end

 for i=1:NUE
    UERR(i).dataBits(snr)=UEs(i).dataBits(snr);
    UERR(i).TP(snr)=UEs(i).TP(snr);
 end
end


   %%%%%%%%%%%%%%
   %%%%%%%%%%%%%
   %%%% proportional fair

 for snr=1:10
    % NUE=sn;
    user_CQI=randperm(6,NUE);
    %%% initialize users
    for i=1:NUE
   % [max_CQI,max_idx]=max(user_CQI);
    UEs(i).index=i;
    UEs(i).rateHistory=1;
    UEs(i).CQI=user_CQI(i);
    UEs(i).avg_CQI=user_CQI(i)/ UEs(i).rateHistory;
    UEs(i).dataRate=zeros(1,length(SNR_vec));
    UEs(i).dataBits=zeros(1,length(SNR_vec));
    UEs(i).TP=zeros(1,length(SNR_vec));
    end

    %initialization
    for users=1:NUE
    UEs(users).avg_CQI=(UEs(users).CQI+user_CQI(users))/
UEs(users).rateHistory;
    avarges_CQI(users)=user_CQI(users)/UEs(users).avg_CQI;
    %UEs(users).dataBits=0;
    end
    BitPerSecPerH=0;
    for nS=3:20   %20 slots in each frame (2 per subframe)
        unAssRB=Nrb;
    while  unAssRB>1
        restCQI=avarges_CQI;
        currentIndex=1;
            while length(restCQI)>=1
             [max_CQI,max_idx]=max(restCQI);
             UEs(max_idx).index=currentIndex;
             restCQI=[restCQI(1:max_idx-1),restCQI(max_idx+1:end)];
             currentIndex=currentIndex+1;
            end

        for users=1:NUE
```

```
            if UEs(users).index==1
                for rbc=1:3
                 unAssRB=unAssRB-1;
                if unAssRB>0
                  [ber,bit]=transceiverSteps(snr-2);
                  bitNum(snr)=bitNum(snr)+bit;
                  berNum(snr)=berNum(snr)+ber;
                  UEs(users).rateHistory=2;
                  UEs(users).dataBits(snr)=(UEs(users).dataBits(snr)+bit);
                  UEs(users).TP(snr)=UEs(users).TP(snr)+(bit-(bit*ber));
                end
                end
             elseif currentIndex==2
                for rbc=1:2
                 unAssRB=unAssRB-1;
                if unAssRB>0
                  [ber,bit]=transceiverSteps(snr-1);
                  bitNum(snr)=bitNum(snr)+bit;
                  berNum(snr)=berNum(snr)+ber;
                  UEs(users).rateHistory=1;
                  UEs(users).dataBits(snr)=(UEs(users).dataBits(snr)+bit);
                  UEs(users).TP(snr)=UEs(users).TP(snr)+(bit-(bit*ber));
                end
                end

              else
              unAssRB=unAssRB-1;
              if unAssRB > 0
              [ber,bit]=transceiverSteps(snr);
              bitNum(snr)=bitNum(snr)+bit;
              UEs(users).rateHistory=1;
              %berNum(sn)=berNum(sn)+ber;
              UEs(users).dataBits(snr)=(UEs(users).dataBits(snr)+bit);
              UEs(users).TP(snr)=UEs(users).TP(snr)+(bit-(bit*ber));
              end
            end
        end
      end

     end
    for i=1:NUE
     UEPF(i).dataBits(snr)=UEs(i).dataBits(snr);
     UEPF(i).TP(snr)=UEs(i).TP(snr);
    end
  end

  hold on;
 xlabel('Number of Frames');
 ylabel('Data Rate');
 title('1x2 LTE System Data Rate');
 %plot(RB,plotDataRateRR, 'r');
 %plot(RB,plotDataRatePF, 'b');
 color=['r' 'm' 'c' ;'y' 'b' 'g'];
 for i=1:NUE
```

```
plot(SNR_vec,UERR(i).dataBits, 'r');
plot(SNR_vec,UEPF(i).dataBits, color(2,i));
end
legend(['Round Robin ' , ' Proportional Fair']);
hold off

figure;
hold on
xlabel('Number of Frames');
ylabel('Throughput');
title('1x2 LTE System Throughput');
for i=1:NUE
plot(SNR_vec,UERR(i).TP,'r');
plot(SNR_vec,UEPF(i).TP, color(2,i));
end
legend(['Round Robin ' , ' Proportional Fair']);
hold off
```

```
init_parm;
snr=5;
bitNum=[];
berNum=[];
slots=1:20;
Frames=1:5;
SNR_vec=1:10;

%%%%%%%%%%%%%
   %%%%%%%%%%%%%
   %%%% round robin
userPlotDRateRR=zeros(1,length(SNR_vec));
plotDataRateRR=zeros(1,length(SNR_vec));
TPRR=zeros(1,length(SNR_vec));
UsedBitPerSecPerH=zeros(1,length(SNR_vec));
plotSpecEffRR=zeros(1,length(SNR_vec));
plotDRateRR=zeros(1,length(SNR_vec));
for snr=1:10

    berNum(snr)=0;
    bitNum(snr)=0;
    TPRR(snr)=0;
    %%% initialize users
    user_CQI=randperm(6,NUE);
    for i=1:NUE
   % [max_CQI,max_idx]=max(user_CQI);
    UEs(i).index=i;
    UEs(i).CQI=user_CQI(i);
    UEs(i).avg_CQI=user_CQI(i);
    UEs(i).dataRate=0;
    UEs(i).dataBits=0;
    UEs(i).TP=0;
    end

for nS=1:20   %20 slots in each frame (2 per subframe)
    unAssRB=Nrb;
    BitPerSecPerH=0;
   % while unAssRB>0
    %while unAssRB>=NUE
    for user=1:NUE
        if unAssRB>0
            unAssRB=unAssRB-1;        %decreasing RB number
            [ber,bit]=transceiverSteps(snr);
            bitNum(snr)=bitNum(snr)+bit;
            berNum(snr)=berNum(snr)+ber;
            UEs(user).dataBits=UEs(user).dataBits+bit;
            UEs(user).TP=UEs(user).TP+(bit-(bit*ber));
            %%% sysetm evaluation
            if unAssRB == 1
               if NUE-user>0
                   unServedUser=NUE-user;
```

```
              end
           end
       end
       %%% sysetm evaluation
       %if user==NUE
       %    unUsedRB=Nrb-unAssRB;
       %end
  % end
   end
  % end
   BitPerSecPerH=BitPerSecPerH+(Nrb-unAssRB)*(12*7*6)/(180* 1e3);
 end
UsedBitPerSecPerH(snr)=BitPerSecPerH/20;
usersDataBits=0;
systemTP=0;
    for i=1:NUE
        usersDataBits=usersDataBits+ UEs(i).dataBits;
        systemTP=systemTP+UEs(i).TP;
    end
userPlotDRateRR(snr)=usersDataBits*1e3/(NUE*10);   %(/10) per second(2
slots=1ms)
TPRR(snr)=systemTP*1e3/10;%1e4;
plotDataRateRR(snr)=bitNum(snr)/(Nrb*12*7*2*4*10);
plotDRateRR(snr)=bitNum(snr)*1e3/10;
plotSpecEffRR(snr)=bitNum(snr)*1e3/(10*(Nrb*2*180)); %(Nrb*2*1e-3*180*1e3
H/s)

end


   %%%%%%%%%%%%%%
   %%%%%%%%%%%%%
   %%%% proportional fair
userPlotDRatePF=zeros(1,length(SNR_vec));
plotDataRatePF=zeros(1,length(SNR_vec));
TPPF=zeros(1,length(SNR_vec));
UsedBitPerSecPerH=zeros(1,length(SNR_vec));
plotSpecEffPF=zeros(1,length(SNR_vec));
plotDRatePF=zeros(1,length(SNR_vec));

 for snr=1:10

    berNum(snr)=0;
    bitNum(snr)=0;
    TPPF(snr)=0;
    user_CQI=randperm(6,NUE);
    %%% initialize users
    for i=1:NUE
   % [max_CQI,max_idx]=max(user_CQI);
    UEs(i).index=i;
    UEs(i).CQI=user_CQI(i);
    UEs(i).avg_CQI=user_CQI(i);
    UEs(i).dataRate=0;
    UEs(i).dataBits=0;
```

```
UEs(i).TP=0;
end

for users=1:NUE
UEs(users).avg_CQI=(UEs(users).CQI+user_CQI(users))/snr;
avarges_CQI(users)=user_CQI(users)/UEs(users).avg_CQI;
UEs(users).dataBits=0;
end

for nS=3:20   %20 slots in each frame (2 per subframe)
    unAssRB=Nrb;
    BitPerSecPerH=0;
    while  unAssRB>1
    restCQI=avarges_CQI;
    currentIndex=1;
        while length(restCQI)>=1
         [max_CQI,max_idx]=max(restCQI);
         UEs(max_idx).index=currentIndex;
        if currentIndex==1
           for rbc=1:3
            unAssRB=unAssRB-1;
           if unAssRB>1
             [ber,bit]=transceiverSteps(snr-2);
             bitNum(snr)=bitNum(snr)+bit;
             berNum(snr)=berNum(snr)+ber;
             UEs(max_idx).dataBits=(UEs(max_idx).dataBits+bit);
             UEs(max_idx).TP=UEs(max_idx).TP+(bit-(bit*ber));
           end
           end
        elseif currentIndex==2
           for rbc=1:2
            unAssRB=unAssRB-1;
           if unAssRB>1
             [ber,bit]=transceiverSteps(snr-1);
             bitNum(snr)=bitNum(snr)+bit;
             berNum(snr)=berNum(snr)+ber;
             UEs(max_idx).dataBits=(UEs(max_idx).dataBits+bit);
             UEs(max_idx).TP=UEs(max_idx).TP+(bit-(bit*ber));
           end
           end

        else
         unAssRB=unAssRB-1;
         if unAssRB>1
         [ber,bit]=transceiverSteps(snr);
         bitNum(snr)=bitNum(snr)+bit;
         %berNum(snr)=berNum(snr)+ber;
         UEs(max_idx).dataBits=(UEs(max_idx).dataBits+bit);
          UEs(max_idx).TP=UEs(max_idx).TP+(bit-(bit*ber));
         end
        end

      restCQI=[restCQI(1:max_idx-1),restCQI(max_idx+1:end)];
      currentIndex=currentIndex+1;
```

```
            end
        end
       BitPerSecPerH=BitPerSecPerH+(Nrb-unAssRB)*(12*7*6)/(180* 1e3);
    end
    UsedBitPerSecPerH(snr)=BitPerSecPerH/18;        %20-2slots for
scheduling
    usersDataBits=0;
    systemTP=0;
    for i=1:NUE
        if UEs(i).index==1
           UEs(i).dataRate= UEs(i).dataBits;%/bitNum(snr);
           plotDRatePF(snr)=UEs(i).dataBits;

        end
        usersDataBits=usersDataBits+ UEs(i).dataBits;
        systemTP=systemTP+UEs(i).TP;
    end
    userPlotDRatePF(snr)=usersDataBits*1e3/(NUE*9); %only 18 slots (9ms)
    TPPF(snr)=systemTP*1e3/(9);
    plotDRatePF(snr)=bitNum(snr)*1e3/10;
    plotDataRatePF(snr)=bitNum(snr)/(Nrb*12*7*2*6*10);     %(4 --> 4QAM
4bit per symbol)   (6 --> QPSKM 4bit per symbol)
plotSpecEffPF(snr)=bitNum(snr)*1e3/(10*(Nrb*2*180)); %(Nrb*2*1e-3*180*1e3
H/s)

 end

hold on;
xlabel('SNR');
ylabel('Data Rate');
title('1x2 LTE System Data Rate');
%plot(RB,plotDataRateRR, 'r');
%plot(RB,plotDataRatePF, 'b');
%plot(SNR_vec,plotDRateRR, 'r');
%plot(SNR_vec,plotDRatePF, 'b');
plot(SNR_vec,userPlotDRateRR, 'r');
plot(SNR_vec,userPlotDRatePF, 'b');
legend(['Round Robin ' , ' Proportional Fair']);
hold off

figure;
hold on
xlabel('SNR');
ylabel('Throughput');
title('1x2 LTE System Throughput');
plot(SNR_vec,TPRR, 'r');
plot(SNR_vec,TPPF, 'b');
legend(['Round Robin ' , ' Proportional Fair']);
hold off

figure;
hold on
xlabel('SNR');
ylabel('Spectral Efficiency');
```

```
title('1x2 LTE System Spectral Efficiency');
plot(SNR_vec,plotSpecEffRR, 'r');
%plot(Frames,UsedBitPerSecPerH, 'r');
plot(SNR_vec,plotSpecEffPF, 'b');
%plot(Frames,UsedBitPerSecPerH, 'b');

legend(['Round Robin ' , ' Proportional Fair']);
hold off
```

```
function [ber, bits]=transceiverSteps(snr)
```

```
%% Constants
FRM=100;    %framelength(dataLength)
Trellis=poly2trellis(4, [13 15], 13);
Indices=randperm(FRM);
M=4;k=log2(M);
R= FRM/(3* FRM + 4*3);
snrdb =10*log10(k) + 10*log10(R);  %EbNo + 10*log10(k) + 10*log10(R);
noise_var = 10.^(-snrdb/10);
%% Initializations
persistent Modulator AWGN DeModulator BitError TurboEncoder TurboDecoder
if isempty(Modulator)
Modulator = comm.QPSKModulator('BitInput',true);
AWGN = comm.AWGNChannel;
DeModulator = comm.QPSKDemodulator('BitOutput',true,...
'DecisionMethod','Log-likelihood ratio',...
'VarianceSource', 'Input port');
BitError = comm.ErrorRate;
TurboEncoder=comm.TurboEncoder(...
'TrellisStructure',Trellis,...
'InterleaverIndices',Indices);
TurboDecoder=comm.TurboDecoder(...
'TrellisStructure',Trellis,...
'InterleaverIndices',Indices,...
'NumIterations',6);
end

%% Processsing loop modeling transmitter, channel model and receiver

numErrs = 0; numBits = 0;results=zeros(3,1);

% Transmitter
user1In = randi([0 1], FRM,1); % Random bits generator

encoded1 =TurboEncoder.step(user1In); % Turbo Encoder
mod_sig = Modulator.step(encoded1); % QPSK Modulator

hLTEChan = comm.LTEMIMOChannel(...
  'Profile',               'EVA 5Hz',...
  'AntennaConfiguration', '1x2',...
  'CorrelationLevel',     'Low',...
  'RandomStream',         'mt19937ar with seed',...
  'Seed',                 99,...
  'PathGainsOutputPort',  true);

[rxFade, chPathG] = step(hLTEChan, mod_sig);
% Add AWG noise
rx_sig=awgn(rxFade,snrdb);% AWGN channel

% Receiver

[row,col]=size(rx_sig);
rx_sig=reshape(rx_sig,row*col/2,2);
demod = DeModulator.step(rx_sig(:,1), noise_var); %Soft-decisionQPSK
Demodulator
```

```
[row,col]=size(demod);

decoded = TurboDecoder.step(-demod); % Turbo Decoder
y = decoded(1:FRM); % Compute output bits

results = BitError.step(user1In, y); % Update BER
numErrs = results(2);
numBits = results(3);

%% Clean up & collect results
ber = results(1);
bits= results(3);%FRM;
```