AALBORG UNIVERSITET

Institute of Electronic Systems
Division of Telecommunications
Niels Jernes Vej 12
http://es.aau.dk/

Title:
Game-Theory Based Policies For Flexible Spectrum Usage in IMT-Advanced Systems

Theme:
Spectrum Sharing
Cognitive radio

Project period:
February 2008 - June 2008

Project group:
08gr1112

Members:
Valentina Palma
Erwann Borgat

Supervisors:
Nicola Marchetti
Sanjay Kumar

Copies: 3

Pages in report: 69

Pages in appendices: 10

Supplement: Nothing

Printed June 4, 2008

Abstract:

The next generation wireless systems, like International Mobile Telecommunications-Advanced (IMT-A), aim to provide high data rates, requiring spectrum allocation in the range of 100 MHz.

Nowadays, we have both a limited available spectrum and inefficieny in its usage. It will be impossible to allocate 100 MHz to each operator in a same geographical area. Therefore there is a need for new spectrum access protocols to share the spectrum among different operators in a fair and efficient way.

A game theory solution to spectrum sharing is discussed in this thesis. We proposed a dynamic game model based on regret lerning test to solve the spectrum sharing problem. We show through realistic simulations that it is possible to share the spectrum in a fair and efficient way.

This report is the outcome of our 10th semester in Mobile Communication, carried out at the department of Communication Technology, Institute of Electronic Systems at Aalborg University, Denmark. It is addressed to our project supervisor, the teachers and the students who are interested in spectrum sharing, cognitive radio and game theory. The aim of the report is to document our 10th semester work.

We would like to thank our project supervisor, Nicola Marchetti and Sanjay Kumar, for the time they spent with us, for their help and patience throughout this project.

Valentina Palma _____________

Erwann Borgat _____________

Contents

Chapter 1

Introduction

Wireless Communications are requiring additional spectrum to satisfy the consumers demand for high data rate applications. At the same time, many of these applications have increasing restrictions to spectrum access. The currently available unlicensed spectrum is reaching its limit. A support of Quality of Service (QoS) is difficult to provide because of the missing coordination between the different radio systems operating in the same frequency band.

In the recent years there is a dramatic increase in the access to the limited spectrum for mobile services and the number of mobile subscribers is expected to exceed 5 billions by the end of 2015. A key to attract such a huge number of users is to make them experiencing new multimedia services with a very high bit rate. In fact, the currently available bit rates do not encourage for usage of today's services like video downloading, as the video downloads take tens of minutes, if not hours.

The International Mobile Telecoomunications-Advanced (IMT-A) systems are expected to provide peak data-rates in the order of 1 Gbps in local area. The new capabilities of these systems are envisaged to handle a wide range of supported data rates according to economic and service demands in multiuser environments [1]. Moreover, the new spectrum allocation for IMT-A decided by the World Radio Conference (WRC) in 2007 is not sufficient to allocate such wide bandwidth to several operators in the same geographical area like in figure 1.1.

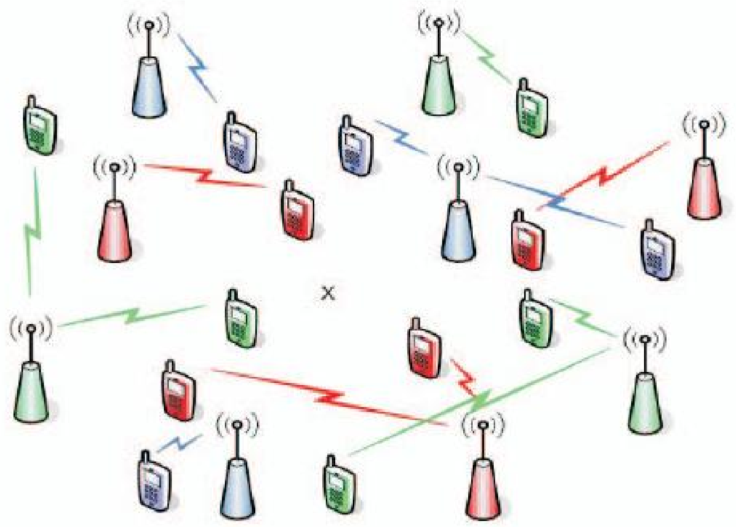

Figure 1.1: Cellular multi-operator environment [2]

Hence, a new approach is desired for Spectrum Allocation/Sharing among several operators over a flexible bandwidth up to 100 MHz for next generation IMT-A systems.

1.1 Problem definition

Problem: How to get an efficient and fair utilization of radio spectrum between different operators ?

The next generation mobile systems will require a very high data rate, that leads to the use of higher bandwidth. For example for IMT-A a peak rate of 1 Gbps is expected in local area that means a spectrum allocation more than 100 MHz with a spectral efficiency of 10 bps/Hz. The spectral efficiency may be enhanced by the means of modern techniques such as OFDM and MIMO, but at a certain point it will be impossible to enhance it any more. So despite of high spectral efficiency, the bandwidth that we need to achieve such high data rate is still high.

Nowadays we have both a limited available spectrum and inefficiency in its usage. Although new spectrum bandwidth will be allocated for the next generation mobile communication system, it could not be enough. It will be impossible to allocate 100 MHz bandwidth to every operator in the same geographical area. A solution is likely to be flexible (Flexible Spectrum Usage)among different operators.

This type of solution could be not well accepted by operators because they are scared of interference as well as of unfairness in spectrum allocation. They have also payed a lot of money for the licenses.

Therefore in our project we try to manage interference and to reach a fair as well as efficient spectrum sharing between several operators that share the spectrum in the same geographical area. Our intent is to achieve these objectives by the means of game theory.

We consider a situation where several operators coexist in the same geographical area and share the same spectrum. We start from the simplest scenario with two base stations in the same area. Each base stations belong to a different operator, so each operators have only one base station.

We would like to analyze whether these operators can coexist and share spectrum in an efficient and fair manner.
Case study: spectrum sharing between two operators in the same geographical area
Goal: to reach a fair and efficient spectrum sharing through a game theory oriented algorithm. Fairness and efficiency are two contradictory objectives so we need to find a suitable trade off.

1.2 Original contributions

The project consists in defining a policy, i.e. a set of rules, taking into account operators/users Quality of Service (QoS) requests as weel as a specific FSU algorithm adopted. The policy has to be aware of the spectrum sensing, which constitutes the first step for an efficient sharing and flexible usage of the spectrum resource. Indeed different operators have to use resources in a way that services of other operators are not interfered.

Operators have to maximize spectral efficiency in mobile network. In a shared spectrum scenario the different operators will be competing for spectrum. We need analytical and simulation tools to obtain a good spectral efficiency in such scenario. One of the most promising approaches is to use game theory, from the analytical point of view. Game theory can be defined as the study of mathematical models of conflict and cooperation between rational decision makers [3]. It has not been a long time since game theory has been applied in telecommunication field. Moreover, we are applying a theory recently used in the last few years.

There is no direct signaling between operators. Therefore there is neither need for new interfaces nor for tight synchronization.

Overlapping allocation: Instead of trying to make orthogonal allocations our approach aims at maximizing spectral reuse and efficiency.

1.3 Report structure

The remaining part of the report is structured as follows:

- Technical background: introducing the spectrum sharing concept between two operators in a local area and in details game theory that will be used.
- Game model chapter explains how the algorithm is designed and what are the most important information we are using.
- Description of the simulation conditions, in which the choices made for the implementation are justified.
- Similation results and analysis showing the results obtained from the simulations and giving an interpretation of them.
- Conclusion summarising the work done in the project and the possible future work.

The literature references are marked with square brackets for the whole report, e.g [14]. Equations are denoted with brackets, taking into account the chapter number and the position of the equation in the chapter, for example (4.1) for the first equation of chapter four. Figures and Tables are numerated in the same way and are always preceded respectively by the term Figure or Table. The lists of references, figures and tables are provided at the end of the report.

Chapter 2

Technical background

In this part we are going to introduce the main functions of cognitive radios and then we will write about FSU and spectrum sharing techniques. The figure 2.1 shows the hierarchy between each approaches.

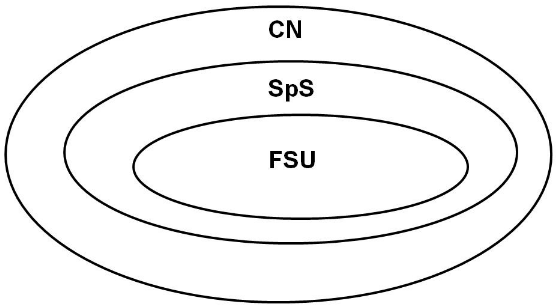

Figure 2.1: Cognitive Networks (CN), Spectrum Sharing (SpS) and Flexible Spectrum Usage (FSU)

2.1 Cognitive radio

Cognitive radio is a new approach to deal with the radio spectrum allocation. This new approach attempts to efficiently use the spectrum by identifying and using under-utilized spectrum [4].

Several techniques are used in cognitive radio:

1. Spectrum sensing: the goal of this technique is to detect spectrum opportunities and then to share it without interference with other users. It is the first step in cognitive radio to be available to sense spectrum detecting used and unused spectrum.

2. Spectrum management: the goal of this technique is to allocate the best available spectrum to satisfy user requirements in terms of Quality of

Service. Cognitive radio system should be able to decide which is the best spectrum allocation after a spectrum analysis.

3. Spectrum mobility: it occurs when a cognitive radio user switches its frequency of operation to use a better spectrum. Basically it defines the possibility to use the spectrum dynamically by allowing the radio terminals to operate in the best available frequency band. During the switching process user should keep the same level of quality of service to satisfy his requirements.

4. Spectrum sharing: the coexistence with licensed users and the wide range of available spectrum are two of the main challenges. Spectrum sharing mainly focuses on two parameters: fairness and efficiency. Actually, it is the main challenge of this thesis as we will see later.

To sum up cognitive radio technology is expected to improve spectrum access through:

- increased spectrum efficiency of licensed spectrum users
- opportunistic spectrum use of unlicensed band

2.2 Spectrum sharing

One of the most important thing for operator is the fairness. The spectrum shares have to be fair between operator sharing the spectrum. Moreover, operators do not have a common goal and then do not cooperate with each other.

The main problen is to set some spectrum sharing rules which allow the systems to share the spectrum in a fair and efficient way.

Efficiency:
A resource allocation is efficient if it is not possible to improve the performance of a given system without degrading the performance of some other system [5]. Hence we can find many efficient operating point where each of them represents different performance trade-off among the different operators.

We can measure efficiency with spectral efficiency. Indeed, spectral efficiency is a measure referring to the amount of information that can be transmitted over a given bandwidth in a specific communication system.

In digital wireless networks, the system spectral efficiency or area spectral efficiency is typically measured in $\frac{bit/s}{Hz}$. It is a measure of the quantity of users or services that can be simultaneously supported by a limited radio frequency bandwidth in a defined geographic area. It may for example be defined as the maximum throughput, summed over all users in the system, divided by the channel bandwidth. This measure is affected not only by the single user transmission technique, but also by multiple access schemes and radio resource management techniques utilized. It can be improved by dynamic radio resource management.

Spectrum efficiency can also be defined as the optimized use of spectrum or bandwidth so that the maximum amount of data can be transmitted with the fewest transmission errors. In a cellular telephone network, spectrum efficiency equates to the maximum number of users per cell that can be accommodated while maintaining an acceptable quality of service (QoS).

Fairness:
Fairness is related to the relative performance among the systems. It can be achieved by optimizing a global utility function over the possible resource allocations. Different utilities represent different fairness goals [5]. We will define fairness more deeply in part 4.2.1.

2.3 Flexible Spectrum Usage

Thanks to Flexible Spectrum Usage (FSU) devices are able to use spectrum in a flexible manner by adapting their operation to the current situation by sensing the environment or based on pre-defined regulatory policies that can vary in time, place, and event-based. So far the main advocates of FSU have been new companies trying to enter the market [6].

For us FSU deals with the use of the same frequency band by radio access systems using the same radio access technology.

In the case where several operators are operating in the same geographical area and in the same frequency band then we can address peaceful coexistence in several ways:

- Time separation: systems transmit at different times;
- Frequency separation: systems transmit at different frequencies;
- Space separation: the distance between the transmit antennas of one system and the receive antennas of another system is sufficiently high to attenuate the interfering signal.

For example in figure 2.2 different operators use resources in a way that services of other operators are not interfered. They can coexist on the frequency-time domain at a certain position X (see in figure 1.1) being able to use a pooled spectrum, differently from current systems, where each operator is assigned a fixed dedicated part of the spectrum [2].

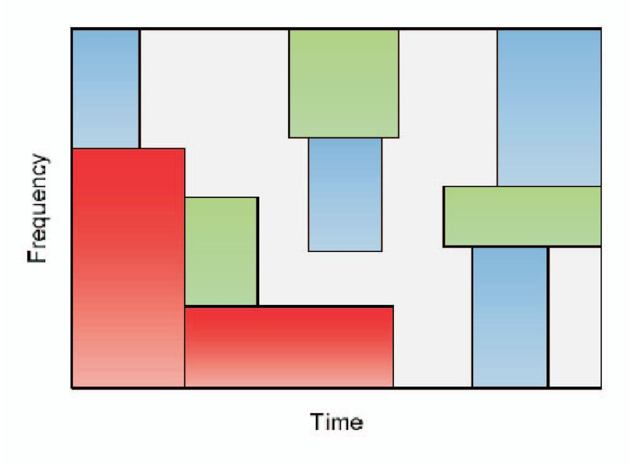

Figure 2.2: Allocation of resources to operators as experienced in position X in figure 1.1. The pink area is vacant for any operator in this position [2]

To enable FSU we need a ressource management function like a spectrum control strategy. Mainly there are two approaches the centralized and the decentralized one.

2.3.1 Spectrum control strategies

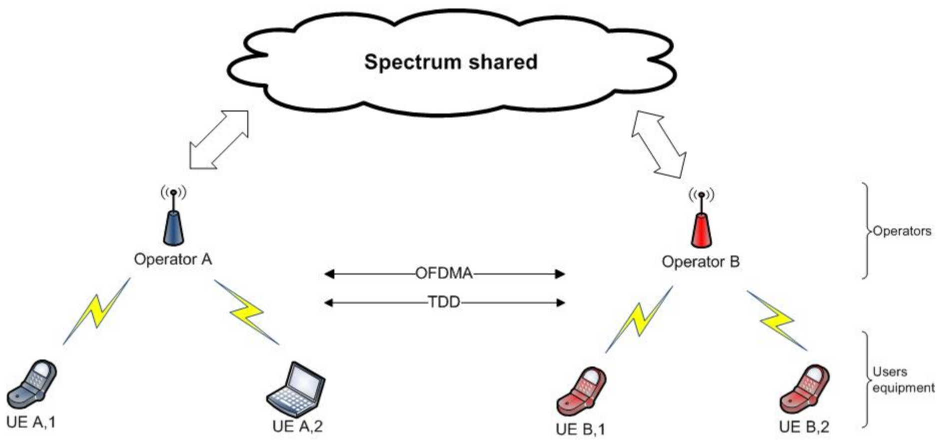

Figure 2.3: Spectrum sharing

Centralized approach

In this approach each operator owns a certain part of the spectrum and decides to share its resources to build a common spectrum pool. It is an agreement between operators. Then, the spectrum is allocated from the centralized spectrum pool in a dynamic way to the differents operators.

Basically, a third entity controls the spectrum allocation and access procedures. Each entity in the network asks based on its requirements an allocation of the spectrum. Then the third entity considers all the requirements and is in charge of the global allocation proccess.

Decentralized approach

In this case, there is no central entity present to control the spectrum management and no signaling between the operators. So, the decisions about management of the spectrum lie with each operator. The decentralized entities can be fully autonomous and uncoordinated, or collaborative and distributed. In the first there is no exchange of information between the operators, in the second case operators form a group inside whitch they collaborate with each other to identify the transmission opportunities and to coexist with other eventual groups.

2.4 Physical Resource blocks

Downlink and uplink transmissions are organized into radio frames with 10ms duration. Each radio frame is $T_f = 10$ ms long and consists of 20 slots of length $T_{slot} = 0.5$ ms.

Physical resource blocks is the resource to share between operators. The downlink transmission scheme is based on conventional OFDM using a cyclic prefix (see appendix C). We assume that the OFDM sub-carrier spacing is $\Delta f = 60kHz$. The transmitted signal in each slot is described by a ressource grid (see 2.4) of $N_{RB}^{DL} N_{sc}^{RB}$ subcarriers and $N_{symb}^{DL} = 7$ OFDM symbols. The $N_{sc}^{RB} = 12$consecutive sub-carriers during one slot correspond to one downlink resource block. In the frequency domain, the total number of resource blocks N_{RB}^{DL} for 100 MHz system bandwidth is 125. Indeed, if we are using a bandwidth about 90MHz. One PRB is equal to 12 consecutives subcarriers where each one has a bandiwdth of about 60kHz. Thus the number of PRBs is equal to:

$$\frac{90000(KHz)}{12 * 60(KHz)} = \frac{90000}{720} = 125 \tag{2.1}$$

Hence, an operator can use a maximum of 125 PRBs.

Moreover, all physical downlink channels (shared data, multicast, broadcast) are processed and mapped to symbols in the downlink resource blocks (resource elements).

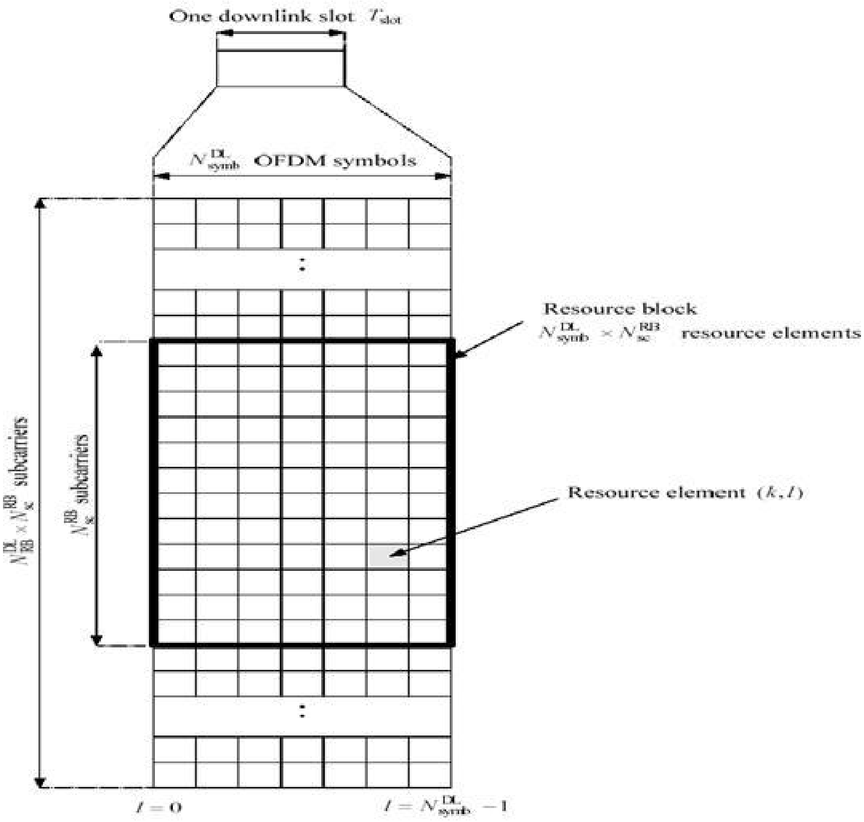

Figure 2.4: Downlink ressource grid

The up-link sub-carrier spacing is $\Delta f = 60kHz$. The $N_{sc}^{RB} = 12$ consecutive sub-carriers during one slot correspond to one uplink resource block. In the frequency domain, the total number of resource blocks N_{sc}^{RB} has the same range as for the downlink transmission.

2.5 Technical background summary

The most important aspect of dynamic spectrum access is dynamic spectrum sharing. It is in charge of efficient and fair spectrum allocation. The key component of network users equipped with cognitive radio is their cognitive intelligence. It enables users to learn from the past (spectrum usage, communication parameters, spectrum sensed and users' allocation) and then to make a smart spectrum allocation. Game theory is one of the approaches used to study the players' behavior (cooperative or selfish) and their influence on the others.

Chapter 3

Game Theory overview

3.1 Game theory approaches

3.1.1 Why game theory ?

Before efficient dynamic spectrum sharing can be achieved, network users' intelligent behaviors and interactions have to be thoroughly studied and analyzed. Game theory studies conflict and cooperation among intelligent rational decision makers, which is an excellent match in nature to dynamic spectrum sharing problems. The importance of studying dynamic spectrum sharing in a game theoretical framework is multifold.

First, by modeling dynamic spectrum sharing among operators as games, the network users' behaviors and actions can be analyzed in a formalized game structure, by which the theoretical achievements in game theory can be fully utilized.

Second, game theory equips us with various optimality criteria for the spectrum sharing problem. Thus game theory provides us well defined equilibrium criteria to measure game optimality under various game settings (operators scenarios in our context).

Third, non-cooperative game theory, one of the most important game theories, enables us to derive efficient distributed approaches for dynamic spectrum sharing using only local information. Such approaches become highly desirable when centralized control is not available or flexible self-organized approaches are necessary. [7]

3.1.2 What is game theory ?

Game theory is a discipline aiming at modeling situations in which actors have to make decisions that have mutual, possibly conflicting, consequences.
It has been used primarily in economics, in order to model competition between companies but also in politics and biology. Telecommunications is one of the

new fields in which it has been studied recently. [8]

In this part, we are going firstly to explain game theory and secondly how situations can be modelled by making use of game theory. Then we will apply game theory to our case-study.

3.2 Classification of games

In the wireless networking context we define players as the users or network operators controlling their devices. Moreover we assume that they are rational and they act according to their strategies. The strategy of a player can be a single move or a set of moves during the game as we will see later. Players try to get the maximum from their utilities in order to maximize their outcomes.

3.2.1 Cooperative and non-cooperative game

We can define a cooperative game as a game where groups of players may enforce cooperative behavior. In cooperative games competition is between coalitions of players. In a cooperative game, players bargain with each other before the game is played. If an agreement is reached, players act according to the agreement reached, otherwise players act in a non-cooperative way. Cooperative games require additional signaling between the decision makers and this is one of the main challenges in telecommunication.

On the contrary, in a non-cooperative game competition is between individual players who have potentially conflicting interests.

3.2.2 Static and dynamic game

In a static game players choose their actions simultaneously without knowing anything about the actions of the other players. In some problems like the Multiple Access Game (see figure 3.4) it is a reasonable assumption.

In most of the games however the players have a sequential interaction. It means that an action from one player happens before or after an action of the other player. So, the second player knows the previous action from the first player. He can adapt his own action to get the best response to the action of the other player. These kind of games are called dynamics in game theory. We can represent them in an strategic form 3.2.3 and in an extensive form (see in 3.2.3). It depends of which visualization we prefer.

3.2.3 Representation of games

Remind that in game theory we assume that players controlling the devices are rational. So, they try to maximize their benefit. There are two graphical representation to formalize games: extensive and strategic.

Extensive form

Sequential game are most of the time represented in extensive form as trees (see figure 3.1). It provides information about the players, payoffs, strategies, and the order of moves. Each node represents an action to choose for a player. The initial node called root represents the first move of the game. The payoffs are specified at the terminal node.

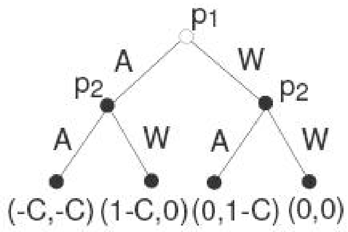

Figure 3.1: Example of extensive form [9]

In the figure 3.1, there are two players. Player 1 (P_1) starts and chooses between action A and W. Then, player 2 (p_2) chooses A or W. Let us suppose that player 1 chooses A and player 2 chooses W. Player 1 gets (1-C) and Player 2 gets 0.

Strategic form

The strategic form of a game G is defined as: $G = (N; S; U)$ [9]. Where:

- N is the set of players : $p_1, ..., p_i$.
- S is the set of strategies
- U is the set of payoff functions.

A dynamic game is more naturally represented by extensive form. However, as we are in a decentralized approach (see chapter 2.3.1) players act without knowing the actions of the others. Therefore it is better to represent the game in strategic form. That is why we will focus only on this graphical representation.

3.2.4 Pure and mixed strategies

Each player can choose between several actions that we call strategies. If one of these actions has a probability of one to be choosen and all the other actions have a probability of zero then it is a pure strategy.

In general and in reality, a player has to choose between differents pure strategies with differents probabilities. These strategies are called mixed strategies in game theory because players choose a probability distribution over several actions. The player will randomly select a pure strategy based on the distribution instead of choosing a particular pure strategy deterministically.

Let us explain with the payoff matrix in 3.1:

		Player 2	
		A	B
Player 1	A	(2,2)	(3,2)
	B	(0,3)	(1,2)

Table 3.1: Payoff matrix

Player 1 chooses a row and receives the first payoff. Player 2 chooses a column and receives the second payoff. Let us focus e.g on player 1. If player 1 chooses to play A with a probability 1 then A is a pure strategy. If player A chooses to play randomly (flipping a coin and play based on the result for example) then it is a mixed strategy.

3.2.5 Complete and incomplete information

A game with complete information is defined as a game where each player knows the game $G = (N; S; U)$, the set of players N, the set of strategies S and the set of utility functions U. On the contrary, the players may have incomplete information. It means they have some assumptions about the outcomes of other players and hence they may try to solve the game on these basis.

Strategies are based on information so they are really different in the case of a game with complete information compared to a game with incomplete information.

3.2.6 Perfect and imperfect information

Perfect information: each player knows the identity of the other players and, for each of them, the payoff resulting from each strategy. We refer to a game with perfect information, if the players have a perfect knowledge of all previous moves in the game at any moment they have to make a new move. On the contrary if players do not know exactly the previous moves like in simultaneous game then it is a game with imperfect information.

3.3 Single Stage Game (SSG)

In game theory we talk about a Single Stage Game (SSG) when we assume there is only one time step (one move) to choose an action. So, players choose only one time a strategy.
Basically, in wireless communications one Single Stage Game consists of three main phases:

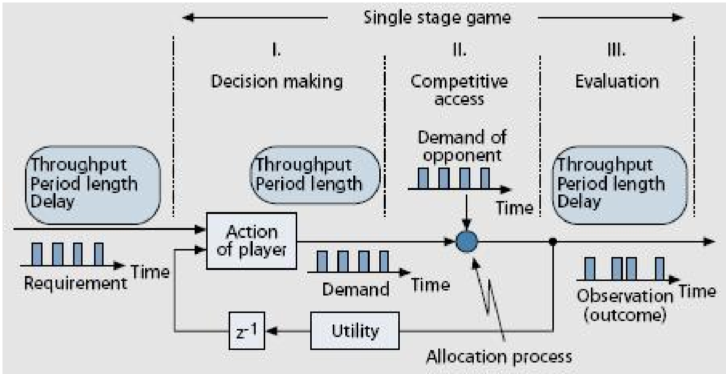

Figure 3.2: Example of Single Stage Game [10]

1. At the beginning the players choose theirs actions and they ask for resource allocation times and duration. This step is instantaneous.
2. Competitive medium access of the allocation process. Each player has asked something but maybe there are some collisions between allocations demanded or the spectrum is already used by an other player. Hence, there will be most of the time a difference between demanded and observed allocations. This step is the one that consumes the most time of the SSG
3. Finaly each players calculates its outcome. We can define the outcome as the difference between what was expected and what has been got [10].

3.3.1 Static forwarder's dilemma game

In this part we are going to explain game theory by using the Forwarder's Dilemma in a static game with a single stage.

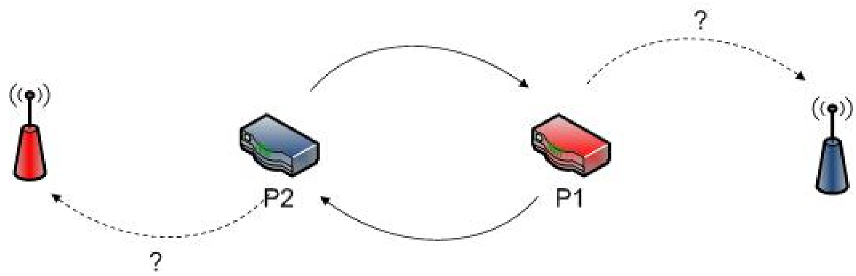

Figure 3.3: Diagram forwarder dilemma

As we have said before in 3.2.2, in a static game players are playing silmutaneously. In the forwarder dilemma they can choose between two actions: forward (F) the packet of the other player or drop it (D). These two actions are the strategies of the player. Which strategy is the best for them ? Let us represent the strategicform in a more convenient approach. We can represent the game by the matrix 3.2.

		Player 2	
		F	D
Player 1	F	(1-C,1-C)	(-C,1)
	D	(1,-C)	(0,0)

Table 3.2: Forward static

In this matrix, p_1 represents the row player and p_2 the column player. As we can see, each cell of the matrix corresponds to a possible combination of the strategies of the players and each pair of values represent respectively the payoffs of players p_1 and p_2

- Reward for packet reaching the destination: 1
- Cost of packet forwarding: C ($0 < C \ll 1$)

Moreover this is a nonzerosum game because the players can mutually increase their outcomes by cooperating. Indeed, if both players forward they will achieve an outcome that is better for both of them than a mutual dropping. In the contrary, in zero-sum game the total outcome to all players in the game, for every combination of strategies, is always equal to zero.

3.3.2 Iterated dominance

After we have expressed the game in a strategic form we need to solve it. It means to be able to predict the strategy of each player considering the information from the game. We still assume players are rationnal and want to get the best payoff from their strategies.

One of the simplest way is to base the game on strict dominance as shown in [11].

Strategy $s_i^{'}$ of player i is said to be stricly dominated by his strategy s_i if:

$$u_i(s_i^{'}, s_{-i}) < u_i(s_i, s_{-i}) \ \ \forall s_{-i} \ \in \ S_{-i} \tag{3.1}$$

Where:

- $-i$ defines all the players, i excepted.
- S_i is the pure strategy space of player i.
- u_i is the utility function that quantifies the outcome for player i given the strategy profile s

Let us apply to game 3.2. From the point of view of player p_1, the D strategy dominates the F strategy. It means that a rational player p_1 will never play the first row of the matrix. We have the same reasoning from the point of view of player p_2. D strategy is still better than F strategy. It means a rational player p_2 will never play the first column of the matrix. Then, the solution of the game is (D,D) and the outcome is (0,0).

As we can see here, we did not get the best solution. Indeed, the pair (F, F) would be better for both players than (D,D). This problem is due to the lack of trust between the players. As we will see later we can add to the game an agreement (trust) between the players and then we will get the best solution.

We cannot use iterated strict dominance techniques to solve every game like the joint packet forwarding game 3.4.

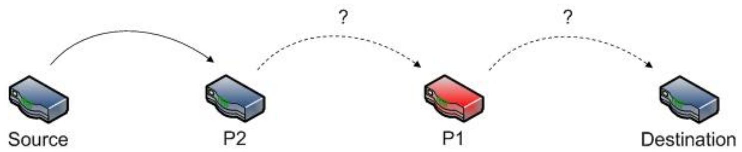

Figure 3.4: Joint packet forwarding game

Now imagine we have two devices. They still have to drop or to forward the package. However they have to decide silmutaneously before the source sends it.

		Player 2	
		F	D
Player 1	F	(1-C,1-C)	(-C,0)
	D	(0,0)	(0,0)

Table 3.3: Joint packet forwarding game

We have not strategy stricly dominating another one. So, we have to define another requirement: weak dominance [11]

Strategy $s_i^{'}$ of player i is said to be stricly dominated by his strategy s_i if:

$$u_i(s_i^{'}, s_{-i}) < u_i(s_i, s_{-i}) \;\; \forall s_{-i} \;\in\; S_{-i} \tag{3.2}$$

From the point of view of player p_2 strategy D is weakly dominated by the strategy F. So, based on iterated weak dominance the solution is the pair (F,F).

Iterated elimination techniques are not working every time to solve a game problem. However they are useful to reduce the size of the matrix. Then it is fastest to find the best strategy.

3.3.3 Nash equilibrium

For example, let us see the Mutiple Access Game represented in the diagram 3.5 and strategic form 3.4:

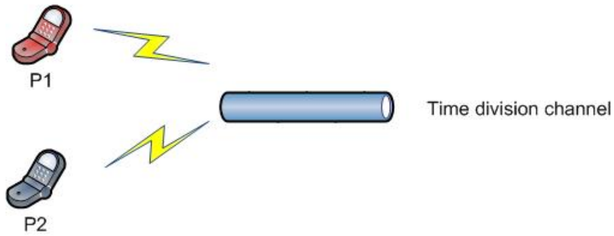

Figure 3.5: Diagram multiple access

		Player 2	
		W	A
Player 1	W	(0,0)	(0,1-C)
	A	(1-C,0)	(-C,-C)

Table 3.4: Multiple Access game

As we can see, player 1 (p_1) and player 2 (p_2) have two differents strategies: access (A) the channel or wait (W). However in the case where both players are transmitting then we will have a collision.

As no strategy is dominated in this game we can not apply previous techniques to solve the problem. So, we are going to define what we call the best response. If p_1 is transmitting the best response (the best payoff) for p_2 is to wait. Same thing if p_2 starts to play.

As shown in [11] ,the best response $br_i(s_{-i})$ of player i to the profile of strategies s_{-i} is a strategy s_i such that:

$$br_i(s_{-i}) = \arg\max u_i(s_i, s_{-i}) \; where \; s_i \in S_i \tag{3.3}$$

So, if two strategies are mutual best responses to each other, then no rational player would have a reason to deviate from the given strategy profile. In the Multiple Access Game, two strategy profiles exist with the above property: (W, A) and (A, W). To identify such strategy profiles in general, Nash introduced the concept of Nash equilibrium [12]. We can formally define the concept of Nash equilibrium (NE) as follows:

The pure strategy profile s^* constitutes a Nash equilibrium if, for each player i:

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*), \forall s_i \in S_i \tag{3.4}$$

In a Nash Equilibrium no player can increase its utility by deviating unilaterally. Moreover, a Nash equilibrium is a strategy profile comprised of mutual best response of the players

If we apply Nash equilibrium in 3.4 we will find two pure strategies (W, A) and (A,W). So, how to choose between several Nash equilibria ?

3.3.4 Pareto optimality

There exist one method for identifying the desired Nash equilibrium point in a game. We have to compare strategy profiles using the concept of Pareto-optimality. To introduce this concept, let us first define Pareto-superiority:

The strategy profile s is Pareto-superior to the strategy profile s' if for any player $i \in N$ [11]:

$$u_i(s_i, s_{-i}) > u_i(s_i^{'}, s_{-i}^{'}) \tag{3.5}$$

with strict inequality for at least one player.

In other words, the strategy profile s is Pareto-superior to the strategy profile s' , if the payoff of a player i can be increased by changing from s' to s without decreasing the payoff of other players. The strategy profile s' is defined as Pareto-inferior to the strategy profile s.

The strategy profile s^{po} is Pareto-optimal (or Pareto-efficient) if there exists no other strategy profile that is Pareto-superior to s^{po}. In other words a strategy profile is Pareto-optimal if it is not possible to increase the payoff of any player without decreasing the payoff of another player

An SSG outcome is called Pareto-optimal or pareto efficient if neither player can gain higher utility without decreasing the utility of at least one other player.

Using the concept of Pareto-optimality, we can eliminate the Nash equilibria that can be improved by changing to a more efficient (i.e. Pareto-superior) strategy profile. The game can have several Pareto-optimal strategy profiles and the set of these profiles is called the Paretofrontier. It is important to stress that a Pareto-optimal strategy profile is not necessarily a Nash equilibrium. We can now use the concept of Pareto-optimality to study the efficiency of pure-strategy Nash equilibria in our running examples [9].

- In the Forwarder's Dilemma game 3.2, the Nash equilibrium (D, D) is not Pareto-optimal. The strategy profiles (F,F), (F, D) and (D, F) are Pareto-optimal, but not Nash equilibria.
- In the Joint Packet Forwarding game 3.3, both strategy profiles (F, F) and (D, D) are Nash equilibria, but only (F,F) is Pareto-optimal.
- In the Multiple Access Game 3.4, both pure-strategy profiles (A, W) and (W, A) are Nash equilibria and Pareto optimal.

3.4 Multi Stage Game (MSG)

Basically Multi Stage Game (MSG) represents several Single Stage Game repeated. The players interact several times and each interaction is called a stage. MSG can be regarded as infinite only if after each period the players believe that the game will continue for an additional period.

So far, we have assumed that the players interact only one time and we modeled this interaction in a static game with the help of the strategic form. In this section, we assume that the players interact several times and hence we model their interaction using a repeated game. Repeated games are a subset of dynamic games and can be expressed in both strategic and extensive form.

In this game players take into account the instantaneous stage but also the effects of their decisions on the utilities of future stages; but they put higher weight to the present utility than the potential utilities in the next stages. A known approach is to use a discounting factor θ, $0 < \theta < 1$.

If θ is near to 1 it implies that future utilities are considered similar to the utility of the current stage. Thus, player should cooperate to reach on a long term a high utility. On the contrary, if θ is near to 0 it means that player should focus on the present utility and then it results in a uncooperative defection. So, that players completely neglect potential future utilities [10].

Now, we are going to use the Forwarder's dilemma (3.3) in a repeated game. We assume the following information:

- move $m_i(t)$: decision in one interaction of player i at stage t
- initial move: the first move with no history
- strategy: defines how to choose the next move, given the previous moves
- history $h(t)$: the ordered set of moves in previous stages
- all past moves are common knowledge at each stage t

We can formally write the history h(t) as follows:

$$h(t) = (m_1(t); ....; m_i(t)); ....; (m_1(0); ....; m_i(0)) \tag{3.6}$$

For example, at the beginning of the third stage of the Repeated Forwarder's Dilemma, if both players have always been defective then the history will be $h(2) = (D; D); (D; D)$. The strategy s_i defines a move for player i in the next stage $(t + 1)$ for a given history $h(t)$ of the game.

$$m_i(t + 1) = s_i(h(t)) \tag{3.7}$$

The payoff in the repeated game might change as well. In repeated games, the users typically want to maximize their payoff for the whole duration T of the game. Hence, they maximize:

$$u_i = \sum_{t=0}^{T} u_i(t, s) \tag{3.8}$$

where $u_i(t; s)$ denotes the stage payoff, the payoff player i receives in stage t.

In some cases, the objective of the players in the repeated game can be to maximize their payoffs only for the next stage (i.e., as if they played a static game). We refer to these games as myopic games as the players are short-sighted optimizers. If the players maximize their total payoff during the game, we call it a long-sighted game. Recall that we refer to a finite-horizon game if the number of stages T is finite. Otherwise, we refer to an infinite-horizon game ([9] and [11]).

In the game theory literature, infinite-horizon games with discounting are used to model a finite-horizon game in which the players are not aware of the duration of the game. Clearly, this is often the case in strategic interactions, in particular in networking operations. In order to model the unpredictable end of the game, one decreases the value of future stage payoffs.

3.4.1 Static strategies

There are two basics static strategies [4]:

- The cooperation strategy (COOP)
- The defection strategy (DEF)

In COOP, player will always cooperate. On contrary, in DEF player will have a defection behavior every time. Theses strategies are independent of the opponent's influence on the player's utility. So static strategies are the continuous applications of one behavior. Note that COOP strategy is better if both operators are cooperating.

3.4.2 Dynamic strategies

One of the most important class of strategies in repeated games is the class of trigger strategies [10]. They consist of strategies where a new behavior starts as soon as a new condition is detected. We are going to introduce two examples of trigger strategies:

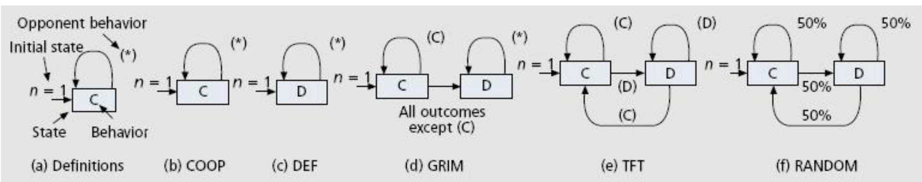

Figure 3.6: Strategies [10]

- Tit-for-tat. Start cooperating. After each round play what your opponent played last round
- Grim. Start cooperating. If the opponent defect for one round, defect forever.

The main difference between these approaches is how long the player will keep the punishment. In Tit for Tat, player will use defect strategy until his opponents want to come back into cooperative strategy. In Grim, one defection will result in a punishment forever. Even in non-cooperative game the fear of future punishment can lead to cooperative behavior [4].

Player should use TFT. Because of a potential punishment it motivates the others players to cooperate. Moreover, in a non-cooperative environment is more adaptive than GRIM.

3.5 Reputation in a decentralized system

One of the strongest assumption in game theory is players are rational. It means they want to maximize their benefit. In our simulation we assume that players want to be fair and respect the policies. So, what is going to happen if one player does not agree with the rules and want to be selfish ? Others players should punish him for its unfair behavior.

A player who plays the same game repeatedly may try to develop a stategy based on reputation. The idea is that if the player always plays in the same way, his opponents will expect him to play that way in the future and will adjust their own play accordingly [13]. It means they have to know the difference between fair and unfair behavior. In a decentralized system we do not know anything about other players like requirements, PRBs used. We are in a game with incomplete information. However we can have some information from interference level. So, we can define unfair behavior when a user is using more than 80% of the spectrum.

Basically if the reputation of a player B is too bad, A can punish him by using more PRBs or all the spectrum in order to interfer on B and to decrease his PRBs quality.

Most prominent games are history - 1 games (players consider only the previous stage). In a reputation system we have to take into account all the previous stages. Of course the last stages are more important than the old ones. If a player has been fair for a long time and he is starting to be unfair we have to be careful and be ready to punish him.

System based on reputation are a part of cognitive radio. There is a learning part and a strategy from historical behavior.

3.6 Game theory limitations

Above, we have presented why game theory is important for our case study. So far, we have made some assumptions about players, game and payoff function. We are going to discuss more deeply about these assumptions.

In wireless networks, the users do not interact with each other on such a fine-granularity basis as forwarding one packet or accessing the channel once. Moreover operators program their devices and base stations to follow a protocol. So we can assume on a long term that these devices are like rational players in our scenario. They do not change after each stage their strategies. [9]

Another problem is how to model payoff function. We have only spoken about a packet to transmit in our previous example. We did not speak for example about the benefits and costs in energy or pricing to get an access to the spectrum. As we will see later in our application we model payoff function as the throughput and later on spectral efficiency with a utility function.

3.7 Game in incomplete information

One of the strongest assumption most of the time used in game theory is to assume we are in complete and perfect information. It means that each player is aware on his influence over the strategies and behavior of the other players. Is it true in wireless network ? In a centralized architecture we can assume this kind of information. However in a decentralized architecture we cannot do that. As we will see later we have some information about the environment but we do not know everything about the other players. Now, we are going to introduce uncoupled game to use game theory with incomplete and imperfect information.

3.7.1 Uncoupled game

In this section we are going to discuss about uncoupled game and then we will see an example.

A learning rule is uncoupled if a player does not condition his strategy on the opponent's payoffs [14]. It is radically uncoupled if a player does not condition his strategy on the opponent's actions or payoffs. We demonstrate a family of simple, radically uncoupled learning rules whose period-by-period behavior comes arbitrarily close to Nash equilibrium behavior in any finite two-persons game. [15]

3.7.2 Regret testing

Regret testing does not depend on observation of the opponent's pattern of play or even on knowledge of the opponent's existence; it depends only on summary statistics of a player's own realized payoffs. In this sense it is similar in spirit to reinforcement and aspiration learning. Response rules that depend only on a player's own payoffs are said to be radically uncoupled.

For example we consider an individual living alone ([15] and [14]). This player has m possible actions. Each action is written on tickets that are stored in a hat. Each hat contains h_m tickets. So, the hat is generating probability distributions over actions. Every probability distribution that is expressible in integer multiples of $1/h$ is represented by exactly one hat. The larger is h, the more closely can any given distribution be approximated by one of these hats. A day consist of s period and we assume s large.

- Step 1 : For each period the player takes a ticket into the current hat and he has to do the action written on the ticket. Then he puts back the ticket into the hat.
- Step 2 : At random times this routine is interrupted by telephone calls. During a call he chooses randomly an action instead of taking a ticket into the hat.
- Step 3: For each action he has done the player receives a payoff. At the end of the day t he calculates the average payoff called b_t. This payoff is the result of actions whenever he was not on the phone. Let us called $b_{j,t}$ the average payoff whenever he was on the phone. So for each action j he compares both average payoff.
- Step 4: If the difference $r_{j,t} = b_{j,t} - b_t > 0$ where 0 is his tolerance level he chooses a new hat. Each hat has the same probability (positive) to be choosen. If $r_{j,t} < 0$ he plays again the next day with the same hat $t + 1$

The previous protocol is called in game theory a regret testing rule. $b_{j,t} =$ is a statical estimate of the payoff on day t, what the player would have received if he had played the action j all the day. $r_{j,t}$ is the regret estimation from not having play the action when player was on the phone. As we are in a game with incomplete information player does not know anything about opponent's actions. So, the regret cannot be directly evaluated.

The logic is simple: if one of the payoff-averages $b_{j,t}$ during the experimental periods is significantly larger than the average payoff in the non-experimental periods, the player becomes dissatisfied and chooses a new strategy, i.e., a new hat from the shelf. Otherwise, out of inertia, he sticks with his current strategy. [16]

The main point with regret testing is *inertia*. Without any particular reason to change, players keep the same strategy as before. So, while information are collected (learning process) the strategy does not change. Player changes his strategy only if a significant improvement is possible. In other words the alternative payoffs should significantly exceed the current average payoff.

As we will see later in part 4 we will use the regret part in one of ours algorithm 4.2.6 to solve the game.

Chapter 4

Game definition

4.1 Game overview

Case study: Spectrum sharing between two operators in the same geographical area.
Goals: reach a fair and efficient spectrum sharing through a game theory based algorithm.

4.1.1 Assumptions

- Intra-system FSU: same RAT, same frequency band shared by different operators.
- Horizontal sharing: all operators have the equal right access to the spectrum.
- Fractional load: each operator is requiring a fraction of the full bandwidth.
- No RAN sharing: every eNB is owned by a different operator.
- Decentralized architecture: there is not a central entity and there is no signaling between the operators.
- Policy: operators have to agree on the same set of rules (number of PRBs per user, spectrum load limit).
- Operators are cooperative. They do not try to be selfish.

PHY layer assumptions:

- Access scheme : Downlink OFDMA, Uplink SC- FDMA (or DFTS-OFDMA).
- Duplexing: TDD (downlink/uplink).
- Frequency reuse factor: one (all cells in the network use the same frequency band, in other words we have a non-orthogonal spectrum allocation among operators).

- Synchronization : yes.

In our work we have often referred to an operator as an eNB because we assumed that each eNB is owned by a different operator.

4.1.2 Game formulation

The allocation process can be define as a game as follows.
The players are the eNBs, users do not participate in the allocation process thus they are not considered players in the game.
The possible actions are the different possible amount of PRBs that the player decides to give to each served user.
Formally:

$$G = M, A, u_i \tag{4.1}$$

where M is the set of player, in our case $M = 1, 2$ or $M = 1, 2, 3, 4$. A is the set of action for player i and u_i is the utility function, defined later.
The outcome of the game is to assign a certain amount of PRBs to each operator. Each stage of our game lasts two frames. Therefore it is a multi-stage game. Moreover it is also a dynamic game with incomplete information, as we are going to explain.

4.1.3 Dynamic game

As we have seen in 3.2.2, in a dynamic game players don't make their decision simultaneously.
Therefore in our case one player starts with his own allocation and the other has to make his decision on top of the previous one's move. Thus the previous one's action influenced the actual player decision by the means of the interference, because, as explained later in 4.2.2, a player cannot choose every PRBs, but only the available ones. Furthermore to not play does not mean wasting the opportunity to allocate some PRBs in the actual frame: it means only that for the actual stage the "non" player keeps his previous allocation. The flowchart for the dynamic game is shown in figure 4.1.

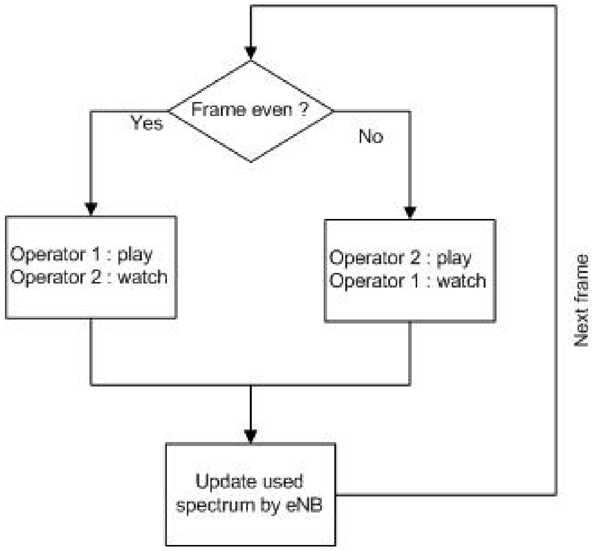

Figure 4.1: Dynamic game

After two frames both the operators make their decisions then a stage is ended.
With this approach we firstly allocate the interference free PRBs and after we will have some overlapping allocation and thus some PRBs shared between different operators.
In appendix B we explain why we have choosen to use a dynamic game instead of a static game.

4.1.4 Incomplete information game

As discussed in 4.1.1 we are in a decentralized approach. Therefore there is not any central entity and there is no signaling between operators. Thus a player cannot state how many PRBs the others have currently allocated. The player cannot guess neither from the interference vector the others' allocation. Thus we have no information on the other's moves (allocation) as well as on the other's payoff. This kind of games are also defined as incomplete information games.

4.1.5 Utility function

A very important step in the definition of a game is to shape an appropriate utility function.
We look for a definition of a utility function such that a eNB can maximize a global utility (as the network total throughput) by only trying to maximize his own utility. We want as well that this utility function leads to a fair and efficient situation. To incorporate the efficiency, the utility function of each eNB is defined such that the eNB can reach the maximum throughput spending the minimum amount of PRBs.
Indeed we observe that it is not true that using more PRBs an eNB can always increase his own throughput. Certainly at the beginning increasing the number

of used PRBs we can see a huge throughput improvement because the eNB is using the interference free PRBs. But at a certain point the eNBs begin to interfere too much each other and increasing the number of PRBs they would have the same or even a worse throughput.
Based on the previous consideration the utility function is of the form given in the equation 4.2:

$$Utility = \frac{(f_1)^g}{f_2} \tag{4.2}$$

Function f_1 quantifies the eNB's actual total sum throughput. The total sum throughput is defined as the sum of users' throughput.

Function f_2 is the total amount of PRBs used by the eNB in the current frame. The total amount of PRBs is simply computed as the number of PRBs per user multiply by the actual operator's number of users.

Coefficient g is a weighting factor. This weighting factor is important in order to achieve a complete spectrum exploitation and a higher throughput in a two-players game. By the means of simulation results we fixed this weighting factor to 2.
Therefore the numerator of the utility function is intended to increase the eNB's throughput. The denominator induces some benefit to being nice to other users as well as it increases the spectral efficiency. Indeed reducing the spectrum exploitation we will reduce the interference to the others. For that reason it is intended to contribute to the fairness in the network as well as to the total spectral efficiency.

4.2 Algorithm

In this section we present our algorithm that, by the means of game theory, achieves a fair and efficient spectral allocation. We will present as well some algorithms used in our work as reference cases in order to show how a game theoretic approach adds some improvement to these basic algorithms.
Before we discuss these algorithms we need to define some general concepts.

4.2.1 Fairness

One of our goals is to have a spectrum shared in a fair way. In this section we are going to discuss about the difficulty to define what we call fairness in our case.

Spectrum load used per operators:
Firstly, we can define the fairness as the spectrum load used by each operators (table 4.1). Let us imagine the scenario 4.1.

As we can see operator A and B have exactly the same number of PRBs used (same spectrum load used). However, the number of PRBs per user is really different from an operator to another one. In a FSU scenario this case is unfair. Operator B has more users than operator A so it should have an higher total number of PRBs.

Operator	Number of users	PRBs per user	PRBs used
A	5	12	60
B	10	6	60

Table 4.1: Fairness: spectrum load used

The number of PRBs per user:
We can also define fairness as the number of PRBs per user (table 4.2). It means each operator should have the same amount of PRBs per user. Let us imagine the scenario 4.2.

Operator	Number of users	PRBs per user	PRBs used
A	5	12	60
B	10	12	120

Table 4.2: Fairness: PRBs per user

In our example A is using less than 50% of the spectrum ($5 * 12 = 60$ PRBs and 50% of $125 = 62.5$) and B is using the 96% of the spectrum ($10 * 12 = 120$ PRBs and 96% of $125 = 120$). Is it normal that an operator uses more spectrum than another one? If we assume they have paid for the same frequency bandwidth this scenario is unfair.

Ratio:
We can also define the fairness as the ratio between the number of PRBs per user and the number of users (table 4.3). Then a game will be fair if both operators have the same ratio to the end.

Operator	Number of users	PRBs per user	PRBs used	Ratio
A	6	9	54	$\frac{9}{6} = 1.5$
B	8	12	96	$\frac{12}{8} = 1.5$

Table 4.3: Fairness: ratio (PRBs per user/ Number of users)

The decentralized problem:
In a centralized case, we are able to calculate each stage the ratio, the number of PRBs per user or the spectrum load used for both operators and then we can base our strategies on these information. However, from our problem definition we focus on a decentralized system, so operator A does not know anything about the allocation of operator B. It does not know if it can increase or decrease its requirements to respect the fairness. Moreover we want to reach fairness on a long term and not on frame basis.
Therefore we define fair a situation where the operator with more users is using more PRBs than the operator with less users on a frame basis. Moreover on a long term basis they should have on average the same number of PRBs per

user. Thus, if they have on average the same number of users then they should have the same average of spectrum load used (see table 4.4).

Operator	Average number of users	Average PRBs per user	Average PRBs used
A	7.3	9.2	67.16
B	7.1	9.7	68.87

Table 4.4: Fairness in a decentralized system

So, we can check the average numbers of PRBs per user and the average spectrum load used to see if our approach is fair for both operators.

4.2.2 Available PRBs

We define the available PRBs as the PRBs in which the sensed interference is under a certain threshold. An operator can allocate all or part of the available PRBs (i.e. it cannot allocate more than 100 PRBs because of the spectrum limit, as shown in figure 4.2), but cannot use the non available PRBs. Moreover this threshold is the same for all the operators.

SINR problem

Our first idea to select the possible available PRBs for each operator was to base this selection on the SINR values.
In doing this we realized that in order to have an SINR value for each PRBs each operator has to allocate it, otherwise we will have no information about its quality. It is true that an eNB can send a dummy signal in order to test the PRB quality before to allocate it, but this solution is not efficient at all. In fact we will waste the spectrum only to know its quality.
Another solution would be to define the available PRBs from the SINR received by the previous allocation process. In other words the PRBs that we can allocate, are a subset of the previous ones. In particular they are the ones over a defined SINR threshold. But in such a way selection after selection the number of allocated PRBs decreases drastically.
This lead us to move for the interference.

Interference

First of all we can start stating that in our systems there is no intra-cell interference because we allocate the PRBs orthogonally within a single cell. In fact we are using OFDMA as access techniques. Moreover we are using TDD, as duplexing scheme, with perfect synchronization thus we have neither intra-cell interference between the uplink and the downlink.
Therefore all the received interference comes from neighboring cells in the same frequency band. It is the so called inter-cell interference.

Our problem is at this point to define who should sense the interference all over the spectrum.
We could make users sense it but then we would have several different interference vectors. In fact users can sense the interference from their perspective that is highly dependent on their position. Then in downlink channel it is not

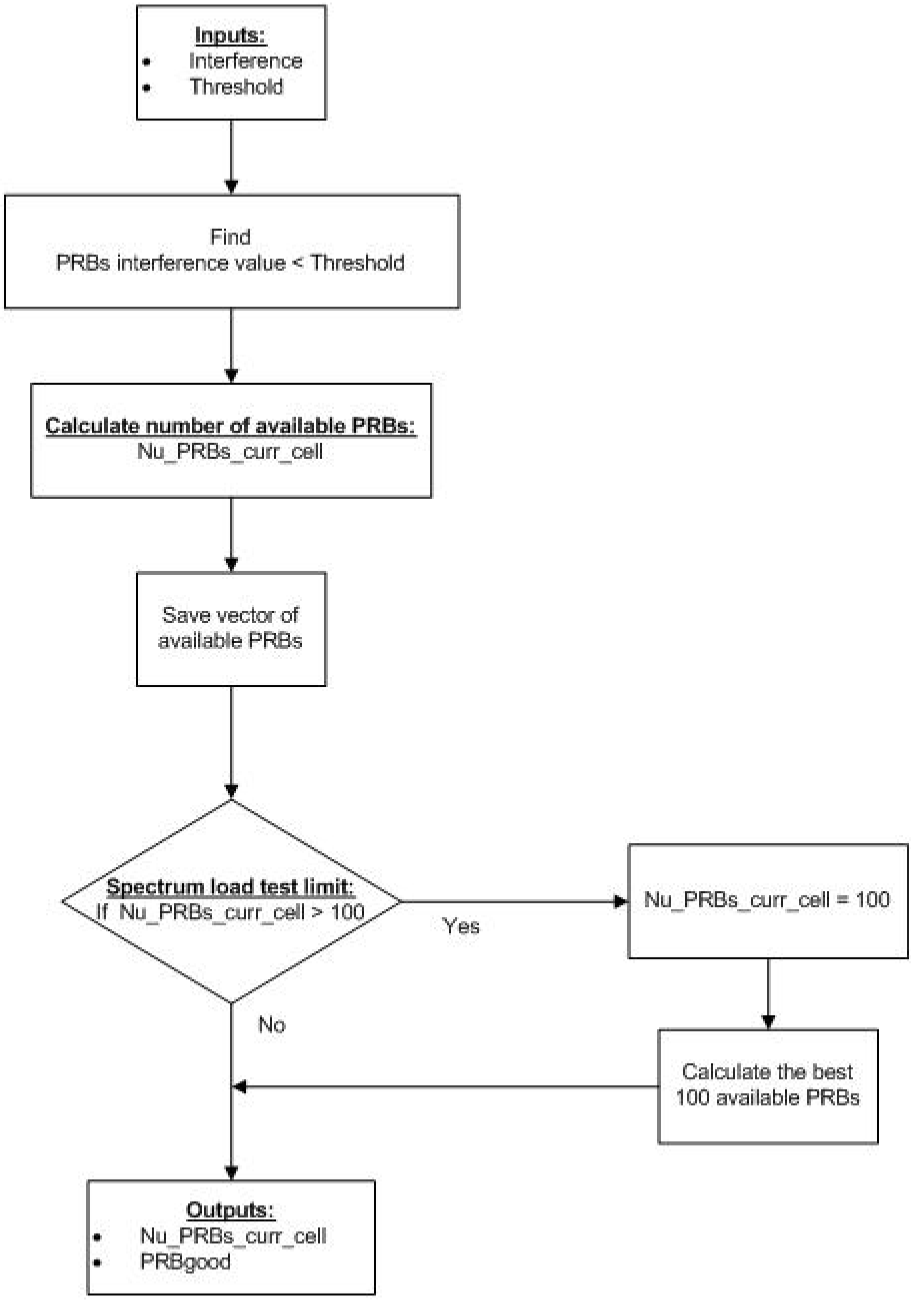

Figure 4.2: PRBs available

true that if a PRB is interference free for one user it will be interference free for another one. The figure can help us to explain this concept.

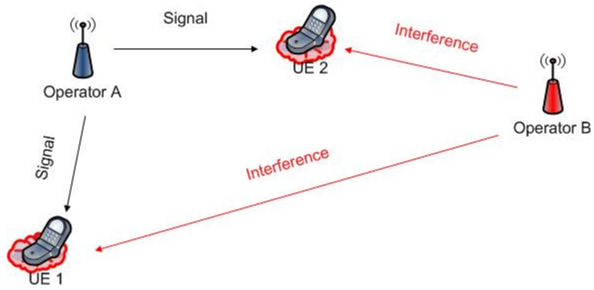

Figure 4.3: Downlink

As we can see in figure 4.3, in dowlink case the interference sensed by UE 2 is not the same as the one sensed by UE 1, because they are in different position with respect to the interfering eNB (operator B).
Therefore these different sensed interference vectors should be sent to their serving eNB (operator A). Operator A should do a sort of average or even something smarter to have a single interference value for each PRBs. This approach can be also power wasting for a user equipment.
However it is not the aim of our project to define some smart algorithms to use the interference vectors sensed by users.
Thus we decide to make the eNB sense all the spectrum and define the interference vector that we will use to state the total amount of available PRBs per eNB. In this case this sensed interference is generally valid for every users in the uplink channel.

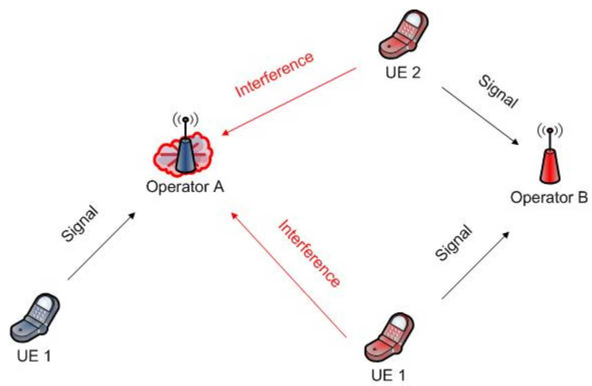

Figure 4.4: Uplink

In fact, as we can see in figure 4.4, the interference in the uplink case is received at the eNB. Therefore if the eNB senses interference in a certain PRB, this PRB is interfered whatever user will use it to transmit his information signal to the eNB.

In conclusion for the previous reasons we choose to sense the uplink channel and according to this choice all our simulation results are referred to the uplink case.
However once defined an appropriate interference vector, our algorithms can be run as well for the downlink case.

4.2.3 Protocol definition

In this section we discuss the general protocol we are using. Our game theory based algorithm as well as all the other discussed algorithms are the core section of this general protocol.
First of all a new game starts whenever the number of users change for one or both the operators.
At the beginning of each stage the actual player is supposed to have as an input his own number of users and a vector stating the interference per PRBs. As explained before, this interference is the sensed interference on the uplink channel coming from the previous frames allocation. Moreover it is still consistent because the opponent is keeping the previous allocation.
In what follows we define the steps of our main protocol.
Protocol steps:

1. The player has to determine if it is his turn to play. For each frame we make only one eNB "playing", therefore only one eNB can change his allocation and the others can only keep the same allocation of the previous frame. The following frame they will switch their roles and so on until the end of the game.

2. The player has to determine how many PRBs are available on the basis of the interference vector. An appropriate threshold is defined in order to allow overlapping allocation and maximize the spectral exploitation. All the PRBs below this threshold can be allocated by the actual player. However also if there are many PRBs available not all of them will be used. In fact we define as a part of our policy a spectral load limit in order to provide fairness: each eNB can use only the 80% of the spectrum as a maximum. Therefore whenever the actual number of available PRBs exceeds the 80% limit (there are more then 100 PRBs) it will be set to 100. In particular they will be the best 100 PRBs in term of interference.

3. At this point we define the number of PRBs per user. Here our algorithm is proposed and later we will discuss it in details. We proposed also other solutions in order to have some reference cases. Also if there are some differences between these algorithms and our proposed one they are all based on a same policy: the number of PRBs per user can't be less than 5 or more than 12. Again this policy is set in order to provide fairness between operators.

4. Once computed how many PRBs per user each operator has, the algorithm provides to the simulator the currently allocated PRBs for the actual eNB, computed as the number of PRBs per user multiplied by the actual operator's number of users. These PRBs are chosen within the available

ones. In particular we will choose the PRBs as the best ones in terms of interference. As well the algorithm will provide the allocation vector for each user that belongs to the eNB. This part could be improved: so far we do not improve the users allocation process. (i.e. giving the interference free PRBs to the users in the worst condition).

The flow chart 4.5 can synthesize the previously described general protocol.

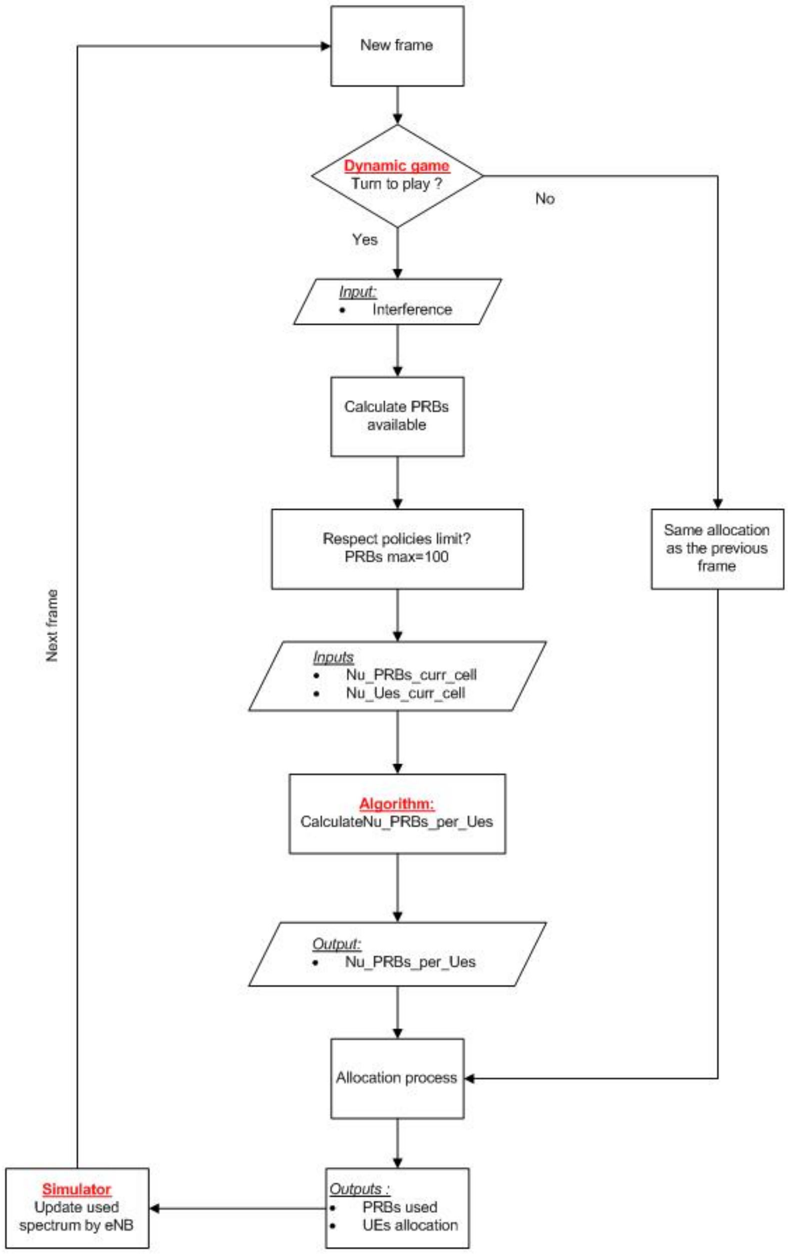

Figure 4.5: Flow chart: Protocol

Algorithm description:
Our goal is to design an algorithm that achieves a fair and efficient allocation. In order to obtain an efficient allocation the spectrum rule must be flexible. Therefore we do not want to assign a fixed amount of PRBs to each operator frame by frame: we want to dynamically change this amount of PRBs according with the actual needs of the operator.

Our idea is simple: if at a certain time one operator has more users than another it is better to give to this operator more resources. Thus when he will have less requirements it will be his turn to be kind to the other and let the other using more resources. Therefore our algorithm provides as an output the number of PRBs per user for each cell in each frame. Then the total amount of PRBs allocated to each operator in each frame is simply computed as the number of PRBs per user multiplies by the actual operator's number of users. This allocation must be fair, in the sense previously defined, and efficient.

The beginning:
The simulator starts running our algorithms from the second frame. Therefore in the first frame we have a blind allocation in order to have a starting point. This allocation is neither fair nor efficient.
As previously discussed the algorithm is initialised with the knowledge of the number of users for the eNB of interest as well as with the interference vector.

4.2.4 Algorithm TMax

First, we worked on a basic algorithm to get another reference case. The basic idea of Tmax is to take as much an operator can get respecting our policies: spectrum load limit, number of PRBs per user and interference threshold.

Basically TMax will set the number of PRBs per user to:

$$Number\ of\ PRBs\ per\ user\ = \left\lfloor \frac{number\ of\ available\ PRBs}{number\ of\ users} \right\rfloor \tag{4.3}$$

As discussed in 4.2.2 the number of available PRBs is the number of PRBs below the interference threshold. Sometimes when the number of PRBs available is low and/or when the requirements are high we cannot allocate a minimum of 5 PRBs per user. So, to solve this problem we drop some users until we can allocate the minumum of PRBs per user. (see flowchart 4.6)

4.2.5 Fair Policies Based Algorithm (FPBA)

Each iteration of the algorithm can be summarised as follows:

Step 1: Inizialization
Set the starting number of PRBs per user.
This step is only for the first two frames: in the second frame the first player will set this starting value as well the second player will set it in the third frame.

1. Start with allocating 6 PRBs per user whatever amount of users the eNB has.

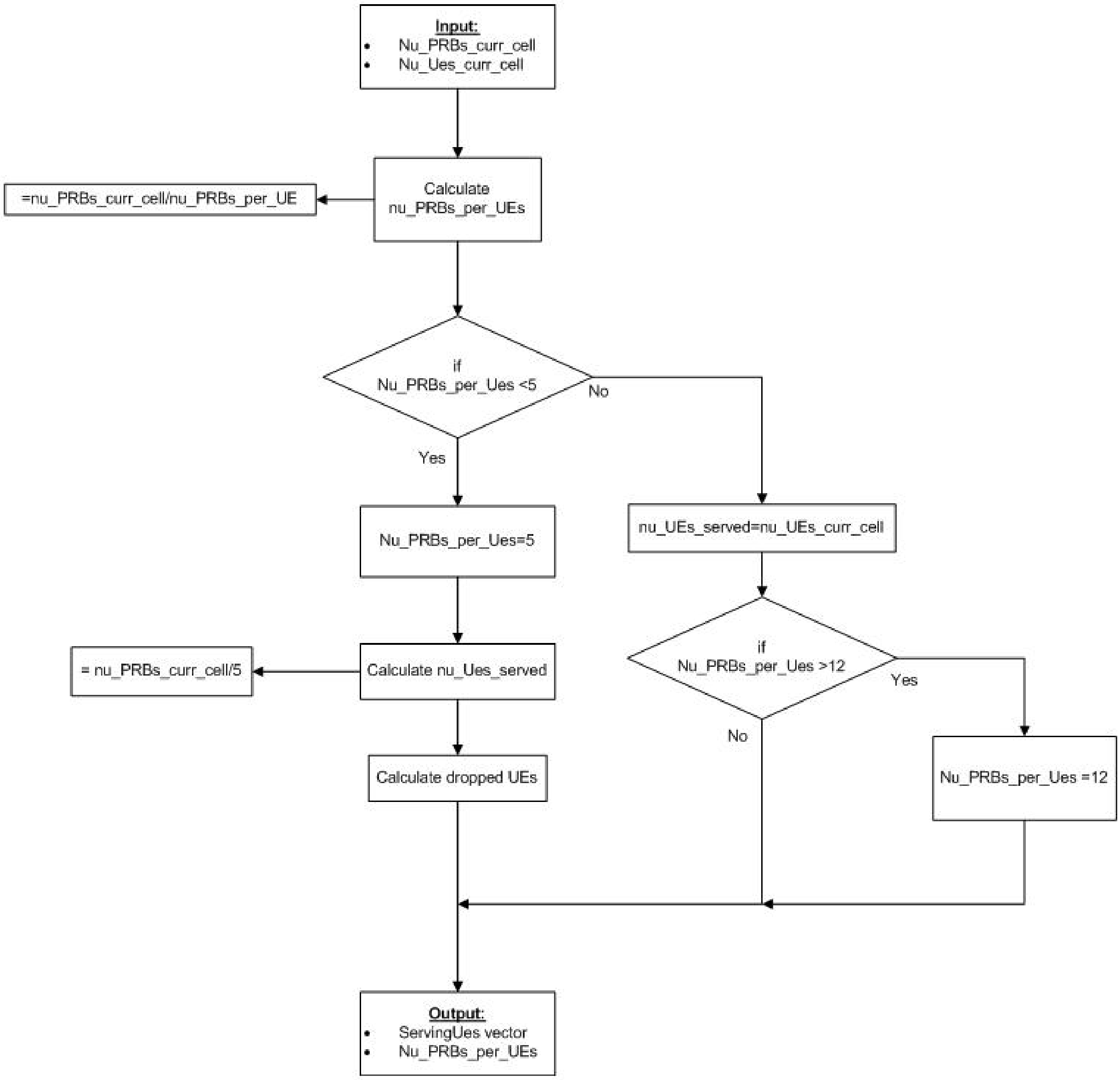

Figure 4.6: Flow chart: TMax

Explanation: this policy is set in order to respect our idea to give more to the eNB with more requirements. In fact if a eNB has more users it will have more allocated PRBs (i.e. 60 PRBs if it has 10 users against 30 PRBs if it has 5 users).

2. It can be possible that the actual player does not have enough PRBs over the interference threshold. In this case the eNB cannot set the number of PRBs per user to the previously discussed value. Therefore it will set the number of PRBs per user to:

$$Number\ of\ PRBs\ per\ user\ = \left\lfloor \frac{number\ of\ available\ PRBs}{number\ of\ users} \right\rfloor \quad (4.4)$$

   As discussed in 4.2.2 the number of available PRBs is the number of PRBs below the interference threshold.

   (a) Check if the number of PRBs per user is under the limit of 5 PRBs per user. If it is true, then set to 5 the number of PRBs per user and drop a user.
   Explanation: This rule is set in order to provide at least 5 PRBs to every user.

Step 2: Increase step by step
From the 3^{rd} frame add two PRBs to the previous number of PRBs per user, whenever you can.

Explanation: this slow increment gives the possibility to the other to increment as well. So it is a rule set to provide some kind of fairness. In this way, in fact, we avoid that a greedy eNB takes as much spectrum as he can, drastically reducing the opponent's available PRBs, also if it has less users than the other. Anyway it is still true that the eNB with more users will be faster than the other getting more PRBs.

1. It can be possible that the actual player does not have enough PRBs below the interference threshold. In this case the eNB cannot give to each user the previous value of PRBs per user plus two new PRBs. Therefore it will try to add only one more PRBs. If there are not still enough PRBs the eNB will set the number of PRBs per user to the result of the equation 4.4.

2. Check if the number of PRBs per user is under the limit of 5 PRBs per user. If it is true, then set to 5 the number of PRBs per user and drop a user.
   Explanation: once again this rule is set in order to provide at least 5 PRBs to every users

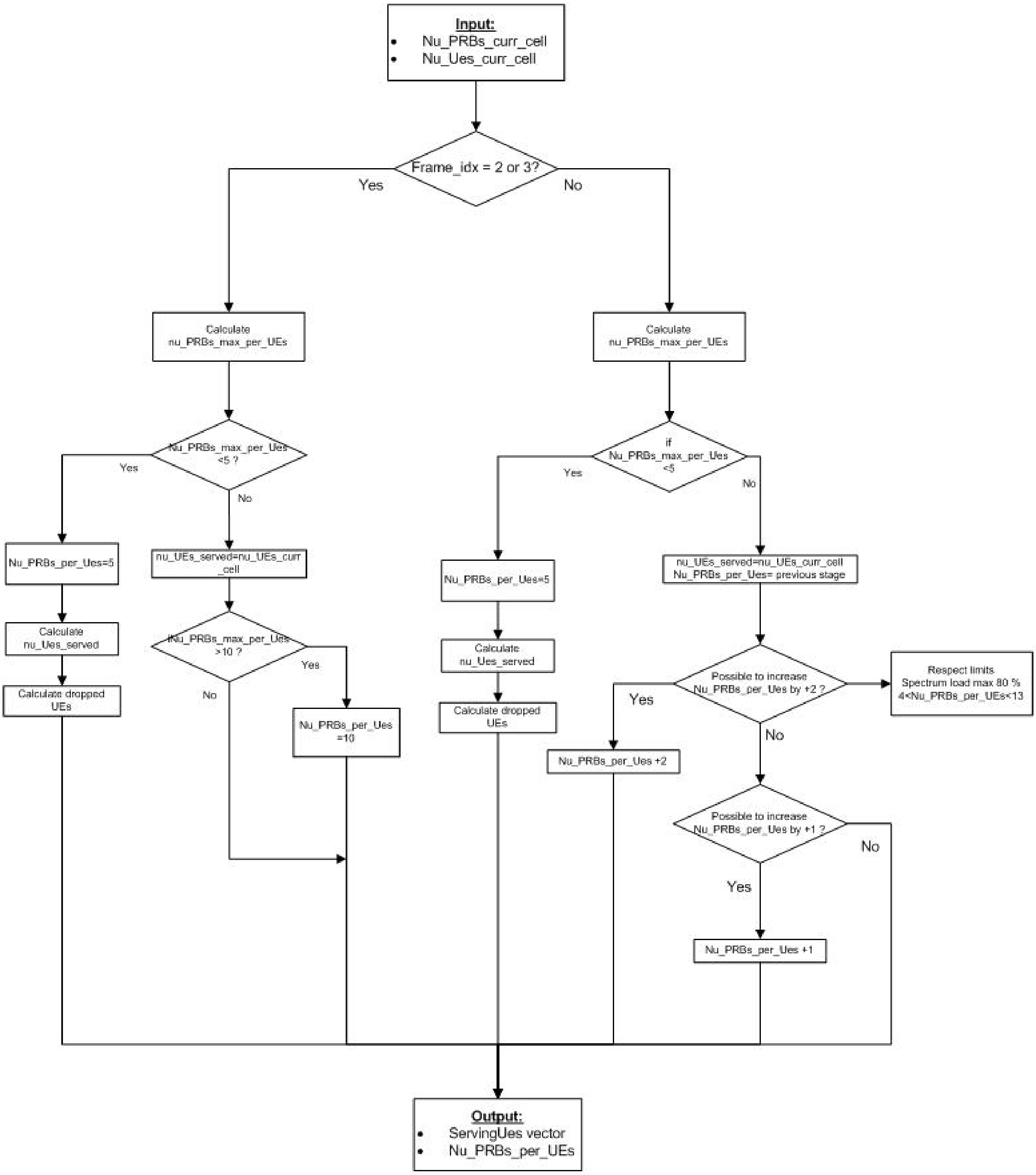

Figure 4.7: Fair Policies Based Algorithm (FPBA)

On one hand it is efficent after some frames when we reach an equilibrium in spectrum load used. It is also fair because each operator has the same opportunities in term of number PRBs available. On another hand during the increasing part not all the spectrum is used.

4.2.6 Game Theory Based Algorithm (GTBA)

In this one we try to achieve the fairness setting some policies and efficiency by the means of game theory. Each iteration of the algorithm can be summarised as follows:

Step 1: Initialization
See FPBA 4.2.5

Step 2: Learning part
From the 3^{rd} till 9^{th} frame add two PRBs to the previous number of PRBs per user, whenever you can. Explanation: see FPBA 4.2.5

Step 3: Revision process
From the 10^{th} frame until the end of the game check out the payoff obtained for each of the previous frames. If the actual payoff is lower than the maximum obtained payoff, change your strategy and choose the number of PRBs chosen in the stage where the maximum payoff is obtained. Moreover do not take into account the first two stages of the game.
Explanation: this is the most important step and it is justified by the means of our game theory approach. Therefore we will explain it in details in the next section. Moreover we do not take into account the first two stages of the game (starting evaluating from the 6^{th} frame) because the realized payoff in the first two stages are not so reliable, in fact it is too few time that our algorithm is running.

In 4.8 you can see a flow chart that synthesizes the previously described algorithm.

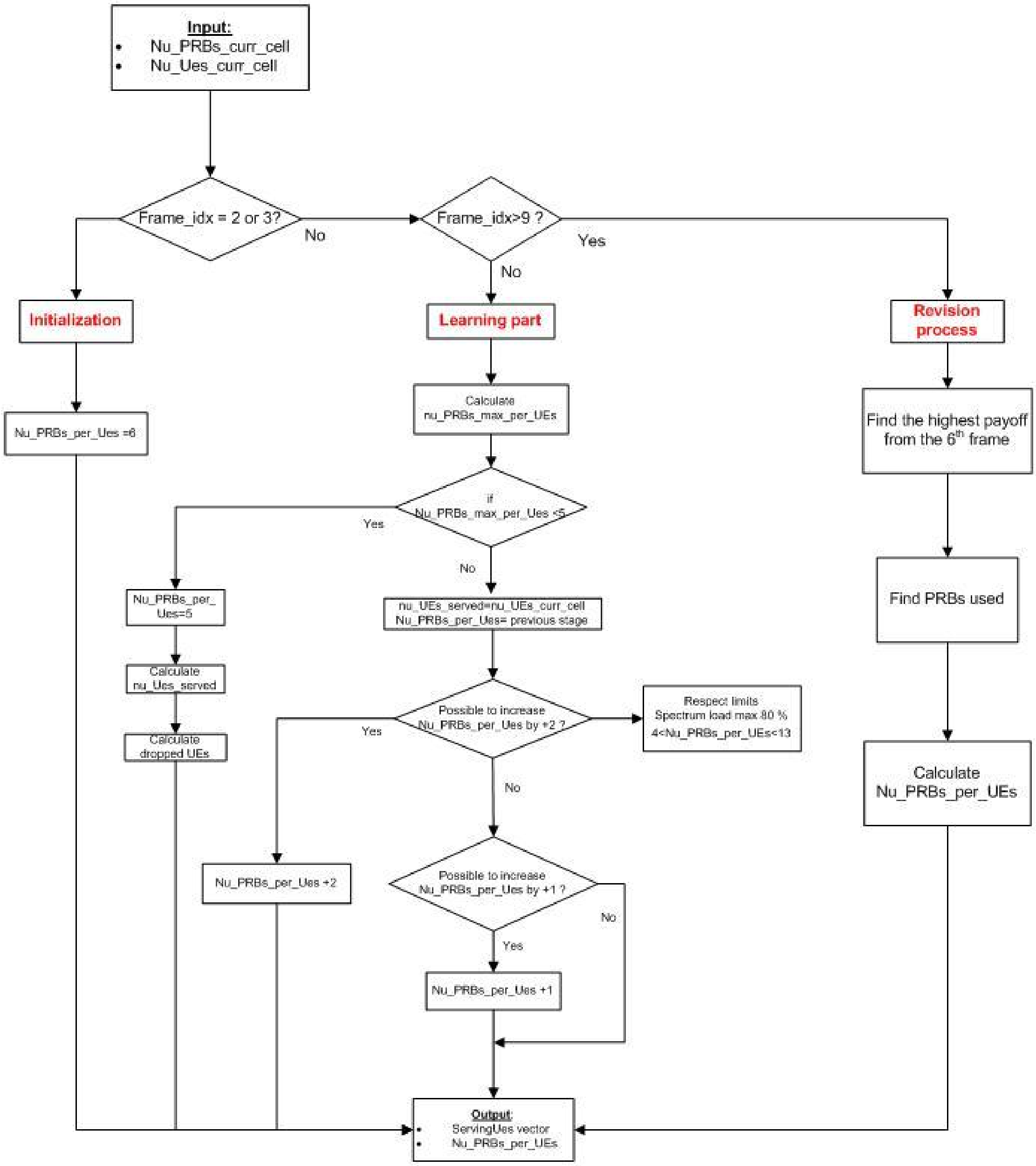

Figure 4.8: Game Theory Based Algorithm (GTBA)

Theoretical validation

As explained in 4.1.4 we are in a incomplete information game. This can lead to situation in which it is impossible to reach a Nash equilibrium (stated in [17]). Starting from this negative result, we tried to find some possible learning rules that lead to Nash equilibrium from an out-of-equilibrium condition.

In [8] is stated that a way to overcome this problem can be to leave the normal rational approach and begin with a sort of random search, trying to use the so called regret testing.
Regret testing is quite useful for us because, as we explained in 3.7.2, it is a method that depends only on a player's realized payoffs and requires no observation of the opponent or even knowledge of the opponent's existence.

Now we are going to explain how we are using this approach matching up the formal regret testing to GTBA. Then by the means of a practical example we will show how it leads to a Nash equilibrium, emphasizing its complete lack of dependence on the actions or payoffs of the opponent.

First of all our players are supposed to be rational, this means that at each stage of the game, they choose the best responses according to the actual model. Furthermore, they do not update after every stage, but only after a substantial amount of stages. With an appropriate choice of parameters, it can be shown that the realized behaviors in such a method reach Nash equilibrium almost every time. Moreover our method leads also to a fair and efficient equilibrium.

As previously defined, GTBA will try different possible allocations from the 2^{nd} until the 9^{th} frame: in regret testing this is the so called experimental period. Every time the player chooses the number of PRBs to allocate he receives a payoff. But he does not know the opponent's actions or payoffs. At the end of this period he will make a decision. This decision is based only on his own observed payoff and it is taken in order to maximize his own utility function. Although this approach is a little bit different from the so called regret testing rule the basic ideas are the same.Indeed a key element of regret testing is the random component that leads to a change in strategy. This random component permits a wide search among different possible actions.

In GTBA this random search is implemented in the following way. As previously discussed the starting number of allocated PRBs per each eNB depends solely on the number of users it has (that can be assumed a number uniformly distributed between 5 and 10). Nevertheless the increasing factor is 2 or 1 PRBs depending on the available PRBs in the actual frame. Therefore different possible allocation are tried within the 1^{st} and the 10^{th} frames. However it is not a random deviation from one strategy it is more a succession of search episodes.

Another key element of regret testing is inertia: if there is no particular reason to change, player continues as before. As well the inertia (see chapter 3.7.2) is present in GTBA: there is no change of actions if this cannot bring an improvement. In fact, a change happens only if the alternative payoff exceeds the current one.

To allow a better understanding of these concepts we can show GTBA in actions by the means of real data obtained from a simulation (table 4.5). As stated in 4.2.3 GTBA is run from the second frame. In this frame the first player will make his own decision based on the number of users he currently has.

For example (table 4.5) he chooses to allocate 48 PRBs because he has 8 users. For this frame the second player simply keeps his previous allocation that comes from the blindly allocated first frame. In the third frame the second player will make his own decision on the top of the opponent's previous move. Therefore it should be possible that he has not enough PRBs available to set the number of PRBs per user to 6. However as we can see this is not the case: he chooses to allocate 42 PRBs, therefore he has 7 users. Now they try to increment their allocated PRBs by step of one or two, depending on the available PRBs. In the 10^{th} frame the first player makes his decision on the basis of his previously observed payoff. In the 11^{th} frame the second player gives his response to the first player's action.

Allocated PRBs 1	48	48	64	64	80	80	88	88	80	80	80	80	80
Allocated PRBs 2	71	42	42	56	56	70	70	77	77	70	70	70	70

Table 4.5: Example of a game

In table 4.6 we show the same game from the first players perspective.

Frame number	Throughput (e+008)	PRBs	Payoff (e+015)
2	1,7940	48	0,6705
3	2,1663	48	0,9776
4	2,4726	64	0,9552
5	3,1639	64	1,1564
6	3,5005	80	1,5317
7	3,0419	80	1,1566
8	3,4672	88	1,3661
9	3,0529	88	1,0591
10	3,1832	80	1,2666
11	3,0419	80	1,1566
12	3,3387	80	1,3934

Table 4.6: Strategic information example 1^{st} player

The first player chooses to allocate 80 PRBs because his payoff is greater than the others considered. As we can see in table 4.6, after a transitory it gets an higher throughput than with the 88 PRBs allocation. Therefore this allocation is more efficient.

Now we show the game from the second player perspective (table 4.7).

As we can see in table 4.7, he chooses to allocate 70 PRBs because his payoff is greater than the others considered (remember we don't consider the payoffs from the beginning). He is the second player so he makes his move on the top of the other's move. Now we will show that also if he does not know the other's move he chose the best response (table 4.8).

Frame number	Throughput (e+008)	PRBs	Payoff (e+015)
2	3,7531	71	0,6705
3	2,0780	42	0,9776
4	2,3322	42	0,9552
5	2,6871	56	1,1564
6	2,4726	56	1,5317
7	3,4658	70	1,1566
8	2,9250	70	1,3661
9	3,4662	77	1,0591
10	3,3251	77	1,2666
11	3,4658	70	1,1566
12	3,1958	70	1,3934

Table 4.7: Strategic information example 2^{nd} player

	56	70	77
80	(1.09 , 1.53)	(1.15, 1.71) - NE	(1.26, 1.43)
88		(1.36, 1.22)	(1.05, 1.56)

Table 4.8: Game in strategic form

As we can see in table 4.8, once the first player chooses to allocate 80 PRBs the best response for the second player is to allocate 70 PRBs.
The equilibrium we reach is also fair (according to our definition of fairness).

Limitations:

1. The first limit of this method is represented by the dependency of players' search episodes. We are in a dynamic game therefore searches are connected via the history of play.
   As we can see in table 4.8 the case (88,63) is not investigate. Moreover if at a certain point of the game one player allocated too many PRBs it can be possible that the other cannot increase anymore his allocated PRBs because of the received interference. Thus there is no guarantee that the joint strategy space will be searched systematically and maybe a more efficient and fair equilibrium should be find.

2. A second limit is that we are in an incomplete information game. Therefore even when the players find an equilibrium, they do not know it. This is because they are unaware of the opponent's payoff, hence they can move away from the equilibrium.

4.2.7 Fair Limit on Average (FLoA)

The basic idea of this algorithm is to have something really fair. So, we set a limit called fair limit in number of PRBs used/ allocated. It means each player is going to increase or decrease its requirements to reach this limit on average and on long term.

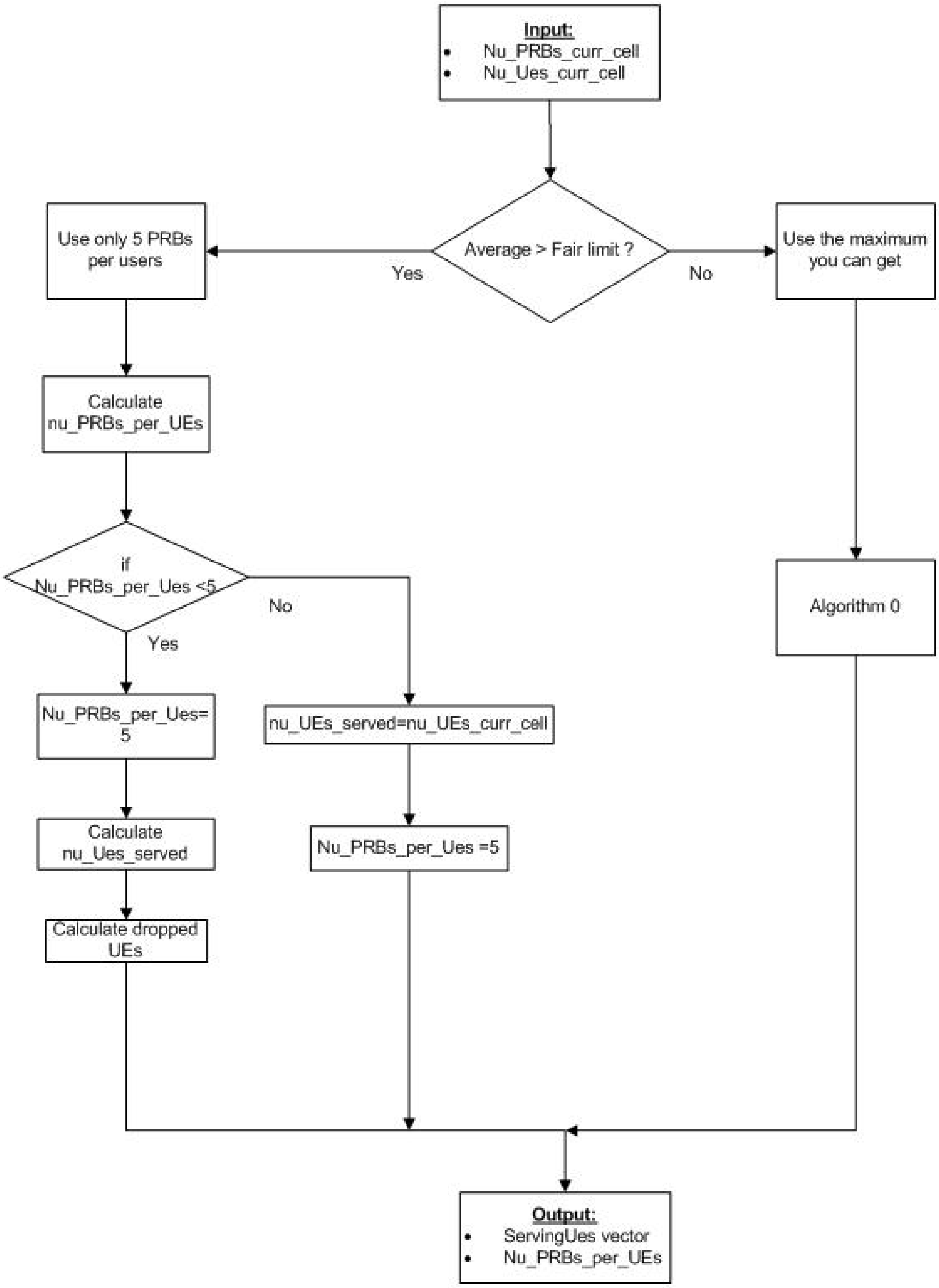

Figure 4.9: Fair Limit on Average (FLoA)

1. Set the fair limit, for example 75 PRBs.
2. From requirements (number of users), the number of PRBs available calculate the number of PRBs per user. It means increase or decrease the previous number of PRBs per user.

The main problem with this algorithm is that not stable at all. Instead of an equilibrium in the number of PRBs used we have what we call yo-yo effect! Indeed when we are close to the fair limit each operator is going to increase and next frame to decrease its requirements to respect the fair limit. To conlude this algorithm takes into account only the fair problem. It is not efficient at all.

4.2.8 Algorithm NoFSU

The basic idea is to give all the spectrum to each operator. They will use a lot of PRBs even if the number of users is low or interferences are high. The problem with this algorithm is we have a lot of overlapping and then a lot of interference between operators. Thus the throughput per cell and users is not so good. This algorithm is one of our references because is not using FSU. Indeed, our goal is to show FSU is better in spectral efficiency as we will see in the chapter 5.

$$Number\ of\ PRBs\ per\ user\ = \left\lfloor \frac{number\ maximum\ of\ PRBs}{number\ of\ users} \right\rfloor \tag{4.5}$$

Where the number maximum of PRBs is equal to 125 as we have seen in chapter 2.4

Chapter 5

Simulation and Results

5.1 Simulation

To study the fairness and the efficiency of our proposed algorithms we used a simulator deployed at Aalborg University and approved by Nokia-Siemens Networks, Aalborg
We integrate our algorithms in the simulator adding flexibility to the basic scheduling version that does not provide any FSU scheme. Therefore our reference cases are the blind random scheduling in frequency domain provided by the simulator in full load mode as well as the fixed spectrum assignment policy.

5.1.1 Scenario

Our simulations are run in an indoor scenario in particular in an office scenario that follows the A1 type specification suggested in [18].

As we can see from figure 5.1 a cell is composed by ten rooms. The eNBs are placed in the middle of each cell, precisely in the corridor and each eNB belongs to a different operator. This is an optimal location for the eNBs and it corresponds to the planned deployment case.
We run the simulations in the case of two as well as four eNBs. Moreover we are in a single floor scenario.
Figure 5.1 is an example of a four eNBs office scenario.

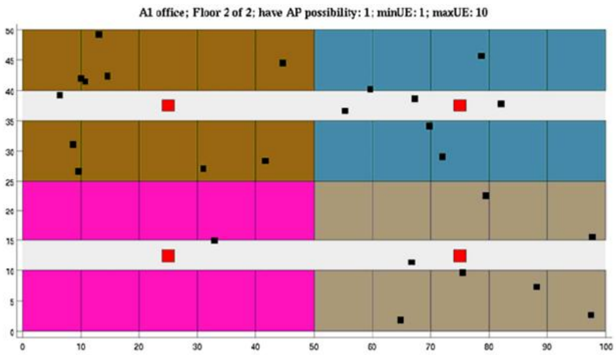

Figure 5.1: Office scenario topography

- Scenario: indoor office
- Number of operators: 2 or 4
- rooms per cell: 10x2 (2 operators) 5x2 (4 operators)
- cell coverage: 100x25 meters (2 operators) 50x25 meters (4 operators)
- Number of users per cell: from a minimum of 5 users up to 10 users

Layout

In the simulator the layout is very flexible. We can choose the number and the position of walls, eNBs and UEs.
The SINR depends a lot on the topography and on the eNBs and UEs deployment.

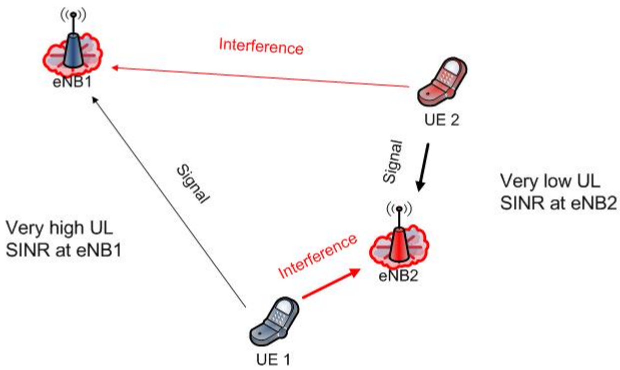

Figure 5.2: Influence of the position on SINR

As shown in figure 5.2 the SINR at eNB 1 and eNB 2 will not be the same. That is why we set the UEs position randomly and we run a lot of time the simulation in order to obtain a realistic SINR distribution. So, we use the following parameters in our simulations:

- Layouts: 30. Random number and position of walls, doors, eNBs; a layout lasts 30 selects.

- Selects: 30. Random number and position of UEs; a select lasts 50 frames.
- Frames: 50.

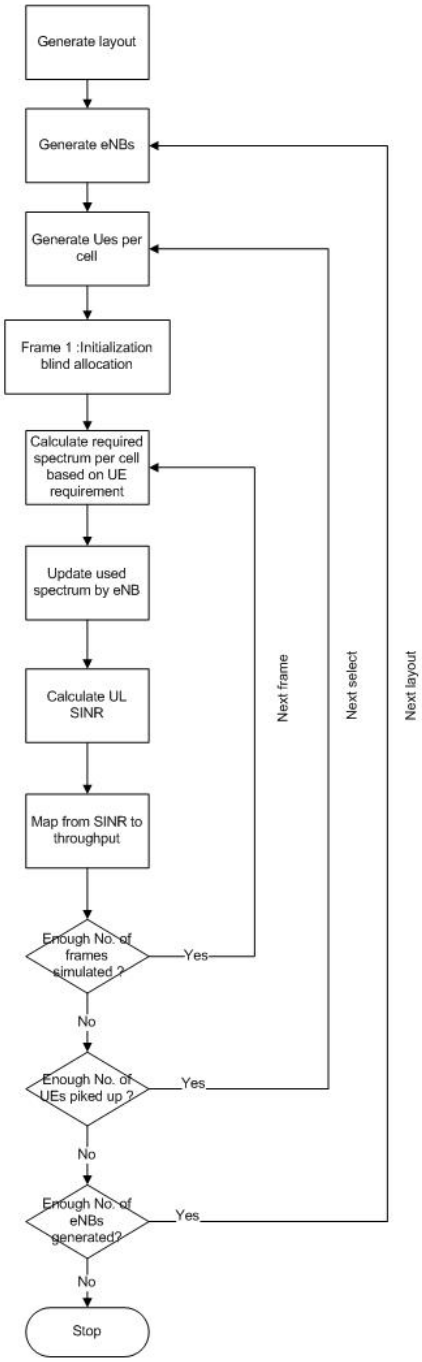

Figure 5.3: Flowchart simulator

As shown in figure 5.3 once the basic layout is generated, eNBs and UEs will be generated. Moreover during each select the users, eNBs and the radio conditions remain constant. Only the PRBs allocation is changed by the means of the FSU algorithm.

UEs and eNB
Users are connected to the eNB in the same cell coverage and they are fixed, they change position only after 50 frames. Each eNB serves from a minimum of 5 users up to a maximum of 10 users.

The eNB's characteristics are:

- trasmission power: from 27 dBm to 30 dBm
- antenna: omni-directional, 3 dBi gain
- height: randomly choosen between 1 and 2.5 meters

The UE's characteristics are:

- trasmission power: from -30 dBm to 24 dBm
- antenna: omni-directional, 0 dBi gain
- height: randomly choosen between 1 and 2.5 meters

Channel model
The simulator models the indoor path-loss according to the A-1 type proposed in [18].
We have both a LOS and a NLOS case. The LOS is corridor-to-corridor and NLOS case is corridor-to-room. In the NLOS case we have a basic path-loss calculation for users in the rooms adjacent to the corridor where the eNB is situated. For users in rooms further away from the corridor a wall-penetration losses is applied.

The following equations summarize the discussed model.

- LOS: $PL = 18.7\ log_{10}\ (d[m]) + 46.8 + 20\ log_{10}\ (f_c[Ghz])/5)$
- NLOS: $PL = 36.8\ log_{10}\ (d[m]) + 43.8 + 20\ log_{10}\ (f_c[Ghz])/5)$
- NLOS with walls penetration factor:
  $PL = 20\ log_{10}\ (d[m]) + 46.4 + 20\ log_{10}\ (f_c[Ghz])/5) + n_w \times L_w[dB]$
  where n_w is the number of walls between the eNB and the UE and L_w is the wall penetration loss factor.

The shadow fading correlation between eNB and users is also computed. It is applied a log-normal model with a standard deviation of 3 for the LOS case, 4 or 6 for the NLOS case depending on the number of walls between users and eNB.

5.1.2 Simulation parameters

Other simulation parameters are:

- Frequency reuse factor: one (all cells in the network use the same frequency band, in other words we have a non-orthogonal spectrum allocation among operators)
- Synchronization : perfect
- Traffic load: fractional (each operator is requiring a fraction of the full bandwidth)
- Systems Bandwidth: 100 Mhz
- Frequency: 3.5 GHz
- Layouts: 30
- Selects: 30
- Frames: 50 (2 operators), 100 (4 operators)

5.1.3 Performance metrics

To evaluate our algorithm performance we will use several kinds of plot: the average cell load, the mean cell throughput, the CDF of the cell throughput and of the user's throughput, the outage user's throughput.

The average cell load plot is particularly important to evaluate the fairness. In fact it states the average allocated number of PRBs per cell, per frames over all the simulations. In particular it provides this result as a percentage of the used PRBs over the total number of PRBs.

The mean cell throughput provides a mean of the total throughput achieved by the eNB frame by frame.

The outage user's throughput is defined as the 5^{th} percentile of the CDF of the user's throughput. In other words it is the minimum throughput achieved by the 95% of the users, so it represents in a certain way the throughput achieved by the user in the worst conditions.

5.1.4 Interference threshold

An important parameter in our simulation is the interference threshold.
We run different simulations in order to set this parameter and we chose the interference threshold: 10^{-11}. In fact from figure 5.4 we can see that when we share too much (interference threshold set to 10^{-10}) the mean cell throughput as well as the outage user's throughput drastically fall down.

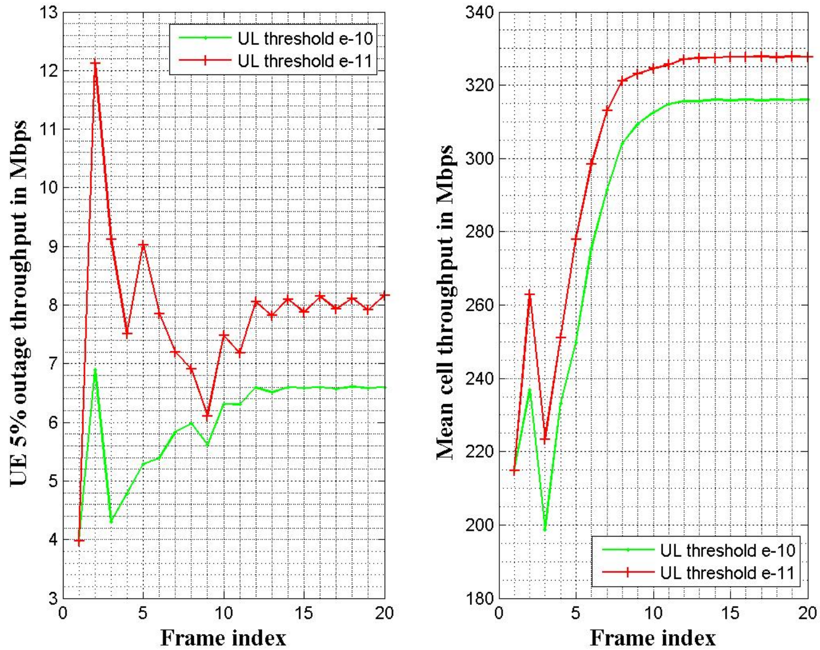

Figure 5.4: Threshold $10^{-11} vs 10^{-10}$

5.2 Results analysis

In this section we present the improvements we can have by using game theory, comparing our GTBA algorithm with several others.

- Fair Policies Based Algorithm (FPBA): FPBA is an algorithm that applies all our spectrum policies (such as to set a starting number of PRBs per user and increase by one or two PRBs per frame until you have enough available PRBs). In other words it is the GTBA without the so called revision process. See also section4.2.5.
- Game theory based Algorithm (GTBA): this algorithm behaves as FPBA until the 10^{th} frame. From this frame the players will make a decision about how many PRBs should allocate. Moreover, this decision is taken on the basis of each player's observed utility function. This is what we call the revision process as explained in section 4.2.6.
- TMax: Tmax algorithm is an algorithm that allows the operators to get as much spectrum as they can get, respecting only our policies on spectrum load and on maximum and minimum numbers of PRBs per user. See also section 4.2.4
- No Flexible Spectrum Usage Algorithm (NoFSU): no flexible spectrum usage algorithm with full load is a basic algorithm already implemented

in the simulator. In this algorithm all the spectrum is exploited by both the eNBs. See also section 4.2.8.

- Fixed Spectrum Allocation Algorithm (FAS): we code this algorithm to have another reference case. It consists in assigning to each operator a fixed part of the spectrum (the 50% in the two players case and the 25% in the four players case).

First of all we show the improvement achieved by the means of game theory, comparing the GTBA with the FPBA and TMax. Then we will compare the GTBA with the reference case (NoFSU algorithm) and with the fixed allocation case (FAA). Finally we will discuss some results for the four players case.

5.2.1 GTBA vs FPBA

Here we make a comparison between the Game Theory Based Algorithm and the Fair Policies Based Algorithm. The figure 5.5 shows the minimum throughput achieved by the 95% of the users for both the GTBA and FPBA.

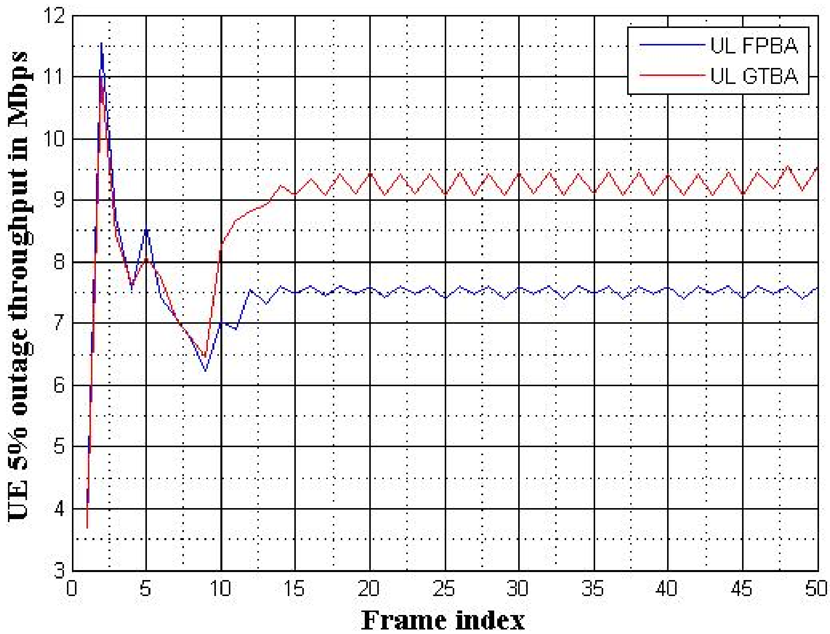

Figure 5.5: Outage user's throughput: GTBA vs FPBA

As we can see in figure 5.5 by using the utility function and the so called revision process we greatly increase the outage user's throughput. As discussed before, after the 10^{th} frame, players will make their decisions about how many PRBs to allocate. Most of the times, this decision leads them to decrease the number of allocated PRBs (as we can see in figure 5.7). Therefore after the 10^{th} frame the outage user's performance significantly improves with respect to the

FPBA.

We can also show that, by the means of the utility function, a player will make a fair and efficient decision.

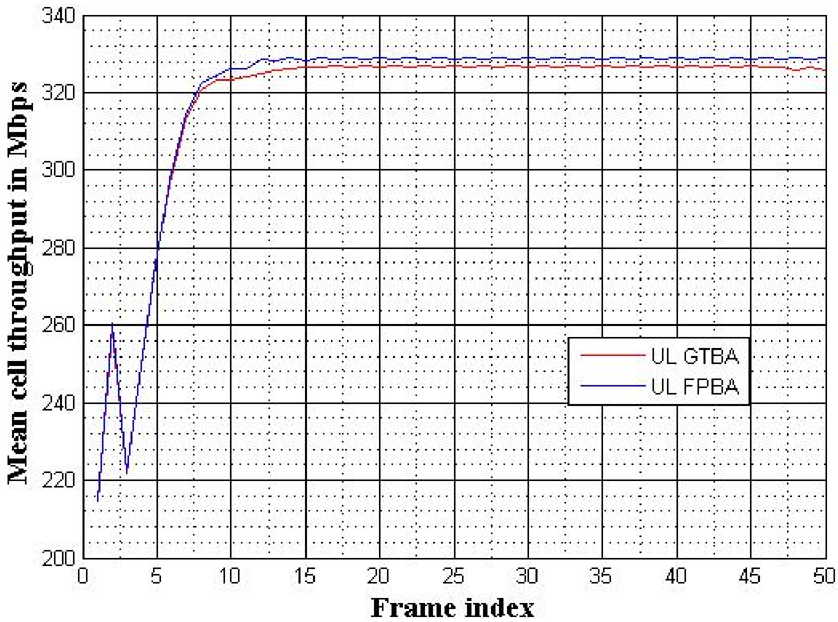

Figure 5.6: Mean cell throughput: GTBA vs FPBA

As we can see in figure 5.6 a eNB reaches almost the same throughput as in FPBA, but it is maximizing the spectral efficiency, using less PRBs to achieve almost the same result (figure 5.7).

From the plot 5.7 we can see also the fairness of our algorithms that lead to a situation where in average the eNBs use almost the same amount of PRBs. Figure 5.7 shows as well that after the 12^{th} frame an equilibrium is reached in the GTBA. In fact players do not change anymore their actions, in other words they keep frame by frame their chosen number of allocated PRBs. As we explained, in this equilibrium point the spectral efficiency is higher than in the equilibrium point reached by the FPBA.

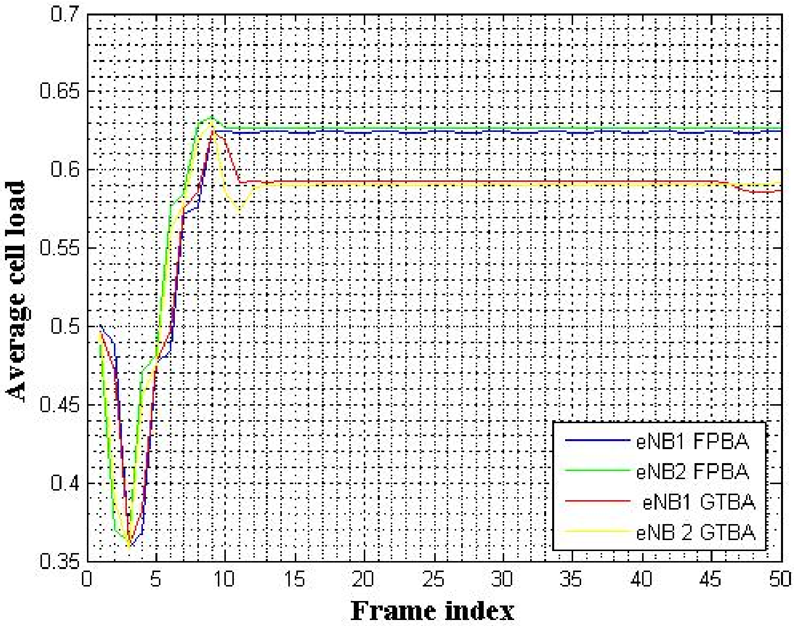

Figure 5.7: Average cell load: GTBA vs FPBA

5.2.2 GTBA vs Tmax

In this section we compare the performance of GTBA and the Tmax algorithm. As explained in section 4.2.4, the Tmax algorithm is an algorithm that allows the operators to get as much spectrum as they can get respecting only our spectrum policies (such as the maximum spectrum load). Once again the cell throughput achieved by the GTBA is similar to the one achieved by the Tmax algorithm (figures 5.10 and 5.9), but we are using less PRBs, as we can see from figure 5.8. So we gain in spectral efficiency.

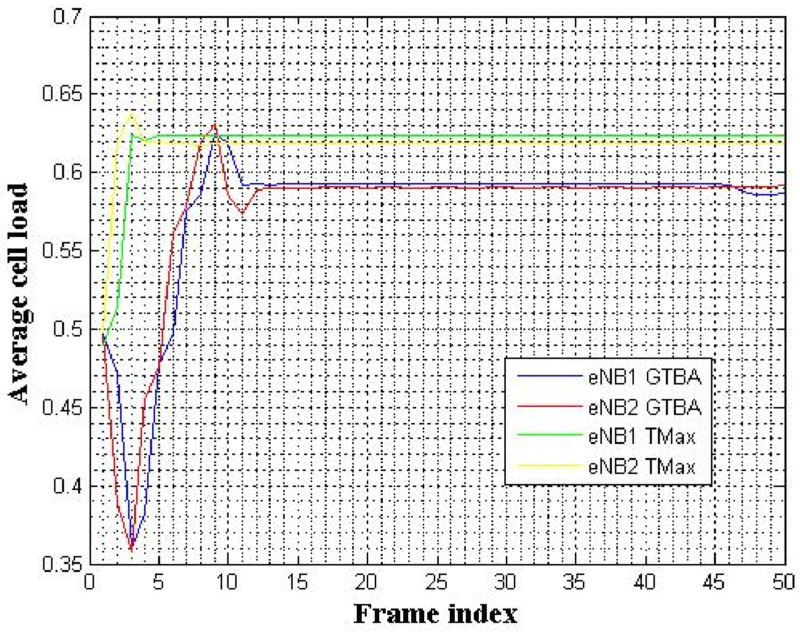

Figure 5.8: Average cell load: : GTBA vs Tmax

Figure 5.9: CDF of cell throughput: GTBA vs Tmax

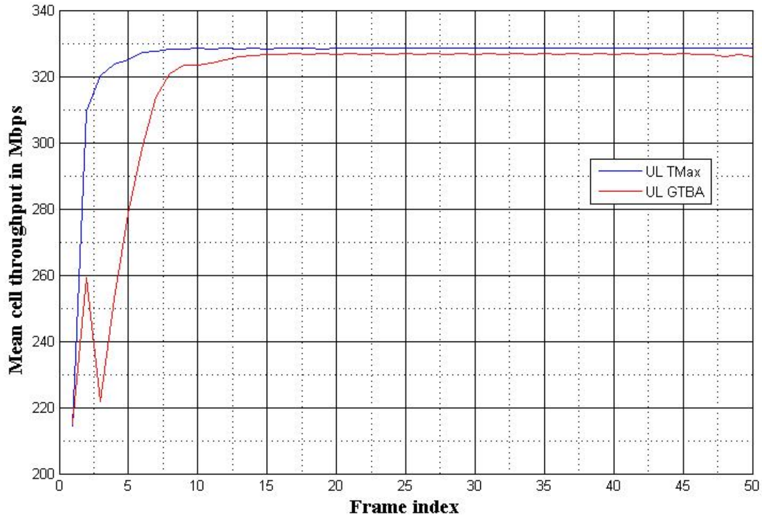

Figure 5.10: Mean cell throughput: GTBA vs Tmax

Moreover the CDF of users' throughput is similar too (figure 5.11), but the outage user's throughput is higher in our case (figure 5.12), because sharing less PRBs the operators interefere less each other.

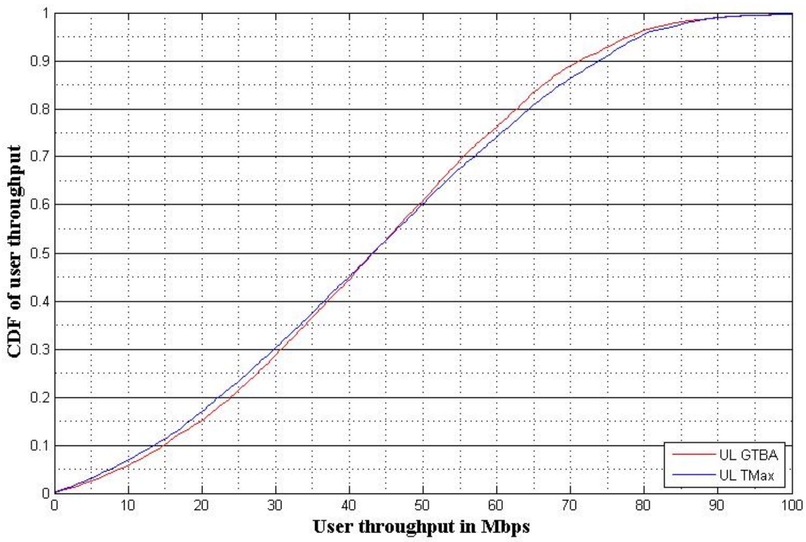

Figure 5.11: CDF of user's throughput: GTBA vs Tmax

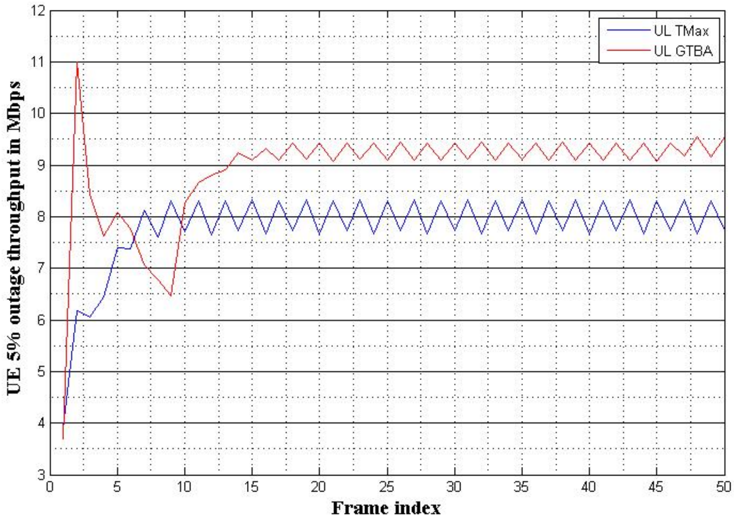

Figure 5.12: Outage user's throughput: GTBA vs Tmax

5.2.3 GTBA vs NoFSU

Here we compare GTBA to the reference case (NoFSU algorithm) in which all the spectrum is exploited by both the eNBs.

In figure 5.13 and 5.14 we have illustrated that both the mean cell throughput and the outage user's throughput are significantly improved by GTBA. This is a confirmation to our idea that it is good to share the spectrum but not over a certain point (as shown in 5.1.4). In fact after a certain point having more PRBs an eNB does not improve anymore its throughput because it gets highly interfered PRBs.

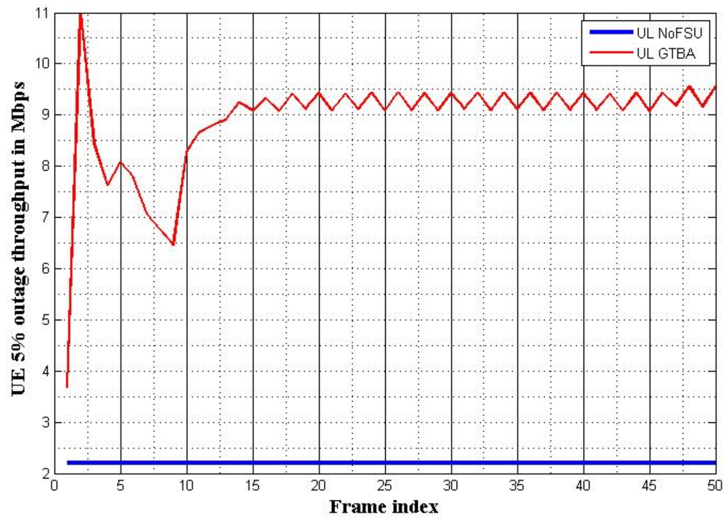

Figure 5.13: Outage user's throughput:GTBA vs NoFSU

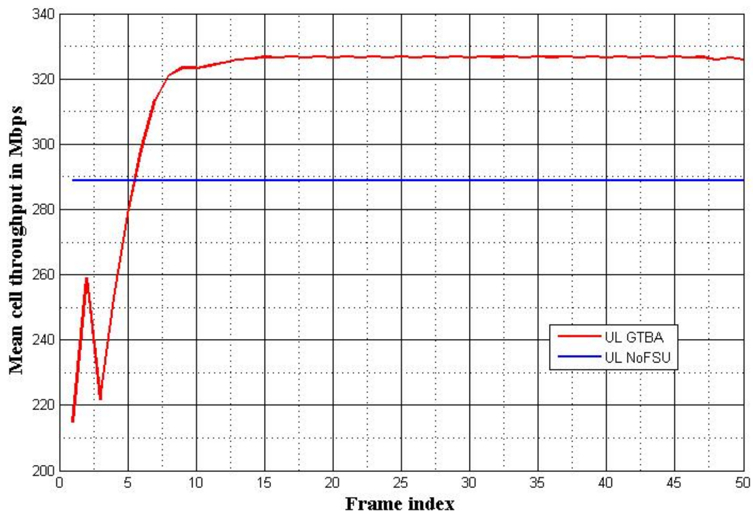

Figure 5.14: Mean cell throughput: GTBA vs NoFSU

From the figure 5.14 we can see how GTBA leads always to an higher mean cell throughput. Moreover using less PRBs we improve the spectral efficiency as well.

However as illustrated in figure 5.15 from the CDF of user's throughput we can state that without any FSU algorithm the 8% of users reach a throughput higher than 80 Mbps against the 4% obtained in our algorithm. This is because the full load algorithm gives to every users the total amount of PRBs divided by the actual number of users. In our case we set some policies (i.e. the maximum number of PRBs per user sets to 12) to achieve the fairness goal that prevents us to reach the same results.

In conclusion using the utility function we have a big improvement in term of mean cell throughput and outage user's throughput, but in order to have a fair situation we pay in term of maximum throughput achieved by users. However GTBA leads also to a more fair situation between users, in fact we reduce the throughput gap between the users in the best situation and the outage user, as we can see from figure 5.15.

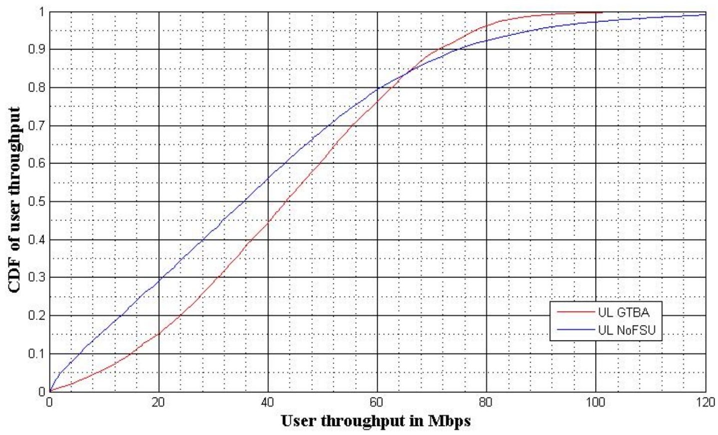

Figure 5.15: CDF of user throughput: GTBA vs NoFSU

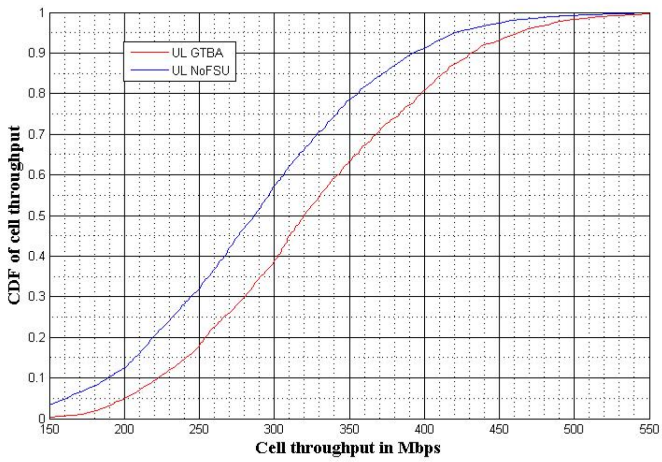

Figure 5.16: CDF of cell throughput: GTBA vs NoFSU

5.2.4 GTBA vs the Fixed Spectrum Allocation case

In this paragraph we compare the GTBA to the FSA case.

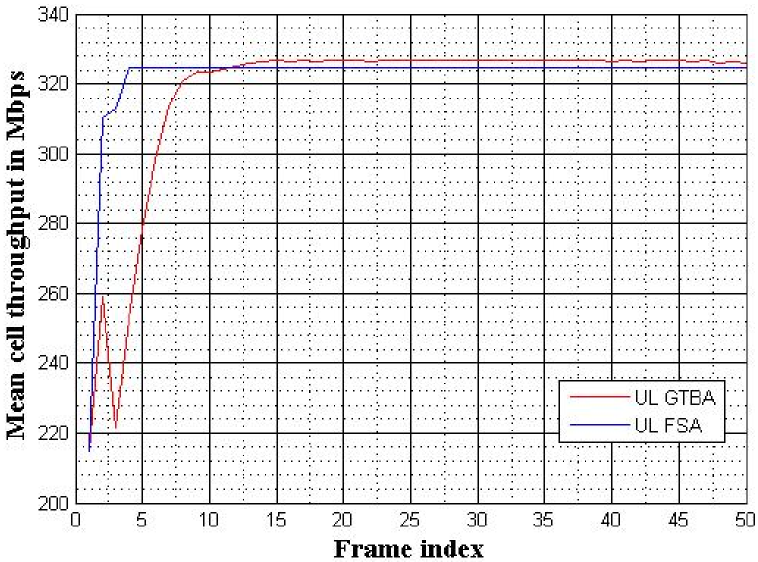

Figure 5.17: Mean cell throughput: GTBA vs FSA

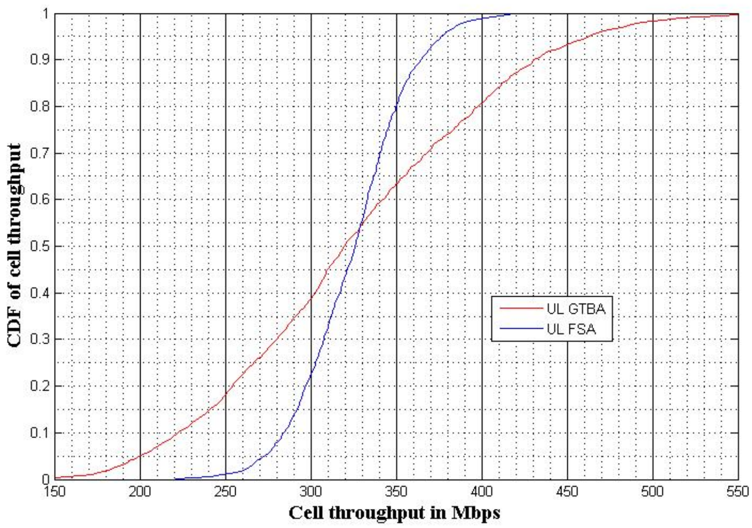

Figure 5.18: CDF of cell throughput: GTBA vs FSA

It is interesting to compare figure 5.17 and figure 5.18. Figure 5.17 shows that with the GTBA the mean cell throughput is still better than in the FSA case, moreover through figure 5.18 we can illustrate the flexibility of the GTBA. In fact in the FSA algorithm the cell throughput does not change too much. On the contrary in the GTBA we have a big variation because the cell throughput depends on the actual number of served users, for example in the 19% of the cases the cell throughput is above 400 Mbps.

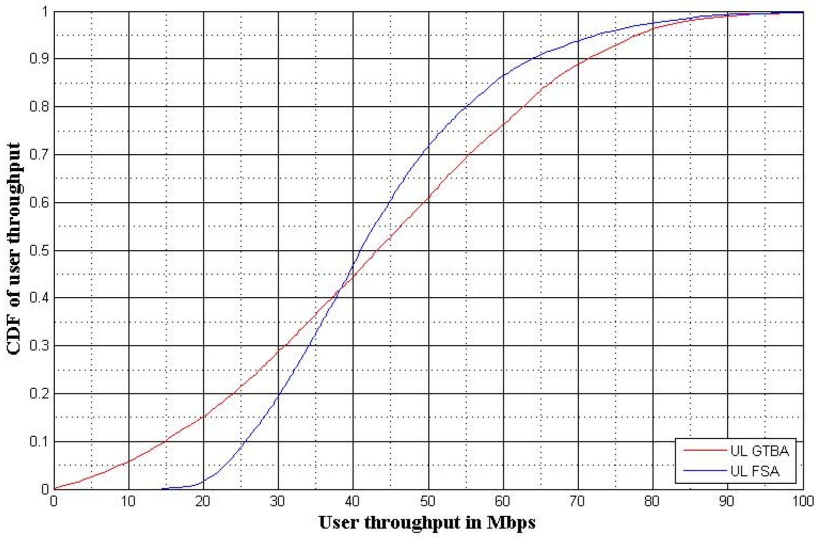

Figure 5.19: CDF of user's throughput: GTBA vs FSA

It can be seen in figure 5.19 that over a certain point also the user's throughput is better. In fact the percentage of users whose throughput is over 50 Mbps is 38% in the GTBA against the 28% reached by the FSA algorithm.

From figure 5.20 we can see that adding the flexibility we paid a lot in term of outage user's throughput. In fact the throughput achieved by the outage user with the GTBA is less than in the fixed spectrum allocation case. This is because in the fixed spectrum allocation users do not experience interference at all, due to the orthogonal allocation.

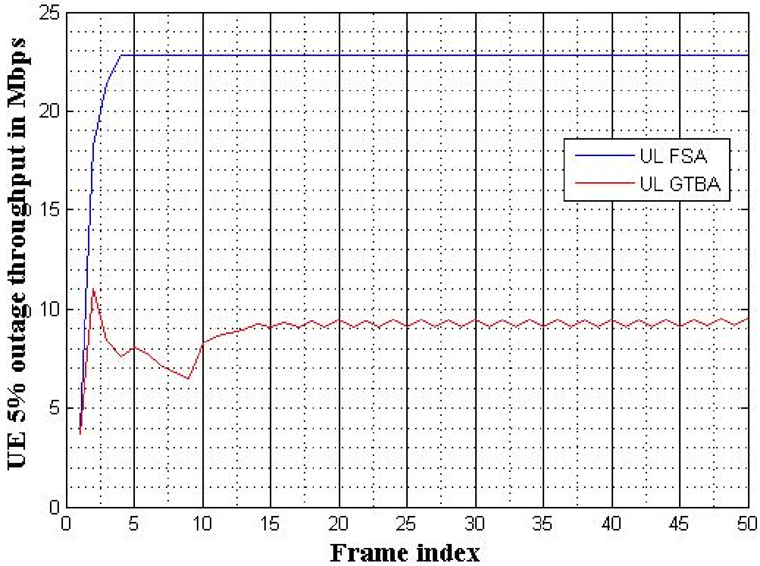

Figure 5.20: Outage user's throughput:GTBA vs FSA

5.2.5 The four players case

In this paragraph we are going to show the results obtained in the four players case.

Increasing the number of operators we obtained a bigger improvement than the one obtained in the two players case. As we can see in figure 5.21 the mean cell throughput is much higher for both our FPBA and TMax algorithms compared to the results obtained with the fixed spectrum allocation (FSA algorithm) and without any flexible spectrum usage algorithms (NoFSU algorithm).

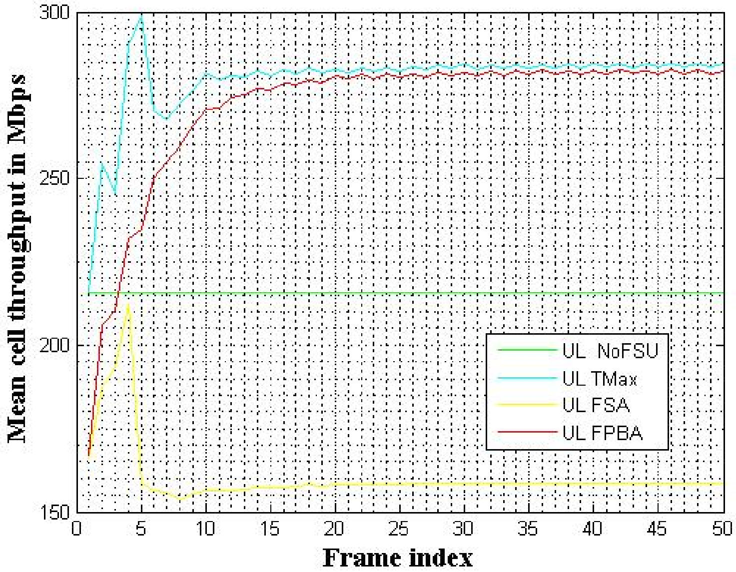

Figure 5.21: Mean cell throughput: the four players case

Figure 5.22 illustrates that in the four players case we can reach a better throughput for the outage user, too. In fact as we showed we reach and overcome the throughput achieved in the fixed allocation case.

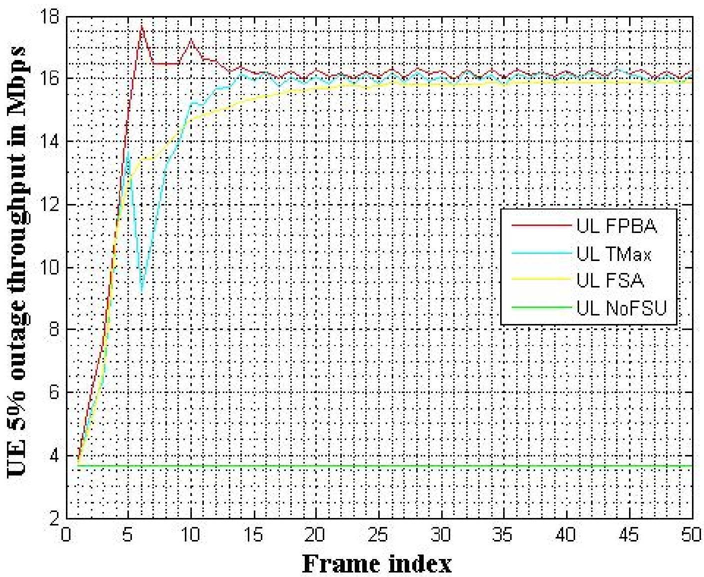

Figure 5.22: Outage user's throughput: the four players case

Figure 5.23 and figure 5.24 show that FPBA and Tmax algorithm lead to an higher throughput both for the users and the total cell throughput compared with the ones achieved by a fixed spectral allocation and without the use of any flexible spectrum algorithm. In fact we can see from figure 5.23 that in the 30% of the cases the cell throughput is higher than 310 Mbps throughout FPBA and Tmax algorithm, against the 240 Mbps reached in the 30% of the cases without any FSU algorithm and the 170 Mbps reached by the FSA algorithm.

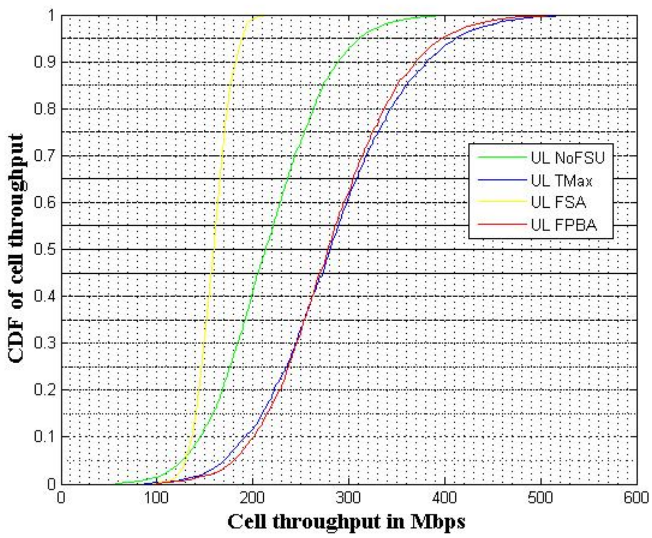

Figure 5.23: CDF of cell throughput: the four players case

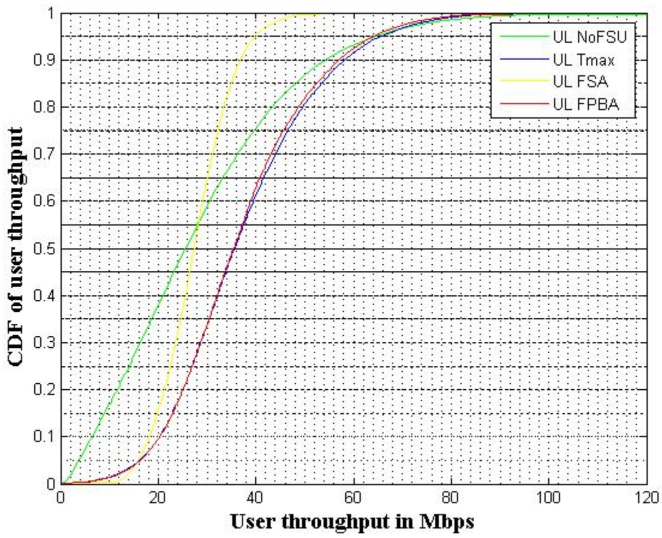

Figure 5.24: CDF of user's throughput: the four players case

Chapter 6

Conclusion and future works

In this chapter, the general conclusion that can be drawn from the project is presented. Moreover, several ideas for future works are proposed.

6.1 What we have done

In this work we have investigated the design of spectrum sharing algorithms for IMT-A networks in a cooperative scenario with two or four operators. Three differents algorithms for the spectrum allocation were proposed: take as much as it is possible respecting our policy (TMax), take as much as you can increasing frame by frame in order to add the fairness (FPBA) and no-regret learning based algorithm to maximize the spectral efficiency (GTBA).

We showed that all the proposed spectrum sharing algorithms converge to a stable equilibrium although in a decentralized system operators do not have any information about the others. Moreover they require no information exchange to work.

Our simulation results have shown that the average achievable throughput is higher for the GTBA, in addition it uses less PRBs than the others reaching an higher spectral efficiency.
TMax is also efficient, but it is not fair at all: in fact the first player can take all the spectrum preventing the others to do the same because of the interference limit.
Through a slow increment we reach a fair equilibrim in FPBA.
We showed that our regret learning approach (GTBA) adds the efficiency to the FPBA. Thus by the means of our game theorectic approach it is possible to share the spectrum in a fair and efficient way.

Through game theory we also improve the outage user's throughput in comparison with the result obtained without any FSU and for both the algorithms TMax algorithm and FPBA.

6.2 Future works

In a centralized system with all the information available it could be a complex optimization problem to share the spectrum in the best efficient and fair way.

We have assumed UL/DL fully synchronized in our approach to model a dynamic game. We could extend the problem to unsynchronized case and study the effect of a static approach.

Users' equipments have been assumed fixed at each simulation loop. Moreover the number of users is changing for each operators in the same time. We could study the case where the number of users is changing independently from an operator to another one. Then we should have a problem with the GTBA. Indeed, in this case both operators reach an equilibrium in term of allocated PRBs and then the number of users of one operator is changing. This one is going to start again the algorithm and the other one to allocate the same spectrum. So it would not know when it has to start again the algorithm and leave the equilibrium.

Finally we have assumed that players are cooperative and thus we did not simulate the case where an operator is selfish. We could include this behavior and play with the reputation and then how to choose credible punishment methods to make a selfish operator becoming cooperative.

List of Figures

List of Tables

Bibliography

[1] Dr. Hyeon Woo Lee. *3G LTE and IMT 3G IMT-Advanced.* Global Standards & Research SAMSUNG ELECTRONICS, February 2006.

[2] Shashi Kant Sanjay Kumar, Gustavo Costa, Flemming Bjerge Frederiksen, Nicola Marchetti, and Preben Mogensen. *Spectrum Sharing for Next Generation Wireless Communication Networks.* Radio Access Technology Section, Aalborg University, Denmark, 2008.

[3] Roger B. Myerson. *Game Theory - Analysis of Conflict.* Harvard University Press, 1991.

[4] Lars Berlemann and Stefan Mangold Bernhard H. Walke. *Policy-based reasoning for spectrum sharing in cognitive radio networks.* IEEE, 2005.

[5] Raul etkin, Abhay Parekh, and David Tse. *Spectrum Sharing for Unlicenced Bands.* University of California, Berkeley, August 2005.

[6] Kimmo Kalliola. *Spectrum Sharing and Flexible Spectrum Use.* Nokia Research Center, August 2004.

[7] Zhu Ji and K. J. Ray Liu. *Dynamic Spectrum Sharing: A Game Theoretical Overview.* IEEE Communications Magazine University of Maryland, May 2007.

[8] Dean P. and H. Peyton Young. *Games and Economic Behavior.* Foster learning, 2003.

[9] M. Felegyhazi. *Thesis: Non-cooperative behavior in wireless networks.* EPFL Switzerland, 2006.

[10] Guido R. Hiertz Lars Berlemann and Bernhard H. Walke. *Radio Resource Sharing Games: Enabling QoS Support in Unlicensed Bands.* IEEE Network, July/August 2005.

[11] M. Felegyhazi and JP. Hubaux. *Game Theory in Wireless Networks: A Tutorial.* EPFL Switzerland, 2006.

[12] J. nash. *Equilibrium points in n-person games.* Proceedings of the National Academy of Sciences, vol. 36, pp. 48Ű49, 1950.

[13] Drew Fudenberg and Jean Tirole. *Game Theory.* The MIT Press, 1991.

[14] Juan E. Surís. *Cooperative Game Theory and Non-Convex Optimization for Distributed Spectrum Sharing.* Virginia Polytechnic Institute and State University, December 2007.

[15] Mark Felegyhazi. *Non-cooperative behavior in wireless networks.* EPFL, Lausanne, December 2007.

[16] Dean P. Foster and H. Peyton Young. *Regret testing: learning to play Nash equilibrium without knowing you have an opponent.* Theoretical Economics 1, December 2006.

[17] Dean P. and H. Peyton Young. *On the impossibility of predicting the behavior of rational agents.* Foster Proceedings of the National Academy of Sciences of the USA, 2001.

[18] Pekka Kyösti and Juha Meinilä. *WINNER II Channel Models.* IST-4-027756 WINNER II, september 2007.

[19] Franck H.P.Fitzek and Marcos D.Katz. *Cognitive Wireless Networks.* Springer, 2007.

[20] Lars Berlemann and Bernhard Walke. *Spectrum load smoothing for optimized spectrum utilization - rationale and algorithm.* IEEE Communications Society, 2005.

[21] Nicola Marchetti, Muhammad Imadur Rahman, and Flemming Bjerge Frederiksen. *Introducing smartness in future wireless systems spectrum utilization: a cognitive radio perspective.* Center for TeleInFrastruktur, Aalborg University, Denmark, 2007.

[22] Pierre de Vries and Amer Hassan. *Spectrum sharing rules for new unlicensed bands.* Microsoft corporation, December 2003.

[23] James O'Daniell Neel. *Analysis and design of cognitive radio networks and distributed radio resource management algorithms.* Virginia Polytechnic Institute and State University, 2006.

[24] Randall Berry. *Spectrum Sharing Games.* Northwestern University Deptartment of ECE, June 2005.

[25] Juan E. Suris, Luiz A. DaSilva, Zhu Han, and Allen B.MacKenzie. *Coopeative game theory for distributed spectrum sharing.* IEEE, 2007.

[26] Tian Feng and Yang Zhen. *A spectrum sharing algorithm based asymmetric-Nash-bargain-solution in cognitive radios.* Institute of signal processing and transmission, Nanjing university, China, 2007.

[27] Nie Nie and Cristina Comaniciu. *Adaptive channel allocation spectrum etiquette for cognitive radio networks.* Stevens Institute of Technology, 2005.

[28] O. Simeone and Y. Bar-Ness. *A game theoretic view on the interference channel with random access.* University Heights USA, 2007.

[29] Allen B.MacKenzie and Stephen B.Wicker. *Game theory in communications: motivation, explanation and application to power control.* Cornell University, 2007.

Appendix A

Glossary

A.1 Abbreviations

DL: Down link
FDD: Frequency Division Duplexing
FSU: Flexible Spectrum Usage
HeNB: Home enhanced Node B
IMT-A: International Mobile Telecommunications
ITU: International Telecommunication Union
NE: Nash Equilibrium
OFDMA: Orthogonal Frequency Division Multiple Access
PRB: Physical Resource Block
QoS: Quality of Service
RF: Radio Frequency
RRM: Radio Resource Management
SC-FDMA: Single Carrier FDMA
SINR: Signal to Interference plus Noise Ratio
TDD: Time Division Duplexing
UL: Up Link
WLAN: Wireless local Area Network

A.2 Definitions

Average Achieved Cell Load: represents the actual number of PRBs utilized by the eHNBs.

Fractional load: each operator is requiring a fraction of the full bandwidth.

Full load: each operator is requiring the full bandwidth.

Mean Cell Throughput: it is the total throughput achieved by the eHNB during one frame after the FSU algorithm is stabilized.

PRBs: it is the smallest units of spectrum sharing. In our game PRBs are the ressource to allocate to each operators.

User Outage Throughput: defined at the 5th percentile of the CDF of user throughput. This gives the minimum throughput achieved by the 95% of the users.

Appendix B

Why are we using dynamic game instead of static game ?

As we have seen in chapter 3 we are using a dynamic approach to solve the problem of spectrum sharing. In this section we are going to explain the reason of this choice.

B.1 Static allocation example: TMax

The diagrams B.1, B.2 and B.3 represent an allocation in a static game. As we can see, all the spectrum is not always used (blue color). Remenber what we have talked about in chapter 3.2.2. In static game players choose their actions silmutaneously. Their strategies are based on their requirements and the number of PRBs available which depends on the interference level and the threshold. It means each operator has to choose which PRBs to allocate from an interference vector calculated before the other play turn. Then, both operator are going to sense a spectrum opportunity (free PRBs with no interference for example). As they sort the interference vector from the best one to the worse one, this PRBs will be one of the best for both operator. Next stage they are going to allocate some users. After the stage , each operator is going to sense interference in the PRBs allocated because they are both using it. So, in the next allocation process they will leave this PRBs and the PRBs will have again a good interference level.

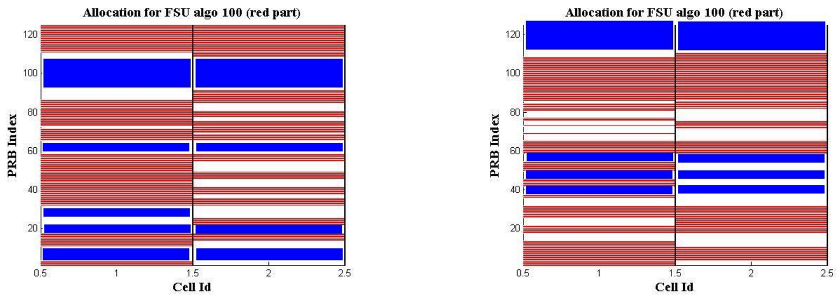

Figure B.1: Allocation static: frame number 1 & 2

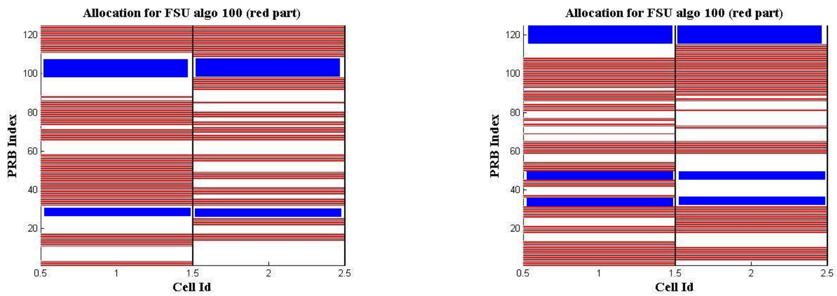

Figure B.2: Allocation static: frame number 3 & 4

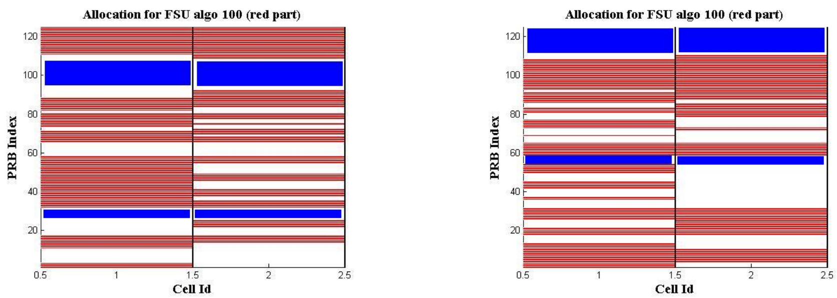

Figure B.3: Allocation static: frame number 5 & 6

As we can see with a static game the allocation of the spectrum is not stable and all the spectrum is not used. Now we are going to compare with a dynamic game.

B.2 Dynamic allocation example: TMax

The diagrams B.4, B.5 and B.5 represent an allocation in a dynamic game. As we have seen in 3.2.2, in a dynamic game players don't make their decision simultaneously. Therefore in our case one player starts with his own allocation and the other has to make his decision on top of the previous one's move. Thus the previous one's action influenced the actual player decision by the means of the interference, because, as explained later in 4.2.2, a player cannot choose every PRBs, but only the available ones. Furthermore to not play does not mean wasting the opportunity to allocate some PRBs in the actual frame: it means only that for the actual stage the "non" player keeps his previous allocation.

The following allocation example use these parameters:

Operators	Number of users
A	8
B	6

Table B.1: Operator parameter

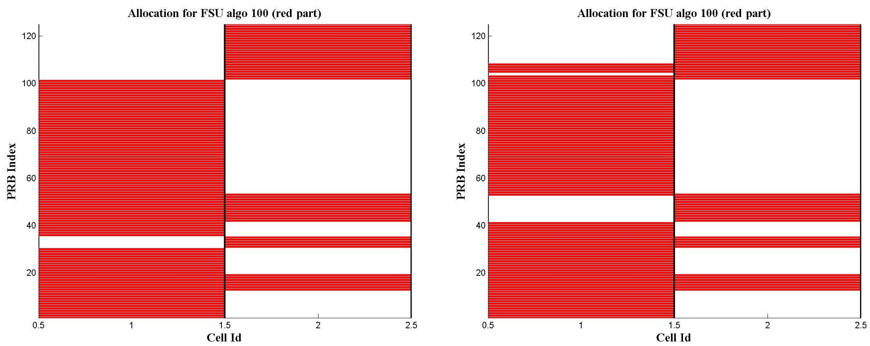

Figure B.4: Allocation dynamic: frame number 1 & 2

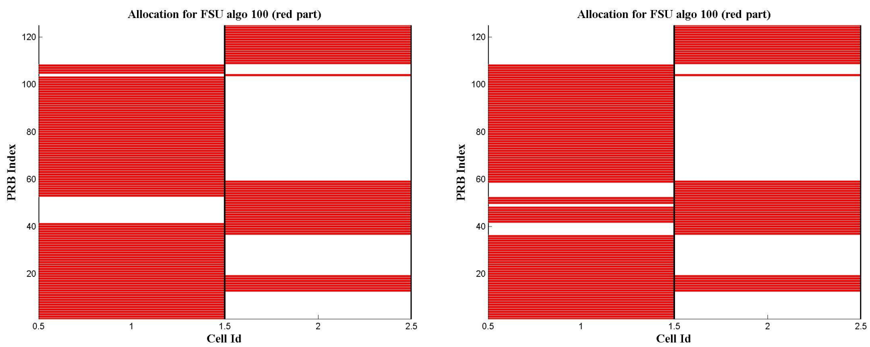

Figure B.5: Allocation dynamic: frame number 3 & 4

And then we reach an equilibrium in frame 5 and 6. It means operators will always use the same number of PRBs and the same allocation until the number of users change.

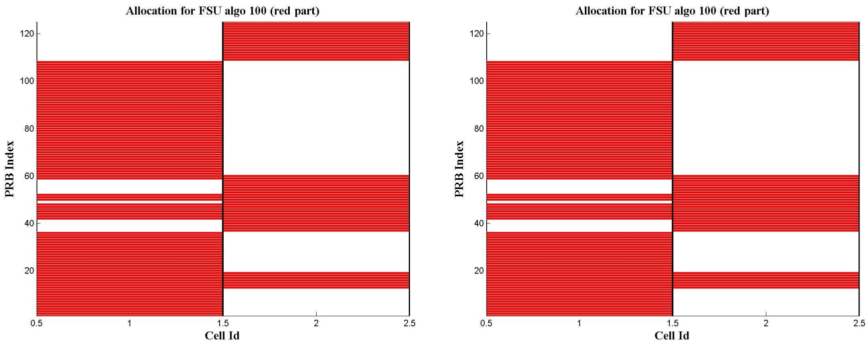

Figure B.6: Allocation dynamic: frame number 5 & 6

Appendix C

OFDM and OFDMA

Orthogonal Frequency Division Multiple Access (OFDMA) is a multiple access technique that uses Orthogonal Frequency Division Multiplexing (OFDM) in order to provide multiple access to the radio resource.
OFDMA techniques is built on OFDM, thus we first take a look on how OFDM works in order to explain this radio access technique. OFDM is a modulation technique that allows to modulate a data stream instead of over a single wide band carrier, over multiple narrow band sub-carriers (fig.).

Figure C.1: OFDM sub-carriers

As showed in figure C.1 there is no band guard between sub-carriers. This is possible because they are orthogonal, in other words the peak of one sub-carrier correspond to the null of the adjacent ones.
Thanks to the orthogonal sub-carriers OFDM reach an high spectral efficiency. Moreover it is possible to use different modulation and coding techniques in each sub-carrier. This allows to use more robust modulations only on a subset of interfered as well as faded sub-carriers, once again increasing the spectral efficiency.

OFDM allows only one user on the channel. In other words at a certain time all the sub-carriers are given to one users. To hold multiple users, a strictly OFDM based system must use a multiple access technique such as the Time Division Multiple Access (TDMA).

OFDMA realize the multiple access by exploiting the orthogonallity among sub-carriers.
The multiple access, in fact, is provided by assigning at a certain time a sub-channel to every user. A sub-channel is a group of contiguous as well as non contiguous sub-carriers.
This kind of multiple access is very flexible because, as the OFDM, it offers the possibility to change the modulation and coding technique for each sub carrier as well as for each sub channel, thus for each user.

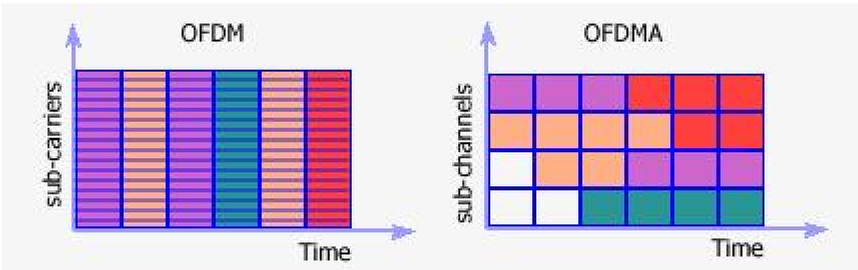

Figure C.2: OFDM/TDMA and OFDMA