UNIVERSITY OF PADUA

SCHOOL OF ENGINEERING

MASTER DEGREE IN TELECOMMUNICATION ENGINEERING

MASTER THESIS

Resource Allocation for Dynamic TDD 5G Systems: A Game Theoretic Approach

Supervisor: STEFANO TOMASIN

Author: CHANDAN KUMAR SHEEMAR

ID No 1148073

ACADEMIC YEAR 2017-2018

9th of July

Abstract

The development of mobile telecommunication systems is a constant process as the number of users and the data traffic required to sustain them are growing exponentially. The evolution will continue with the 5th generation (5G) radio technology, which will bring tremendous improvements in data rates, reliability, energy efficiency and security. With all these drastic improvements, one of the biggest challenges that arises is the interference generated by the neighboring base stations (BSs), as they will be deployed independently. Recent studies have shown that Time Division Duplexing (TDD) results a very good approach to mitigate this type of interference if the switching points between Uplink and Downlink are optimized in a proper way.

In the first part of this thesis, we deal with the problem of sum rate maximization of a cellular system in which all the BSs serve their user equipments (UEs) by using the TDD mechanism. Moreover, all BSs are allowed to choose their transmission directions between Uplink or Downlink independently. In order to find its optimal solution, the Multi-start algorithm has been used. In the second part of this thesis we exploited the theoretical tools offered by game theory and formulated our problem of sum rate maximization as a dynamic TDD assignment game. All the BSs are treated as players and various games are formulated based on the quantity of knowledge possessed by each BS.

Contents

Chapter 1

Introduction

In the last few decades, Mobile Wireless Communication Networks have experienced a tremendous change. From the first communication devices to the latest cellular systems, their complexity, number of users and the amount of data traffic sent and received have experienced an exponential growth. Since early 1970, its creation, revolution and evolution started and approximately every ten years new generations of mobile wireless communication systems have appeared to satisfy new network requirements and to sustain the data traffic demands.

1.1 Evolution

The first generation (1G) of mobile wireless technology based on analog signals was launched in the 1980s. The key idea was deployment of multiple cells and ability to adapt when users travelled between cells during a conversation. The voice call modulation for each mobile device was performed to a higher frequency of about 150 MHz and then transmitted to a nearest Base Station (BS). Modulation was based on the so-called Frequency Division Multiple Access (FDMA) technique. Various standards were established for 1G wireless communication systems, among which the most important were: Advanced Mobile phone service (AMPS), employed in the United States for the first time ever and then adapted by many other countries, Total Access Communication System (TACS), used by the United Kingdom, Malta, Singapore and by few Arabic countries and Nordic Mobile Telephone (NMT)-450/900, employed firstly by Nordic countries and then by few European countries [1]. In Italy, the Radio Telephone Mobile System (RTMS) was employed, which operated in the band of 450 MHz with more than 200 radio channels. Around the 1990s, technology improved and the second generation (2G) of mobile wireless technology was introduced. It differed from 1G systems in the use of digital radio signals instead of analog signals and in the use of new digital multiple access technologies like the Time Division Multiple Access (TDMA) and the Code Division Multiple Access (CDMA). Firstly, 2G was based on the Global System for Mobile Communication (GSM) standard [2], in which 125 allocations and eight time slots per channel were available, providing a total of 1000 channels. GSM-based networks were circuit switched and their data transfer rates were very low, around 30-35 Kbps. Later on, an alternative to GSM, a new standard called the Interim Standard-95 (IS-95) was introduced. It was the first ever CDMA-based digital cellular technology developed by Qualcomm, in which audio band data signals were multiplied by a high rate spreading signal. IS-95-based systems can be thought of as having layers of protection against interference which let many users co-exist with minimal mutual interference. In these days, number of users started to grew very fast as the use of 2G mobile systems became viral and the telecommunication industry responded by improving further the existing 2G systems by introducing 2.5 G [3]. It was based on the General Packet Radio Service (GPRS) technology, ratified in 1997 and able to support a very large scale of users with higher data rates (from 56 kbps up to 115 kbps) compared to the previous 2G based systems. For the first time ever, it offered services like Short Message Service (SMS), Multimedia Messaging Service (MMS) and new type of Internet communication services such as the Electronic mail (e-mail) and the World Wide Web (WWW). After few years, a technological evolution of GSM standard called the Enhanced Data Rates for GSM Evolution (EDGE) (2.75G) was introduced, which pushed further the data transfer rates, improved the spectral efficiency, allowed the existence of new applications/services and increased the system capacity. As the use of mobile systems became more widespread, demand of data traffic and services grew very fastly and in 2001 the third generation (3G) of mobile wireless communication systems was introduced to sustain these demands. This generation brought with itself very high data transfer rates compared to its predecessors, video conferencing, wide area wireless voice telephony and entertaining services like 3D gaming and mobile TV. The 3G based mobile systems enabled network operators to offer users a wider range of more advanced and sophisticated services while achieving great system capacity through improved spectral efficiency [3]. These systems resulted much flexible than its predecessors, able to support also the 5 main radio technologies which operate based on TDMA, CDMA and FDMA techniques. After few years, the 3G evolved in the High Speed Downlink Packet Access (HSDPA) as 3.5G and later in the High Speed Uplink Packet Access(HSUPA) as 3.75G. In the former one, Downlink performance was enhanced by exploiting techniques like Adaptive Modulation and Coding (AMC), Multiple-Input-Multiple-output (MIMO), Hybrid Automatic Request (HARQ), fast cell search and an advanced receiver design. The achieved peak data transfer rates were around 10 Mbps. In the latter one, Uplink performance has undergone an increase and the maximum data transfer rates achieved were around 5.8 Mbps. HSUPA is directly related to HSDPA as they both are complimentary to one another, which means HSUPA exploits same techniques mentioned above. Around 2010, the fourth generation (4G) was introduced, which indicates a change in the nature of services, non-backward compatible transmission technology and new frequency bands, which later evolved into the 4G-Long Term Evolution (4G-LTE). It can be seen as an extension of the 3G with much more bandwidth and with a large variety of services to offer. For pure extrapolation of history, the fifth generation (5G), which is the next major phase of mobile telecommunications standards, is expected to be launch onto the market around 2020.

1.2 Motivation

According to Cisco, the Internet is going to be flooded by up to 50 billions devices by 2020 and all these devices will need to access and share data, anywhere and anytime. With this expected rapid increase in the number of connected devices to be served, the Internet will face new challenges which will be responded by increasing the system capacity, a better spectrum utilization, employing low power consumption and low cost technologies. The total mobile broadband traffic will increase up to a factor of 1000. These figures assume a 100 times higher data traffic per user and 10 times increase in the mobile broadband subscribers, thus 5G will bring an improvement of 10x in data transfer rates compared to 4G-LTE Advanced [4]. An ideal 5G network will have to satisfy these requirements:

- Peak data rate of 10 Gbps.
- Latency of (0.1-1) ms.
- Spectral efficiency almost 2 times better than the 4G based systems.
- Very low power consumption (up to ten years battery life time for low power devices).
- Efficient support for machine type communication (MTC).
- Simple and low cost design.
- Self-optimization of ultra-densely deployed access points.

- Flexible spectrum usage.

It is expected to become a cornerstone of many economic sectors and will benefit tremendously the areas of home automation, smart transportation, healthcare, virtual and augmented reality and wearable devices. Application of 5G based systems in these areas will generate a huge amount of data traffic that will be analyzed in real time to infer useful information. Apart form this, it has been also envisaged that everyone and everything will be communicating to build a connected society under the shadow of Internet of Things (IoT), where tens to hundred of devices will serve every person. It will also support Big Data which will enable cities to be smart. Mobile communication will also play a vital role in Machine-to-machine (M2M) type communication, e.g., it will enable efficient and safe transportation, alert or help vehicle drivers if any road is congested and will also improve autonomous self-driving. The Device to Device (D2D) communication will also be integrated with much smaller latency, increased data transfer rates and very low energy consumption, by letting two devices communicate correctly also in the case of infrastructure damage. A widespread increase in the use of these applications will create a demand of data transfer rates of the order of many Gbps. Therefore, 5G will have to face new type of challenges and much more complex and efficient systems will be required. To get an idea, an example of a general 5G cellular network architecture is shown in Fig. 1.1.

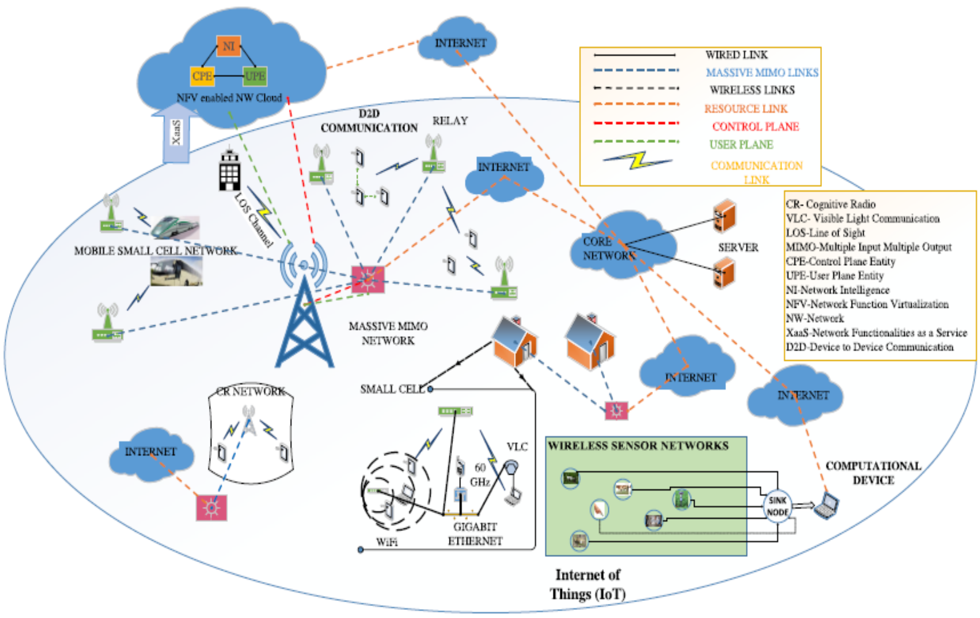

Figure 1.1: An example of the Fifth Generation (5G) cellular network architecture.

Increasing the total number of smart phones, devices with powerful multimedia capabilities, M2M and IoT are the main factors that will contribute in the expected data traffic. As a first solution, the multiple-input-multiple-output (MIMO) technology was proposed to improve the spectrum efficiency of 5G mobile communication systems. A MIMO system consists of m transmitting and n receiving antennas. During an on going communication between these multiple antennas, every receiving antenna receives not only the direct components from the intended antenna but also the indirect components from other antennas. By putting all the received signals into a vector $\mathbf{y}$ and all the transmitted signals into a vector $\mathbf{x}$, an equation that characterizes the transmission of a MIMO system can be written as:

$$\mathbf{Y} = \mathbf{H}\,\mathbf{x} + \mathbf{n}$$

where **H** is the transmission matrix that comprehend the direct and indirect paths and **n** is the noise vector. The policy used to transmit the intended data is that of dividing it into multiple independent data streams, where the number of data streams M has to be always less or equal to the number of antennas. In this way the capacity of a MIMO system is increased linearly with the number of independent streams :

$$C \ = \ M \ B \ log_2(1 + \frac{S}{N})$$

where S/N is the signal-to-noise ration and B is the bandwidth. Later, the millimeter-wave (mm-wave) communication technology was proposed to increase the bandwidth of 5G based systems. The mm-wave carrier frequency allows larger bandwidth allocations, which directly increase the systems data transfer rates. The mm-wave spectrum will let service providers to expand channel bandwidths far beyond the currently 20 MHz channels used by 4G systems. This major breakthrough in the communication systems' bandwidth and new capabilities offered by mm-wave will let backhaul links and BS-to-Device links to handle much greater capacity. Recently, the ultra-densely deployed small cells solution has been proposed, which tends to increase systems' throughput and decrease power consumption in a cellular scenario. The ultra-dense small cell deployments are considered to be a key technology for 5G systems which will provide short range coverage areas. When densely deploying multiple small cells, traffic demands per cell might variate significantly from one cell to another. Beside that, telecommunication society is thinking about making these small cells or group of small cells (called clusters) independent. In this way a malfunction of a single cell will not affect other cells and independent decisions on the transmission direction, i.e., between Uplink or Downlink, can be made to satisfy data traffic demands. Making all these cells independent and adjusting dynamically their transmission directions can generate interference due to an opposite transmission directions of neighbouring cells. Since the amount of total Downlink and Uplink traffic per cell may also vary over space and time drastically, it might not be easy to know what kind of interference will be generated by neighbouring cells and to cope with these issues, new interference avoiding strategies are also needed.

1.3 Time Divison Duplex

Recent studies [9]- [13] have shown that Time Division Duplexing (TDD) results a very good approach to mitigate interference generated by an opposite transmission direction of the neighbouring BSs. It can be shown, mitigation is possible only if switching points between Uplink and Downlink are selected cleverly. Most of the wide-band Internet Protocols use TDD in which a common frequency carrier is shared between Uplink and Downlink and allocation of different time slots is used to satisfy data traffic demands. User equipments (UEs) can be allocated one or more time slots depending on their need. Well known examples of already existing TDD systems are:

- Half-duplex packet mode networks based on carrier sense multiple access.
- TD-CDMA for indoor mobile telecommunications.
- IEEE 802.16 WiMAX.
- PACTOR (Radio Modulation Mode).
- Universal Mobile Telecommunications System 3G supplementary air interfaces.
- Digital Enhanced Cordless Telecommunication (DECT).

- TDD W-CDMA (Wideband Code Division Multiple Access) systems.
- 4G TDD LTE

TDD-based systems presents numerous advantages compared to the Frequency Division Duplex (FDD) systems. First of all, a complex duplexer can be substituted with a simple low cost solid state switch at the both sides. The TDD systems use same frequency for Uplink and Downlink and instantaneous propagation, fading characteristics other propagation parameters strictly depend on the frequency used for transmission. Therefore, all these parameters undergo a very slow variation and as a consequence the complexity of equipments can be reduced significantly. The characteristics of Uplink and Downlink are very similar, hence a channel equalization can also be allocated on the transmission side. The use of adaptive channel equalization can be combined with transmitter predistortion in order to improve resistance to multipath propagation. The TDD systems allow the capacity of Downlink and Uplink transmission to be more flexible, on the contrary of FDD systems, where the capacity of Downlink and Uplink is determined by the portion of spectrum allocated to the respective sub-bands. Moreover, TDD systems are also able to support voice, symmetrical and asymmetrical data traffics. The main advantage of TDD is that it allows asymmetric traffic flow which is more suited for data transmission, the reason why it's an optimal candidate for 5G systems. Despite TDD systems have numerous advantages over FDD, its efficiency still suffers as the cell size increases. A reduction in efficiency comes from an unpredictable nature of delay times of transmission from the BS over air channels to the UEs, and from the UEs over air channels back to the BS. This unpredictable nature is caused by the fact that UEs can be fixed or move anywhere within the radius of cell. A BS might not be able to know in advance how long a propagation delay could be while communicating with a particular UE. By considering the worst case, TDD makes use of round-trip guard time to ensure that communication will be completed with the first UE before starting it with another. This round-trip guard time is present in each slot independent of UEs' positions and adds extra overhead. These overheads can decrease the efficiency of TDD systems and limit the maximum numbers of UEs. A comparison between FDD and TDD is shown in Fig. 1.2, in which it can be clearly seen how FDD uses different channels for Downlink and Uplink and TDD adjusts frames' portions to satisfy data traffic demands.

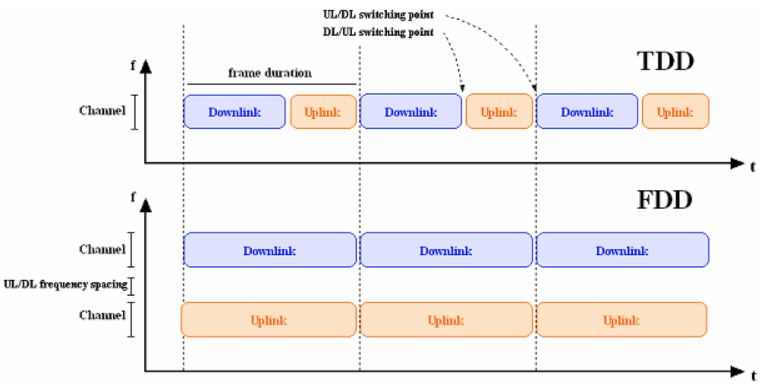

Figure 1.2: A comparison between TDD and FDD.

In 4G-LTE(Long Term Evolution) seven configurations are available for a flexible Downlink/Uplink reconfiguration to dynamically adapt to traffic demands, but these configurations are not enough for 5G because each cell can face asymmetries of different levels. The TDD is capable of handling traffic asymmetries in which resources between Downlink and Uplink can be redistributed very easily by adjusting the frames' portion durations allocated for both communication direction. But dynamically adjusting the communication directions can also cause the so-called **cross-link** interference. Several techniques to cope with cross-link interference problem have been already proposed. In [9], a procedure for joint user scheduling, precoding design, and transmit direction selection in dynamic TDD MIMO small cell networks has been studied, in which transmission directions are optimized at every frame. In [8], a procedure to optimize the bi-directional schedule of a MIMO two-way links that operate using TDD with a fixed switching times has been proposed. The schemes used to reduce the cross-link interference can be classified into two categories :

1. Centralized schemes.
2. Decentralized schemes.

In the former one, more cells can be grouped together to form a cluster and then a cluster controller is applied, which takes all the decisions for its cells. A cluster controller knows exactly what is going on between its cells and possesses all the knowledge to take optimal decisions to mitigate the cross-link interference. In [10], the authors have studied a centralized scheme, in which a novel resource allocation framework in dynamic TDD systems for interference management and asymmetric traffic adaptation scheme has been proposed. Firstly, a cluster specific cooperative dynamic Downlink/Uplink reconfiguration approach for cross-link interference cancellation was explained and then an efficient algorithm to eliminate the residual intra-cluster cross-link interference was proposed. In the decentralized schemes, every BS takes decision by its own, relying on a certain principle, e.g., each BS can decide to minimizes his Uplink/Downlink delay, maintains its SINR above a certain threshold or maximizes its cells throughput. To decide when to switch between the transmission directions, each cell can gain some knowledge about its neighbours in order to take the best decision for himself, thus mitigating to minimum the cross-link interference. In [11], a decentralized scheme for adapting the switching point positions of TDD frames for small cell networks with strong interference has been proposed. This decentralized scheme doesn't require full knowledge of its network and relies solely on low-rate signaling information exchange among the neighbouring BSs.

Chapter 2

Mathematical Formulation

Channel capacity is the most-used metric in information theory, which stands for the maximum amount of traffic that can move over a particular infrastructure. The problem of sum rate maximization of the cellular system when all parameters (channel gains and noise power) are known, can be formulated as the maximization of channel capacity. The problem can be written down by exploiting the mathematical tools offered by *calculus* and then solved by using the optimization theory. Sum rate maximization under some constraints is equivalent to maximizing so-called the **Lagrangian function**. This function consists of the objective function which we wish to maximize, together with all the constraints added up by multiplying each of them with new variables. These variables are called the **Lagrange multipliers**, which assure us the non violation of constraints. The method offers us a very operational way by transforming the maximization problem with constraints into a maximization problem without constraints. Our aim is to formulate the maximization of sum rate problem for the most general case in which we have N base stations (BSs) and M user equipments (UEs) and each BS serves his UEs by using the Time Division Duplex (TDD) mechanism. Fig. 2.1 gives a graphical idea of the scenario in the case of $N = 5$ and $M = 6$.

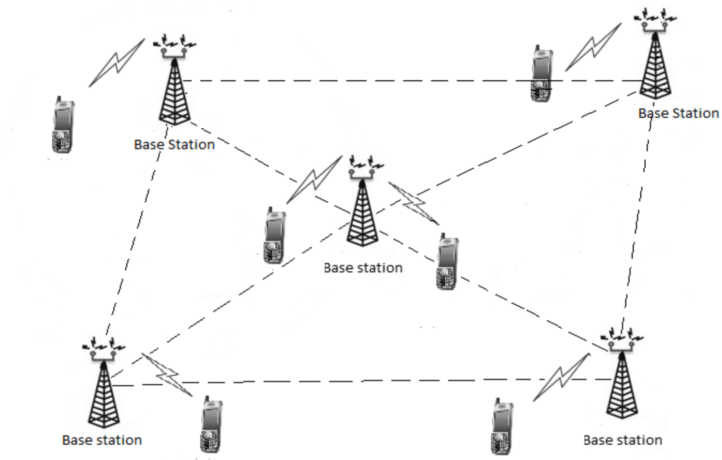

Figure 2.1: A general scenario with $N = 5$ BSs and $M = 6$ UEs.

Let $\mathcal{N}$ and $\mathcal{M}$ denote the sets which contain the indices of BSs and UEs in the system, i.e.,

$$\mathcal{N} \triangleq \{1, ..., N\}, |\mathcal{N}| = N, \tag{2.0.1}$$

$$\mathcal{M} \triangleq \{1, ...M\}, |\mathcal{M}| = M. \tag{2.0.2}$$

Each BS n ($n \in \mathcal{N}$) has a set $\mathcal{U}_n$ of associated UEs.

$$\mathcal{U}_n \triangleq \{u_{n,1}, ..., u_{n,j_n}\}, |\mathcal{U}_n| = J_n. \tag{2.0.3}$$

For example, UE-BS association is done on the minimum distance criteria where each UE is associated to its nearest BS. By creating the set $\mathcal{U}_n$ for each $n \in \mathcal{N}$, the total number of UEs in our cellular system can be obtained as $M = |\cup_{n\in\mathcal{N}} \mathcal{U}_n| = \sum_{n\in\mathcal{N}} |\mathcal{U}_n|$. We define a $N \times M$ matrix $\mathbf{H}$ called the channel matrix, which contains the channel gains among all the BSs and UEs. All of its elements are real-valued and greater or equal to zero and each element can be identified with two subscripts x and y as $H_{x,y}$, where $x \in \mathcal{N}$ identifies the x-th BS and $y \in \mathcal{M}$ identifies the y-th UE. In this way, each row $x \in \mathcal{N}$ represents all the channel gains between the BS x and all the other UEs. A further assumption that we make is that element $H_{x,y}$ of the channel matrix represents also the channel gain from UE $y \in \mathcal{M}$ back to BS $x \in \mathcal{N}$. To include the interference due to unwanted radiated emission form adjacent BSs and UEs, we introduce two matrices which will represent the interference channel gains. From now on, we will use the convention $\mathcal{M}_{\setminus i}$ to denote the set $\mathcal{M}$ without its element i. Let a $M \times M$ matrix $\mathbf{U}$ denotes the channel gains among all the UEs $\in \mathcal{M}$. The $\mathbf{U}$'s elements can be identified with two subscripts h and s as $U_{h,s}$, where $h \in \mathcal{M}$ identifies the UE which is being interfered and $s \in \mathcal{M}_{\setminus h}$ identifies the UE which is generating interference. The matrix $\mathbf{U}$ is symmetric and all of its elements are real-valued and greater than zero, which means that the interference channel gain from any UE $h \in \mathcal{M}$ to any UE $s \in \mathcal{M}_{\setminus h}$ and from UE $s \in \mathcal{M}_{\setminus h}$ back to UE $h \in \mathcal{M}$ is the same. The diagonal elements of this matrix are also zero as any UE cannot interfere with itself. Also the off-diagonal elements can be zero if there is no interference relation between the UEs identified with the subscripts of the elements which are zero. Let a $N \times N$ matrix $\mathbf{B}$ define the channel gains among all the BSs. All of its elements are real-valued and greater than zero and its characterized by being symmetric. Each of its element can be identified with two subscripts p and q as $B_{p,q}$ where $p, q \in \mathcal{N}$. In this way each row p represents the interference channel gains among the BS $p \in \mathcal{N}$ and all the other BSs. Now we define the variable P_n as the total power used by BS $n \in \mathcal{N}$ to communicate with all of his UEs $\in \mathcal{U}_n$ and let Q_n be the total power used by all the UEs $\in \mathcal{U}_n$ to communicate with their BS $n \in \mathcal{N}$. The power used by BS $n \in \mathcal{N}$ to communicate with his UE $j \in \mathcal{U}_n$ can be denoted as P^n_j. Each UE $j \in \mathcal{U}_n$ will use power Q^n_j to communicate with their BS $n \in \mathcal{N}$. Note that under the previous assumptions, the following equalities also hold: $\sum_{j\in\mathcal{U}_n} Q^n_j = Q_n$ and $\sum_{j\in\mathcal{U}_n} P^n_j = P_n$. Let X_n be a continuous variable within the interval $[0, 1]$ which identifies a fraction of the TDD's frame allocated for *Downlink* and $1 - X_n$ the fraction allocated for *Uplink*. By exploiting all the previous notations and assumptions, we can write the sum rate maximization problemas follows:

$$
\begin{aligned}
\max_{P_n, Q^n_j, X_n} \sum_{n\in\mathcal{N}} w_n \Bigg(X_n \Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \Bigg(1 + \frac{H_{n,j}\ P^n_j}{\sigma_d^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\setminus n}}} H_{k,j}\ P_k + \sum\limits_{\substack{m\in\mathcal{N} \\ m\neq n}} \sum\limits_{x\in\mathcal{U}_m} U_{x,j} Q^m_x} \Bigg)\Bigg) \\
+ (1 - X_n) \Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \Bigg(1 + \frac{H_{n,j}\ Q^n_j}{\sigma_u^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\setminus n}}} B_{k,n} P_k + \sum\limits_{m\in\mathcal{N}} \sum\limits_{\substack{y\in\mathcal{U}_m \\ y\neq j}} H_{n,y} Q^m_y} \Bigg)\Bigg)\Bigg)
\end{aligned}
\tag{2.0.4}
$$

where w_n are the weights associated to each BS $n \in \mathcal{N}$, $\sum_{\substack{k\neq n \\ k\in\mathcal{N}}} H_{k,j}\ P_k$ is the Downlink-to-Downlink interference at the UE $j \in \mathcal{U}_n$ generated by the BSs $k \in \mathcal{N}_{\setminus n}$, $\sum_{\substack{m\in\mathcal{N} \\ n\neq n}} \sum_{x\in\mathcal{U}_m} U_{x,j} Q^m_x$ is the Uplink-to-Downlink interference at the UE $j \in \mathcal{U}_n$ generated by the UEs $x \in \mathcal{M}_{\setminus \mathcal{U}_n}$, $\sum_{\substack{k\neq n \\ k\in\mathcal{N}_{\setminus n}}} B_{k,n} P_k$ is the Downlink-to-Uplink interference at the BS $n \in \mathcal{N}$ generated by the

BS $k \in \mathcal{N}_{\backslash n}$ and $\sum\limits_{m\in\mathcal{N}} \sum\limits_{\substack{y\in\mathcal{U}_m \\ y\neq j}} H_{n,y} Q_y^m$ is the Uplink-to-Uplink interference at the BS $n \in \mathcal{N}$ generated by all the UEs $y \in \mathcal{M}_{\backslash j}$. The σ_d^2 is the received noise component in Downlink and σ_u^2 is the received noise component in Uplink. The product $H_{n,j} P_j^n$ is the useful received signal power observed at the UE $j \in \mathcal{U}_n$ in Downlink and the product $H_{n,j} Q_j^n$ is the useful received power observed at the BS $n \in \mathcal{N}$ in Uplink. The fraction term of the first logarithm represents the Signal-To-Iterference Noise Ratio ($SINR$) observed at the UE $j \in \mathcal{U}_n$ in Downlink and the fraction term of the second logarithm is the SINR observed at the BS $n \in \mathcal{N}$ in Uplink.

We are considering a practical scenario, thus we impose limits on the total amount of power used by all the BSs and the UEs. In order to consider these limits, we simply restrict the total power that can be used by each BS and each UE. Mathematically speaking, the problem (2.0.4) can be rewritten as:

$$\max_{P_n, Q_j^n, X_n} \sum_{n\in\mathcal{N}} w_n \left(X_n \left(\sum_{j\in\mathcal{U}_n} \log_2 \left(1 + \frac{H_{n,j}\ P_j^n}{\sigma_d^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\backslash n}}} H_{k,j}\ P_k + \sum\limits_{\substack{m\in\mathcal{N} \\ m\neq n}} \sum\limits_{x\in\mathcal{U}_m} U_{x,j} Q_x^m} \right)\right)\right.$$
$$\left. + (1 - X_n) \left(\sum_{j\in\mathcal{U}_n} \log_2 \left(1 + \frac{H_{n,j}\ Q_j^n}{\sigma_u^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\backslash n}}} B_{k,n} P_k + \sum\limits_{m\in\mathcal{N}} \sum\limits_{\substack{y\in\mathcal{U}_m \\ y\neq j}} H_{n,y} Q_y^m} \right)\right)\right) \tag{2.0.5}$$

$$\textbf{subject to constraints} \begin{cases} P_n \leq P,\ \forall n \in \mathcal{N} \text{ and } P \geq 0 \\ Q_j^n \leq Q, \forall j \in \mathcal{U}_n,\ \forall n \in \mathcal{N} \text{ and } Q \geq 0 \\ 0 \leq X_n \leq 1, \forall n \in \mathcal{N}. \end{cases}$$

We have specific constraints imposed on our objective function which takes into account the behavior of these three variables. By looking at the (2.0.5), it can be easily noticed that our objective function has a very tough structure and solving it is not a trivial task. In order to proceed, we first write down our objective function's Lagrangian function by including all the constraints as:

$$\mathcal{L}(P_n, Q_n, X_n, \vec{\lambda}) =$$
$$\max_{P_n, Q_j^n, X_n} \sum_{n\in\mathcal{N}} w_n \left(X_n \left(\sum_{j\in\mathcal{U}_n} \log_2 \left(1 + \frac{H_{n,j}\ P_j^n}{\sigma_d^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\backslash n}}} H_{k,j}\ P_k + \sum\limits_{\substack{m\in\mathcal{N} \\ m\neq n}} \sum\limits_{x\in\mathcal{U}_m} U_{x,j} Q_x^m} \right)\right)\right.$$
$$\left. + (1 - X_n) \left(\sum_{j\in\mathcal{U}_n} \log_2 \left(1 + \frac{H_{n,j}\ Q_j^n}{\sigma_u^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\backslash n}}} B_{k,n} P_k + \sum\limits_{m\in\mathcal{N}} \sum\limits_{\substack{y\in\mathcal{U}_m \\ y\neq j}} H_{n,y} Q_y^m} \right)\right)\right)$$
$$+ \sum_{n\in\mathcal{N}} \lambda_{P_n} (P - P_n) + \sum_{n\in\mathcal{N}} \sum_{j\in\mathcal{U}_n} \lambda_{Q_{j,n}}\ (Q - Q_j^n) +$$
$$\sum_{n\in\mathcal{N}} \lambda_n (1 - X_n) + \sum_{n\in\mathcal{N}} \lambda_{N+n}\ X_n \tag{2.0.6}$$

where the first and second terms are our objective function, the third and fourth terms take into account the constraints stated under (2.0.5) by multiplying each constraint with their associated Lagrange multipliers λ_i and $\vec{\lambda}$ is a vector which contain all the multipliers. The function (2.0.5) is constrained and by expressing it through its Lagrangian (2.0.6) it has become a non-constrained function and the multipliers make sure that the constraints are satisfied. Now, instead of looking for the optimum of (2.0.5), we will focus our search on the optimum of (2.0.6). The solution can be found if and only if the so-called **Stationarity condition** and the **Complementarity conditions** of (2.0.6) are satisfied. Both of the conditions stated above are called the first order optimality conditions and can be expressed as follows:

Stationarity condition

$$\nabla_{P_n,Q_j^n,X_n,\vec{\lambda}}(\mathcal{L}) = 0 \Leftrightarrow \begin{cases} \sum_{n\in\mathcal{N}} w_n \quad \nabla\Bigg(X_n \Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \left(1 + \frac{H_{n,j}\ P_j^n}{\sigma_d^2 + \sum\limits_{\substack{k\neq n\\ k\in\mathcal{N}_{\setminus n}}} H_{k,j}\ P_k + \sum\limits_{\substack{m\in\mathcal{N}\\ m\neq n}} \sum\limits_{x\in\mathcal{U}_m} U_{x,j} Q_x^m} \right)\Bigg)\Bigg) \\ (1-X_n)\Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \left(1 + \frac{H_{n,j}\ Q_j^n}{\sigma_u^2 + \sum\limits_{\substack{k\neq n\\ k\in\mathcal{N}_{\setminus n}}} B_{k,n} P_k + \sum\limits_{m\in\mathcal{N}} \sum\limits_{\substack{y\in\mathcal{U}_m\\ y\neq j}} H_{n,y} Q_y^m} \right)\Bigg)\Bigg)\Bigg) \\ +\nabla\Bigg(\sum_{n\in\mathcal{N}} \lambda_{P_n}(P-P_n) + \sum_{n\in\mathcal{N}} \sum_{j\in\mathcal{U}_n} \lambda_{Q_{j,n}}\ (Q-Q_j^n) + \\ \sum_{n\in\mathcal{N}} \lambda_n(1-X_n) + \sum_{n\in\mathcal{N}} \lambda_{N+n}\ (X_n)\Bigg) = 0 \end{cases} \tag{2.0.7}$$

Complementarity conditions

$$\begin{aligned} &\lambda_{P_n}(P-P_n) = 0,\ \forall n \in \mathcal{N} \\ &\lambda_{Q_{j,n}}\ (Q-Q_j^n) = 0, \forall j \in \mathcal{U}_n \text{ and } \forall n \in \mathcal{N} \\ &\lambda_n(1-X_n) = 0,\ \forall n \in \mathcal{N} \\ &\lambda_{N+n}\ (X_n),\ \forall n \in \mathcal{N} \end{aligned} \tag{2.0.8}$$

2.1 Discretization process

The objective function (2.0.5) is non-convex due to the interference terms present at the denominator. This implies that it is characterized by having multiple local optimum. Finding all of them by satisfying the conditions (2.0.7)-(2.0.8) is a very difficult task. Notice that, the variables P_n, Q_j^n and $X_n, \forall n \in \mathcal{N}$ are continuous and finding their solution can be very laborious. In order to reduce the complexity of the (2.0.6) we discretize the continuous variable X_n to the values 0 and 1 corresponding to the case in which each BS have always one or the other behavior, i.e.,

$$X_n = \begin{cases} 1 & \text{BS } n \text{ always transmits in Downlink} \\ 0 & \text{BS } n \text{ always transmits in Uplink.} \end{cases}$$

In this way, we are restricting the solution space to a subspace where X_n assumes only two values. By discretizing X_n, we can assume its behavior like a switch which indicates if the BS $n \in \mathcal{N}$ is transmitting in *Downlink* or *Uplink* for the whole duration of the frame. It may be worth emphasizing that as we are in a TDD scenario, X_n assumes value 1 (Downlink) if and only if P_n is strictly positive and Q_j^n is zero $\forall j \in \mathcal{U}_n$, or assumes value 0 (Uplink) if and only if Q_j^n $(\forall j \in \mathcal{U}_n)$ is strictly positive and P_n is zero $\forall n \in \mathcal{N}$. Notice that the values P_n and Q_j^n are defined as greater or equal to zero, which means it may also present situations in which P_n and Q_j^n are zero and there is no exchange of data traffic. Suppose we fix the values of P_j^n and Q_j^n, one solution of our objective function is obtained where X_n assumes values 0 or 1. It is easy to understand the previous statement, as the values of P_j^n and Q_j^n are fixed and the channel gains are known, it is easy to evaluate the SINRs. As the values of X_n are not part of the argument of logarithms, it is easy to choose the value of X_n which maximizes the sum rate. Which simply means that every BS can choose if he should transmits in *Uplink* or *Downlink* by simply evaluating the SINRs. It may be worth mentioning that by discretizing X_n we are reducing the solution space of our objective function but in this way the complexity to find the solution reduces significantly as the variable X_n is fixed to two particular values and only the value of P_n and Q_j^n are continuous.

2.2 Discretized version and its solution

Now we consider the case in which X_n is being discretized and can assume only the values 0 or 1. We are sure that the solution space is reduced by discretizing X_n, however it still yields a solution which is the optimum one. Now the original problem (2.0.5) can be reformulated as a function of continuous variables P_n and Q_j^n and a discrete variable X_n. Everything remain the same except for the constraint which takes into account the behavior of the variable X_n. The objective function (2.0.5) can be rewritten with a slightly modification of the constraints regarding X_n as:

$$\max_{P_n,Q_j^n,X_n} \sum_{n\in\mathcal{N}} w_n \Bigg(X_n \Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \Bigg(1 + \frac{H_{n,j}\ P_j^n}{\sigma_d^2 + \sum\limits_{\substack{k\neq n\\ k\in\mathcal{N}_{\setminus n}}} H_{k,j}\ P_k + \sum\limits_{\substack{m\in\mathcal{N}\\ m\neq n}} \sum\limits_{x\in\mathcal{U}_m} U_{x,j} Q_x^m} \Bigg)\Bigg)$$
$$+ (1-X_n) \Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \Bigg(1 + \frac{H_{n,j}\ Q_j^n}{\sigma_u^2 + \sum\limits_{\substack{k\neq n\\ k\in\mathcal{N}_{\setminus n}}} B_{k,n} P_k + \sum\limits_{m\in\mathcal{N}} \sum\limits_{\substack{y\in\mathcal{U}_m\\ y\neq j}} H_{n,y} Q_y^m} \Bigg)\Bigg)\Bigg), \tag{2.2.1}$$

$$\textbf{subject to constraints} \begin{cases} P_n \leq P,\ \forall n\in\mathcal{N} \text{ and } P\geq 0 \\ Q_j^n \leq Q, \forall j\in\mathcal{U}_n,\ \forall n\in\mathcal{N} \text{ and } Q\geq 0 \\ X_n = 0 \text{ or } X_n = 1, \forall n\in\mathcal{N}. \end{cases}$$

The structure of the conditions (2.0.7)-(2.0.8) do not change and we simply keep treating everything as it is. The only thing that changes is the variable X_n, which has become discrete and we cannot compute its gradient ∇X_n in the condition (2.0.7). Discretization of X_n also restricts the possible values of multipliers λ_n and λ_{N+n}.

Now after the discretization of the variables $X_n, \forall n\in\mathcal{N}$ which reduced the complexity of our objective function, let's move towards it solution procedure. Let's recall that Lagrangian function (2.0.6) includes in itself all the constraints and can be treated as a function which is non-constrained. It is non-convex because of the interference terms present at the denominator, thus it is characterized by having multiple local optimum. We will strive to find them all and the one which will make our objective function assumes the highest value, will be declared as the global optimum. The solution can be found by using the **Multi-start algorithm** offered by the optimization toolbox of MATLAB. This algorithm commits to find all the local minimums, thus we need to write our objective function (2.2.1) as $-(2.2.1)$ and by doing so, all the local minimums will correspond to our local maximums. It also generate by itself the Lagrangian function by multiplying each constraint with its multipliers. The local maximum which will make assume our objective function (2.2.1) the highest value, it is returned in a separate vector. In short words, the Multi-start algorithm works as follows:

1. It requires in the input, number of different feasible starting points.
2. It starts moving from all the starting points towards the negative gradient direction $-\nabla f(\vec{x})$, which is the direction of the fastest decrease, with a very small and decreasing step size ϵ and it stops when it reaches the condition of null gradient for all the starting points.
3. When the algorithm terminates, it returns the vector called GlobalOptimumSolutions vector which includes all the local optimum of the objective function. It also return the local minimum which can be declared as the global minimum (our maximum).

To get an idea of how the multi-start algorithm works, its flowchart is shown in Fig. 2.2 and its behavior is shown in Fig. 2.3.

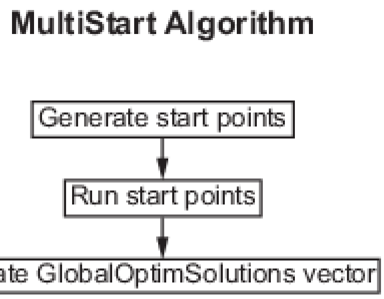

Figure 2.2: Flowchart of the Multi-start algorithm.

In order to exploit the Multi-start algorithm in MATLAB, the objective function's structure (2.2.1) has to be well defined and the following steps have to be followed:

Procedure to define the objective funtion's structure

1. Define the objective function (2.2.1) as a file or an anonymous function.
2. Create the constraints structure.
3. Define a feasible start point vector $\vec{x_o}$.
4. Create an options structure using *optimoptions*. For example,
   options = optimoptions(@fmincon,,'Algorithm','specify Algorithm');
   where *@fmincon* is a gradient based local solver and *Algorithm* is the algorithm exploited by *@fmincon* to converge.
5. Create a problem structure which includes the objective function, constraints and options. For example, *problem = createOptimProblem('fmincon','xo', $\vec{xo}$,'objective',...*

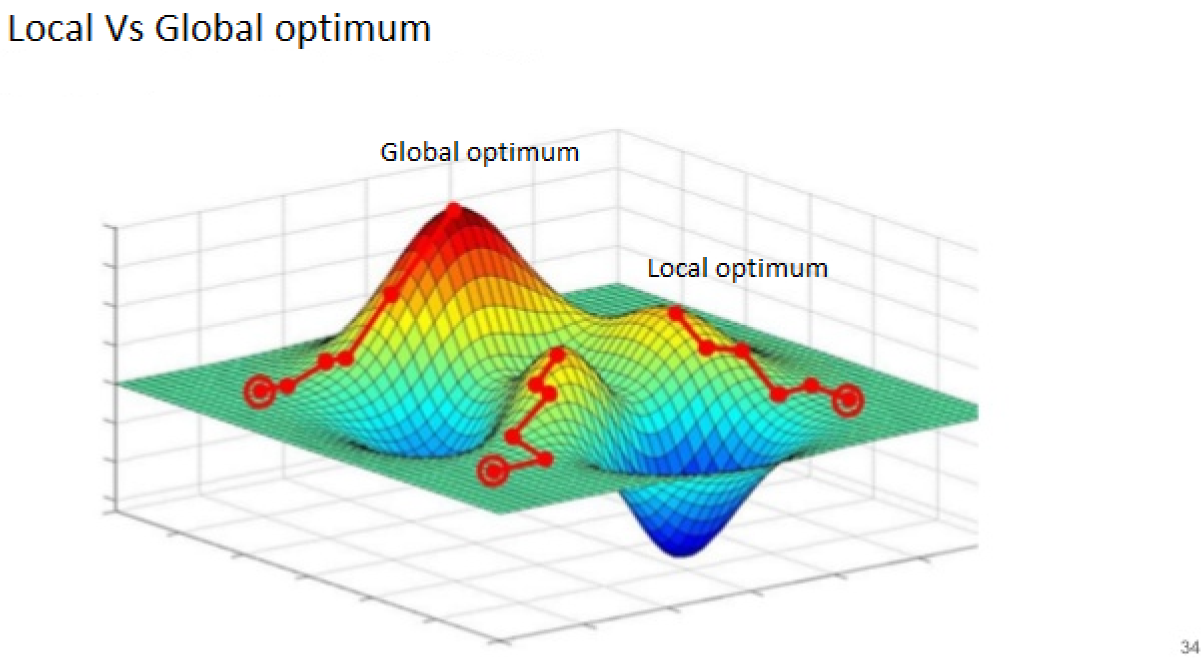

Figure 2.3: The Multi-start algorithm's behavior when we have to deal with a simple multivariate function, characterized by having three local optimum.

fun,'Aineq',A,'bineq',b,'Aeq',Ae,'beq',be, 'options',opts); where fun is the objective function (2.2.1), A is the inequality constraints' matrix, b is the upper bounds' vector for A, Ae is the equality constraints' matrix and b is its equality bounds'.

Let's recall that the local solver *@fmincon* finds the minimums of a constrained multivariate function $f(\vec{x})$, thus in our case the (2.2.1) has to be written as -(2.2.1) at step 1. By doing so, all the local minimums found by *@fmincon* will match with the local maximums of our objective function. At step 4, we need to specify the algorithm that we want that the local solver *@fmincon* uses. The MATLAB offers us four fundamental algorithms which can be used by *@fmincon* to find the local minimums of any multivariate problems. All of them use the first order optimality measure to declare the convergence. As briefly stated before, all these algorithms move towards the direction of negative gradient until they reach the local minimum. This is their principle of operation, however they distinguish among them because of the different techniques they use to satisfy the non violation of constraints and to reduce the complexity of any complex function.

Algorithms used by *@fmincon*

1. The **Interior point optimization** algorithm, which exploits the slack variables and the barrier functions and try to solve a sequence of approximate minimization (- maximization) problems.

2. The **Active set** optimization algorithm, which uses a sequential quadratic programming (SQP) method. In this method, the algorithm tries to solve a quadratic programming subproblem at each iteration. The *@fmincon* updates an estimate of the Hessian of the Lagrangian function by using the **BFGS** formula. The **BFGS** stands for **Broyden–Fletcher–Goldfarb–Shanno** formula which is widely used for solving unconstrained non-linear problems.

3. The **SQP and SQP-legacy optimization** algorithm, which is very similar to the Active set algorithm with just few differences.

4. The **Trust-Region-Reflective optimization** algorithm, which is a subspace of trust region method and based on the interior-reflective Newton method.

Once all the 5 steps mentioned before have been followed, to solve the optimization problem we need to create a Multi-start object with the command *ms = MultiStart;* and the **Multi-start** algorithm can be launched with the command *run(ms,problem,k)*. The *ms* is the Multi-start object, *problem* is the objective function's structure created at step 5, and k is an integer for which the algorithm randomly generates $k-1$ start points. It solves the optimization problem for all the k points ($k-1$ randomly generated + $\vec{x_0}$) by moving in the direction of negative gradient for all the k points and it stops when it runs out of starting points reaches the condition of gradient being zero for all the k points. The conditions (2.0.7) and (2.0.8) are called the first order optimality conditions and the local solver *@fmincon* verifies if they are satisfied with some tolerance. More precisely, it verifies if:

Stationarity condition's verification by *@fmincon*

$$||\nabla_{P_n,Q_j^n,\vec{\lambda}}(\mathcal{L})|| \Leftrightarrow \begin{cases} ||\sum_{n\in\mathcal{N}} w_n \quad \nabla\Bigg(X_n \Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \Bigg(1 + \frac{H_{n,j}\ P_j^n}{\sigma_d^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\setminus n}}} H_{k,j}\ P_k + \sum\limits_{\substack{m\in\mathcal{N} \\ m\neq n}} \sum\limits_{x\in\mathcal{U}_m} U_{x,j} Q_x^m}\Bigg)\Bigg) \\ \quad +(1-X_n)\Bigg(\sum_{j\in\mathcal{U}_n} \log_2 \Bigg(1 + \frac{H_{n,j}\ Q_j^n}{\sigma_u^2 + \sum\limits_{\substack{k\neq n \\ k\in\mathcal{N}_{\setminus n}}} B_{k,n} P_k + \sum\limits_{m\in\mathcal{N}} \sum\limits_{\substack{y\in\mathcal{U}_m \\ y\neq j}} H_{n,y} Q_y^m}\Bigg)\Bigg)\Bigg) \\ \quad + \sum_{n\in\mathcal{N}} \nabla(\lambda_{P_n}(P-P_n)) + \sum_{n\in\mathcal{N}} \sum_{j\in\mathcal{U}_n} \nabla(\lambda_{Q_{j,n}}\ (Q-Q_j^n)) \\ \quad + \sum_{n\in\mathcal{N}} \nabla(\lambda_n(1-X_n)) + \sum_{n\in\mathcal{N}} \nabla(\lambda_{2n}\ X_n)|| \leq 1e^{-6} \end{cases} \tag{2.2.2}$$

Complementarity conditions' verified by *@fmincon*

$$\begin{aligned} &||\ \overrightarrow{\nabla(\lambda_{P_n}(P-P_n))}|| \leq 1e^{-6},\ \forall n \in \mathcal{N} \\ &||\ \overrightarrow{\nabla(\lambda_{Q_{j,n}}\ (Q-Q_j^n))}|| \leq 1e^{-6},\ \forall j \in \mathcal{U}_n \text{ and } \forall n \in \mathcal{N} \\ &||\ \overrightarrow{\lambda_n(1-X_n)}|| \leq 1e^{-6},\ \ \forall n \in \mathcal{N} \\ &||\ \overrightarrow{\lambda_{N+n}\ (X_n)}|| \leq 1e^{-6}.\ \ \forall n \in \mathcal{N} \end{aligned} \tag{2.2.3}$$

The local solver uses the infinity norm of (2.0.7) and (2.0.8) to verify their validity under the tolerance of $1e^{-6}$. It is also possible to choose smaller tolerance values but the time to solve the problem increases rapidly. The algorithm works properly under the only hypothesis that our multivariate objective function (2.2.1) is well defined and differentiable in the neighborhood of the initial feasible point $\vec{x_0}$. For the other $k-1$ random points, if for any point the objective function is not differentiable in its neighborhood, this point is automatically discarded.

2.3 Practical scenario

Now that we have understood how to find the solution of our optimization problem (2.2.1), we consider a particular scenario made of 16 BSs and 16 UEs. More formally we can define it as follows:

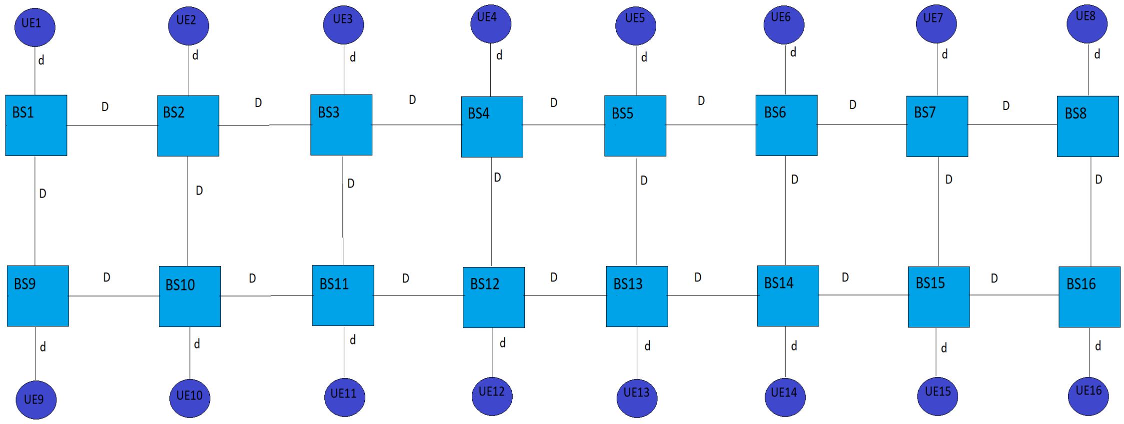

Figure 2.4: Practical Scenario considered for the simulation made with 16 BSs and 16 UEs. Each BS has just one UE at distance d and distance among the BSs are of step D.

Let $\mathcal{N}$ and $\mathcal{M}$ be the BSs' and UEs' sets:

$$\mathcal{N} = \{1, 2, ..., 16\}, |\mathcal{N}| = 16, \tag{2.3.1}$$

$$\mathcal{M} = \{1, 2, ..., 16\}, |\mathcal{M}| = 16. \tag{2.3.2}$$

The UEs and BSs are positioned as shown in Fig. 2.4, where each UE is distant d from its BS and the distance among the BSs is discretized to the step size D and the $|\mathcal{U}_n| = 1$. We assume that UE $i \in \mathcal{M}$ is associated to the BS $i \in \mathcal{N}$. We want to simulate a practical scenario in a free space, thus the *FRIIS* formula which takes into account the ***Free-space path loss (FSPL)*** effect and the gains of transmission and receiving antennas, will also be included. The formula is defined as follows:

$$\frac{P_{rx}}{P_{tx}} = G_{tx}\, G_{rx}\, (\frac{\lambda}{4\pi d})^2 \tag{2.3.3}$$

where the parameters have the following meaning:

1. P_{rx} is the power received by the receiving antenna.
2. P_{tx} is the transmitted power.
3. G_{tx} is the transmitting antenna's gain.
4. G_{rx} is the receiving antenna's gain.
5. λ is the wave length, calculated as $\lambda = \frac{C}{f}$, where C is the speed of light in a vacuum and f is the frequency.
6. d is the distance between the transmission antenna and the reception point.

By looking at the (2.3.3), it can be clearly notice that the transmitted power decay as the square of the distance. The FSPL is defined as:

$$\mathbf{FSPL} = (\frac{4\pi d}{\lambda})^2 \tag{2.3.4}$$

which is a loss in signal strength of an electromagnetic wave that would result from a line-of-sight path through a free space with no obstacles nearby to cause reflection or diffraction. To take into account the FRIIS formula in our objective function, we will include the (2.3.3) within the channel matrix $\mathbf{H}$. For the sake of simplicity, we will model the channel gains as $H_{n,j} = \frac{C}{d_{n,j}^2}$, where C is a constant value which includes all the constants terms which appear in (2.3.3) and $d_{n,j}^2$ is the distance between the BS n and its solely UE j. As for the matrices $\mathbf{B}$ and $\mathbf{U}$, their interference channel gains will be modeled in the same way. In order to keep things simple, we will use the normalized version of channel gains to the maximum distance and for simplicity we will assume the constant C equal to 1. The maximum available powers for BSs and UEs allowed by the regulations which deal with wireless communication standards are P = 46 dBm and Q=30 dBm, respectively. We suppose that each BS and UE uses the maximum available power P and Q to communicate. We normalize these values to the maximum available power P for the BSs which yields $P = 1$ and $Q = 2.5\ 10^{-2}$. We also assume the values noise powers in *Uplink* and *Downlink* equal to 1.

To normalize the channel gains, we limit the maximum coverage area of each BS to 1 km. Consider for example the channel gain between the BS n and its solely UE j:

$$H_{n,j}(d_{max}) = \frac{1}{d_{max}^2} = 1 \rightarrow d_{max} = 1\ km. \tag{2.3.5}$$

In this way, we obtain the value of channel gain $H_{n,j}$ at the maximum distance of 1 km, which yields each of his UE a SNIR of 0 dB.

We assume that all the BSs can be accessed by their UE directly starting from the minimum distance d_{min} which yields each of his UE an SNIR of 15 dB. By solving:

$$H_{n,j}(d_{min}) = \frac{1}{d_{min}^2} = 10^{\frac{15}{10}} \rightarrow d_{min} = \sqrt{10^{-1.5}} = 0.178\ km \tag{2.3.6}$$

By adopting the previous criteria, we can rewrite the objective function (2.2.1) in this particular scenario as follows:

$$\max_{P_n, Q_j^n, X_n} \sum_{n \in \mathcal{N}} \sum_{\substack{j \in \mathcal{U}_n, \\ |\mathcal{U}_n|=1}} w_n \Bigg(X_n \log_2 \Bigg(1 + \frac{H_{n,j}\ P}{\sigma_d^2 + \sum\limits_{\substack{k \neq n \\ k \in \mathcal{N}_{\setminus n}}} H_{k,j}\ P + \sum\limits_{\substack{m \in \mathcal{N} \\ m \neq n}} \sum\limits_{x \in \mathcal{U}_m} U_{x,j} Q} \Bigg) \Bigg)$$
$$+ (1 - X_n) \log_2 \Bigg(1 + \frac{H_{n,j}\ Q}{\sigma_u^2 + \sum\limits_{\substack{k \neq n \\ k \in \mathcal{N}_{\setminus n}}} B_{k,n} P + \sum\limits_{m \in \mathcal{N}} \sum\limits_{\substack{y \in \mathcal{U}_m \\ y \neq j}} H_{n,y} Q} \Bigg) \Bigg) \tag{2.3.7}$$

$$\textbf{subject to constraints} \begin{cases} P_n \leq P,\ \forall n \in \mathcal{N} \text{ and } P \geq 0 \\ Q_j^n \leq Q, \forall j \in \mathcal{U}_n,\ \forall n \in \mathcal{N} \text{ and } Q \geq 0 \\ X_n = 0 \text{ or } X_n = 1, \forall n \in \mathcal{N}. \end{cases}$$

Where $P_n = P$ and $Q_j^n = Q$ and channel gains contain the path-loss term and P and Q are limited to the values previously specified. Now we will search for it's solution with the MATLAB code which implements the Multi-start algorithm. We will use the optimal values of P_n, Q_n and X_n found by the Multi-start algorithm and see how the mean rate for each UE varies as the distance d between the BSs and their UEs increases.

In Table 2.1 we present the parameter values used for the scenario of our interest and Table 2.2 shows the optimal values of P_n, Q_j^n, X_n found by the Multi-start algorithm for each BS and each UE which maximize their sum rate. Finally, Fig. 2.5 shows how the mean rate changes as a function of d, which is the distance between each BS and their only associated UE.

Table 2.1: Simulation parameters

Parameter	symbol	value
Maximum power for BSs	P	1
Total power for UEs	Q	$2.5\ 10^{-2}$
Distance among the BSs	D	0.20 km
UEs' max distance from the BSs	d	1 km
UEs' min distance from the BSs	d	0.178 km
Noise variance in Downlink	σ_d^2	1
Noise variance in Uplink	σ_u^2	1

BS/UE	P	Q	X_n
1	1	0	1
2	0.89	0	1
3	0.93	0	1
4	0.93	0	1
5	0.93	0	1
6	0.92	0	1
7	0.89	0	1
8	1	0	1
9	1	0	1
10	0.89	0	1
11	0.92	0	1
12	0.93	0	1
13	0.92	0	1
14	0.89	0	1
15	1	0	1
16	1	0	1

Table 2.2: Decision table obtained from simulation results of multi-start algorithm.

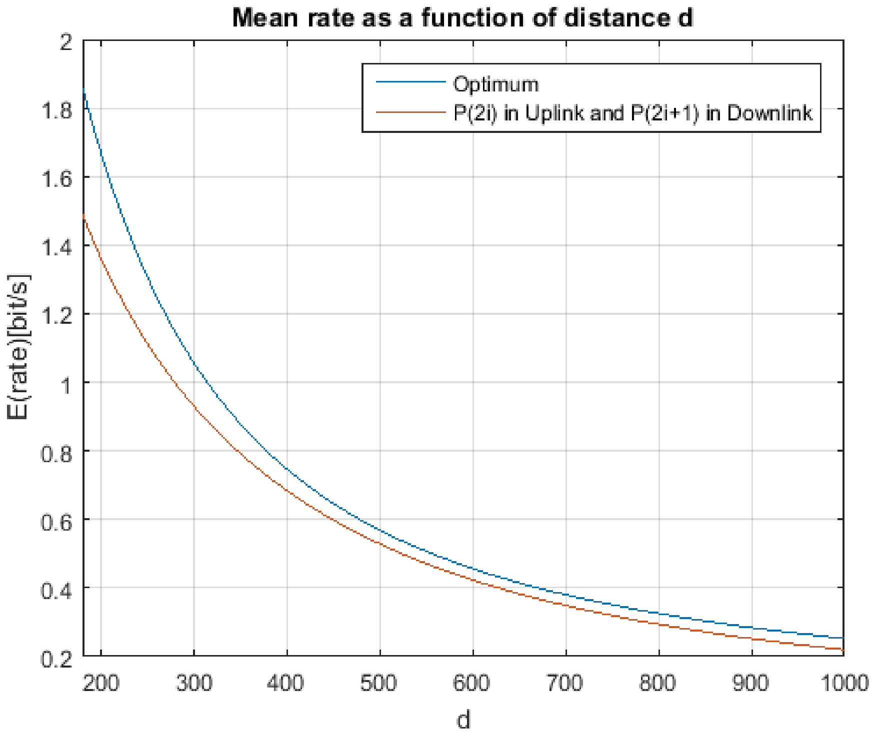

Figure 2.5: Mean rate as a function of distance between the BSs and their associated UEs.

Chapter 3

Game Formulation

The problem can also be formulated in the terms of a game and then solved with the theoretical tools offered by the **game theory**. In this chapter, we will exploit these tools to formulate our problem as a game in which each Base-Station (BS) will act as a player and will have to choose among the strategies : *Uplink or Downlink*, in order to maximize his own *well-being* (throughput). This chapter will be structured as follows: First of all, a brief introduction to game theory will be given and then we will try to formulate our problem as a game in different ways depending on its distinctive features. These games will differ depending on the quantity of knowledge possessed by each player (BS) about his or other players' SINRs and the channel situated among his and other players' User-Equipments (UEs).

3.1 Brief introduction to the Game theory

Imagine yourself waking up in the morning and deciding what to eat for breakfast. You may go to a nice cafeteria in your neighborhood, in which case you could have a large variety of foods from which to choose or you just prefer to have a breakfast at home, so you end up by choosing among just two or three types of cereals. This trivial yet ubiquitous situation is an example of a *decision problem.* Decision problems confront us daily, as individuals and also as groups. All these problems share a similar structure: an individual (or a player) faces a situation in which he has to choose from one of the several alternatives and every choice will result in some outcome. Every decision has a consequence which will be borne by the player himself and might influence other individuals (players) too. Any player who wants to take a decision regarding any decision problem in an intelligent way, must be aware of three fundamental features of the problem: his possible choices, results of each of those choices and the influence that every decision has on his will being. If a player is able to understand these three things, he can make the best decision for himself or for the whole group, depending on his interests. So, we can say that any decision problem consists of these three main features:

1. **Actions** are all the alternatives from which the player can choose.
2. **Outcomes** are the possible consequences that can result from any of the actions.
3. **Preferences** describes how the player ranks the set of possible outcomes, from most desired to least desired.

To express the player's preference between any two possible outcomes, a **preference relation** symbol $\succsim$ is used, e.g. a preference between two possible outcome x and y can be expressed as $x \succsim y$, which means " x is at least good as y", which is consistent with saying that x is better than y or equally as good as y. To distinguish between these two scenarios, a **strict preference relation** " $\succ$ " can be used to state that "x is better than y" and **indifference relation** " $\sim$ " for "x and y are equally good". Lets denote with $\mathcal{A}$ the actions set and with $\mathcal{X}$ the outcomes set.

$$\mathcal{A} = \{a_1, ..., a_n\},\ a_i \in \mathcal{A} \text{ are the actions} \tag{3.1.1}$$

$$\mathcal{X} = \{x_1, ..., x_j\},\ x_i \in \mathcal{X} \text{ are the outcomes} \tag{3.1.2}$$

Both of them may be finite or infinite and to be able to represent the player's preference over outcomes, two assumptions are needed which can help him to think through a decision problem. First, the player must be able to rank any two possible outcomes from $\mathcal{X}$, which can be expressed more formally by the following axiom:

The Completeness Axiom := The preference relation $\succsim$ is **complete**, hence any two outcomes $x, y \in \mathcal{X}$ can be ranked by the preference relation.

The completeness axiom imposes that the player can always decide between two possible outcomes. The second assumption imposes that the player is able to decide among any number of possible outcomes and can be stated as follows:

The Transitivity Axiom := The preference relation $\succsim$ is **transitive**, hence more than two outcomes can also be ranked, e.g. if $x, y, z \in \mathcal{X}$ and $x \succsim y$ and $y \succsim z$ then $x \succsim z$.

These two assumptions make sure that when the player has to make a decision among any number of outcomes, the completeness axiom guarantees that two outcomes can be ranked and the transitivity axiom guarantees that there will be no contradictions in the ranking, which could create an indecisive cycle. A preference relation which satisfies both of the assumptions stated above is called a **rational preference relation**. When we deal with players with rational preferences, we can replace with preference relation with a much friendlier and more operational function, so called a **payoff function**:

Definition 1. *A **payoff function** $u : \mathcal{X} \to \mathbb{R}$ represents the preference relation $\succsim$ if for any pair $x, y \in \mathcal{X}$, $u(x) \geq u(y)$ if and only if $x \succsim y$.*

In short words, the preference relation $\succsim$ is represented by the payoff function u which assigns to each outcome in $\mathcal{X}$ a real number if and only if the function assigns a higher value to higher-ranked outcomes. It might be useful to highlight that the payoff function is convenient but the payoff values by themselves have no meaning, e.g. if $x \succsim y$ then we can assign values to their payoff functions $u(x) = 5$ and $u(y) = 1$ or $u(x) = 50$ and $u(y) = 10$. The reason of using payoff functions instead of preferences is that we can build a theory of how decision makers with rational preferences ought to behave. They will choose actions that maximize a payoff function that represents their preferences over the set $\mathcal{X}$. The only requisite to define a payoff function over the rational preferences is :

Proposition 1. *If the set of outcomes $\mathcal{X}$ is finite then any rational preference relation over $\mathcal{X}$ can be represented by a payoff function u.*

We can now introduce the concept of *Homo economicus* or *"economic man"*, which is widely used in *economics* and *game theory.* A *Homo economicus* is rational in that he chooses actions that maximize his well-being as defined by his payoff function over the resulting outcomes. The assumption that the player is rational lies on a fundamental paradigm which is known as **rational choice paradigm**, which implies that when a player is choosing among his actions, he will be guided by rationality to choose the best action for himself. By adopting this paradigm, we are strictly imposing some assumptions that can be listed as follows:

Rational choice assumptions The player completely understands the decision problem by knowing :

1. All possible actions $\mathcal{A}$.

2. All possible outcomes $\mathcal{X}$.

3. Exactly how each action affects the outcomes.

4. His rational preferences over outcomes.

All of these four conditions must be satisfied if we expect that the player makes rational choices, thus he chooses an action that maximizes his well being. If we drop any of these four assumptions, the player may not be able to decide which action to play or what could be the consequences related to his chosen actions, e.g. if 1) is unknown then the player could not be able to decide which action to play or if 2) or 3) are unknown then he may not correctly foresee the actual consequences. Finally, if 4) is unknown then the player may perceive incorrectly the effect of his choice's consequences on his well-being. To operationalize the paradigm of rationality the player must choose the best action from the set $\mathcal{A}$, yet the payoff function has been defined over outcomes. It would be more useful if the payoff function is defined on the actions instead over the outcomes. To be precise, let $f : \mathcal{A} \to \mathcal{X}$ be the function that maps actions into outcomes, and let u be the payoff function over outcomes as defined above. We can define the composite function v as $v = u \circ f : \mathcal{A} \to \mathbb{R}$, where $v(a) = u(x(a))$ and a is an action choosen from the set $\mathcal{A}$. Now, that we understood the concept of rational paradigm, the definition of *Homo economicus* mentioned before, can be defined more formally as:

Definition 2. *A player facing a decision problem with a payoff function $v(\cdot)$ over actions is rational if he chooses an action $a \in \mathcal{A}$ that maximizes his payoff. That is, $a^* \in \mathcal{A}$ is chosen if and only if $v(a^*) \geq v(a)$, $\forall a \in \mathcal{A}$.*

By this definition we are defining a player who has rational preferences and is rational in that he understands all the aspects of the decision problem, the consequences of his actions and will always choose an action $a \in \mathcal{A}$ that yields him the highest payoff.

Even if this structure can be very useful to analyze a wide variety of decision problems (or games), it still lacking of one important element which is the probabilistic relationship between actions and outcome, so called randomness. In order to understand it, consider the following example: A division manager has to decide whether he should embark on a new Research and Development (R&D) project or not. His actions can be denoted as g for going ahead and n for not going ahead, so that $\mathcal{A} = \{g, n\}$. Imagine that there are only two possible outcomes: his new product line is successful which is equivalent to a profit of 10, or his product line is obsolete which is equivalent to a profit of 0, so that $\mathcal{X} = \{0, 10\}$. His final outcome is influenced by so many other factors which are out of his control, in addition to his choice. When he takes the decision, he has to take into account many aspects, e.g., if he decides not to go ahead, overtime his main product could become obsolete and outdated and his company would not be able to compete in the market or maybe profits will still continue to flow in. And if he goes ahead, he could have a vast improvement in his profits or perhaps the research could fail and no new products will emerge. In this case he will be left with a long list of bills to pay for the expensive R&D miserably failed. Imagine that a successful product line is more likely to be created if the player chooses to go ahead with the R&D project, while it is less likely to be created if he does not. More precisely, the odds are 3 to 1 that success happens if g is chosen, while odds are 50-50 if s is chosen. By using the formal language of probabilities, we can think of it as: If the player chooses g then the probability of a payoff of 10 is 0.75 and the probability of a payoff of 0 is 0.25. If he chooses s then the probability of a payoff of 10 is 0.5 and the probability of a payoff of 0 is 0.5. We can think of the division manager as the player choosing between two **lotteries**, which is exactly defined by a random payoff. In our example, selecting g by the manager is like choosing a lottery that pays zero with probability 0.25 and pays 10 with probability 0.75. On the other hand, choosing n yields him a payoff of 10 or 0 with probabilities 0.5 and 0.5, respectively. We can think of these lotteries as choices made by another player that is often called the "Nature", which can be seen as en external individual who makes decisions for the actual players of the game and has no strategic interests in the outcomes. First of all, the player choose an action from the set $\mathcal{A}$ and conditioned on this action, Nature chooses a probability distribution over the outcomes $\mathcal{X}$. Now that we have understood the meaning of lotteries and the role played by the Nature, a formal definition of a lottery can be stated more formally as follows:

Definition 3. *A simple lottery over outcomes $\mathcal{X} = \{x_1, x_2, ..., x_n\}$ is defined as a probability distribution $p = (p(x_1), p(x_2), ..., p(x_n))$, where $p(x_k) \geq 0$ is the probability that x_k occurs and $\sum_{k=1}^{n} p(x_k) = 1$.*

To be precise, the lottery chosen by a Nature is conditional on the action taken by a player. Hence, given any action $a \in \mathcal{A}$, a conditional probability that $x_k \in \mathcal{X}$ occurs is given by $p(x_k|a)$, where $p(x_k|a) \geq 0$ and $\sum_{k=1}^{n} p(x_k|a) = 1$. When a decision problem contains randomness, the definition of payoff has to be rewritten in order include these random events. The intuitive idea is about averages. Sometimes the actions chosen by the player can make him gain some profit or make him loose some of his well being, but if on average things turn to out on the positive side, then we can view the player's actions as pretty good because the gains will be more than the losses. To put this in a formal way, an **expected payoff** can be defined as follows:

Definition 4. *Let $u(x)$ be the player's payoff function over outcomes in $\mathcal{X} = \{x_1, x_2, ..., x_n\}$ and let $p = (p_1, p_2, ..., p_n)$ be a lottery over $\mathcal{X}$ such that $p_k = Pr\{x = x_k\}$. Then we define the player's* ***expected payoff from the lottery*** *p as*

$$E[u(x)|p] = \sum_{k=1}^{n} p_k u(x_k)$$

It is worth mentioning that this is the general definition of the *expected payoff*, and will be restated from case to case when we will proceed with the classifications of our games based on the quantity of information that each player will hold. As mentioned before, a rational player chooses actions that maximize his payoff among the set of all possible actions. When the outcomes are stochastic, the player must know that by choosing actions he is choosing lotteries, and he knows exactly what the probability of each outcome is, conditioned on the choice of an action. A natural way to define rationality for decision problems with random outcomes is as follows:

Definition 5. *A player facing a decision problem with a payoff function $u(\cdot)$ over outcomes is rational if he chooses an action $a \in \mathcal{A}$ that maximizes his expected payoff. That is, a^* is chosen if and only if $v(a^*) = E[u(x)|a^*] \geq E[u(x)|a] = v(a)$ for all $a \in \mathcal{A}$.*

Notice the notation $v(a) = E[u(x)|a], \forall a \in \mathcal{A}$, $\forall x \in \mathcal{X}$ defines the expected payoff of an action given the distribution over outcomes. This is a convention that can be used, as the case of concern is what a player should do, and this notation implies that his ranking should be based on his actions. The definition 5 states that, the player that is rational and understand the stochastic consequences of each of his actions, will choose an action that offers him the highest expected payoff [5]- [6].

The framework is introduced in the world of decision problem in which outcomes that determines the player's well-being as consequences of actions played by him with some randomness which is beyond his control. There might be some situations in which the player's well-being can also be influenced by the choices made by other players which are part of the same game. A classification can be made based on: the effect that each player's choice has on other players' well being, the type of knowledge possessed by each of them and the quantity of knowledge they each of them holds. From now on, the convention of confusing the word *decision problem* with **games** will be adapted and followed for the rest of the thesis. Now a brief introduction about the theory that explains us how the action of one player is correlated to the outcomes of other players is outlined. This is exactly what we need as the final payoffs of our players (BSs) will be strongly influenced by decisions taken by other player.

3.2 Static game of complete information

We now consider a simplest case, which is very useful to capture simple strategic situations. When one player is trying to obtain maximum profit to increase his well being, others players are trying to do the same and each player is trying to guess what other players are doing, and how to act accordingly. In this scenario, all players are engaged in a so called *strategic environment* in which each of them must have a set of strategies from which to choose depending on what others are doing. The essence of this type of games can be captured by a framework called **static games of complete information** [6]. In the static games, a set of players choose actions independently and once-and-for-all which in turn cause the realization of an outcome. Hence, a static game can be thought of as a game played in two steps:

1. Each player chooses an action *simultaneously* and *independently.*
2. Conditional on the players' chosen actions, payoffs are distributed to each player.

In game theory, by simultaneously and independently we mean that each player must take their actions without observing what actions their counterparts take and without interacting among them. The second step captures the fact that, once the player have made their choices, these choices will result in a particular outcome or a probabilistic distributions over outcomes. Steps 1) and 2) defines what we called *static.* It remains to specify the meaning of *complete information.* With this word, we want to emphasizes that all the players understand the environment they are in, the game they are playing, in every way [6]. This can be defined more formally as follows:
Games of complete information In a game of complete information all these four component are of a common knowledge among all the players:

1. all the possible actions of all the players,
2. all the possible outcomes,
3. how each combination of actions affects the outcomes,
4. the preference of every player over the outcomes.

Now that we have understood the basic ingredients of a static game of complete information, it is possible to develop a formal framework to represent it with a so called **normal-form game**. This is the most common way used to represent a game and it consists of :

1. A set of **players**.
2. A set of **actions** for each player.

3. A set of **payoff functions** for each player that give a payoff value to each combination of the players' chosen actions.

It is now time to introduce the concept of **strategy**, which can be defined as a plan of action intended to accomplish a specific goal. We can think of this plan as a some kind of list structured as: "If someone asks me question $q1$ then i will respond with answer $a1$; if I have been asked question $q2$, i will answer with $a2$ " and so on. A formal definition can be given for the case in which we consider only pure strategies as follows:

Definition 6. *A **pure strategy** for a player i is a deterministic plan of action. The set of all **pure strategies** is denoted by S_i. A **profile of pure strategies** $\{s_1, s_2, ..., s_n\}, s_i \in S_i$ for all $i = 1, 2, .., n$ describes a particular combination of pure strategies chosen by all n players.*

Here with **pure** we mean that all the actions chosen by all the players are deterministic and there is no randomness involved. From now on, our focus will be on strategies instead of actions as this will let us represent also the games in which there is a relevance to conditioning one's actions on events that may also fold over time. The definition of *normal-form game* can be rewritten to comprehend the strategies as follows:

Definition 7. *A **normal-form game** includes three components as follows:*

1. *A finite **set of players**, $\mathcal{P} = \{1, 2, ..., n\}$.*
2. *A **collection of sets of pure strategies**, $\{S_1, S_2, ..., S_n\}$.*
3. *A **set of payoff functions**, $\mathcal{V} = \{v_1, v_2, ..., v_n\}$, each assigning a payoff value to each combination of chosen strategies, which is a set of functions $v_i : S1 \times S_2 ..... \times S_n \to \mathbb{R}, \forall i \in \mathcal{P}$.*

This representation captures games in which every player chooses simultaneously strategy $s_i \in S_i$. A n-player game can be treated with the **Decision tree** [6]. When we are dealing with a 2-player game, which is the simplest case, an alternative graphical method can be used, which is called the **Matrix representation**. The first one gives us a very intuitive and graphically simple tool to depict a game and capture the essence of play. It includes two kind of nodes: **decision nodes** and **terminal nodes**, which are connected among them through **edges**. Decision nodes corresponds to choices about strategies and terminal nodes represent the final payoffs.

The second type of representation is very useful for a two players game. It can be exploited very easily to find the **Nash Equilibrium** of a game, which is a profile of strategies for which each player is choosing the best response to the strategies chosen by all other players and nobody deviates from their choice. The *Nash equilibrium* is named after his inventor, John Nash, an American mathematician. It is considered one of the most important concepts of game theory, which attempts to determine mathematically and logically the actions (or strategies) that each player should take to secure the best payoff for themselves. It will be defined later more formally for our cases of interest as its definition varies from game to game. In Fig. 3.1 and Table 3.2, an illustrative idea of the two kinds of graphical representation is given. The game presented in both cases is formed by two players who decide their strategy simultaneously. First of all, player 1 chooses his strategy from the set $S_1 = \{x_1, x_2\}$ and player 2 chooses his strategy from the set $S_2 = \{y_1, y_2\}$. The dotted line that connects the nodes at the second player's level, highlights the idea that player 2 has no idea about the strategy chosen by player 1. Thus, player 2 can be in any of the nodes at his level. Here we are implying the definition of *simultaneous* moves mentioned above.

If there is no randomness involved, all the theory presented above is sufficient to formulate games with simultaneous moves.

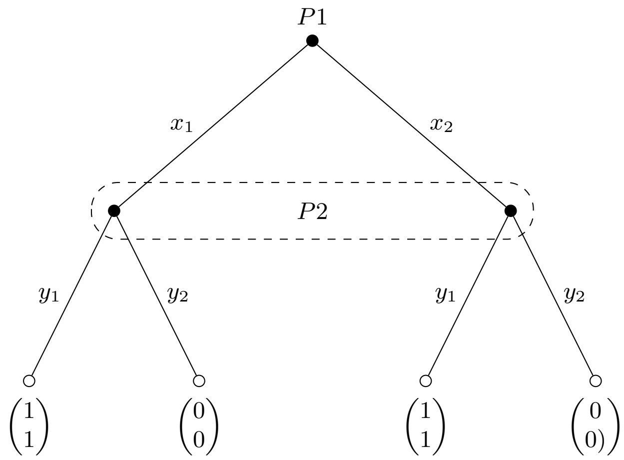

Figure 3.1: Decision tree of a simultaneously played two player game where player 1 chooses strategies from set $\{x_1, x_2\}$ and player 2 chooses from set $\{y_1, y_2\}$.

		Player 2	
		y1	y2
Player 1	x1	1,1	0,0
	x2	1,1	0,0

Figure 3.2: Matrix representation of a simultaneously played two player game where player 1 chooses strategies from set $\{x_1, x_2\}$ and player 2 chooses from set $\{y_1, y_2\}$.

3.2.1 Static game of complete information with mixed strategies

It is worth shedding some light on the framework we presented in the previous section, its drawbacks and the necessity of its extension. Suppose that a static game of complete information represents a practical scenario and is played many times (finite or infinite) over time and there is just one strategy for each player that predicts the Nash Equilibrium. Which clearly means that there is just one Nash equilibrium of the game. Once we have discovered the strategy profile which assure us the equilibrium, all the players will be forced to play the same strategy profile over and over again (finite or infinite times). Thus, in the case of repeated game the previous framework does not result very useful. It could also happen that when all the players use pure strategies, the equilibrium might not exists. To illustrate the idea, an example of the famous child's game called *rock-paper-scissors* is given below.

Example 3.2.1. The child's game named *rock-paper-scissors* is played as follows: rock beats scissors, scissors beats paper and paper beats rock. If winning gives a payoff of 1 and losing the game yields a payoff -1, and if we assume that a tie is worth 0, then the game can be described in the matrix form as shown in Fig. 3.3.

It is quite easy to write down the best response (deterministic strategies) correspondence for player 1 when he believes that player 2 will play one of his pure strategies as:

		Player 2		
		Rock	Paper	Scissor
Player 1	Rock	0,0	-1,1	1,-1
	Paper	1,-1	0,0	-1,1
	Scissor	-1,1	1,-1	0,0

Figure 3.3: Matrix representation of the *rock-paper-scissors* game.

$$s_1(s_2) = \begin{cases} P \text{ when } s_2 = \text{R} \\ S \text{ when } s_2 = \text{P} \\ R \text{ when } s_2 = \text{S}, \end{cases} \tag{3.2.1}$$

and a similar list of best responses can be written for the player 2 given his believes that player 1 will play from one of his pure strategies:

$$s_2(s_1) = \begin{cases} P \text{ when } s_1 = \text{R} \\ S \text{ when } s_1 = \text{P} \\ R \text{ when } s_1 = \text{S}. \end{cases} \tag{3.2.2}$$

By examining the two best response correspondences, it implies that starting from any pair of pure strategies there is no *Nash equilibrium*, as at least one player is not playing a best response and will for sure want to change his strategy in response.

It can be shown that in the game *rock-paper-scissors*, if both players choose stochastically among their strategies then the Nash equilibrium exists and if any player decides to use only pure strategies, it doesn't exists. The purpose of extending our framework is to avoid the inconvenience of the non existence of Nash equilibrium in some particular cases. Now, we proceed by extending the whole framework presented in the section 3.1 by including mixed strategies and then we will proceed with our game formulation in which players (BSs) will use probability distributions over their available pure strategies. At the end, we will announce the theorem which states the conditions that every n-player game must satisfy for the existence of *Nash equilibrium* [6].

Here we introduce the idea of players choosing stochastically among their available strategies, which means that every player will play mixed strategies. By doing so, as we are interested in a game which is repeated over time, our players will have more degree of freedom in choosing among their strategies. After a brief theory introduction about the mixed strategies, in which we will extend the definition of expected payoff, the definition of the *Nash equilibrium* in the case of mixed strategies will be given. The main reason of extending our framework by including **mixed strategies** is the following one: In the static game of a complete information some times may occur that the Nash equilibrium doesn't exits. This may be occur due to the fact that players are forced to choose their strategies deterministically. By letting them choose stochastically, players have much more degree of freedom and the possibility of the existence of the equilibrium increases. Notice also that the pure strategy case is a particular case of the mixed strategies in which only one strategy is chosen by each player with probability 1 and each player assigns null probability to all the other strategies [7].

We start with the basic definition of random play in the case in which all the players have finite strategies, which is exactly our case of interest, as the only strategies available by the players are : *Uplink* or *Downlink*.

Definition 8. *Let $S_i = \{s_{i,1}, s_{i,2}, ..., s_{i,m}\}$ be the player i's finite set of pure strategies. We define ΔS_i as a* ***simplex*** *of S_i, which is the set of all probability distributions over S_i. A* ***mixed strategy*** *for player i is an element $\sigma_i \in \Delta S_i$ so that $\sigma_i = \{\sigma_i(s_{i,1}), \sigma_i(s_{i,2}), ..., \sigma_i(s_{i,m})\}$ is a probability distribution over S_i, where $\sigma_i(s_i)$ is the probability that player i plays $s_i \in S_i$.*

The definition states that a mixed strategy for player i is just a probability distribution $\sigma_i \in \Delta S_i$ over his pure strategies. As now we are dealing with probability distributions, the following properties must also be satisfied :

1. $\sigma_i(s_i) \geq 0, \text{for all } s_i \in S_i$
2. $\sum_{s_i \in S_i} \sigma_i(s_i) = 1$

The first property tell us that the probability of any event must be greater than or equal to zero and the sum of the probabilities of all the possible events must add up to one. The definition 4 of **expected payoff** can be rewritten more formally in the case of mixed strategies. From now on, the convention of referring to all the mixed strategies chosen by the i-th player's opponents as $\vec{\sigma_{\setminus i}}$ will be used, which belong to the simplexes which will be referred as $\Delta S_{\setminus i}$, which allow us to write $\vec{\sigma_{\setminus i}} \in \Delta S_{\setminus i}$. We will also denote with $S_{\setminus i}$ the strategic profile of the i-th player's opponents.

Definition 9. *The* ***expected payoff*** *of player i when he chooses the mixed strategy $\sigma_i \in \Delta S_i$ and his opponents play the mixed strategies $\sigma_{-i} \in \Delta S_{-i}$ is*

$$v_1(\sigma_i, \vec{\sigma_{\setminus i}}) = \sum_{s_i \in S_i} \sigma_i(s_i) v_i(s_i, \vec{\sigma_{\setminus i}}) = \sum_{s_i \in S_i} \left(\sum_{s_{\setminus i} \in S_{\setminus i}} \sigma_i(s_i) \vec{\sigma_{\setminus i}}(\vec{s_{\setminus i}}) v_i(s_i, \vec{S_{\setminus i}}) \right) \tag{3.2.3}$$

Definition 9 can be stated also for players using pure strategies, but it will be omitted, as we are strictly interested in players using only mixed strategies. Now we can state the definition of *Nash equilibrium* in the case of players using only mixed strategies as follows:

Definition 10. *The mixed strategy profile $\vec{\sigma}^* = (\sigma_1^*, \sigma_2^*, ..., \sigma_n^*)$ is a* Nash equilibrium *if for each player i, σ_i^* is a best response to $\vec{\sigma_{\setminus i}}^*$. That is, $\forall i \in \mathcal{P}$,*

$$v_i(\sigma_i^*, \vec{\sigma_{\setminus i}}^*) \geq v_i(\sigma_i, \vec{\sigma_{\setminus i}}^*) \quad \forall\, \sigma_i \in \Delta S_i$$

Definition 10 states that each player will be choosing a mixed strategy $\sigma_i^* \in \Delta S_i$ which is the best choice he can make when his opponents are choosing some mixed strategy profiles $\vec{\sigma_{\setminus i}}^* \in \Delta S_{\setminus i}$. In short words, each player i should choose the mixed strategy σ_i^* which yields him the highest expected payoff. Now we are ready to state the theorem which is the cornerstone of the game theory and generalizes the condition under which it is possible to find the Nash equilibrium of any n-player game [6]- [7].

Theorem 1. *Any n-player normal-form game with finite strategy sets S_i for all players has a Nash equilibrium in mixed strategies.*

The proof of this theorem is omitted as this is not the primary focus of this thesis. The theorem states sufficient conditions for the existence of the Nash equilibrium of any n-player game when players use mixed strategies. Now we have come far enough to precisely understand what the Nash equilibrium means in the case of mixed strategies. After gaining all the theoretical basis necessary to formulate the game of our interest, we now proceed by formulating our problem as a static game of complete information with mixed strategies.

3.2.2 Game formulation in the case of static game of complete information using mixed strategies

Suppose we are in a strategic environment and let the BSs be the players who have to choose between the *Uplink* or *Downlink* strategy in order to maximize their payoff (throughput). As we are formulating a game of complete information, we assume that every player knows exactly all the channels situated among their UEs and among all the other BSs and their UEs. Suppose also that every base station transmits signal with maximum available power P and all the user equipments transmits signal with maximum available power Q. As everything is available to each player's knowledge, assume also that every player can compute his SINR and also the SINR values of his opponents, i.e. every player possesses his payoff value and also the payoffs of other players. In this scenario, which is quite unrealistic or hardly encountered, every player can determine which strategy to play by mixing (randomizing) between their strategies in order to achieve the Nash equilibrium of Definition 10. The reason why we wish to determine the Nash equilibrium of this game is: Players may predict the best for themselves, but at the end they might end up by getting a very little portion of it. To be more precise, as we are in a strategic environment in which payoff of each player is correlated by the strategies chosen by other players, the perfect strategy's choice may not be trivial. For example, one player can simply select the strategy that gives him the highest payoff. But as his final payoff is strongly influenced by decisions made by other players, he may end up by getting a very small amount of payoff that he predicted before choosing his strategy. The Nash equilibrium makes prediction of the best strategic profile for which all the players are satisfied and nobody wants to deviate from his strategy given the strategies chosen by other players. We strictly impose the assumption that each player is interested in achieving the equilibrium. Otherwise, if there is a possibility of having players not interested in achieving the equilibrium then players who are interested in, may pay the price. Another important assumption required by theorem 1 to allow the existence of Nash equilibrium is that the strategy sets S_i for each player i is finite. In our case, each player has only two strategies available: *Uplink* and *Downlink*, which means the previous assumption is always satisfied. We also assume that every player is willing to mix between his pure strategies, otherwise there may be no existence of Nash equilibrium as in the case of *rock-paper-scissors*. Finally, under the assumptions :

Assumptions

1. Each player transmits signal with power P.

2. Each UE transmit signal with power Q.

3. Every player knows the channels situated among their UEs and among all the other players and their UEs.

4. Every player is able to compute his SINR and the SINRs of other players, so the payoffs.

5. Everyone involved in the game is rational.

This precisely state that every player knows everything about the game environment and is able compute his and other players' payoff. We can give the normal-form representation of our game in the case of n-player game as follows:

Normal-form representation of the dynamic TDD assignment n-player game

1. Set of Players $\mathcal{P} = \{1, ..., n\}$ (the Base stations) and each player i has his own UEs set $\mathcal{U}_i = \{1_i, ...j_i\}$. Moreover, the total UEs belong to set $\mathcal{M} = \{1, ...m\}$

2. A collection set of pure strategies $\{S_1, ..., S_n\}$, where each $S_i = \{Uplink, Downlink\}$

3. A set of payoff functions, $\{v_1, ..., v_n\}$, each assigning a payoff value to each combination of chosen strategies, that is $v_i : S_1 \times S_2 \times ... \times S_n \rightarrow$

$$
\begin{aligned}
w_i \Bigg(X_i \Bigg(\sum_{j \in \mathcal{U}_i} \log_2 \Bigg(1 + \frac{H_{i,j}\ P}{\sigma_d^2 + \sum\limits_{\substack{k \neq n \\ k \in \mathcal{N}_{\setminus i}}} H_{k,j}\ P + \sum\limits_{\substack{m \in \mathcal{N} \\ m \neq i}} \sum\limits_{x \in \mathcal{U}_m} U_{x,j} Q} \Bigg) \Bigg) \\
+ (1 - X_i) \Bigg(\sum_{j \in \mathcal{U}_i} \log_2 \Bigg(1 + \frac{H_{i,j}\ Q}{\sigma_u^2 + \sum\limits_{\substack{k \neq i \\ k \in \mathcal{N}_{\setminus i}}} B_{k,i} P + \sum\limits_{m \in \mathcal{N}} \sum\limits_{\substack{y \in \mathcal{U}_m \\ y \neq j}} H_{i,y} Q} \Bigg) \Bigg) \Bigg)
\end{aligned}
\tag{3.2.4}
$$

As we are dealing with mixed strategies, we also include the simplexes ΔS_i for each strategy set S_i:

4. A collection set of simplexes $\{\Delta S_1, ..., \Delta S_n\}$, where $\sigma \in \Delta S_i$ is a mixed strategy, $\sigma_i = \{\sigma(Uplink), \sigma(Downlink)\}$ and $\sigma_i(Uplink) + \sigma_i(Downlink) = 1$.

Now we have a normal-form representation of our game static game of complete information with mixed strategies. However, the solution of this game in the case of n-players is not trivial. As the theorem 1 states: the Nash equilibrium exists if the strategy set of each player is finite, but it doesn't tell us how to find it. We can decrease the difficulty by limiting ourself to case in which we have only 2 players and find its Nash equilibrium by exploiting the theory presented before.

3.2.3 Solution of the 2-player dynamic TDD assignment game

We now consider the simplest case in which there are only 2 players with strategy sets $S_i = \{Uplink, Downlink\}, i = 1, 2$. First of all, we proceed by giving the normal-form representation of this game under the assumption stated above. Then we will depict our game by using both decision tree and Matrix representation methods. Finally, a method to find the Nash equilibrium for our game will be proposed to predict the optimal behavior of both players.

Normal-Form representation of the dynamic TDD assignment 2-player game

1. Set of Players $\mathcal{P} = \{1, 2\}$ (the Base stations),
2. A collection set of pure strategies $\{S_1, S_2\}$, where $S_i = \{Uplink, Downlink\}$ for $i = 1, 2$.
3. A set of payoff functions, $\{v_1, v_2\}$, that is $v_i : S_1 \times S_2 \to$ (3.2.4).
4. A collection set of simplexes $\{\Delta S_1, \Delta S_2\}$, where $\sigma_1 = \{q, 1-q\}$, for $0 \leq q \leq 1$ is an element of ΔS_1 and $\sigma_2 = \{p, 1-p\}$, for $0 \leq p \leq 1$ is an element of ΔS_2.

As there are many possible values of p and q, there are many σ_1 and σ_2 elements of the simplexes. As it can be noticed, a normal-form representation of the 2-player game can be easily depicted as the cardinalities of all the sets are very small. Below, we present the 2-player game by using the decision tree method first and then by exploiting the matrix representation method. The convention of specifying player 1's payoff as $P^1_{a,b}$ and players 2's payoff as $P^2_{a,b}$, where $a \in S_1$ and $b \in S_2$ will be used in the decision tree. The adoption of this convention is justified by maintaining the structure of decision tree compacted and the use of subscripts a and b allow us to emphasize the strategies chosen by both players.

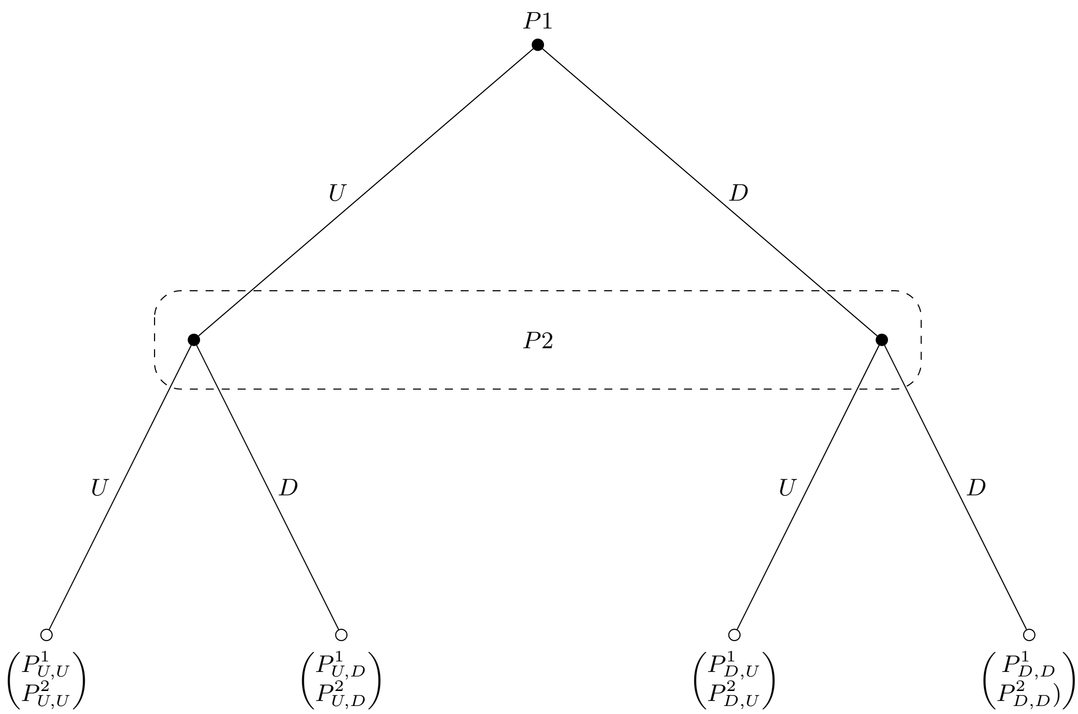

Figure 3.4: Decision tree of a 2-player game. The dotted line represents the idea of both players making moves simultaneously and without observing moves made by other player.

		P2
		Uplink
P1	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H_{1,y}Q}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H_{2,y}Q}\right)$
	Downlink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_2} U_{x,j}Q}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\,Q}{\sigma_d^2+B_{1,2}P+\sum_{x\in\mathcal{U}_{2\setminus j}} H_{2,x}Q}\right)$

		P2
		Downlink
P1	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\,Q}{\sigma_u^2+B_{2,1}P+\sum_{y\in\mathcal{U}_{1\setminus j}} H_{1,y}Q}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_1} U_{x,j}Q}\right)$
	Downlink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\,P}{\sigma_d^2+H_{2,j}\,P}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\,P}{\sigma_d^2+H_{1,j}P}\right)$

Figure 3.5: Matrix form representation of the 2 player TDD assignment game. Because of the space related issues, the matrix is rived into two different pieces.

After getting graphically an idea of our 2-player game, its now time to move towards the equilibrium condition. As stated by Definition 10, a mixed strategy profile is a Nash equilibrium if it yields each player the highest expected payoff given the strategies played by other players. The decision tree graphical method captures the essence of play but the matrix representation can be easily used to find the equilibrium. Let's recall that searching for the Nash equilibrium, corresponds to looking for the mixed strategy profile of $\sigma_1^* = \{q^*, 1-q^*\}$ and $\sigma_2^* = \{p^*, 1-p^*\}$, which yields each player the highest expected payoff. As we are playing a 2-player game of complete information, every player knows his and his opponents payoffs. Suppose that player 1 chooses with probability q the strategy $s_1 = Uplink$, thus with probability $1-q$ the strategy $s_1 = Downlink$. On the other hand, suppose that player chooses with probability p the strategy $s_2 = Uplink$, hence with probability $1-p$ the strategy $s_2 = Downlink$. We suppose that the values of p and q are of common knowledge. In order to proceed, we start first by writing the expected payoffs from the point of view of each player.

From the point of view of player 1:

$$\begin{aligned} v_1(U, s_2) = p\; w_n \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H_{1,y}Q}\right) \\ + (1-p) w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\,Q}{\sigma_u^2+B_{2,1}P+\sum_{y\in\mathcal{U}_{1\setminus j}} H_{1,y}Q}\right) \end{aligned} \quad (3.2.5)$$

$$v_1(D,s_2) = p\,w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\;P}{\sigma_d^2+\sum\limits_{x\in\mathcal{U}_2} U_{x,j}Q}\right) + (1-p)w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}\;P}{\sigma_d^2+H_{2,j}\;P}\right) \tag{3.2.6}$$

The first equation represents player 1's expected payoff when he chooses strategy $s_1 = \mathit{Uplink}$ and in the second equation we have his expected payoff when he decides to play $s_1 = \mathit{Downlink}$. It must be noticed that in both cases the expected payoff of player 1 is obtained as the weighted sum of the mixed strategy profile of player 2. We know do the same from the point of view of player 2 as follows.

From the point of view of player 2:

$$\begin{aligned} v_2(s_1,U) = q\,w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\;Q}{\sigma_u^2+\sum\limits_{m=1,2}\sum\limits_{\substack{y\in\mathcal{U}_m\\ y\neq j}} H_{2,y}Q}\right) \\ + (1-q)w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\;Q}{\sigma_d^2+B_{1,2}P+\sum\limits_{x\in\mathcal{U}_{2\setminus j}} H_{2,x}Q}\right) \end{aligned} \tag{3.2.7}$$

$$\begin{aligned} v_2(s_1,D) = q\,w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\;P}{\sigma_d^2+\sum\limits_{x\in\mathcal{U}_1} U_{x,j}Q}\right) \\ + (1-q)\;w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\;P}{\sigma_d^2+H_{1,j}P}\right) \end{aligned} \tag{3.2.8}$$

The first equation represents player 2's expected payoff when he chooses the strategy $s_2 = \mathit{Uplink}$ and the second equation shows his expected payoff when he decides to play $s_2 = \mathit{Downlink}$. It is worth recalling, as we are playing a game of complete information, (3.2.5)-(3.2.8) are known by both players. If we observe (3.2.5)-(3.2.6), the task of player 2 is to choose the value of p such that the equality between these equations hold. Which means, he should choose his optimal mixed strategy in such a way that his opponent is indifferent of his choice. Otherwise player 1 could play the mixed strategy which yield him a bigger payoff with higher probability and player 2 may loose the game. The same task has to be computed by player 1 by looking at (3.2.7) - (3.2.8). He must choose the value of q in order to render both expected payoffs of player 2 equal. To be more precise, player 2 should select the value p^* and player 1 should choose the value q^* which makes their opponent indifferent of his choice. The optimal values p and q can be found by satisfying the equalities:

$$v_1(U,s2) = v_1(D,s_2), \tag{3.2.9}$$

$$v_2(s_1,U) = v_2(s_1,D), \tag{3.2.10}$$

and then by solving (3.2.9) for p and (3.2.10) for q, we can obtain the values p^* and q^*. As stated by Definition 10, the pair of values (p^*,q^*) are the one which assures us the *Nash* equilibrium.

3.2.4 Solution of the 3-player dynamic TDD assignment game

In the previous subsection we learned how to solve the dynamic TDD assignment game in the case of two players. Almost all the examples available in the literature (see [5], [6] and [7]) take into account only the static game of complete information with two players.

The reason could be the following one: by exploiting the matrix form representation of any 2-player game we can easily find the Nash equilibrium in the case of pure strategies, whose explanation has been omitted as we are strictly interested in the mixed strategies case. While, by using the method explained in the previous, the Nash equilibrium can be easily found also in the case of mixed strategies. However, our game represents a practical scenario which can normally include more than 2 players. By following this belief, we would like to build a method which allows us to play the dynamic TDD assignment game with three players. So, now we consider the case in which we have 3 players (BSs) and each of them has their own UEs' set. Each player has his own strategy's set $S_i = \{Uplink, Downlink\}$, for $i = 1, 2, 3$. We assume that all the players strictly play mixed strategies and no one is willing to play any pure strategy. First of all, we proceed by giving the normal-form representation of the game under the assumptions (5). Then we will depict our game by using the decision tree method. It should be clear that as we are considering a 3 players game, it won't be possible to use the matrix representation method to depict the game. Finally, a concept of finding the Nash equilibrium of the 3-player dynamic TDD assignment game will be proposed.

Normal-Form representation of the dynamic TDD assignment 3-player game

1. Set of Players $\mathcal{P} = \{1, 2, 3\}$ (the Base stations).
2. A collection set of pure strategies $\{S_1, S_2, S_3\}$, where $S_i = \{Uplink, Downlink\}$ for $i = 1, 2, 3$.
3. A set of payoff functions, $\{v_1, v_2, v_3\}$,as from $\rightarrow$ (3.2.4).
4. A collection set of simplexes $\{\Delta S_1, \Delta S_2, \Delta S_3\}$, where $\sigma_1 = \{q, 1-q\}$, for $0 \leq q \leq 1$ is an element of ΔS_1, $\sigma_2 = \{p, 1-p\}$, for $0 \leq p \leq 1$ is an element of ΔS_2 and $\sigma_3 = \{r, 1-r\}$, for $0 \leq r \leq 1$ is an element of ΔS_3.

As there are many possible values of p,q and r, there are many σ_1, σ_2 and σ_3 elements of the simplexes $\Delta S_1, \Delta S_2$ and ΔS_3. The normal-form representation of the 3-player game can be depicted with the decision tree method. For the sake of simplicity and compactness, we will use again the notation of Section 3.2.3 with little extensions. To be precise, we will use the notation $P^i_{a,b,c}$ for $i = 1, 2, 3$ to specify player i's payoff, which is dependent on the strategy chosen by player i and by all the other player. The subscripts a, b, c denote the strategies chosen by all the three player, i.e. $a \in S_1$, $b \in S_2$ and $c \in S_3$. We will also use the abbreviation of the strategy Uplink as U and the strategy Downlink as D in the payoffs. This notion will be used used for being able represent the 3-player game's decision tree in a compact way. We will also use the notation U to denote the strategy Uplink and D to denote the strategy Downlink on the edges of the decision tree. By using this simplified notation the decision tree for the 3-player game is presented in Fig. 3.6.

As it can be clearly noticed by comparing the 3-players' decision tree in the Fig. 3.6 with the 2-players' in Fig. 3.4, the graphical complexity is increased considerably. Not only, also the solution of the 3-players' game is quite more difficult than the solution of the 2-players' game. The reason is because each player's payoff is now dependent on the strategies chosen by other two players and as the consequence, we will have to deal with one more variable while searching for the Nash equilibrium. The expected payoff of each player will strictly depend on the product of probabilities with which they choose their strategy and we may end up solving non-linear equations. After all these observations, we now proceed by looking at the solution of the 3-player's game. The idea of making each player indifferent of his choice will be followed again and we will look for its Nash equilibrium. Let's recall that searching for the equilibrium corresponds to looking for the

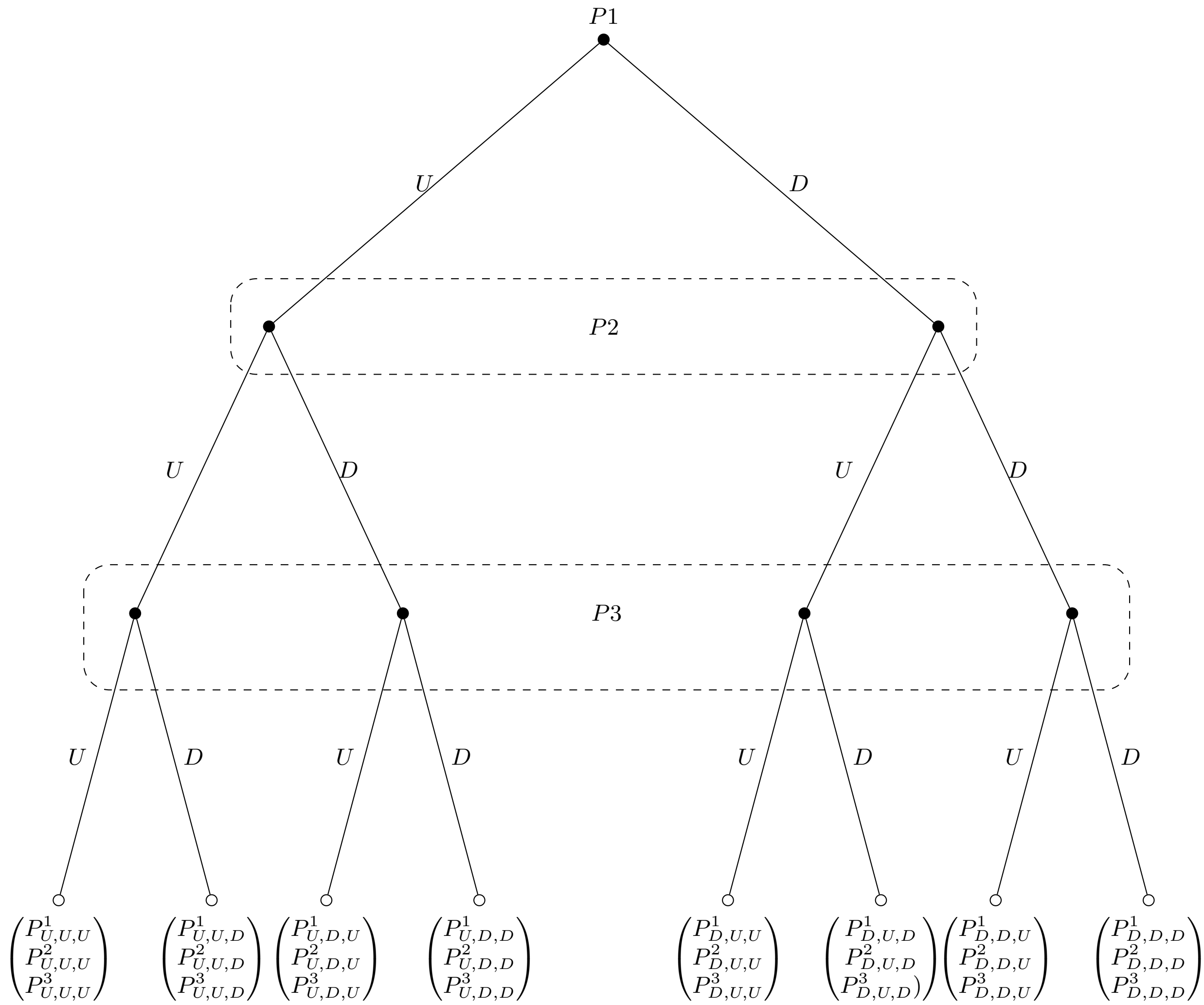

Figure 3.6: Decision tree of the 3-player dynamic TDD assignment game. The dotted line represents the idea of all the 3 players making moves simultaneously and without observing moves made by other player. The abbreviation of D to D and U to U is used because of space reasons.

mixed strategy profile $\sigma_1 = \{q^*, 1-q^*\}$,$\sigma_2 = \{p^*, 1-p^*\}$ and $\sigma_3 = \{r^*, 1-r^*\}$, where $\sigma_1 \in \Delta S_1$, $\sigma_2 = \Delta S_2$ and $\sigma_3 \in \Delta S_3$, which yields each player the highest expected payoff according to the payoff function (3.2.4). Moreover, we are playing a static game of complete information with all the payoffs and channel gains of common knowledge. As we did in the 2-player's game, we will look at the each player i's payoff $v(i, \vec{i}_{\setminus i})$ from i's point of view and will try to search for the optimal values q^*, p^*, r^* which will render their choice of mixing among their strategies indifferent.

From the point of view of player 1:

$$
\begin{aligned}
v_1(U,s_2,s_3) =& r\Bigg(p\, w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}\ Q}{\sigma_u^2+\sum\limits_{\substack{m=1,2\\,3}}\sum\limits_{\substack{y\in\mathcal{U}_m\\y\neq j}} H_{1,y}Q}\Bigg)\\
&+(1-p)w_1\sum_{j\in\mathcal{U}_1}\log_2\Bigg(1+\frac{H_{1,j}\ Q}{\sigma_u^2+\sum\limits_{k=2} B_{k,1}P+\sum\limits_{m=1,3}\sum\limits_{\substack{y\in\mathcal{U}_m\\m\neq j}} H_{1,y}Q}\Bigg)\Bigg)\\
&+(1-r)\Bigg(p\, w_1\sum_{j\in\mathcal{U}_1}\log_2\Bigg(1+\frac{H_{1,j}\ Q}{\sigma_u^2+\sum\limits_{k=3} B_{k,1}P+\sum\limits_{m=1,2}\sum\limits_{\substack{y\in\mathcal{U}_m\\y\neq j}} H_{1,y}Q}\Bigg)\\
&+(1-p)w_1\sum_{j\in\mathcal{U}_1}\log_2\Bigg(1+\frac{H_{1,j}\ Q}{\sigma_u^2+\sum\limits_{k=2,3} B_{k,1}P+\sum\limits_{\substack{y\in\mathcal{U}_1\\y\neq j}} H_{1,y}Q}\Bigg)\Bigg),
\end{aligned}
\tag{3.2.11}
$$

$$
\begin{aligned}
v_1(D,s_2,s_3) =& r\Bigg(p\, w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}\ P}{\sigma_d^2+\sum\limits_{m=2,3}\sum\limits_{y\in\mathcal{U}_m} U_{y,j}Q}\Bigg)\\
&+(1-p)w_1\sum_{j\in\mathcal{U}_1}\log_2\Bigg(1+\frac{H_{1,j}\ P}{\sigma_d^2+\sum\limits_{k=2} H_{k,j}\ P+\sum\limits_{y\in\mathcal{U}_3} U_{y,j}Q}\Bigg)\Bigg)\\
&(1-r)\Bigg(p\, w_1\sum_{j\in\mathcal{U}_1}\log_2\Bigg(1+\frac{H_{1,j}\ P}{\sigma_d^2+\sum\limits_{k=3} H_{k,j}\ P+\sum\limits_{y\in\mathcal{U}_2} U_{y,j}Q}\Bigg)\\
&+(1-p)w_1\sum_{j\in\mathcal{U}_1}\log_2\Bigg(1+\frac{H_{1,j}\ P}{\sigma_d^2+\sum\limits_{k=2,3} H_{k,j}P}\Bigg)\Bigg).
\end{aligned}
\tag{3.2.12}
$$

We will write the equations (3.2.5)-(3.2.6) in a more compact way, which will allow us to manipulate them more easily. Note that, even if the structure of the equations seems very difficult, in our static game of complete information all the quantities within the brackets are of common knowledge. By specifying the payoffs of player one as a function of strategies chosen by all the three players, we can write the payoffs of player one as:

$$
\begin{aligned}
v_1(U,s_2,s_3) =& r\Big(p\ v_1(U,U,U)+(1-p)\ v_1(U,D,U)\Big)\\
&+(1-r)\Big(p\ v_1(U,U,D)+(1-p)\ v_1(U,D,D)\Big),
\end{aligned}
\tag{3.2.13}
$$

$$
\begin{aligned}
v_1(D,s_2,s_3) =& r\Big(p\ v_1(D,U,U)+(1-p)\ v_1(D,D,U)\Big)\\
&+(1-r)\Big(p\ v_1(D,U,D)+(1-p)\ v_1(D,D,D)\Big).
\end{aligned}
\tag{3.2.14}
$$

From the point of view of player 2:

$$\begin{aligned} v_2(s_1,U,s_3) =& r\Bigg(q\ w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\ Q}{\sigma_u^2+\sum\limits_{\substack{m=1,2\\,3}}\sum\limits_{\substack{y\in\mathcal{U}_m\\y\neq j}} H_{2,y}Q}\right) \\ &+(1-q)w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}\ Q}{\sigma_u^2+\sum\limits_{k=1}B_{k,2}P+\sum\limits_{m=2,3}\sum\limits_{\substack{y\in\mathcal{U}_m\\m\neq j}} H_{2,y}Q}\right)\Bigg) \\ &+(1-r)\Bigg(q\ w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}\ Q}{\sigma_u^2+\sum\limits_{k=3}B_{k,2}P+\sum\limits_{m=1,2}\sum\limits_{\substack{y\in\mathcal{U}_m\\y\neq j}} H_{2,y}Q}\right) \\ &+(1-q)w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}\ Q}{\sigma_u^2+\sum\limits_{k=1,3}B_{k,2}P+\sum\limits_{\substack{y\in\mathcal{U}_2\\y\neq j}} H_{2,y}Q}\right)\Bigg), \end{aligned} \tag{3.2.15}$$

$$\begin{aligned} v_2(s_1,D,s_3) =& r\Bigg(q\ w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}\ P}{\sigma_d^2+\sum\limits_{m=1,3}\sum\limits_{y\in\mathcal{U}_m} U_{y,j}Q}\right) \\ &+(1-q)w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}\ P}{\sigma_d^2+\sum\limits_{k=1}H_{k,j}\ P+\sum\limits_{y\in\mathcal{U}_3} U_{y,j}Q}\right)\Bigg) \\ &(1-r)\Bigg(q\ w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}\ P}{\sigma_d^2+\sum\limits_{k=3}H_{k,j}\ P+\sum\limits_{y\in\mathcal{U}_1} U_{y,j}Q}\right) \\ &+(1-q)w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}\ P}{\sigma_d^2+\sum\limits_{k=1,3}H_{j,k}P}\right)\Bigg), \end{aligned} \tag{3.2.16}$$

and as we did for player 1, we can write these equations in a more compact way as:

$$\begin{aligned} v_2(s_1,U,s_3) =& r\Big(q\ v_2(U,U,U)+(1-q)\ v_2(D,U,U)\Big) \\ &+(1-r)\Big(q\ v_2(U,U,D)+(1-q)\ v_2(D,U,D)\Big), \end{aligned} \tag{3.2.17}$$

$$\begin{aligned} v_2(s_1,D,s_3) =& r\Big(q\ v_2(U,D,U)+(1-q)\ v_2(D,D,U)\Big) \\ &+(1-r)\Big(q\ v_2(U,D,D)+(1-q)\ v_2(D,D,D)\Big). \end{aligned} \tag{3.2.18}$$

From the point of view of player 3:

$$
\begin{aligned}
v_3(s_1,s_2,U) =& p\Bigg(q\ w_3 \sum_{j\in\mathcal{U}_3} \log_2\left(1+\frac{H_{3,j}\ Q}{\sigma_u^2+\sum\limits_{\substack{m=1,2\\,3}}\sum\limits_{\substack{y\in\mathcal{U}_m\\ y\neq j}} H_{2,y}Q}\right)\\
&+(1-q)w_3\sum_{j\in\mathcal{U}_3}\log_2\left(1+\frac{H_{3,j}\ Q}{\sigma_u^2+\sum\limits_{k=1}B_{k,2}P+\sum\limits_{m=2,3}\sum\limits_{\substack{y\in\mathcal{U}_m\\ m\neq j}}H_{2,y}Q}\right)\Bigg)\\
&+(1-p)\Bigg(q\ w_3\sum_{j\in\mathcal{U}_3}\log_2\left(1+\frac{H_{3,j}\ Q}{\sigma_u^2+\sum\limits_{k=2}B_{k,2}P+\sum\limits_{m=1,3}\sum\limits_{\substack{y\in\mathcal{U}_m\\ y\neq j}}H_{2,y}Q}\right)\\
&+(1-q)w_3\sum_{j\in\mathcal{U}_3}\log_2\left(1+\frac{H_{3,j}\ Q}{\sigma_u^2+\sum\limits_{k=1,2}B_{k,2}P+\sum\limits_{\substack{y\in\mathcal{U}_3\\ y\neq j}}H_{2,y}Q}\right)\Bigg),
\end{aligned}
\tag{3.2.19}
$$

$$
\begin{aligned}
v_3(s_1,s_2,D) =& p\Bigg(q\ w_3 \sum_{j\in\mathcal{U}_3} \log_2\left(1+\frac{H_{3,j}\ P}{\sigma_d^2+\sum\limits_{m=1,2}\sum\limits_{y\in\mathcal{U}_m} U_{y,j}Q}\right)\\
&+(1-q)w_3\sum_{j\in\mathcal{U}_3}\log_2\left(1+\frac{H_{3,j}\ P}{\sigma_d^2+\sum\limits_{k=1}H_{k,j}\ P+\sum\limits_{y\in\mathcal{U}_2}U_{y,j}Q}\right)\Bigg)\\
&(1-p)\Bigg(q\ w_3\sum_{j\in\mathcal{U}_3}\log_2\left(1+\frac{H_{3,j}\ P}{\sigma_d^2+\sum\limits_{k=2}H_{k,j}\ P+\sum\limits_{y\in\mathcal{U}_1}U_{y,j}Q}\right)\\
&+(1-q)w_3\sum_{j\in\mathcal{U}_3}\log_2\left(1+\frac{H_{3,j}\ P}{\sigma_d^2+\sum\limits_{k=1,2}H_{k,j}P}\right)\Bigg),
\end{aligned}
\tag{3.2.20}
$$

and again as we did for player 1 and player 2, we can write the strategy dependent payoff of player 3 as:

$$
\begin{aligned}
v_3(s_1,s_2,U) =& p\Big(q\ v_3(U,U,U)+(1-q)\ v_3(D,U,U)\Big)\\
&+(1-p)\Big(q\ v_3(U,D,U)+(1-q)\ v_3(D,D,U)\Big),
\end{aligned}
\tag{3.2.21}
$$

$$
\begin{aligned}
v_3(s_1,s_2,D) =& p\Big(q\ v_3(U,U,D)+(1-q)\ v_3(D,U,D)\Big)\\
&+(1-p)\Big(q\ v_3(U,D,D)+(1-q)\ v_3(D,D,D)\Big).
\end{aligned}
\tag{3.2.22}
$$

Let's recall that we are looking for the Nash equilibrium such that each player is indifferent between their strategies. Making all the players indifferent yield the following conditions:

$$v_1(U, s_2, s_3) = v_1(D, s2, s3), \tag{3.2.23a}$$
$$v_2(s_1, U, s_3) = v_2(s_1, D, s_3), \tag{3.2.23b}$$
$$v_3(s_1, s_2, U) = v_3(s_1, s_2, D). \tag{3.2.23c}$$

By satisfying all these three condition, the following equality holds:

$$\begin{aligned}
&r\,p\Bigg(v_1(U,U,U) - v_1(D,U,U) - v_1(U,D,U) + v_1(D,D,U)\\
&- v_1(U,U,D) + v_1(D,U,D) + v_1(U,D,D) - v_1(D,D,D)\Bigg)+\\
&p\Bigg(v_1(U,U,D) - v_1(D,U,D) - v_1(U,D,D) + v_1(D,D,D)\Bigg)-\\
&r\Bigg(v_1(U,D,D) - v_1(D,D,D)\Bigg) = v_1(D,D,U) - v_1(U,D,U)
\end{aligned} \tag{3.2.24}$$

$$\begin{aligned}
&r\,q\Bigg(v_2(U,U,U) - v_2(U,D,U) - v_2(D,U,U) + v_2(D,D,U)\\
&- v_2(U,U,D) + v_2(U,D,D) + v_2(D,U,D) - v_2(D,D,D)\Bigg)+\\
&q\Bigg(v_2(U,U,D) - v_2(U,D,D) - v_2(D,U,D) + v_2(D,D,D)\Bigg)-\\
&r\Bigg(v_2(D,U,D) - v_2(D,D,D)\Bigg) = v_2(D,D,D) - v_2(D,U,D)
\end{aligned} \tag{3.2.25}$$

$$\begin{aligned}
&p\,q\Bigg(v_3(U,U,U) - v_3(U,U,D) - v_3(D,U,U) + v_3(D,U,D)\\
&- v_3(U,D,U) + v_3(U,D,D) + v_3(D,D,U) - v_3(D,D,D)\Bigg)+\\
&q\Bigg(v_3(U,D,U) - v_3(U,D,D) - v_3(D,D,U) + v_3(D,D,D)\Bigg)-\\
&p\Bigg(v_3(D,D,U) - v_3(D,D,D)\Bigg) = v_3(D,D,D) - v_3(D,D,U)
\end{aligned} \tag{3.2.26}$$

As it can be noticed, we have ended up with a system with three equations and three unknowns, q, p and r. All the quantities that appear within the parenthesis are of common knowledge, which means everyone posses its knowledge. Only the unknown variable p, q, r

are need to be found and once we have solved the system, we would obtain q^*, p^*, r^* which specifies the probabilities with which each player should choose the strategy *Uplink*. On the other hand, the strategy *Downlink* is chosen by player 1 with probability $1 - q^*$, by player 2 with with probability $1 - p^*$ and by player 3 with probability $1 - r^*$, respectively. The strategy profile (q^*, p^*, r^*) is the Nash equilibrium of the 3-player dynamic TDD assignment game and yields each player the highest expected payoff given the strategies chosen by the other players.

3.3 Static game of incomplete information

In the previous section we analyzed the 2-player and the 3-player dynamic TDD assignment game in which our BSs played the role of players. By exploiting the tools offered by game theory we proceeded directly in the search of *Nash equilibrium* under one particular hypothesis: every one knows everything about every one. This means that the game which is being played is common knowledge and players are aware of all the payoffs of their opponents. This scenario is quite unrealistic or maybe hardly encountered in reality. To formulate the game, we supposed that every player knows all the channel gains among all the players and all the UEs and everyone is able to compute the SINR of their opponents, which is not so realistic. Maybe it is more convincing that every player has a reasonably good idea about their opponents' payoffs, without knowing them precisely. In this situation the theoretical toolbox developed in Section 3.1 is not adequate to address such situations. When players have an idea of the characteristics of other players, this situation is very similar to the case of *simultaneous move game* in which players do not know what actions their opponents are taking, but they know what their action sets are. In the mid-1960s John Harsanyi found the similarity between beliefs over strategies, which is the probability distribution over players' opponents' strategies and beliefs over their characteristics. Harsanyi introduced a new operational way to capture the idea that beliefs (probability distribution) over the characteristics of other players can be embedded into the framework that we presented before. From now, we will use the convention to refer players having different characteristics as different *types*. Players having different types can be classified based on the different kind of knowledge they posses, different type of preferences over outcomes they have or how well each player knows his opponents' payoffs. However, they all lead to the case in which players can have multiple type dependent payoffs. This kind of games are called *games of incomplete information*, also known as *Bayesian games* [5]- [6]. As we did before, we will proceed by giving a brief introduction to the theory related to the games of incomplete information to find the Nash equilibrium in the case of mixed strategies. To develop the concept, let's assume that the set of players $\mathcal{P} = \{1, 2...., n\}$ and the collection of sets of strategies $\{S_1, ..., S_n\}$ are still common knowledge. The component which is being missing is the exact knowledge of the type dependent payoff functions or the preference that each player has over the outcomes. To capture the idea that players' characteristics maybe unknown to other players, we introduce the idea of having *uncertainity over the preferences* of players. The main requisite to solve these kind of games is that every player has well-defined beliefs over the types of his opponents. Harsanyi proposed the following framework: Imagine that before the game starts, Nature chooses players' types from their set of types, therefore Nature decides how to associate players' payoffs to the their types. Once every player has learned his type, which is his private information, everyone can form a well defined beliefs about their opponents by exploiting some kind of information which is common knowledge among all players. It may be important to emphasize that two different types of same player may not differ in his preferences, but they may differ in the knowledge they have about the types or the preferences of other players or about the characteristics of the game. The choice made by

Nature can be commented also in a different manner, where Nature is choosing a game from a large set of games, with same players and strategies but with different payoff functions. If Nature is choosing randomly among many possible games, then there must be a well defined joint probability distribution over these different games which is exploited by Nature to choose the player' types. As we will see, this is the common knowledge information cited in Harsanyi's framework which will allow every player to form well defined beliefs about their opponents.

Before moving toward the analysis of the game of incomplete information, an important point that must be addressed is about the behavior of players depending on their strategies. In order to impose their optimal behavior, we will let them choose the optimal strategy to maximize their type dependent payoffs given their beliefs over their opponents' strategies. This is the same approach we followed in the static game of complete information of the previous section where we didn't have to deal with type dependent payoffs. In order to do so, we assume that players know their own preferences, which in turn will allow us to analyze a player's best response given his assumptions about the behavior of his opponents. A last important issue that is still needed to answer is, even if the players know their own preferences and they do not know the preferences (or types) of other players, what must each player know in order to allow them to have a well-defined best response, to perform the *Nash equilibrium* analysis. This may not be a trivial question to answer and as Harsanyi realized, players must form *correct-beliefs* about the preferences and types of their opponents. This will allow players to form rational conjecture about the way in which their opponents will play the game. Due to this reason, we assume that even if players do not know the actual preferences of their opponents, they all know how Nature chooses these types. Which means, each player knows the joint probability distribution over players' types and this is included in the common knowledge possessed by each player of the game. This assumption is called the *common prior assumption* and it means that all players agree on the way in which Nature chooses players' types described by some joint probability distribution. To define the framework of incomplete information, the following steps will be followed: First of all, we will model the game of incomplete under the assumption that players have uncertainty about the preferences of other players, thus their types. Second, we will assume that players share same beliefs about this uncertainty, which will allow us to formulate the game of our interest. Finally, its normal-form representation will be given. After that, we will use all these theoretic tools to formulate the dynamic TDD assignment game of incomplete information and we will try to perform the equilibrium analysis.

3.3.1 Strategic representation of Bayesian Games

Now we extend the framework presented in Section 3.2, in which normal-form representation of a static game of complete information was given, for the case of static game of incomplete information. Our aim of this section is to formulate the dynamic TDD assignment static game of incomplete information and then be able to perform the equilibrium analysis in the case in which players strictly play mixed strategies. The definition of mixed strategy previously given, will be extend to our case of interest because now we will have to deal with imperfect information. Before proceeding, let's recall that the normal-form representation of a game of complete information in given in Definition 7. We wish to extend this framework in the scenario in which players know their own type dependent payoffs from outcomes, but they do not know the type dependent payoffs of their opponents with certainty. More precisely, everyone knows theirs own type dependent payoffs with certainty and of theirs opponents type dependent payoffs statistically, according to their beliefs. This particular assumption is quite important, if it is not so, then players will not be able to act rationally and perform the equilibrium analysis. Regarding the knowledge that players posse about their opponents, it can be of different types. For example, in the dynamic TDD assignment game of incomplete information, which we will define later in a formal way, if we fix the values of powers used by BSs and UEs, then the following classification can be made to distinguish between the players' types depending on knowledge of channel gains that each player possesses as follows:

1. Players may know with certainty only the channel gains situated among their UEs and statistically between other players and their UEs;
2. Players may possess the knowledge of all the channels gains among their UEs and among the other players and their UEs in the neighborhood with certainty and only statistically the channel gains among the players which are far.
3. Players may know the channel gains of half of the players of the game with certainty, which is their private information and of other halfs statistically.

The classification can go on and for every point players will have different types, thus the different type dependent payoffs. Notice that once Nature chooses the types of players, there is one and only game of incomplete information which corresponds to the selected types' profile. It is like having multiple dynamic TDD assignment games of incomplete and Nature decides which game will be played. The previous classification can be done under the *common prior assumption*, i.e. nobody will question how Nature chooses among the types of different players and this choice is common knowledge. It is now time to extend the framework of Section 3.2 by including three tools which will allow us to represent our game in the normal-form. First, a player's preference will be associated with his **type**. If a player is allowed to have many preferences over outcomes, each of them will be associated with a different type. More generally information that the player has about his own payoffs, or information he might have about other relevant attributes of the game, is also part of what defines a player's type. Second, the randomness over types is described by Nature choosing players' types profile. Hence, we introduce the **type spaces** for each player, which are the sets from which Nature chooses the players' types with some probability distribution. Finally, there is a common knowledge how the Nature choose between profiles of types of players. This is represented by a **common prior**, which is the probability distribution over types that is common knowledge among all the players playing the game. Now we can define a general normal-form representation of the n-player static game of incomplete information, as follows:

Definition 11. *The normal-form representation of n-player* ***static Bayesian game of incomplete information*** *is:*

1. *Set of players* $\mathcal{P} = \{1, 2, ..., n\}$,
2. *A collection of sets of actions* $\{A_1, A_2, ..., A_n\}$, *where* A_i *is the action set of player* i.
3. *A collection of sets of types spaces* $\{\Theta_1, \Theta_2, ..., \Theta_n\}$, *where* $\Theta_i = \{\theta_{i,1}, ..., \theta_{i,n}\}$ *is the type space of player i.*
4. *A collection of sets* $\{v_i(\cdot; \theta_i), \theta_i \in \Theta_i\}_{i=1}^n$, *where* $v_i : A \times \Theta_i \to \mathbb{R}$ *is the type dependent payoff function of player i and* $A \equiv A_1 \times A_2 ..... \times A_n$.
5. *A set* $\{\phi_1, ..., \phi_n\}$, *where* ϕ_i *describes the* belief *of player i with respect to the uncertainty over the other players' types.*

By looking at definition 11, aside from three basic components: players, actions and payoffs, the addition of types, type dependent payoffs and beliefs about the types of other players, captures the idea of *a static game of incomplete information* or so called the *Bayesian game*. A Bayesian game can be thought of as a game played by following these steps in order:

1. Nature chooses the players' types profile $(\theta_1, ...\theta_n)$.
2. Each player i learns his own type θ_i, which is his *private information* and then he can used his prior ϕ_i, to form posterior beliefs about all the other players.
3. All players choose simultaneously their actions $a_i \in A_i$.
4. Given each player's choice, the type-dependent payoffs $v_i(a;\theta_i)$ are realized for each player $i \in \mathcal{P}$.

It is worth shedding some light on what "posterior beliefs" means. We introduced the concept of a common prior which is the information available to all the players and describes how Nature chooses the players' type profile. Once every player has learned his type, all players are able to compute the posterior beliefs about their opponents. This can be done very easily by exploiting the conditional probabilities. They allow each player to update their prior believes, in the light of new evidence into posterior beliefs. In order to show how this mathematical tool will be used throughout the game, it may be useful to illustrate an example: Suppose that there are two players and each of them have two possible type, $\theta_1 \in \{a, b\}$ and $\theta_2 \in \{c, d\}$. Nature chooses these types according to common prior over their four possible combinations, which can be represented as a joint distribution 2×2 matrix as follows:

		Player 2's type	
		c	d
Player 1's type	a	$(\frac{1}{6})$	$(\frac{1}{3})$
	b	$(\frac{1}{3})$	$(\frac{1}{6})$

The common prior assumption implies that every player takes as given Nature's choice made by using the joint distribution matrix. Now, imagine that player 1 learns his type is a then he can use the conditional probabilities to update his prior beliefs into posterior beliefs as follows:

$$\phi_1(\theta_2 = c|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{6}}{\frac{1}{6} + \frac{1}{3}} = \frac{1}{3} \tag{3.3.1}$$

$$\phi_1(\theta_2 = d|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = d\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{3}}{\frac{1}{6} + \frac{1}{3}} = \frac{2}{3} \tag{3.3.2}$$

The equation (3.3.1) shows the probability of player 2 being c and the equation (3.3.2) shows the probability of player 2 being type d. In this way every player can update his beliefs about the other players, once each of them has learned his type. Now, we can move on and define the concept of *pure strategies* and *mixed straetgies*. Their formal definition has to be stated more carefully compared to the case of a static game, as we have to deal with players' types.

Definition 12. *Consider a static Bayesian game defined by following the definition 11. A* ***pure strategy*** *for player i is a function* $s_i : \Theta_i \to A_i$ *that specifies a pure action* $s_i(\theta_i)$ *that player i will choose when his type is* θ_i. *A* ***mixed strategy*** *is a probability distribution over a player's pure strategies.*

Now that we have precisely defined what is a static Bayesian game and defined the meaning of strategies for each player, we can state the solution concept that is derived from the Nash equilibrium in Bayesian games when players uses pure strategies as follows:

Definition 13. *In a Bayesian game defined under Definition 11. a strategy profile $s^* = (s_1^*, s_2^*, ..., s_N^*)$ is a* ***pure-strategy Bayesian Nash equilibrium****, if for every player i, for each of player i's type $\theta_i \in \Theta_i$, and for every $a_i \in A_i$, the strategy profile s^* solves:*

$$\sum_{\theta_{\setminus i} \in \Theta_{\setminus i}} \phi_i(\theta_{\setminus i}|\theta_i) v_i(s_i^*(\theta_i), s_{\setminus i}^*(\theta_{\setminus i}); \theta_i) \geq \sum_{\theta_{\setminus i} \in \Theta_{\setminus i}} \phi_i(\theta_{\setminus i}|\theta_i) v_i(a_i, s_{\setminus i}^*(\theta_{\setminus i}); \theta_i) \tag{3.3.3}$$

Just for formality, the definition has been stated for the case in which players use only pure strategies. The definition of our interest, which is when players use only mixed strategies can be stated as follows:

Definition 14. *In a Bayesian game defined under the definition 11, a strategy profile $\sigma^* = (\sigma_1^*, \sigma_2^*, ..., \sigma_N^*)$ is a* ***mixed-strategy Bayesian Nash equilibrium****, if for every player i, for each of player i's type $\theta_i \in \Theta$, and for every $a_i \in A_i$, the mixed strategy profile σ^* solves:*

$$\sum_{\theta_{\setminus i} \in \Theta_{\setminus i}} \phi_i(\theta_{\setminus i}|\theta_i) v_i(\sigma_i^*(\theta_i), \sigma_{\setminus i}^*(\theta_{\setminus i}); \theta_i) \geq \sum_{\theta_{\setminus i} \in \Theta_{\setminus i}} \phi_i(\theta_{\setminus i}|\theta_i) v_i(\sigma_i(\theta_i), \sigma_{\setminus i}^*(\theta_{\setminus i}); \theta_i) \tag{3.3.4}$$

As it can be clearly noticed, the definition of a mixed-strategy Nash equilibrium in the case of a static game of incomplete information is an extension of Definition 13 in which, instead of $s_i(\cdot)$ that maps types into pure actions, we are using σ_i that maps types into probability distribution over actions.

3.3.2 Game formulation in the case of static game of incomplete information using mixed strategies

Suppose we are in a strategic environment and let the BSs be the players of our game and each BS can choose between the actions *Uplink* or *Downlink*. As we are formulating a game of incomplete information, suppose also that each player does not possesses a complete information about his opponents' payoffs. More precisely, suppose that each player knows what are the types of their opponents statistically and what are their type-dependent payoffs but nobody knows for sure which is the type profile of their opponents which is being selected by Nature. With this assumption we want to capture the idea of incomplete information as there is an uncertainty about the players' opponents' type dependent payoffs. The role of Nature in this game is to choose players' types with some joint probability distribution. We suppose that the joint probability distribution exploited by Nature to chooses players' types is common knowledge, which later will be used by players to form posterior beliefs. This is the only element which is known by all the players with certainty. By assuming this property being true, we are invoking the so-called *common prior assumption.* Since each player may have different types, which means everyone may have different type dependent payoffs which are function of the power used by players, by UEs and depends also on the channel gains. We impose that the type of each player are only a function of the channel gains and as for the other variables, we assume that all the players and UEs use the maximum available power P and Q, respectively.

Under the previous assumption, we can write each player's type as a solely function of his channel gains, thus what defines player's types is his different channel gains which yield him different payoffs. As there can be different channel gains depending on the players' types, we extend the notation of channel matrix $\mathbf{H}$, UE-to-UE channel matrix $\mathbf{U}$ and the BS-to-BS channel matrix $\mathbf{B}$. The extension is required to include the possibility that players having different types will have different channel matrices, therefore different payoffs, as the other variables are fixed for each type. Let's recall that as the dynamic TDD assignment game of incomplete information starts, Nature chooses the players' types profile and each types profile will correspond to one and only one channel matrix, UE-to-UE channel matrix and BS-to-BS channel matrix from the collection of sets of matrices:

$$\mathcal{H} = \{\mathbf{H}^{\theta_1,\theta_2\cdots}\}, \text{ where } \theta_i \in \Theta_i \text{ for } i = 1, ..|\mathcal{N}|. \tag{3.3.5}$$

$$\mathcal{UE} = \{\mathbf{U}^{\theta_1,\theta_2\cdots}\}, \text{ where } \theta_i \in \Theta_i \text{ for } i = 1, ..|\mathcal{N}|. \tag{3.3.6}$$

$$\mathcal{B} = \{\mathbf{B}^{\theta_1,\theta_2\cdots}\}, \text{ where } \theta_i \in \Theta_i \text{ for } i = 1, ..|\mathcal{N}|. \tag{3.3.7}$$

where $\mathcal{H}$,$\mathcal{UE}$ and $\mathcal{B}$ contain the type dependent matrices and have the same cardinality. After Nature's move, there will be one and only matrix from each collection of sets which will correspond to the actual matrix of the game. So now, we can think of Nature choosing from different games as he is choosing among the collection of sets of matrices which will result in different type dependent payoffs for each player. As the game which is being played is of incomplete information, each player will have a statistical idea about the opponents payoffs, thus about the channel gains, which will allow every player to compute their opponents' expected type dependent payoffs. Each player will also be able to compute his own type-dependent payoff, as everyone can learn his own type. They will know with some probability distribution the type-dependent payoffs of their opponents, thus their channel gains. To be more precise, each player will know his own type and his channel gains, i.e. if $\theta_i = \{a, b\}$ and Nature chooses $\theta_i = a$ then player i will be able to learn that he is in the game $\mathbf{H}^{\dots,\theta_i=a\dots}$, $\mathbf{U}^{\dots,\theta_i=a\dots}$, $\mathbf{B}^{\dots,\theta_i=a\dots}$ and will just posses the information about his opponents' type profile statistically, thus about their payoffs. Notice that, in the previous clarification we implicitly meant that all the type dependent collection of sets are common knowledge among the players and each player have beliefs over the other players' types. It may be worth mentioning that the difficulty of our game is strictly related to the cardinality of the previous collection of sets because if it is bigger then also beliefs will be spread over bigger sets. As we mentioned in the previous Section, our game represents a practical scenario in which BSs have to alternate among their strategies to satisfy data traffic demands. Therefore, we suppose that nobody is willing to play any pure strategy and everybody uses only mixed strategies. If we would allow players to choose their strategy deterministically, we may end up in a situation in which Nash equilibrium predicts only one pure strategy for each player. In this way, all the players will be forced to play the only predicted pure strategy between *Uplink* or *Downlink* forever and will never be able to meet the data traffic demands. Thus the only way to satisfy the traffic requirements is to allow players to alternate between their strategies. The another reason to impose the previous assumption is also justified by the Theorem 1 which states sufficient conditions for the existence of the Nash Equilibrium. Notice that we are again trying to formulate our game as the *rock-paper-scissors* game and the only difference is that there can be many payoffs of each players, which means there can be many different tables like the Table 3.3 for the same game and only one table will correspond to the actual game, which strictly depends on Nature's choice. Finally, under the following assumptions:

Assumptions

1. Every player (BS) transmits signal with the maximum available power P,
2. Every UE transmits his signal with the maximum available power Q,
3. All the players mix among their strategies,
4. Every player is rational,
5. Everyone posses the common prior distribution with which Nature chooses players' types,
6. All types' sets are common knowledge,

we can write the normal-form representation of the dynamic TDD assignment static game of incomplete information by following the structure defined in Definition 11.

Normal-Form representation of dynamic TDD assignment game of incomplete information

1. Set of players $\mathcal{P} = \{1, 2, ...N\}$ (the Base station).
2. A collection of sets of actions $\{A_1, A_2, ..., A_N\}$ where $A_i = \{Uplink, Downlink\}$ is the action set of player i.
3. A collection of sets of types spaces $\{\Theta_1, \Theta_2, ..., \Theta_N\}$, where $\theta_i = \{\theta_{i,1}, \theta_{i,2}, \dots\}$ is the type space of player i.
4. A collection of sets $\{v_i(\cdot; \theta_i), \theta_i \in \Theta_i\}_{i=1}^{N}$, where $v_i : A \times \Theta_i \rightarrow$ (3.2.4) is a type dependent payoff function of player i and $A \equiv A_1 \times A_2 ..... \times A_N$.
5. A set $\{\phi_1, ..., \phi_N\}$, where ϕ_i describes the *belief* of player i with respect to the uncertainty over other players' types.

   As we are playing a game of incomplete information where every player mixes between his actions, we extend the Definition 11 to include simplexes.

6. A collection set of type dependent simplexes $\{\Delta A_1, ..., \Delta A_n\}$, where $\sigma(i, \vec{\theta}) \in \Delta A_i(\vec{\theta})$ is a type-dependent mixed strategy, $\sigma(i, \vec{\theta}) = \{\sigma_{i,\vec{\theta}}(Uplink), \sigma_{i,\vec{\theta}}(Downlink)\}$ and $\sigma_{i,\vec{\theta}}(Uplink) + \sigma_{i,\vec{\theta}}(Downlink) = 1$, where $\vec{\theta}$ is the type' vector containing the types profile of all players and $i \in \mathcal{P}$.

Notice that elements $1, 2$ and 3 are same of normal-form representation of the static game of complete information, with only difference that at step 2 now we are dealing with actions. The extension of this framework by including types spaces and set of beliefs, together with the extension of the payoff functions being type dependent, captures the idea of the static game of incomplete information. Now, we have normal-form representation of the dynamic TDD assignment game of incomplete information in which all players choose mixed strategies and we would like to find its Nash equilibrium. However, its solution in the case of n-player game is not trivial. The Theorem 1 tells us which are sufficient conditions under which the equilibrium exists and that is exactly our case. However, the solution for n-player games would result very laborious and in order to proceed, we consider the simplest 2-player case.

3.3.3 Solution of the 2-player dynamic TDD assignment game of incomplete information

We now consider the simplest case in which there are only two players (BSs) and both of them have the actions' set $S_i = \{Uplink, Downlink\}$, for $i = 1, 2$. In order to formulate our game, first of all we will proceed by writing the normal-form representation of our game under the assumptions stated before. As we are considering the 2-player game, we will be able to represent it through the matrix and the decision tree representation. Finally, a method to find the Nash equilibrium of dynamic TDD assignment game of incomplete information in the case of 2 players will be proposed. Notice that, the complexity of the 2-player game is also strictly related to the number of types that each player may have. For simplicity, we assume that both players can have only two different types, i.e, $\theta_1 = \{a, b\}$ and $\theta_2 = \{c, d\}$, which implies that the collection of sets of matrices are the following:

$$\mathcal{H} = \{\mathbf{H}^{\theta_1=a,\theta_2=c}, \mathbf{H}^{\theta_2=a,\theta_2=d}, \mathbf{H}^{\theta_1=b,\theta_2=c}, \mathbf{H}^{\theta_1=b,\theta_2=d}\}, \tag{3.3.8}$$

$$\mathcal{UE} = \{\mathbf{U}^{\theta_1=a,\theta_2=c}, \mathbf{U}^{\theta_2=a,\theta_2=d}, \mathbf{U}^{\theta_1=b,\theta_2=c}, \mathbf{U}^{\theta_1=b,\theta_2=d}\}, \tag{3.3.9}$$

$$\mathcal{B} = \{\mathbf{B}^{\theta_1=a,\theta_2=c}, \mathbf{B}^{\theta_2=a,\theta_2=d}, \mathbf{B}^{\theta_1=b,\theta_2=c}, \mathbf{B}^{\theta_1=b,\theta_2=d}\}. \tag{3.3.10}$$

As mentioned before, both players will learn their type once Nature has made his move by exploiting some joint probability distribution. It will result in one and only one matrix from each collection of set which will correspond to the actual game that is being played. Remember, players having different types yield them different payoffs which can be thought as having different games from which Nature chooses. In our case, as there are only 2 players and each player can have only two different types, we will have a total of four games. We will use the convention of identifying each player's type dependent mixed strategy profile as $\sigma(n, \theta_i)$, $\forall n \in \mathcal{N}$ and for $i = 1, 2$.

Normal-Form representation of the dynamic TDD assignment 2-player game of incomplete information

1. Set of Players $\mathcal{P} = \{1, 2\}$ (the Base stations),

2. A collection of sets of actions $\{A_1, A_2\}$ where $A_i = \{Uplink, Downlink\}$ for $i = 1, 2$.

3. A collection of sets of type spaces $\{\Theta_1, \Theta_2\}$ where $\Theta_i = \{\theta_i\}$ is the type space of player i for $i = 1, 2$ and $\theta_1 = \{a, b\}$ and $\theta_2 = \{c, d\}$.

4. A collection of sets $\{v_i(\cdot, \theta_i), \theta_i \in \Theta_i\}_{i=1}^{2}$, where $v_i : A \times \Theta_i \rightarrow$ (3.2.4) is the type dependent payoff function of player i and $A \equiv A_1 \times A_2$.

5. A set $\{\phi_1, \phi_2\}$ of beliefs of each player.

6. A collection set of type dependent simplexes $\{\Delta A_1, \Delta A_2\}$ where $\sigma(i, \vec{\theta}) \in \Delta S_i$ and $\sigma(1, \theta_1 = a, \theta_2 = c) = \{q, 1 - q\}$, $\sigma(1, \theta_1 = a, \theta_2 = d) = \{r, 1 - r\}$, $\sigma(1, \theta_1 = b, \theta_2 = c) = \{s, 1 - s\}$, $\sigma(1, \theta_1 = b, \theta_2 = d) = \{p, 1 - p\}$ are the elements of ΔS_1, $\sigma(2, \theta_1 = a, \theta_2 = c) = \{y, 1 - y\}$, $\sigma(2, \theta_1 = a, \theta_2 = d) = \{z, 1 - z\}$, $\sigma(2, \theta_1 = b, \theta_2 = c) = \{x, 1 - x\}$, $\sigma(2, \theta_1 = b, \theta_2 = d) = \{t, 1 - t\}$ are the elements of ΔS_2 and $0 \leq q, r, s, p \leq 1$ and $0 \leq y, z, x, t \leq 1$ are the probabilities with which player 1 and player 2 play the strategy Uplink, respectively.

Let's recall that as there are many possible values of q, r, p, s, there are many possible mixed strategies for both players. Finding the Nash equilibrium of our game consist in search of the one optimal pair from the pairs $(q^*, p^*), (q^*, s^*), (r^*, p^*), (r^*, s^*)$ and which pair has to be found depends strictly on the players' type profile which is being selected by Nature. To represent our game in the matrix form, we will use four matrix representations for the four possible type dependent games and then we will adopt the same methodology that we used in the static game of complete information to find its Nash equilibrium. In the matrix form representation we will use the notations $H_{n,j}^{\theta_1,\theta_2}, B_{k,n}^{\theta_1,\theta_2}, U_{x,j}^{\theta_1,\theta_2}$, $\forall n \in \mathcal{N}, k \in \mathcal{N}_{\backslash n}, \forall x \in \mathcal{U}_k$ and $j \in \mathcal{U}_n$, to represent the different type dependent channel gains. In order to highlight the type dependent payoffs in the decision tree representation, we will use the notation $P_{x,y}^{i,s,t}$, where $x \in A_1$, $y \in A_2$, $i \in \mathcal{P}$, $s \in \theta_i$ and $t \in \theta_2$, to identify players' type dependent payoffs. In this way we will be able to represent our game in the decision tree representation in a compact way by taking into account the players' type dependence. The decision tree of our 2-player dynamic TDD assignment game is presented in Fig. 3.11, which consists of $2^4 = 8$ terminal nodes. Now, we proceed by analyzing our game:

		P2($\theta_2 = c$)
		Uplink
P1($\theta_1 = a$)	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}^{a,c}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H_{1,y}^{a,c}Q}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}^{a,c}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H_{2,y}^{a,c}Q}\right)$
	Downlink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}^{a,c}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_2} U_{x,j}^{a,c}Q}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}^{a,c}\,Q}{\sigma_d^2+B_{1,2}P+\sum_{x\in\mathcal{U}_{2\backslash j}} H_{2,j}^{a,c}Q}\right)$

		P2($\theta_2 = c$)
		Downlink
P1($\theta_1 = a$)	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}^{a,c}\,Q}{\sigma_u^2+B_{2,1}^{a,c}P+\sum_{y\in\mathcal{U}_{1\backslash j}} H_{1,y}^{a,c}Q}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}^{a,c}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_1} U_{x,j}^{a,c}Q}\right)$
	Downlink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}^{a,c}\,P}{\sigma_d^2+H_{2,j}^{a,c}\,P}\right), w_2 \sum_{j\in\mathcal{U}_2} \log_2\left(1+\frac{H_{2,j}^{a,c}\,P}{\sigma_d^2+B_{1,j}^{a,c}P}\right)$

Figure 3.7: Matrix form representation of the 2-player TDD assignment game when the player 1 is of type $\theta_1 = a$ and player 2 is of type $\theta_2 = c$. We name this game as game AC.

Now we will use the same mechanism that we used in the dynamic TDD assignment game of complete information in which we wrote the expected payoffs of each player from their point of view. As this information was available to all the players, everyone could choose the mixed strategy which make his opponents indifferent of their choice. But now the things are different as we are in a game of incomplete information, the expression that each player will write to make his opponent indifferent will not be common knowledge. More precisely, every player will try to write the expected payoff of his opponent which dependents on the combination of actions and types. In order to proceed, suppose that Nature chooses players' types by exploiting the joint probability matrix presented in Table 3.1, which is common knowledge between the players.

It may worth mentioning that the game of incomplete information is played by following

		P2($\theta_2 = d$) Uplink
P1($\theta_1 = a$)	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H^{a,d}_{1,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H^{a,d}_{1,y}Q}\right), w_2 \sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,d}_{2,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j}H^{a,d}_{2,y}Q}\right)$
	Downlink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H^{a,d}_{1,j}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_2}U^{a,d}_{x,j}Q}\right), w_2 \sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,d}_{2,j}\,Q}{\sigma_d^2+B^{a,d}_{1,2}P+\sum_{x\in\mathcal{U}_{2\setminus j}}H^{a,d}_{2,x}Q}\right)$

		P2($\theta_1 = d$) Downlink
P1($\theta_1 = a$)	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H^{a,d}_{1,j}\,Q}{\sigma_u^2+B^{a,d}_{2,1}P+\sum_{y\in\mathcal{U}_{1\setminus j}} H^{a,d}_{1,y}Q}\right), w_2 \sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,d}_{2,j}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_1}U^{a,d}_{x,j}Q}\right)$
	Downlink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H^{a,d}_{1,j}\,P}{\sigma_d^2+H^{a,d}_{2,j}\,P}\right), w_2 \sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,d}_{2,j}\,P}{\sigma_d^2+B^{a,d}_{1,j}P}\right)$

Figure 3.8: Matrix form representation of the 2-player TDD assignment game when the player 1 is of type $\theta_1 = a$ and player 2 are of type $\theta_1 = d$. We name this game representation as game AD.

		Player 2's type	
		c	d
Player 1's type	a	$(\frac{1}{6})$	$(\frac{1}{3})$
	b	$(\frac{1}{3})$	$(\frac{1}{6})$

Table 3.1: The joint probability matrix used by Nature to select players' type profile.

these steps in order:

1. Nature chooses the players' type profile $(\theta_1, ...\theta_N)$.
2. Each player i learns his own type θ_i, which is his *private information* and then he can used his prior ϕ_i to form posterior beliefs about his opponents.
3. All players choose simultaneously their actions $a_i \in A_i$.
4. Given each player's choice, the type-dependent payoffs $v_i(a;\theta_i)$ are realized for each player $i \in \mathcal{P}$.

Suppose that Nature makes his move and chooses types $\theta_1 = a$ $\theta_2 = c$ for player 1 and player 2, respectively. Now both players have learned their type and now by using the common prior distribution of Table 3.1 and by using the conditional probability tool, both of them can update their prior beliefs as follows:

Player 1's Update

$$\phi_1(\theta_2 = c|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{6}}{\frac{1}{6}+\frac{1}{3}} = \frac{1}{3} \tag{3.3.11}$$

		P2($\theta_2 = c$)
		Uplink
P1($\theta_1 = b$)	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H_{1,j}^{b,c}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H_{1,y}^{b,c}Q}\right), w_2 \sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}^{b,c}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j}H_{2,y}^{b,c}Q}\right)$
	Downlink	$w_1\sum_{j\in\mathcal{U}_1}\log_2\left(1+\frac{H_{1,j}^{b,c}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_2}U_{x,j}^{b,c}Q}\right), w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}^{b,c}\,Q}{\sigma_d^2+B_{1,2}^{b,c}P+\sum_{x\in\mathcal{U}_{2\setminus j}}H_{2,j}^{b,c}Q}\right)$

		P2($\theta_2 = c$)
		Downlink
P1($\theta_1 = b$)	Uplink	$w_1\sum_{j\in\mathcal{U}_1}\log_2\left(1+\frac{H_{1,j}^{b,c}\,Q}{\sigma_u^2+B_{2,1}^{b,c}P+\sum_{y\in\mathcal{U}_{1\setminus j}}H_{1,y}^{b,c}Q}\right), w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}^{b,c}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_1}U_{x,j}^{b,c}Q}\right)$
	Downlink	$w_1\sum_{j\in\mathcal{U}_1}\log_2\left(1+\frac{H_{1,j}^{b,c}\,P}{\sigma_d^2+H_{2,j}^{b,c}\,P}\right), w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H_{2,j}^{b,c}\,P}{\sigma_d^2+B_{1,j}^{b,c}P}\right)$

Figure 3.9: Matrix form representation of the 2-player TDD assignment game when the player 1 is of type $\theta_1 = b$ and player 2 are of type $\theta_2 = c$. We name this game representation as game BC.

$$\phi_1(\theta_2 = d|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = d\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{3}}{\frac{1}{6}+\frac{1}{3}} = \frac{2}{3} \tag{3.3.12}$$

Player 2's Update

$$\phi_1(\theta_1 = a|\theta_2 = c) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c\}}{P_r\{\theta_2 = c\}} = \frac{\frac{1}{6}}{\frac{1}{6}+\frac{1}{3}} = \frac{1}{3} \tag{3.3.13}$$

$$\phi_1(\theta_1 = b|\theta_2 = c) = \frac{P_r\{\theta_1 = b \cap \theta_2 = c\}}{P_r\{\theta_2 = c\}} = \frac{\frac{1}{3}}{\frac{1}{6}+\frac{1}{3}} = \frac{2}{3} \tag{3.3.14}$$

Note that players are able to compute the equation (3.3.11)-(3.3.14) because they have statistical knowledge about their opponents' type space. We now proceed by writing down the expected payoffs for each player's opponent from their point of view as a function of their own mixed strategies. Player 1 knows that his type is $\theta_1 = a$ and according to him, he can be in game AC or game AD. On the other hand, player 2 knows his type being $\theta_2 = c$ and according to him he can be in the game AC or BC.

		P2($\theta_2 = d$)
		Uplink
P1($\theta_1 = b$)	Uplink	$w_1 \sum_{j\in\mathcal{U}_1} \log_2\left(1+\frac{H^{b,d}_{1,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j} H^{b,d}_{1,y}Q}\right), w_2 \sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{b,d}_{2,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j}H^{b,d}_{2,y}Q}\right)$
	Downlink	$w_1\sum_{j\in\mathcal{U}_1}\log_2\left(1+\frac{H^{b,d}_{1,j}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_2}U^{b,d}_{x,j}Q}\right), w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{b,d}_{2,j}\,Q}{\sigma_d^2+B^{b,d}_{1,2}P+\sum_{x\in\mathcal{U}_{2\setminus j}}H^{b,d}_{2,j}Q}\right)$

		P2($\theta_2 = d$)
		Downlink
P1($\theta_1 = b$)	Uplink	$w_1\sum_{j\in\mathcal{U}_1}\log_2\left(1+\frac{H^{b,d}_{1,j}\,Q}{\sigma_u^2+B^{b,d}_{2,1}P+\sum_{y\in\mathcal{U}_{1\setminus j}}H^{b,d}_{1,y}Q}\right), w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{b,d}_{2,j}\,P}{\sigma_d^2+\sum_{x\in\mathcal{U}_1}U^{b,d}_{x,j}Q}\right)$
	Downlink	$w_1\sum_{j\in\mathcal{U}_1}\log_2\left(1+\frac{H^{b,d}_{1,j}\,P}{\sigma_d^2+H^{b,d}_{2,j}\,P}\right), w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{b,d}_{2,j}\,P}{\sigma_d^2+B^{b,d}_{1,j}P}\right)$

Figure 3.10: Matrix form representation of the 2-player TDD assignment game when the player 1 is of type $\theta_1 = b$ and player 2 are of type $\theta_2 = d$. We name this game representation as game BD.

From the point of view of player 1:

$$
\begin{aligned}
v_2(a_1, U; a, \theta_2) = & \frac{1}{3}\Bigg[q\Bigg(w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,c}_{2,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j}H^{a,c}_{2,y}Q}\right)\Bigg)\\
& (1-q)\Bigg(w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,c}_{2,j}\,Q}{\sigma_d^2+B_{1,2}P+\sum_{x\in\mathcal{U}_{2\setminus j}}H^{a,c}_{2,j}}\right)\Bigg)\Bigg]_1\\
& \frac{2}{3}\Bigg[r\Bigg(w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,d}_{2,j}\,Q}{\sigma_u^2+\sum_{m=1,2}\sum_{y\in\mathcal{U}_m, y\neq j}H^{a,d}_{2,y}Q}\right)\Bigg)\\
& +(1-r)\Bigg(w_2\sum_{j\in\mathcal{U}_2}\log_2\left(1+\frac{H^{a,d}_{2,j}\,Q}{\sigma_d^2+B^{a,d}_{1,2}P+\sum_{x\in\mathcal{U}_{2\setminus j}}H^{a,d}_{2,x}Q}\right)\Bigg)\Bigg]_2
\end{aligned}
\tag{3.3.15}
$$

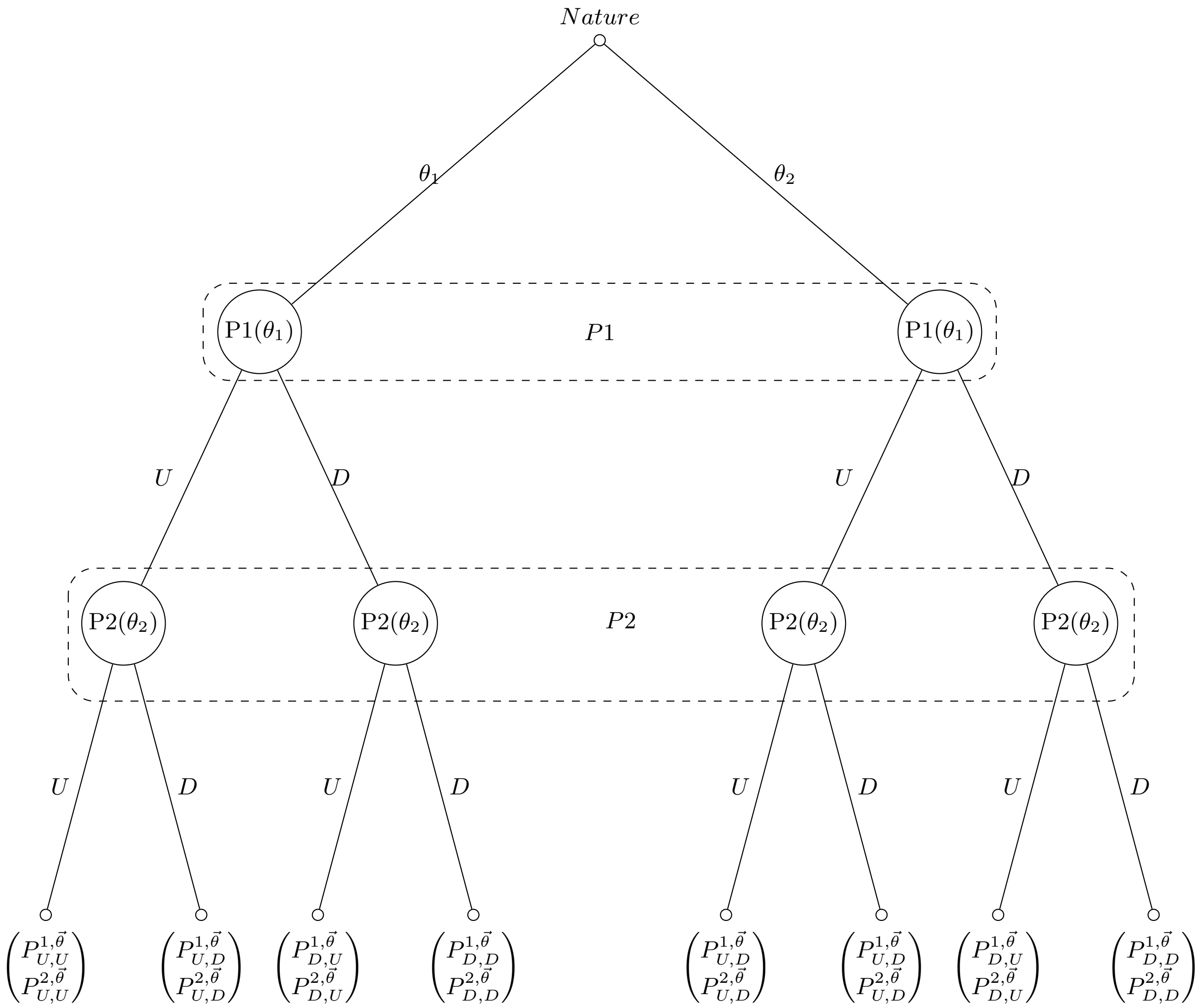

Figure 3.11: Decision tree of the 2-player dynamic TDD assignment game of incomplete information. The dotted line represents the idea of both players being unconscious of his opponent's move and type. The vector $\vec{\theta}$ is the types' profile vector.

$$
\begin{aligned}
v_2(a_1, D; a, \theta_2) = & \frac{1}{3}\Bigg[q\Bigg(w_2 \sum_{j\in\mathcal{U}_2} \log_2\Bigg(1 + \frac{H_{2,j}^{a,c}\ P}{\sigma_d^2 + \sum\limits_{x\in\mathcal{U}_1} U_{x,j}^{a,c} Q}\Bigg)\Bigg) \\
& + (1-q)\Bigg(w_2 \sum_{j\in\mathcal{U}_2} \log_2\Bigg(1 + \frac{H_{2,j}^{a,c}\ P}{\sigma_d^2 + B_{1,j}^{a,c} P}\Bigg)\Bigg)\Bigg]_3 \\
+ & \frac{2}{3}\Bigg[r\Bigg(w_2 \sum_{j\in\mathcal{U}_2} \log_2\Bigg(1 + \frac{H_{2,j}^{a,d}\ P}{\sigma_d^2 + \sum\limits_{x\in\mathcal{U}_1} U_{x,j}^{a,d} Q}\Bigg)\Bigg) \\
& + (1-r)\Bigg(w_2 \sum_{j\in\mathcal{U}_2} \log_2\Bigg(1 + \frac{H_{2,j}^{a,d}\ P}{\sigma_d^2 + B_{1,j}^{a,d} P}\Bigg)\Bigg)\Bigg]_4
\end{aligned}
\tag{3.3.16}
$$

From the point of view of player 2:

$$v_1(U,a_2;\theta_1,c) = \frac{2}{3}\Bigg[y\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{a,c}\,Q}{\sigma_u^2+\sum\limits_{m=1,2}\sum\limits_{\substack{y\in\mathcal{U}_m\\ y\neq j}} H_{1,y}^{a,c}Q}\Bigg)\Bigg)$$
$$+(1-y)\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{a,c}\,Q}{\sigma_u^2+B_{2,1}^{a,c}P+\sum\limits_{y\in\mathcal{U}_{1\setminus j}} H_{1,y}^{a,c}Q}\Bigg)\Bigg)\Bigg]_5$$
$$+\frac{2}{3}\Bigg[x\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{b,c}\,Q}{\sigma_u^2+\sum\limits_{m=1,2}\sum\limits_{\substack{y\in\mathcal{U}_m\\ y\neq j}} H_{1,y}^{b,c}Q}\Bigg)\Bigg)$$
$$+(1-x)\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{b,c}\,Q}{\sigma_u^2+B_{2,1}^{b,c}P+\sum\limits_{y\in\mathcal{U}_{1\setminus j}} H_{1,y}^{b,c}Q}\Bigg)\Bigg)\Bigg]_6 \tag{3.3.17}$$

$$v_1(D,a_2;\theta_1,c) = \frac{2}{3}\Bigg[y\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{a,c}\,P}{\sigma_d^2+\sum\limits_{x\in\mathcal{U}_2} U_{x,j}^{a,c}Q}\Bigg)\Bigg)$$
$$+(1-y)\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{a,c}\,P}{\sigma_d^2+H_{2,j}^{a,c}\,P}\Bigg)\Bigg)\Bigg]_7$$
$$+\frac{1}{3}\Bigg[x\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{b,c}\,P}{\sigma_d^2+\sum\limits_{x\in\mathcal{U}_2} U_{x,j}^{b,c}Q}\Bigg)\Bigg)$$
$$+(1-x)\Bigg(w_1 \sum_{j\in\mathcal{U}_1} \log_2\Bigg(1+\frac{H_{1,j}^{b,c}\,P}{\sigma_d^2+H_{2,j}^{b,c}\,P}\Bigg)\Bigg)\Bigg]_8 \tag{3.3.18}$$

Now from the point of view of player 1, it can be clearly notice that once he has learned his type $\theta_1 = a$, he knows that the possible games from which Nature could choose are the game AC or AD. In order to choose his optimal type dependent mixed strategy to make his opponent indifferent of his choice he can solve the sub-equations of the equations (3.3.15)-(3.3.16) as follows:

$$\frac{1}{3}[q(\ldots)+(1-q)(\ldots)]_1 = \frac{1}{3}[q(\ldots)+(1-q)(\ldots)]_3, \tag{3.3.19}$$

$$\frac{2}{3}[r(\ldots)+(1-r)(\ldots)]_2 = \frac{2}{3}[r(\ldots)+(1-r)(\ldots)]_4. \tag{3.3.20}$$

On the other hand, player 2 can do the same by solving the equations:

$$\frac{1}{3}[y(...) + (1-y)(...)]_5 = \frac{1}{3}[y(...) + (1-y)(...)]_7, \tag{3.3.21}$$

$$\frac{1}{3}[x(...) + (1-x)(...)]_6 = \frac{1}{3}[x(...) + (1-x)(...)]_8. \tag{3.3.22}$$

Which yields the Nash equilibrium (a^*, c^*) of our game AC which is the actual game that has been selected by the Nature.

3.3.4 Solution of the 3-player dynamic TDD assignment game of incomplete information

As we saw in the previous section, each player can select his optimal mixed strategy by using his private information based on the common prior knowledge. However, for the case in which number of players increase, thus for $|\mathcal{P}| > 2$, things do not work in the same way. The reason lies in the degree of freedom that each player has in order to make his opponents indifferent of their choice. As every player can choose just one optimal mixed strategy, which means each of them has just one degree of freedom, it is not possible to make more than one opponents indifferent of their choice. Therefore we introduce the idea of an average game. For us, an average game is a game in which once each player has learned his type, everyone can transform their common prior believes into posterior believes which gives them a statistical idea of which game is being played. By exploiting this statistical knowledge, everyone can create for himself an average game in which all the payoffs are averaged over all the possible games. Before proceeding for its solution, we give a formal representation by following the same guidelines that we used in the previous Section. Then we will try to find its solution by developing a new mechanism which consider the game as an average game, where all the type dependent payoffs of each player will be reduced to an average payoff which will result type independent. In order to represent it in a compact way, we will use the notation $\sigma(i, \theta_1, \theta_2, \theta_3)$ to indicate the type dependent mixed strategy of player i and the notation $\vec{\theta}$ will be used to indicate players' types profile.

Normal-Form representation of the dynamic TDD assignment 3-player dynamic game of incomplete information

1. Set of Players $\mathcal{P} = \{1, 2, 3\}$ (the Base stations),

2. A collection of sets of actions $\{A_1, A_2, A_3\}$ where $A_i = \{Uplink, Downlink\}$ for $i = 1, 2, 3$.

3. A collection of sets of type spaces $\{\Theta_1, \Theta_2, \Theta_3\}$ where $\Theta_i = \{\theta_i\}$ is the type space of player i for $i = 1, 2$ and $\theta_1 = \{a, b\}$,$\theta_2 = \{c, d\}$ and $\theta_3 = \{e, f\}$.

4. A collection of sets of payoff functions $\{v_i(\cdot, \theta_i), \theta_i \in \Theta_i\}_{i=1}^3$, where $v_i : A \times \Theta_i \rightarrow$ (3.2.4) is the type dependent payoff function of player i and $A \equiv A_1 \times A_2 \times A_3$.

5. A set $\{\phi_1, \phi_2, \phi_3\}$ of beliefs for each player.

6. A collection set of type dependent simplexes $\{\Delta A_1, \Delta A_2, \Delta A_3\}$ where $\sigma(i, \vec{\theta}) \in \Delta S_i$ and $\sigma(1, a, c, e) = \{q_1, 1 - q_1\}$, $\sigma(1, a, c, f) = \{q_2, 1 - q_2\}$, $\sigma(1, a, d, e) = \{q_3, 1 - q_3\}$, $\sigma(1, a, d, f) = \{q_4, 1 - q_4\}$,$\sigma(1, b, c, e) = \{q_5, 1 - q_5\}$, $\sigma(1, b, c, f) = \{q_6, 1 - q_6\}$, $\sigma(1, b, d, e) = \{q_7, 1-q_7\}$, $\sigma(1, b, d, f) = \{q_8, 1-q_8\}$ are the mixed strategies of player 1, $\sigma(2, a, c, e) = \{p_1, 1 - p_1\}$, $\sigma(2, a, c, f) = \{p_2, 1 - p_2\}$, $\sigma(2, a, d, e) = \{p_3, 1 - p_3\}$, $\sigma(2, a, d, f) = \{p_4, 1 - p_4\}$,$\sigma(2, b, c, e) = \{p_5, 1 - p_5\}$, $\sigma(2, b, c, f) = \{p_6, 1 - p_6\}$,

$\sigma(2,b,d,e) = \{p_7, 1-p_7\}$, $\sigma(2,b,d,f) = \{p_8, 1-p_8\}$ are the mixed strategies of player 2, $\sigma(3,a,c,e) = \{r_1, 1-r_1\}$, $\sigma(3,a,c,f) = \{r_2, 1-r_2\}$, $\sigma(3,a,d,e) = \{r_3, 1-r_3\}$, $\sigma(3,a,d,f) = \{r_4, 1-r_4\}$,$\sigma(3,b,c,e) = \{r_5, 1-r_5\}$, $\sigma(3,b,c,f) = \{r_6, 1-r_6\}$, $\sigma(3,b,d,e) = \{r_7, 1-r_7\}$, $\sigma(3,b,d,f) = \{r_8, 1-r_8\}$ are the mixed strategies of player 3 and all these variable are ranging from 0 to 1

Let's recall that as there are many possible values that all these variables can assume, there are many different possible mixed strategies and we are interested to find the optimum ones which gives us the equilibrium condition.

Solution of the 3-player dynamic TDD assignment game of incomplete information

In order to look for its solution, suppose that Nature uses the following joint probability distribution table to select the players' types.

Players' type profile$(\theta_1, \theta_2, \theta_3)$	**Probabilities**
a, c, e	1/8
a, c, f	1/8
a, d, e	1/8
a, d, f	1/8
b, c, e	1/8
b, c, f	1/8
b, d, e	1/8
b, d, f	1/8

Now suppose that Nature chooses the type profile $(\theta_1 = a, \theta_2 = c, \theta_3 = e)$. At this point, each player can learn his own type and update his believes about his opponents as follows:

Player 1's Update

$$\phi_1(\theta_2 = c, \theta_3 = e|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c \cap \theta_3 = e\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.23}$$

$$\phi_1(\theta_2 = c, \theta_3 = f|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c \cap \theta_3 = f\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.24}$$

$$\phi_1(\theta_2 = d, \theta_3 = e|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = d \cap \theta_3 = e\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.25}$$

$$\phi_1(\theta_2 = d, \theta_3 = f|\theta_1 = a) = \frac{P_r\{\theta_1 = a \cap \theta_2 = d \cap \theta_3 = f\}}{P_r\{\theta_1 = a\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.26}$$

Player 2's Update

$$\phi_2(\theta_1 = a, \theta_3 = e|\theta_2 = c) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c \cap \theta_3 = e\}}{P_r\{\theta_2 = c\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.27}$$

$$\phi_2(\theta_1 = a, \theta_3 = f|\theta_2 = c) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c \cap \theta_3 = f\}}{P_r\{\theta_2 = c\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.28}$$

$$\phi_2(\theta_1 = b, \theta_3 = e|\theta_2 = c) = \frac{P_r\{\theta_1 = b \cap \theta_2 = c \cap \theta_3 = e\}}{P_r\{\theta_2 = c\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.29}$$

$$\phi_2(\theta_1 = b, \theta_3 = f|\theta_2 = c) = \frac{P_r\{\theta_1 = b \cap \theta_2 = c \cap \theta_3 = f\}}{P_r\{\theta_2 = c\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.30}$$

Player 3's Update

$$\phi_3(\theta_1 = a, \theta_2 = c|\theta_3 = e) = \frac{P_r\{\theta_1 = a \cap \theta_2 = c \cap \theta_3 = e\}}{P_r\{\theta_3 = e\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.31}$$

$$\phi_3(\theta_1 = a, \theta_2 = d|\theta_3 = e) = \frac{P_r\{\theta_1 = a \cap \theta_2 = d \cap \theta_3 = e\}}{P_r\{\theta_3 = e\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.32}$$

$$\phi_3(\theta_1 = b, \theta_2 = c|\theta_3 = e) = \frac{P_r\{\theta_1 = b \cap \theta_2 = c \cap \theta_3 = e\}}{P_r\{\theta_3 = e\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.33}$$

$$\phi_3(\theta_1 = b, \theta_2 = d|\theta_3 = e) = \frac{P_r\{\theta_1 = b \cap \theta_2 = d \cap \theta_3 = e\}}{P_r\{\theta_3 = e\}} = \frac{\frac{1}{8}}{(\frac{1}{8})4} = \frac{1}{4} \tag{3.3.34}$$

As it can be seen from the previous computations, the values of believes result same. This may be caused by the fact that Nature uses a joint probability distribution table with uniform distribution. Once Nature has made his move, all players know for sure their type and everyone has a statistical idea of which game is being played. Each player knows that there are four possible games and only one game is actually being played. As we are in a game of incomplete information, no player can know which is the actual game. As we found the Nash equilibrium for the 2-player case, if we now proceed in the same way, we will end up having two equations as function of just one variable for each players' point of view. This occur because each player cannot make two players indifferent of their choice by selecting just one optimum mixed strategy. The only thing that can be done from now on is that we can try to solve the average game for each player, where everyone has his own game in which the payoffs as given as the average payoffs of all the four games from each players' point of view. In this way, we will end up having a game from which Nature's role will be removed and each player can solve his own average game in order to find the average optimal mixed strategy which can give everyone on average satisfactory results. It's worth emphasizing that in the average game everyone plays mixed strategy which is not type dependent as by averaging over types we are making players type independent.

As we have three players and each player can be of two different types, we have a total of eight possible games from which Nature can choose, which means:

$$\mathcal{H} = \{\mathbf{H}^{\theta_1,\theta_2,\theta_3}\}, |\mathcal{H}| = 8, \tag{3.3.35}$$

$$\mathcal{UE} = \{\mathbf{U}^{\theta_1,\theta_2,\theta_3}\}, |\mathcal{U}| = 8, \tag{3.3.36}$$

$$\mathcal{B} = \{\mathbf{B}^{\theta_1,\theta_2,\theta_3}\}, |\mathcal{B}| = 8. \tag{3.3.37}$$

As we said before, we are now trying to solve an average game, where everyone on average chooses a mixed strategy which is optimum. In our case, if player i has learned his type, his opponents' possible types $\vec{\theta}_{\setminus i}$ are four, and player i will solve a game which will be an average of four games seen from his point of view. Now, we can write the average payoffs, which will result type independent from each player's point of view. For the sake of simplicity and space, we will use the abbreviation U and D to denote the actions Uplink and Downlink, respectively.

From the point of view of player 1:

Now player 1 knows that he is of type a, he can compute his and his opponents' average payoffs in Uplink and Downlink as follows:

$$\begin{aligned} E_1[v_1(U,a_2,a_3)] =& \phi_1(a,c,e)v_1(U,a_2,a_3;a,c,e) + \phi_1(a,c,f)v_1(U,a_2,a_3;a,c,f) \\ &+ \phi_1(a,d,e)v_1(U,a_2,a_3;a,d,e) + \phi_1(a,d,f)v_1(U,a_2,a_3;a,d,f) \end{aligned} \tag{3.3.38}$$

$$\begin{aligned} E_1[v_1(D,a_2,a_3)] =& \phi_1(a,c,e)v_1(D,a_2,a_3;a,c,e) + \phi_1(a,c,f)v_1(D,a_2,a_3;a,c,f) \\ &+ \phi_1(a,d,e)v_1(D,a_2,a_3;a,d,e) + \phi_1(a,d,f)v_1(D,a_2,a_3;a,d,f) \end{aligned} \tag{3.3.39}$$

$$\begin{aligned} E_1[v_2(a_1,U,a_3)] =& \phi_1(a,c,e)v_2(a_1,U,a_3;a,c,e) + \phi_1(a,c,f)v_2(a_1,U,a_3;a,c,f) \\ &+ \phi_1(a,d,e)v_2(a_1,U,a_3;a,d,e) + \phi_1(a,d,f)v_2(a_1,U,a_3;a,d,f) \end{aligned} \tag{3.3.40}$$

$$\begin{aligned} E_1[v_2(a_1,D,a_3)] =& \phi_1(a,c,e)v_2(a_1,D,a_3;a,c,e) + \phi_1(a,c,f)v_2(a_1,D,a_3;a,c,f) \\ &+ \phi_1(a,d,e)v_2(a_1,D,a_3;a,d,e) + \phi_1(a,d,f)v_2(a_1,D,a_3;a,d,f) \end{aligned} \tag{3.3.41}$$

$$\begin{aligned} E_1[v_3(a_1,a_2,U)] =& \phi_1(a,c,e)v_3(a_1,a_2,U;a,c,e) + \phi_1(a,c,f)v_3(a_1,a_2,U;a,c,f) \\ &+ \phi_1(a,d,e)v_3(a_1,a_2,U;a,d,e) + \phi_1(a,d,f)v_3(a_1,a_2,U;a,d,f) \end{aligned} \tag{3.3.42}$$

$$\begin{aligned} E_1[v_3(a_1,a_2,D)] =& \phi_1(a,c,e)v_3(a_1,a_2,D;a,c,e) + \phi_1(a,c,f)v_3(a_1,a_2,D;a,c,f) \\ &+ \phi_1(a,d,e)v_3(a_1,a_2,D;a,d,e) + \phi_1(a,d,f)v_3(a_1,a_2,D;a,d,f) \end{aligned} \tag{3.3.43}$$

Notice that we used the notation $E_1[P^i_{a_1,a_2,a_3}]$, where a_1, a_2, a_3 are the actions of player 1,2 and 3, to indicate the expected payoffs of all the player and the subscript indicated that this knowledge is possessed only by player 1. These are the average payoffs which are now type independent and their knowledge is only possessed by player 1. Naturally, the final outcome depends on the choices made by other payers too and at the end we will end up having a game which can be represented by the decision tree with eight end nodes as we saw for the case of 3-player game of complete information. We using the notation for the expected payoffs previously mentioned, the decision tree can be depicted as follows:

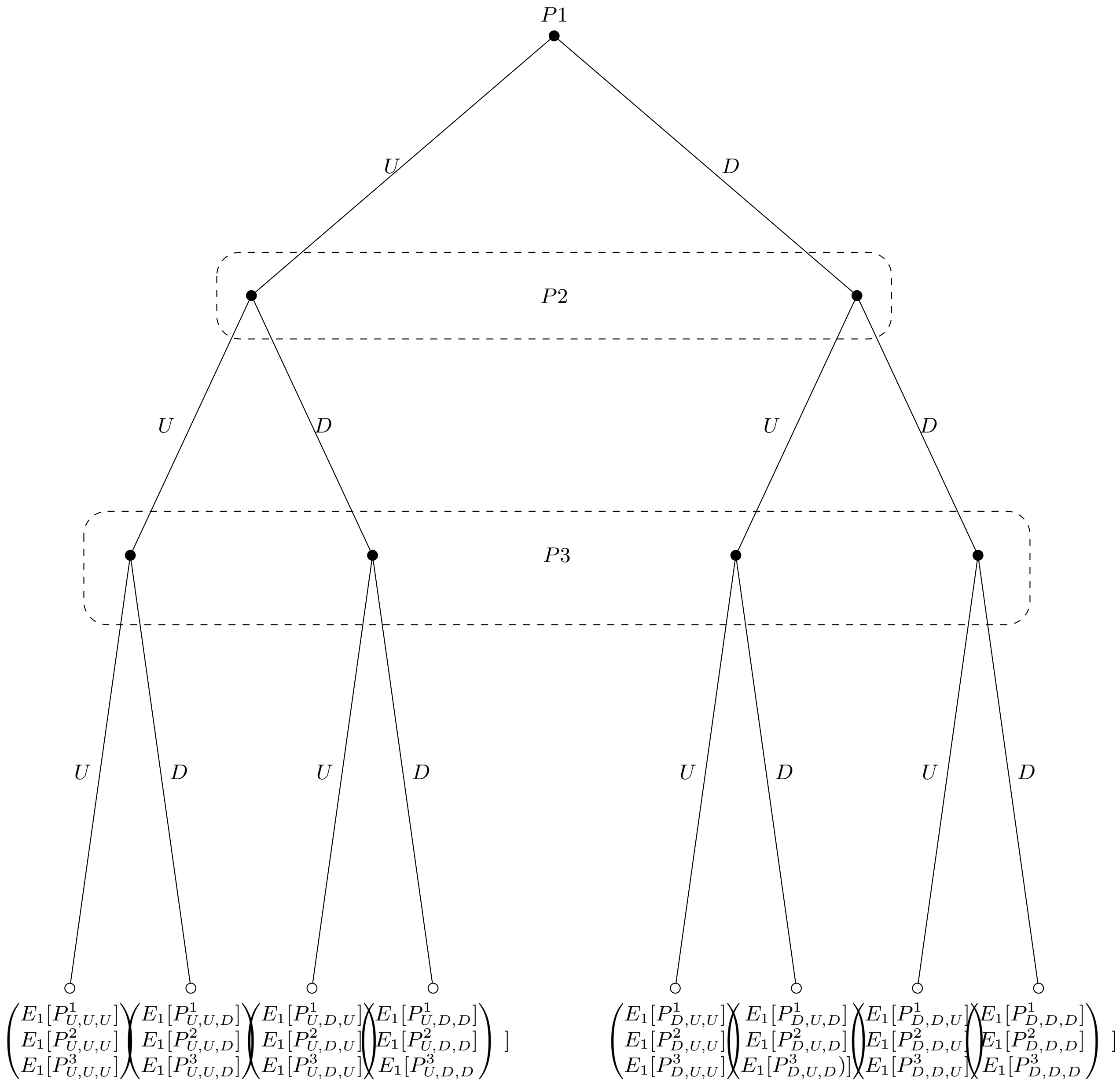

Figure 3.12: Decision tree of the average 3-player dynamic TDD assignment game of incomplete information from player 1's point of view.

Now this is the idea of the average game that player 1 has and he cannot know more than this. By following the previous steps, a similar decision tree can be created for player 2's and player 3's point of view, which would of course result their private knowledge. Now, suppose that the type independent mixed strategies to play the average game are $\sigma_1 = \{q, 1-q\}$,$\sigma_2 = \{p, 1-p\}$ and $\sigma_3 = \{r, 1-r\}$ and player 1 possesses an estimate, which could be totally wrong, about the optimum values of p^* and r^*. The optimum mixed strategy q is what he has to choose in order to obtain on average satisfactory results by making his opponents indifferent of their choice. For here, player 1 possesses all the information he needs to solve the average game, therefore, he sees the game of incomplete information as a game of complete information and can solve it by following the same procedure described in (3.2.23).

From the point of view of player 1:

$$
\begin{aligned}
& r^* \, p^* \Bigg(E[v_1(U,U,U)] - E[v_1(D,U,U)] - E[v_1(U,D,U)] + E[v_1(D,D,U)] \\
& - E[v_1(U,U,D)] + E[v_1(D,U,D)] + E[v_1(U,D,D)] - E[v_1(D,D,D)] \Bigg) + \\
& p^* \Bigg(E[v_1(U,U,D)] - E[v_1(D,U,D)] - E[v_1(U,D,D)] + E[v_1(D,D,D)] \Bigg) - \\
& r^* \Bigg(E[v_1(U,D,D)] - E[v_1(D,D,D)] \Bigg) = E[v_1(D,D,U)] - E[v_1(U,D,U)]
\end{aligned}
\tag{3.3.44}
$$

$$
\begin{aligned}
& r^* \, q \Bigg(E[v_2(U,U,U)] - E[v_2(U,D,U)] - E[v_2(D,U,U)] + E[v_2(D,D,U)] \\
& - E[v_2(U,U,D)] + E[v_2(U,D,D)] + E[v_2(D,U,D)] - E[v_2(D,D,D)] \Bigg) + \\
& q \Bigg(E[v_2(U,U,D)] - E[v_2(U,D,D)] - E[v_2(D,U,D)] + E[v_2(D,D,D)] \Bigg) - \\
& r^* \Bigg(E[v_2(D,U,D)] - E[v_2(D,D,D)] \Bigg) = E[v_2(D,D,D)] - E[v_2(D,U,D)]
\end{aligned}
\tag{3.3.45}
$$

$$
\begin{aligned}
& p^* \, q \Bigg(E[v_3(U,U,U)] - E[v_3(U,U,D)] - E[v_3(D,U,U)] + E[v_3(D,U,D)] \\
& - E[v_3(U,D,U)] + E[v_3(U,D,D)] + E[v_3(D,D,U)] - E[v_3(D,D,D)] \Bigg) + \\
& q \Bigg(E[v_3(U,D,U)] - E[v_3(U,D,D)] - E[v_3(D,D,U)] + E[v_3(D,D,D)] \Bigg) - \\
& p^* \Bigg(E[v_3(D,D,U)] - E[v_3(D,D,D)] \Bigg) = E[v_3(D,D,D)] - E[v_3(D,D,U)]
\end{aligned}
\tag{3.3.46}
$$

All the values within the parenthesis are known by player 1 and he also possesses an estimate of what are the optimal values of p^* and r^*, which could be totally wrong. In order to obtain his optimal mixed strategy q^*, player 1 can directly discard the equation (3.3.44) and solve the equation (3.3.45) or the (3.3.46) for $q*$ which could give him on average satisfactory results. The same procedure has to be followed by player 2 and player 3, where they use the joint probability distribution to know their and their opponents' expected payoffs. Then, by using an estimate their opponents' mixed strategies, they can find their own optimal mixed strategy for the average game. It may be worth mentioning again that nothing prevents that the estimate of players' opponents mixed strategies being wrong.

3.4 Dynamic game of complete information

We analyzed in the previous Sections how to find the optimal mixed strategies with which each player should choose their moves in order to achieve the Nash equilibrium. But, we strictly restricted our attention to strategic scenarios in which all players were allowed to choose among their strategies/actions only simultaneously. We can widen our study, e.g., by extending the case of static game of complete information to the case of dynamic game of complete information [5]. Therefore, in this Section we derive the most common representation for games that unfold over time and in which some players can learn the actions of other players. First of all, we need to extend the framework of normal-form representation to the so-called *extensive-form* representation. As with the normal-form game, three elements must be part of any extensive-form game's representation:

1. Set of players, $\mathcal{N} = \{..\}$.
2. A collection set of pure actions.
3. Players' payoffs as a function of outcomes $\{v_i(\cdot)\}_{i\in\mathcal{N}}$.

To overcome the limitation of normal-form representation and capture sequential play, we need to expand rather simplistic concept of pure and mixed strategy to a more complex organization of actions. In order to do so, we comprehend in our representation the elements which captures the idea of what players can do and when they can do:

4. Order of moves.
5. Actions of players when they can move.

Because some players move after choices made by other players, we need to be able to describe the knowledge that players have about the history of game when it is their turn to move. More precisely, what is relevant to players who move later is not the chronological order of play, but the information they posses in order to make their decision. To represent the way in which information and knowledge unfold and the players' mixed strategies, we add another two components to the description of an extensive-form game:

6. The knowledge that players have when they can move.
7. A collection set of simplexes which comprehend the players' mixed strategies.

Finally, to be able to analyze these situations with the methods and concepts to which we have already been introduced, we add a final and familiar requirement:

8. The structure of the extensive form-game represented by 1-7 is a common knowledge among all the players.

Now all the elements mentioned above are what can allow us to represent a game which unfolds over time. As it is very intuitive to understand, representing it in matrix form might not be a good idea as it would not capture the essence of play and moreover, it would be able to represent just the 2-player case. As a better choice, we will only use the decision tree representation which perfectly captures the idea of a sequentially played game, e.g, if a player has to make his move at i-th place, he will find himself at the level i of decision tree. Once all the players have made their choices, a final outcome will be realized, depending on the players' action profile [7].

Until here, we extended the concepts developed in the previous sections to be able to represent a game which unfolds over time. Now, we proceed by extending the notations and definitions which will allow us to describe our game in a precise manner. Let K_i be the collection of all information sets at which player i plays, and let $k_i \in K_i$ be one of $i's$ information sets. Let $A_i(k_i)$ be the action that player i can take at k_i, and let A_i be the set of all actions of player i, $A_i = \bigcup_{k_i \in K_i} A_i(k_i)$.

Definition 15. *A **pure strategy** for player i is a mapping $s_i : K_i \rightarrow A_i$ that assigns an action $s_i(k_i) \in A_i(k_i)$ for every information set $k_i \in H_i$. We denote by S_i the set of all pure strategy mapping $s_i \in S_i$.*

We again want to formulate a game where all the players choose only mixed strategies, thus we define the mixed strategies in an extensive form game as follows:

Definition 16. *A **mixed strategy** for player i is the probability distribution over his pure strategies.*

Now we can define what is called a sub-game, which will allow us to analyze our game into smaller games by reducing the complexity. A sub-game's decision tree consists of a number of nodes less than the actual game's decision tree and can be defined as follows:

Definition 17. *A proper **sub-game** G of any extensive form game Γ consists of a single node and all it's successors.*

The idea of a proper sub-game is simple and it will allow us to dissect an extensive-form game into a sequence of smaller games, an approach that will allow us to apply the concept of sequential rationality [6]. A player is said to be sequentially rational who plays an optimal strategy based on the information he has. We also define a set which contains the indices of the nodes at which players will find themselves to play their mixed strategies as:

$$\mathcal{N}_{\mathcal{O}} = \{1, 2, ...d\} \tag{3.4.1}$$

where d is the cardinality of our decision tree.

Extensive-Form representation of the dynamic TDD assignment n-player game of complete information

1. Set of players $\mathcal{N} = \{1, ...n\}$.
2. A collection set of pure actions $\{A_1, A_2, ...A_n\}$
3. A set of payoff functions $\{v_i(\cdot)\}_{i=1}^{n}$, each assigning a payoff values to each combination of chosen actions, that is $v_i = A_1 \times A_2 .... \times A_n \rightarrow$ (3.2.4).
4. Orders of move $\mathcal{M} = \{P_{i_t}\}_{i,t=1}^{n}$, where $i \in \mathcal{N}$ and $1 \leq t \leq n$ specifies player i's turn.
5. Actions of players when they can move $A_i = \{Uplink, Downlink\}$, for $i = 1, ..., n$.
6. A collection of information sets $\mathcal{K} = \{K_1, ..., K_n\}$, where $K_i = \{k_{i,j}\}$ for $i \in \mathcal{N}$ and $j \in \mathcal{N}_{\mathcal{O}}$ specifies the node at which player i is playing.
7. A collection set of simplexes $\{\Delta A_1, \Delta A_2, ..., \Delta A_n\}$, where $\sigma(i, k_{i,j}) \in \Delta A_i$, for $i \in \mathcal{N}$ and $j \in \mathcal{N}_{\mathcal{O}}$ is the information set dependent mixed strategy of player i, $\sigma(i, k_{i,j}) = \{\sigma_{k_{i,j}}(Uplink), \sigma_{k_{i,j}}(Downlink)\}$ and $\sigma_{k_{i,j}}(Uplink) + \sigma_{k_{i,j}}(Downlink) = 1$.

As we did in the previous cases, we now proceed to look for its solution by starting from a simple 2-player game.

3.4.1 Solution of the 2-player dynamic TDD assignment game of complete information which unfolds over time

We know consider the simplest case in which there are only 2 players and both of them have a action set $A_i = \{Uplink, Downlink\}$, for $i = 1, 2$, and at each information set where player 1 plays, he can mix by using the mixed strategies $\sigma(1, k_{1,1}) = \{q_1, 1 - q_1\}$ and player 2 plays with $\sigma(2, k_{2,2}) = \{p_2, 1-p_2\}$ and $\sigma(2, k_{2,3}) = \{p_3, 1-p_3\}$. By using the previous notations we are imposing that player 1 plays at node 1 and player 2 at nodes 2 and 3, respectively. First of all, we proceed by giving its extensive-form representation and then we will represent it by exploiting the decision tree representation. Finally, we will develop a method to find its Nash equilibrium.

Extensive-Form representation of the dynamic TDD assignment 2-player game of complete information

1. Set of players $\mathcal{N} = \{1, 2\}$.
2. A collection set of pure actions $\{A_1, A_2\}$
3. A set of payoff functions $\{v_i(\cdot)\}_{i=1}^2$, each assigning a payoff values to each combination of chosen actions, that is $v_i = A_1 \times A_2 \rightarrow$ (3.2.4).
4. Orders of move $\mathcal{M} = \{P_{1_1}, P_{2_2}\}$, which means player 1 and player 2 find themselves at level 1 and level 2 of the decision tree, respectively.
5. Actions of players when they can move $A_i = \{Uplink, Downlink\}$, for $i = 1, 2$.
6. A collection of information sets $\mathcal{K} = \{K_1, K_2\}$, where $K_1 = \{k_{1,1}\}$ and $K_2 = \{k_{2,2}, k_{2,3}\}$.
7. A collection set of simplexes $\{\Delta A_1, \Delta A_2\}$, where $\sigma(1, k_{1,1}) = \{q_1, 1 - q_1\}$ for $0 \leq q \leq 1$ is an element of ΔA_1 and $\sigma(2, k_{2,2}) = \{p_2, 1 - p_2\}$ and $\sigma(2, k_{2,3}) = \{p_3, 1 - p_3\}$ for $0 \leq p_2, p_3 \leq 1$ are the elements of ΔA_2.

Note that $k_{i,j} \in K_i$, for $i \in \mathcal{N}$ and $j \in \mathcal{N}_{\mathcal{O}}$ contains all the information which brought player i at the node j. We again use the notation $P^i_{a_1,a_2}$, where $i = 1, 2$, $a_1 \in A_1$ and $a_2 \in A_2$ to indicates the player i's action dependent payoff and its decision tree can be represented as shown in Fig 3.13.

It may be worth mentioning that as we are playing a game of complete information, the previous decision tree can be drawn by each player even before the game starts. This is possible because everybody knows the moves and levels at which all the players are, as everything is of common knowledge. Solving this kind game can be complicated, what can be done is that every player can solve sub-games starting from their level for each node that belongs to their level. The sub-games will be made with lesser nodes than the game presented in the decision tree 3.13, thus their complexity will be definitely reduced. To proceed, what can be used is the so called the *backward induction* method. It exploits the fact that player 2 who immediately precede the terminal nodes, has all the information available for himself. Therefore, he can directly choose the optimal action for himself at both sub-games, starting from nodes 2 and 3, respectively. After that, player 1 finds himself at node 1 and he has two actions from which to choose. Moreover, he also knows what player 2 would do and the payoffs where he may end up. Therefore, he can also make the best decision for himself as he knows perfectly what happens in the nodes that succeed his level. In order to achieve the equilibrium, these kind of games require that at each level players are sequentially rational. Let's recall that we are strictly interested in players playing only mixed strategies, $\sigma(1, k_{1,1}) = \{q_1, 1-q_1\}$ and $\sigma(2, k_{2,2}) = \{p_2, 1-p_2\}$,

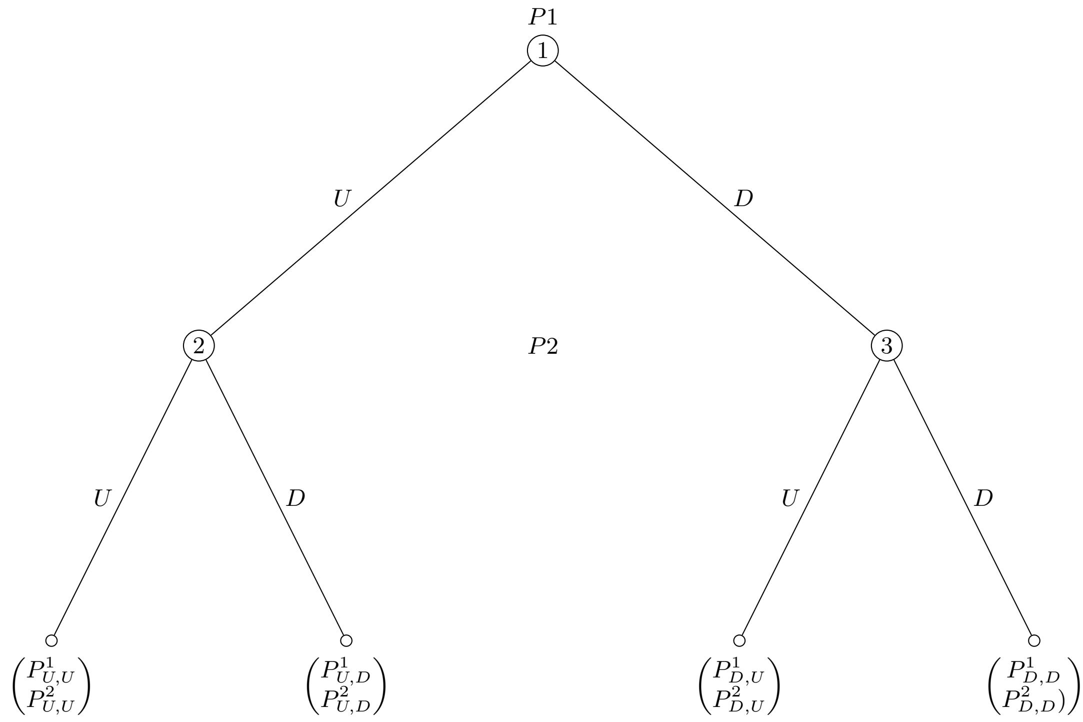

Figure 3.13: Decision tree of the 2-player game dynamic game of complete information.

$\sigma(2, k_{2,3}) = \{p_3, 1 - p_3\}$ for player 1 and player 2, respectively. The way how our dynamic game is solved, the values of q_1 p_2 and p_3 assumes only the values 0 or 1. Therefore, we have ended up in a case in which players' mixed strategies tends to become pure strategies. To get a graphical idea of how the game is solved, its solution is presented in Fig. 3.14-3.16.

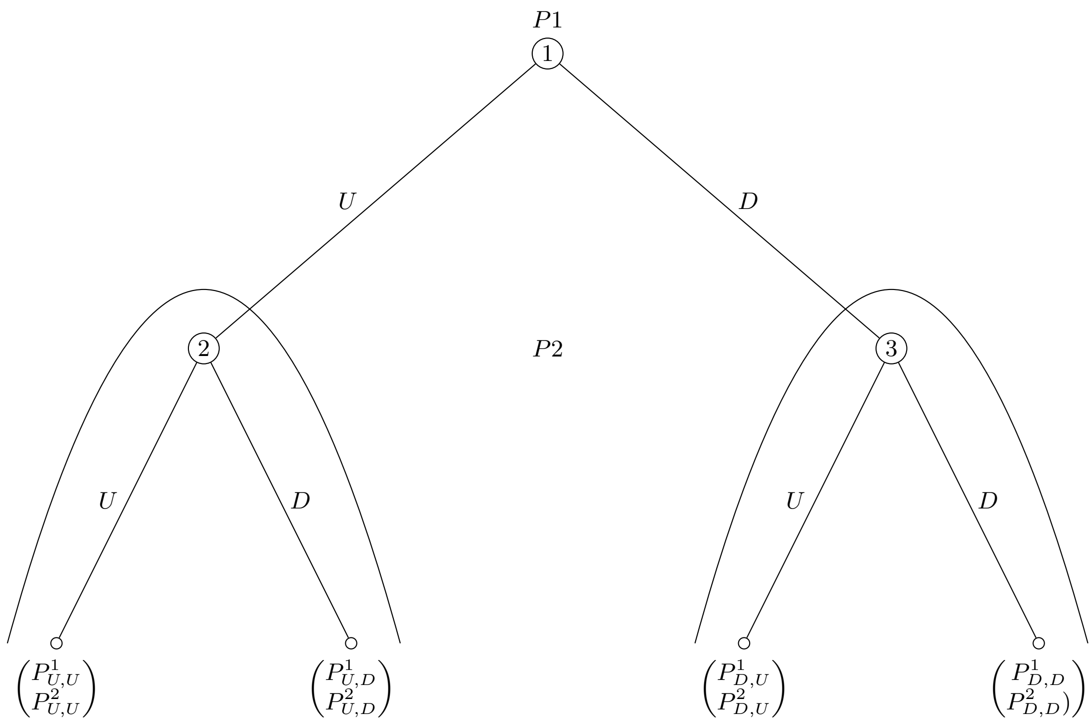

Figure 3.14: Step 1: player 2 solves two sub-games for himself starting from nodes 2 and 3, respectively.

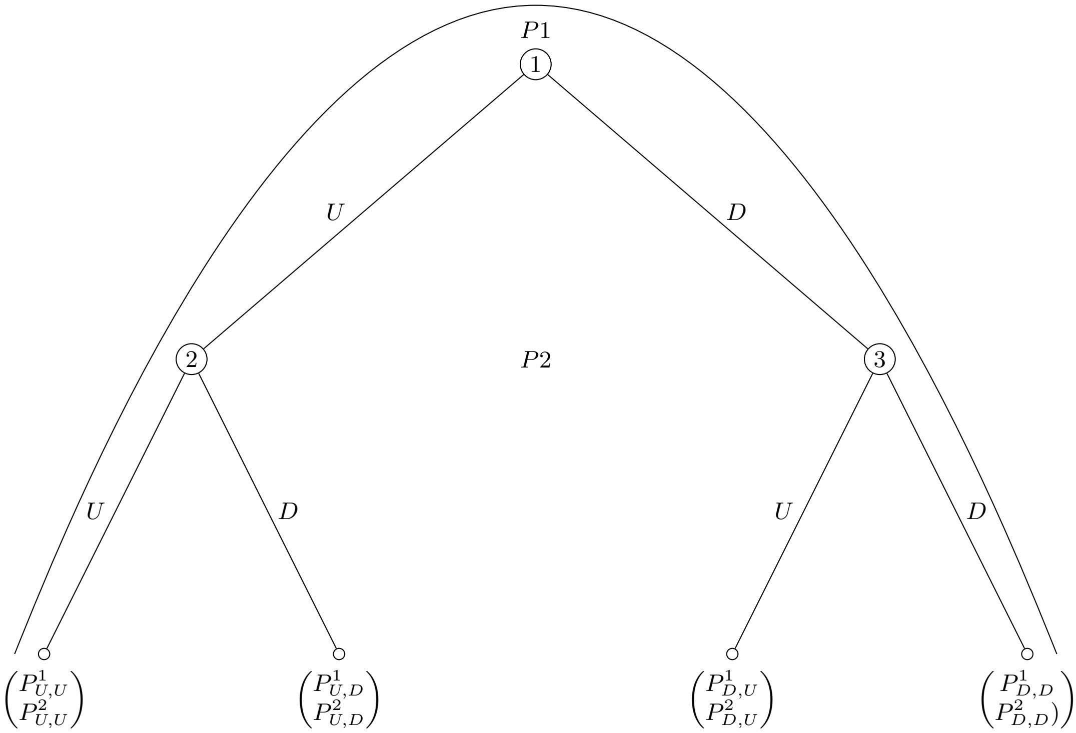

Figure 3.15: Step 2: player 1 solves his sub-game starting from node 1, which also corresponds to be the whole game.

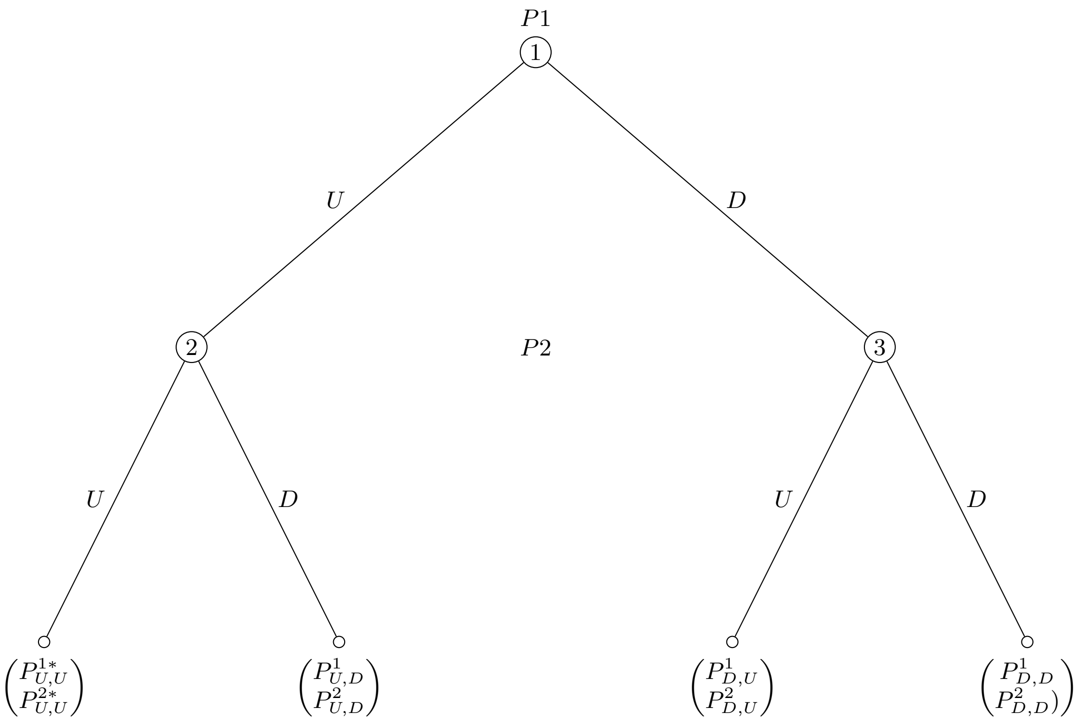

Figure 3.16: The decision which emphasize the Nash equilibrium of our 2-player game.

As it can be seen in Fig. 3.14, player 2 solves 2 sub-games starting from the nodes 2 and 3, respectively. Suppose that at node 2, he discovers that the payoff $P^2_{U,U}$ is greater than $P^2_{U,D}$ and at node 3, $P^2_{D,U}$ is greater than $P^2_{D,D}$. Therefore, player 2 chooses at the nodes 2 and 3 the mixed strategies $p^*_2 = 1$ and $p^*_3 = 1$, respectively. It is shown in Fig. 3.15, the only sub-game solved by player 1 at node 1 and solution of his sub-game corresponds also to the actual solution of the game. Player 1 knows that at node 2 and 3 player 2 chooses Uplink. Now, player 1 has to evaluate between the payoffs $P^1_{U,U}$ and $P_{D,U}$ and he should choose the move which gives him the highest payoff. Suppose that $P^1_{U,U}$ is greater than $P_{D,U}$, then player 1 chooses the mixed strategy $q_1^* = 1$. It may also present the case at step 1 or at step 2 in which players may not be able to decide which action to play if there are payoffs which are equal. This question can be answered by the following proposition:

Proposition 2. *Any finite game of perfect information has a backward induction solution that is sequentially rational. Furthermore, if no two terminal nodes prescribe the same payoffs to any player then the backward induction solution is unique.*

Thus for a solution to be unique, there shouldn't be equal payoffs. The decision taken by player 1 and 2, shown in Fig. 3.14-3.15 are called the sub-game perfect Nash equilibrium. The path which brings us to the end node which contains the payoffs $P^{1*}_{U,U}$ and $P^{2*}_{U,U}$ are called the Nash equilibrium path.

3.4.2 Solution of the 3-player dynamic TDD assignment game of complete information which unfolds over time

We now extend the previous game by considering the case of 3 players, where everyone has a action set $A_i = \{Uplink, Downlink\}$, for $i = 1, 2, 3$ and their mixed strategies are $\sigma(1, k_{1,1}) = \{q_1, 1 - q_1\}$, $\sigma(2, k_{2,2}) = \{p_2, 1 - p_2\}$, $\sigma(2, k_{2,3}) = \{p_3, 1 - p_3\}$, $\sigma(3, k_{3,4}) = \{r_4, 1 - r_4\}$,$\sigma(3, k_{3,5}) = \{r_5, 1 - r_5\}$ and $\sigma(3, k_{3,6}) = \{r_6, 1 - r_6\}$. As we proceeded in the 2-player case, we start by giving its extensive form representation and then we will represent it by using the decision tree representation. Finally, we will develop a solution method to solve our game and achieve the equilibrium.

Extensive-Form representation of the dynamic TDD assignment 3-player game of complete information

1. Set of players $\mathcal{N} = \{1, 2, 3\}$.
2. A collection set of pure actions $\{A_1, A_2, A_3\}$
3. A set of payoff functions $\{v_i(\cdot)\}_{i=1}^3$, each assigning a payoff values to each combination of chosen actions, that is $v_i = A_1 \times A_2 \times A_3 \rightarrow$ (3.2.4).
4. Orders of move $\mathcal{M} = \{P_{1_1}, P_{2_2}, P_{3_3}\}$, which specifically means player 1 is at level one, player 2 is at level 2 and player 3 is at level 3.
5. Actions of players when they can move $A_i = \{Uplink, Downlink\}$, for $i = 1, 2, 3$.
6. A collection of information sets $\mathcal{K} = \{K_1, K_2, K_2\}$, where $K_1 = \{k_{1,1}\}$ and $K_2 = \{k_{2,2}, k_{2,3}\}$ and $K_3 = \{k_{3,4}, k_{3,5}, k_{3,6}\}$.
7. A collection set of simplexes $\{\Delta A_1, \Delta A_2, \Delta A_3\}$, where $\sigma(1, k_{1,1}) = \{q_1, 1 - q_1\}$ for $0 \le q \le 1$ is an element of ΔA_1, $\sigma(2, k_{2,2}) = \{p_2, 1-p_2\}$ and $\sigma(2, k_{2,3}) = \{p_3, 1-p_3\}$ for $0 \le p_2, p_3 \le 1$ are the elements of ΔA_2 and $\sigma(3, k_{3,4}) = \{p_4, 1 - p_4\}, \sigma(3, k_{3,5}) = \{p_5, 1 - p_5\}$, $\sigma(3, k_{3,6}) = \{p_6, 1 - p_6\}$ and $\sigma(3, k_{3,7}) = \{p_7, 1 - p_7\}$ are the elements of ΔA_3.

We can represent it in the decision tree form as shown in Fig. 3.17 and to find the Nash equilibrium path we can adopt the same procedure we used in the 2-player case. Let's recall that as everything is of common knowledge, every player is able to draw the decision tree of this game and can analyze it in order to choose his best move. In order to solve the game, again the backward induction procedure can be exploited under the hypothesis that all the players are sequentially rational. To proceed, the first move is made by player 3 who has all the information available for himself and can directly choose the optimum mixed strategy for all the sub-games seen from his level at nodes the 4,5,6 and 7. These sub-games are shown in Fig. 3.18. Suppose that by evaluating directly the payoffs, player 3 notices that at node 4 $P^3_{U,U,U}$ is greater than $P^3_{U,U,D}$, at node 5 $P^3_{U,D,U}$ is greater than $P^3_{U,D,D}$, at node 6 $P^3_{D,U,U}$ is greater than $P^3_{D,U,D}$ and at node 7 $P^3_{D,D,U}$ is greater than $P^3_{D,D,D}$. At this point, he will choose the mixed strategies $r^*_4 = 1, r^*_5 = 1, r^*_6 = 1$ and $r^*_7 = 1$. Now by following the backward induction, player 2 finds himself at the nodes 2 and 4 and he knows what player 3 will choose and the payoffs where he may end up. Suppose that $P^2_{U,U,U}$ is greater than $P^2_{U,D,U}$ and at node 3 $P^2_{D,U,U}$ is greater than $P^2_{D,D,U}$ and at this point player 2 will choose the mixed strategies $p^*_2 = 1$ and $p^*_3 = 1$. Now at node 1, player 1 has to choose his solely mixed strategy and his decision will yield the final outcome of our game. Suppose that by evaluating he notices that $P^1_{U,U,U}$ is greater than $P^1_{D,U,U}$, thus he chooses the mixed strategy $q^*_1 = 1$. All these equilibriums of our sub-games are called the sub-game Nash equilibriums and the path which brings player to the end node which contains the payoffs $\{P^1_{U,U,U}, P^2_{U,U,U}, P^3_{U,U,U}\}$ is called the Nash equilibrium path which is the actual equilibrium of our game.

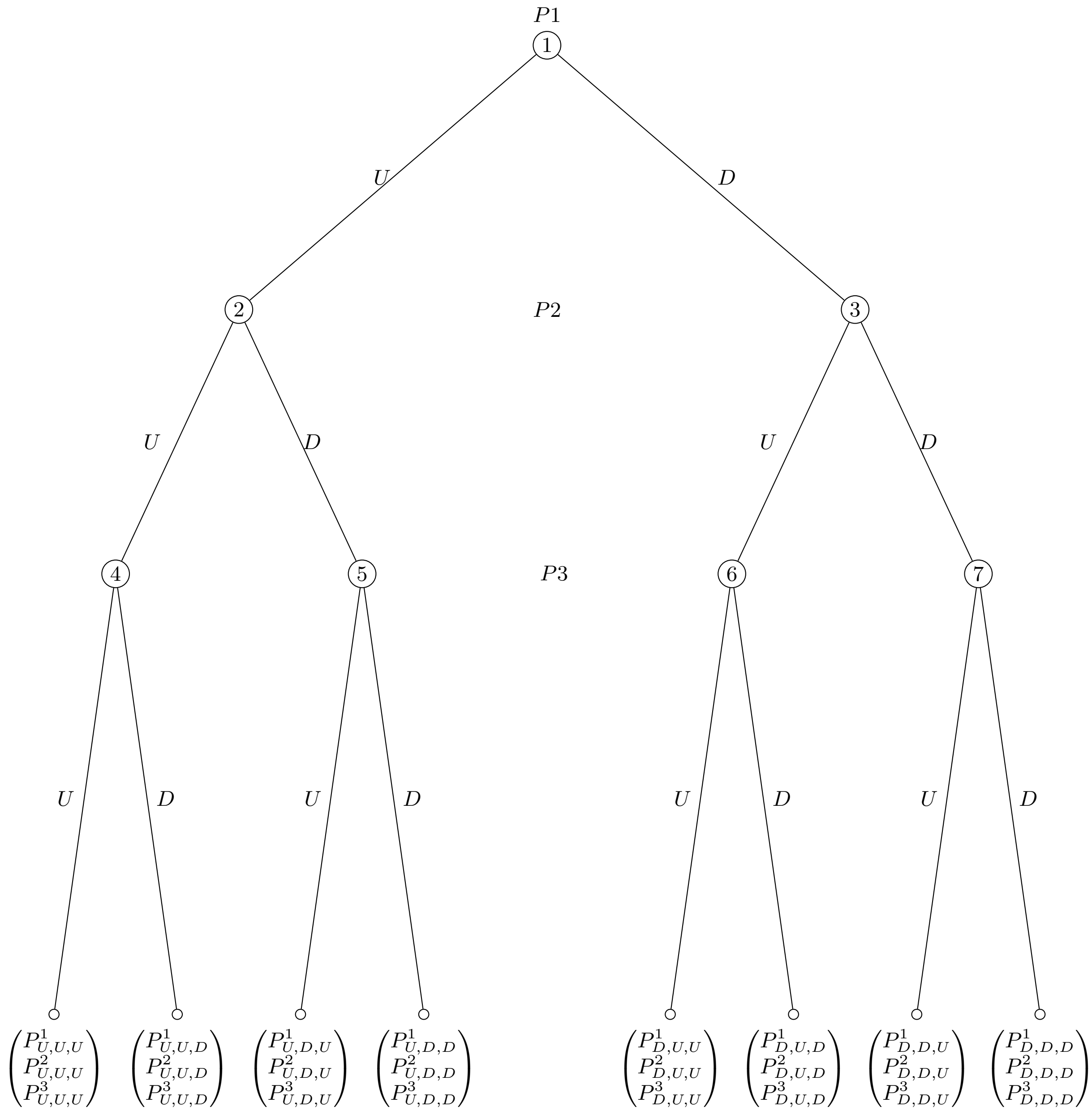

Figure 3.17: Decision tree of the 3-player dynamic TDD assignment game of complete information which unfold over time.

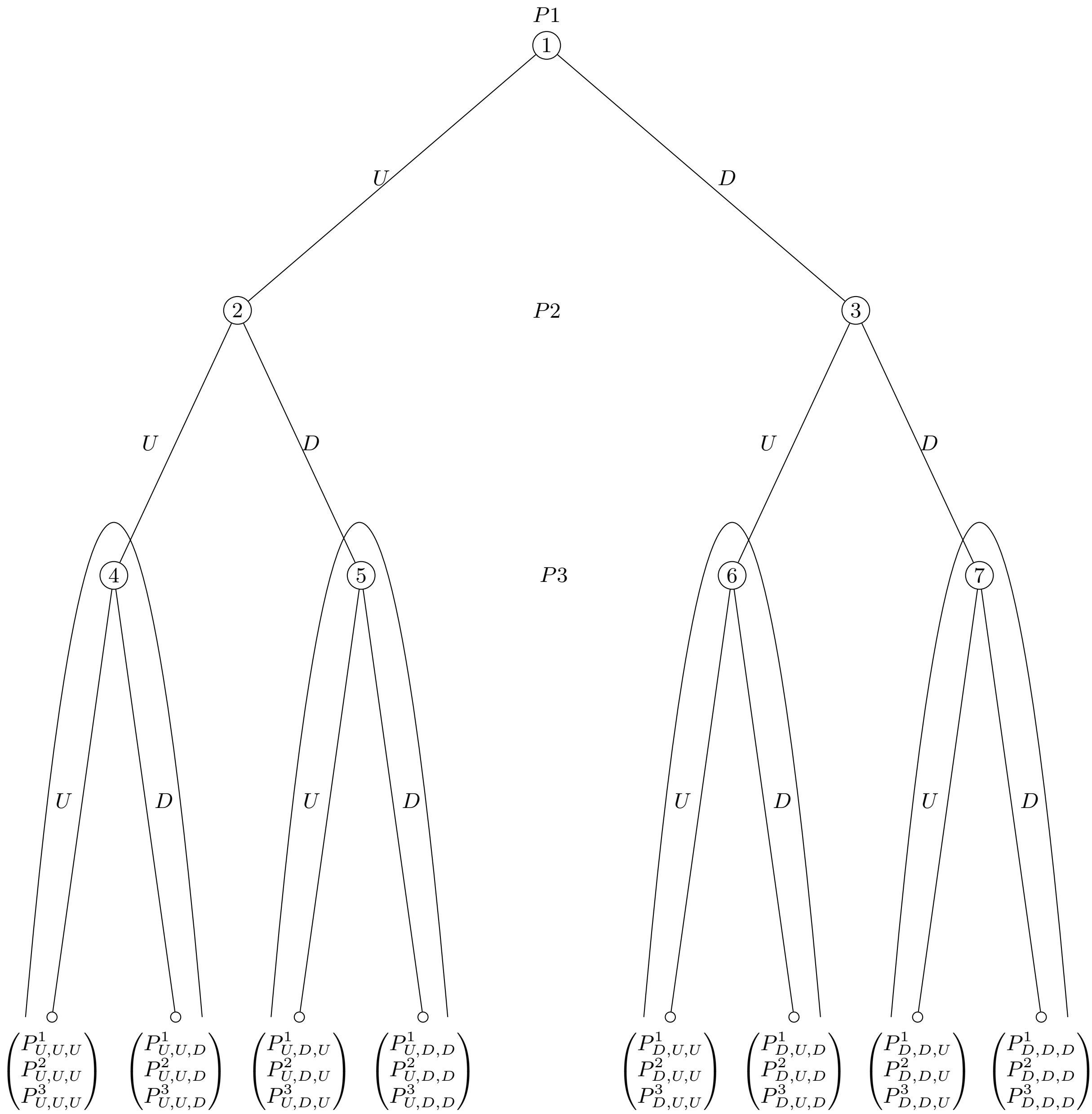

Figure 3.18: Step 1: Player 3 solves all the sub-games he sees from his level at the nodes 4,5,6 and 7.

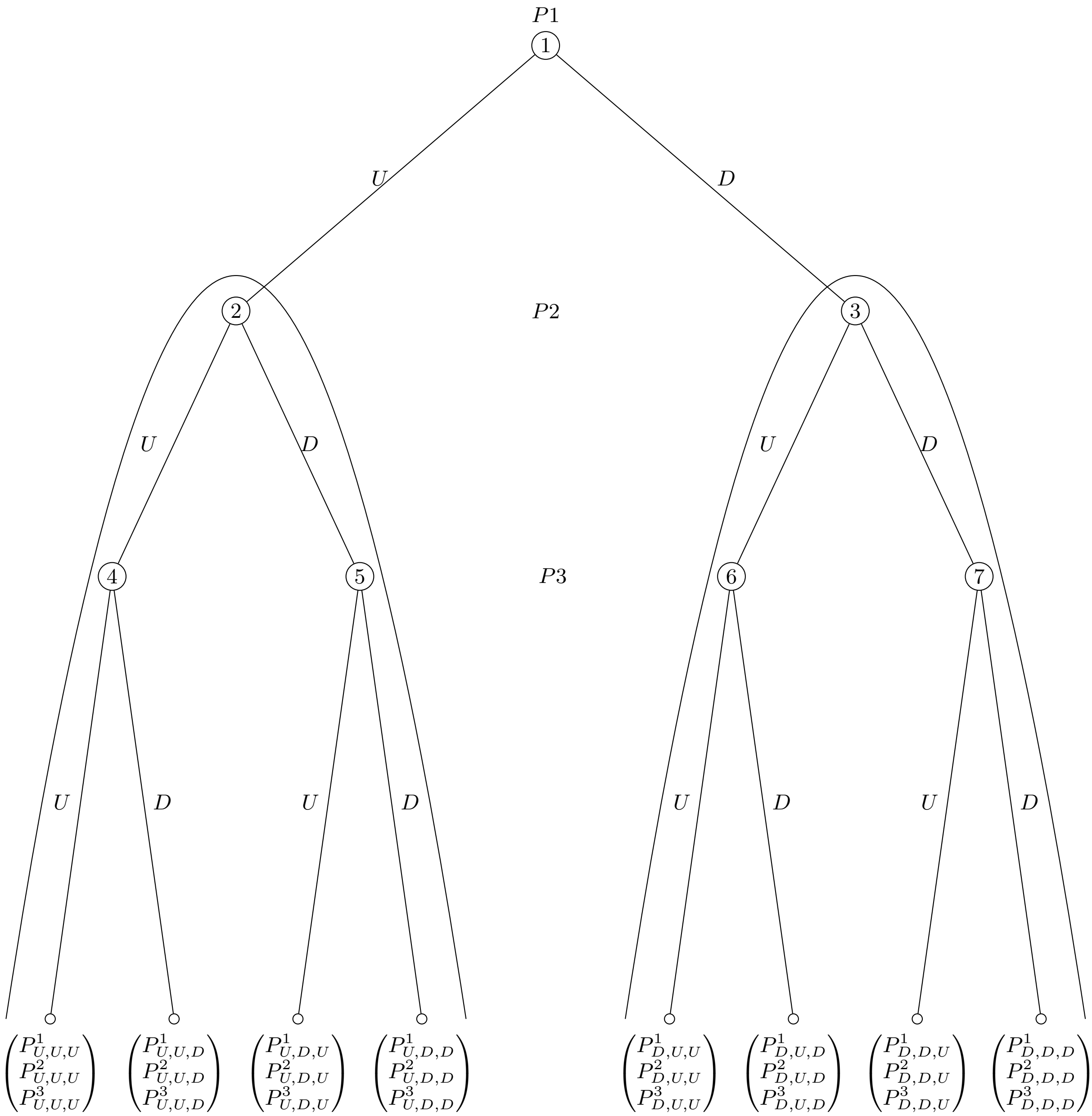

Figure 3.19: Step 2: Player 2 solves all the sub-games he sees from his level at the nodes 2 and 3.

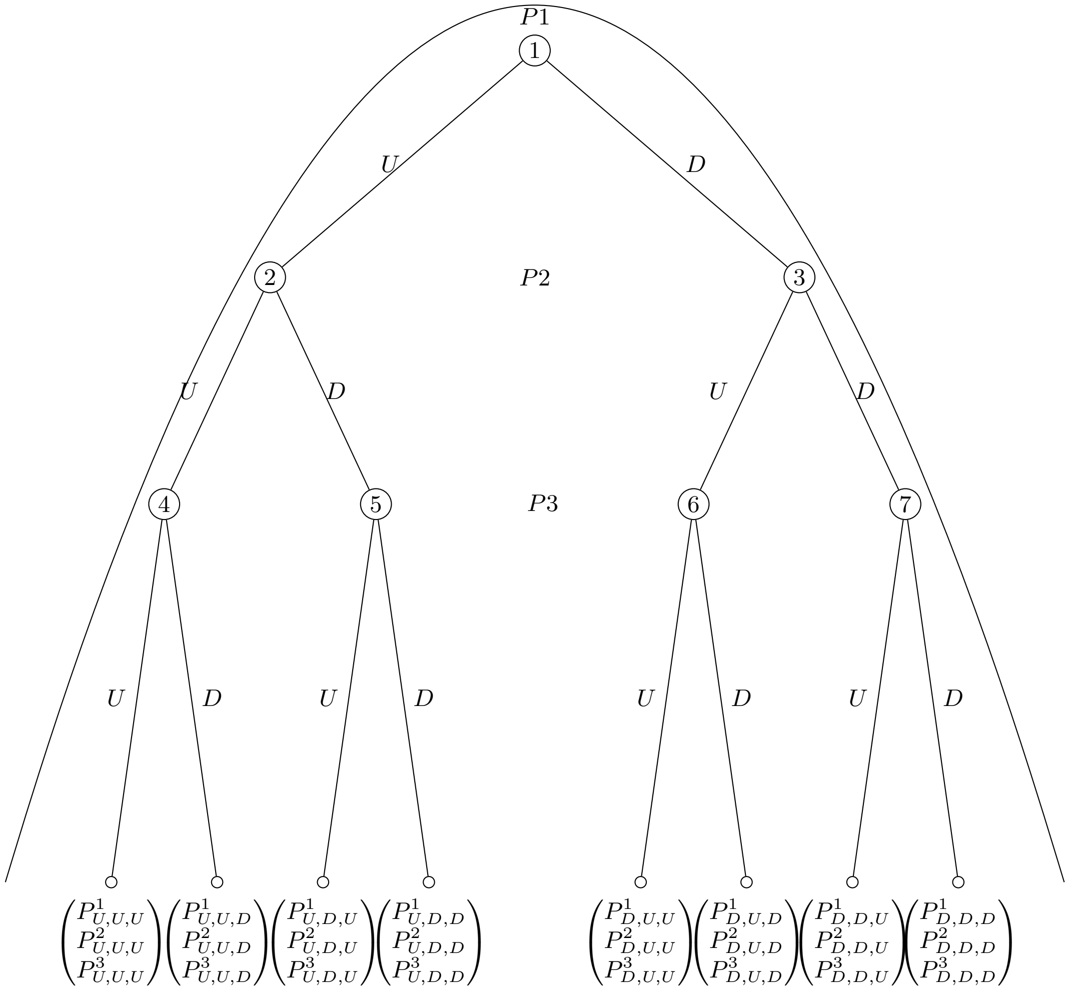

Figure 3.20: Step 3: Player 1 solves the only sub-games he sees from his level at the node 1.

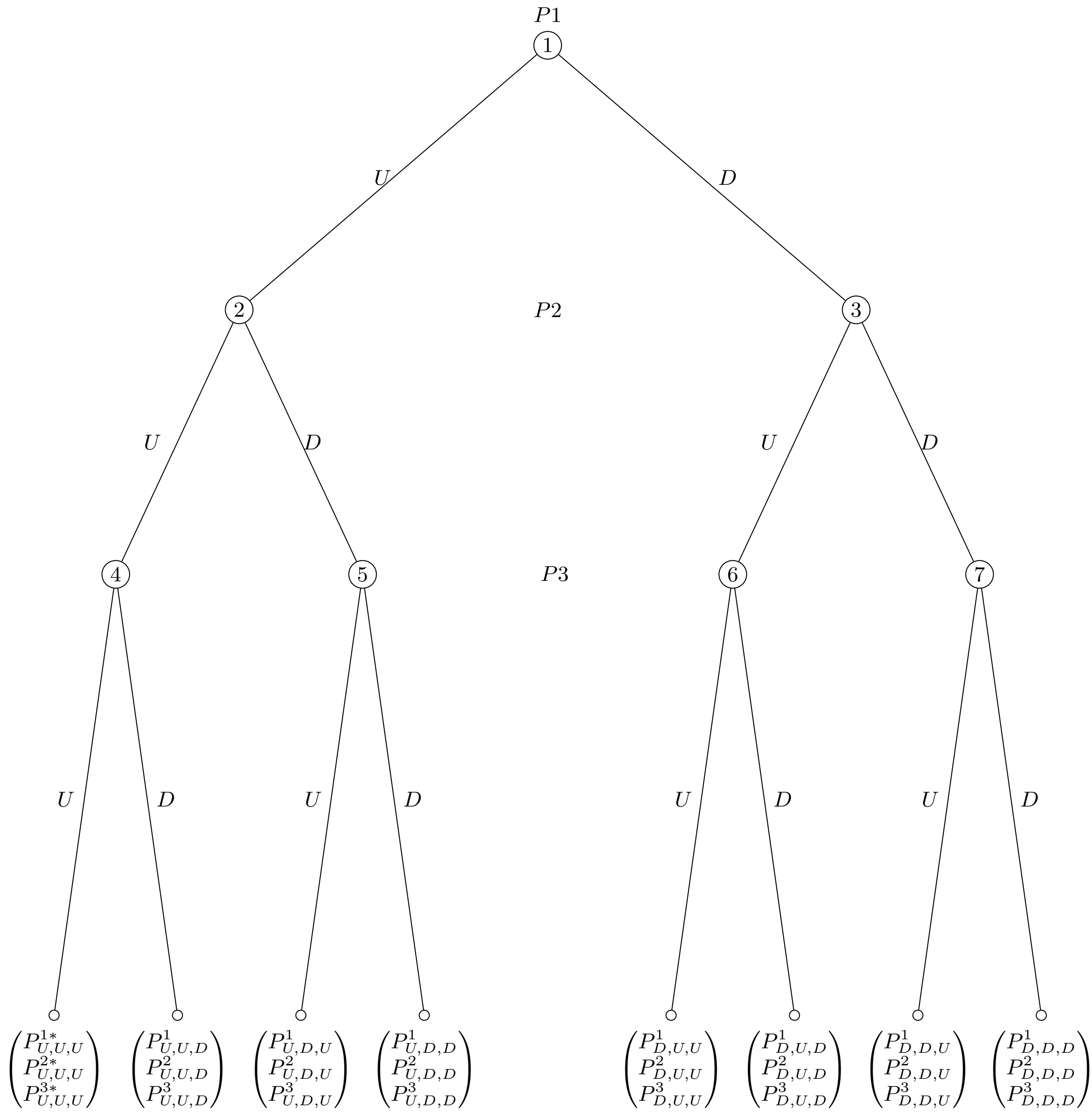

Figure 3.21: The Decision tree illustrate the Nash equilibrium for the 3-player case.

Chapter 4

Simulation Results

In the previous chapter we faced the problem of sum rate maximization from game theoretic point of view. We analyzed various cases by limiting our only to two and three players' scenarios. In this chapter we report the simulation results that we obtained to evaluate the performances and the usefulness of the solution method we developed to achieve the Nash equilibrium. All the simulation that have been done, we considered only the 2-player scenario as this allowed us to write simple scripts in MATLAB. All the scripts can be easily extended also to the 3-player case.

As we limited ourself only to the case of 2-player, first of all we report in Fig. 4.1 how the mean rate per player varies as the distance between the BSs and their solely associated UEs increase. The blue curve represents how the mean rate varies when both players choose the optimal moves obtained by the Multi-start algorithm. As it has already been said in Chapter 2, Multi-start commits to find all the possible optimums and returns also a vector which contains the coordinates of a local optimum which can be declared as the global optimum. If players are willing to choose pure strategies, then if any player chooses a strategy different from the one found by Multi-start, then the mean rate can not be bigger than the case in which everybody choose the optimum pure strategies found with Multi-start. In our simulation, the optimum pure strategies found by our algorithm were Uplink for both players. It can be clearly noticed by the red curve, when player 1 chooses Uplink and Player 2 chooses Downlink, that the mean rate is evidently lower from the optimum case.

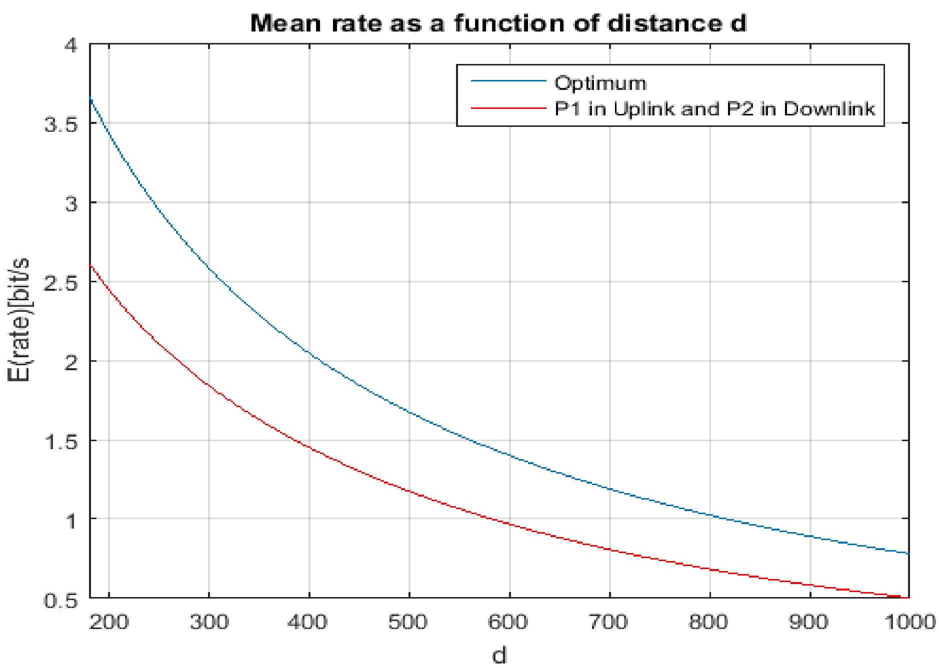

Figure 4.1: Mean rate as a function of distance between BSs and their UEs.

As we learned from the previous analysis, global optimum found by Multi-start are the only pure strategies that players should choose in order to obtain the maximum mean rate. When we considered the 2-player case in which all the players strictly choose only mixed strategies, thing didn't work exactly how we imagined. We first considered the case in which the standard values of P and Q were used. In the simulation, we considered the normalized version in which P was normalized to 1 and Q to 0.025. By solving the equations (3.2.5)-(3.2.8), we found that in this case the Nash equilibrium doesn't exists as the values of optimum mixed strategies don't belong to the interval 0-1. The non existence of equilibrium is due to the fact that when try to put equations (3.2.5)-(3.2.6) equal, which would give use the optimum value of q which could make players 1 indifferent of his choice results negative. This is because the terms one the left and the right hand side of the equality sign are totally unbalanced and when we try to subtract and solve it for q, its results -4.46. By analyzing in depth, we found that the equilibrium exists in two cases: 1) The value of P starts to be comparable to Q, 2) The SINR at the BSs in Uplink is reduced of factor 100.

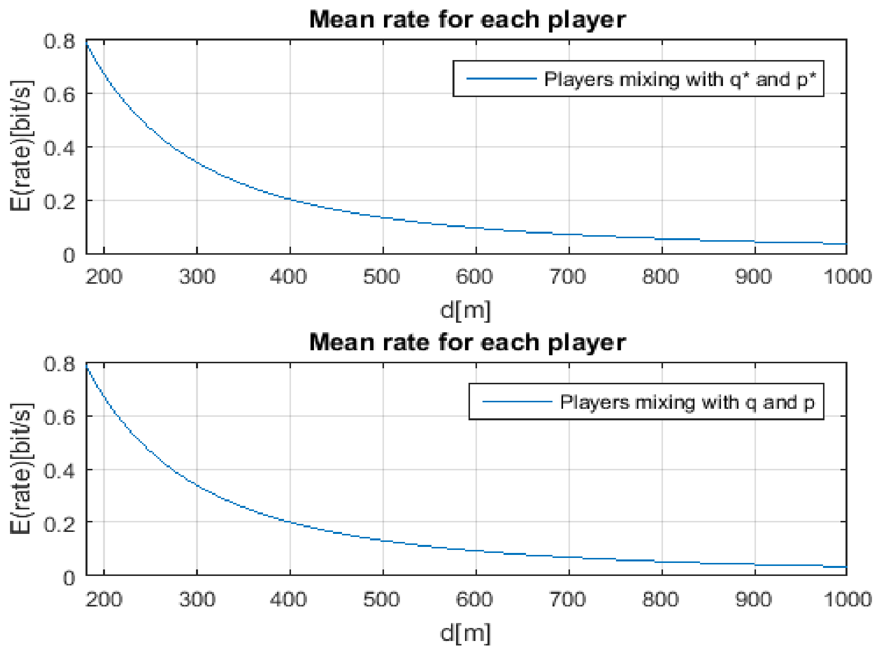

Figure 4.2: Mean rate as a function of distance between BSs and their UEs in the case in which P is comparable to Q. We report the two different cases for the optimal values and the non optimal values of mixed strategies.

In Fig. 4.2 we report the mean rate as function of distance for the first case in which P is comparable to Q, for q and p which are the mixed strategies of player 1 and player 2, respectively. In the upper plot, its is shown how the mean rate varies when both players choose the optimal mixed strategies obtained by solving the equation (3.2.5)-(3.2.8). In the lower plot, the behavior of mean rate for non optimum values of P and Q is reported. It can be clearly seen that even if the players choose non optimum mixed strategies, the mean rate results the same as on average the payoff is same because P is comparable to Q. Which means, independent of players' mixed strategy choice, the mean rate doesn't changes. For the second case, in which the SINR at the BSs in Uplink has been reduced of factor 100, the results are reported in Fig. 4.3. It can be clearly noticed that when the players choose the optimal mixed strategies, the mean rate results greater. As for the decay, it is visible that even if the mean rate is greater in the case of optimal mixed strategies, they both decay more or less in the same way.

As for the case in which P=1 and Q=0.025, we analyzed the behavior of mean rate for different values of q and p. We know from the analysis of Fig. 4.1 that the mean rate is

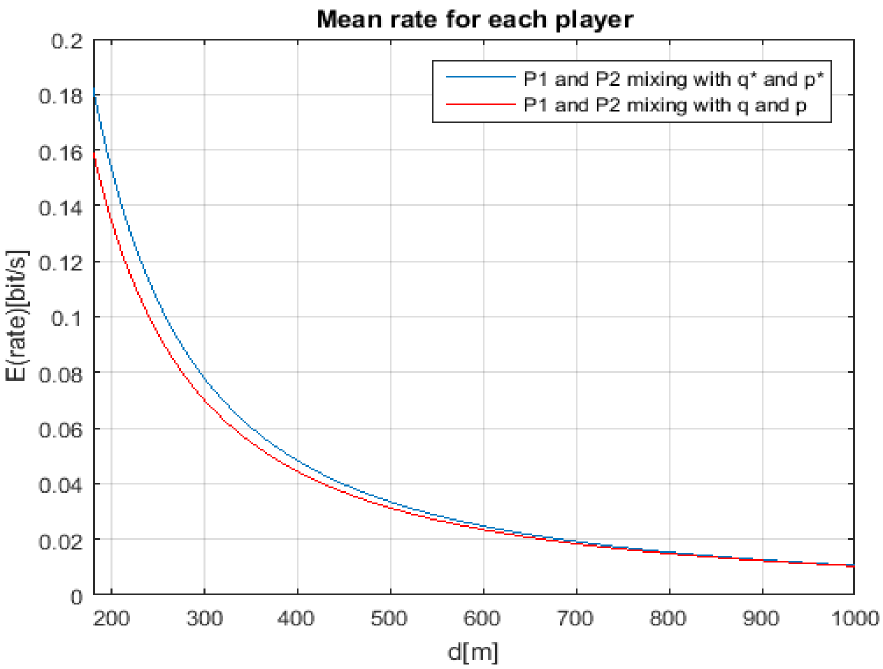

Figure 4.3: Mean rate as a function of distance between BSs and their UEs.

higher from any case when both players choose mixed strategy Uplink, which specifically means $q = 1$ and $p = 1$. Thus we analyzed the case where $q = 0.3$ and $p = 0.3$ with the case in which both variables assume value 1. The results are reported in Fig. 4.4, it is evident from the red curve that in the case $q, p = 0.3$, mean rate is very far from the ideal as the probability of choosing Downlink is strictly higher. In the case in which $p, q = 1$, we tend to approach the ideal case found by the Multi-start algorithm and definitely the mean rate is considerably higher.

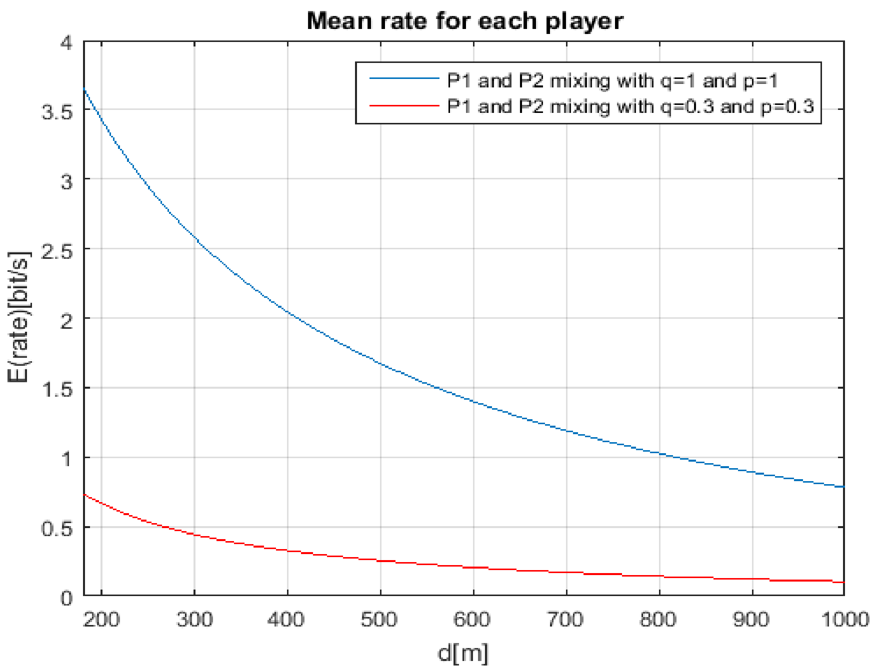

Figure 4.4: Mean rate as a function of distance between BSs and their UEs.

We also analyzed the case with standard values of P and Q in which player 1 used the mixed strategy $q = 0.5$ and $q = 1$ while player 2 was fixed with mixed strategy Uplink. The same thing has been done for player 2 and we concluded that for both cases the mean rate tends to increase as the probability of choosing the mixed strategy Uplink increases. The results are shown in Fig.

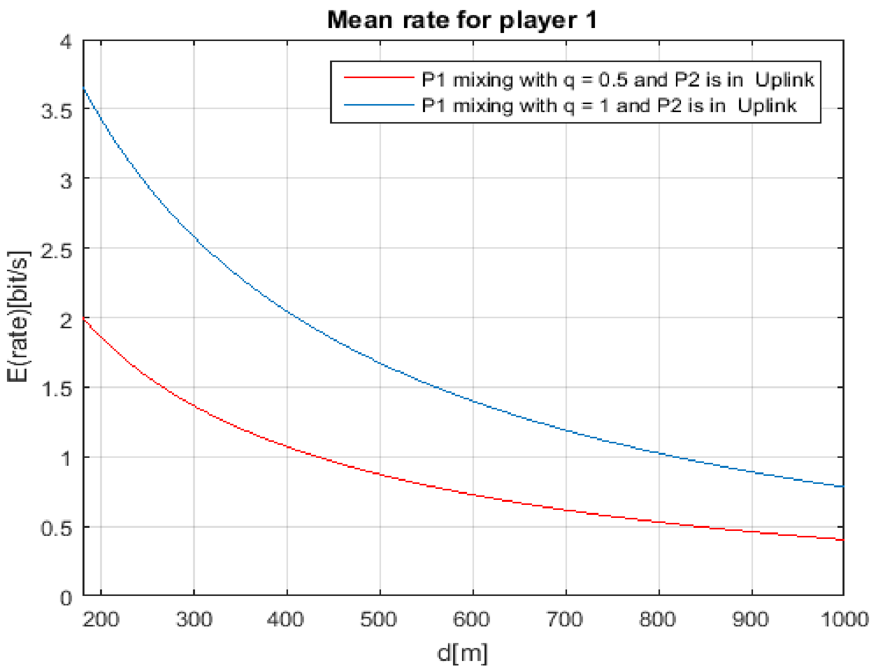

Figure 4.5: Mean rate of player 1 with different mixed strategies when player 2 plays Uplink.

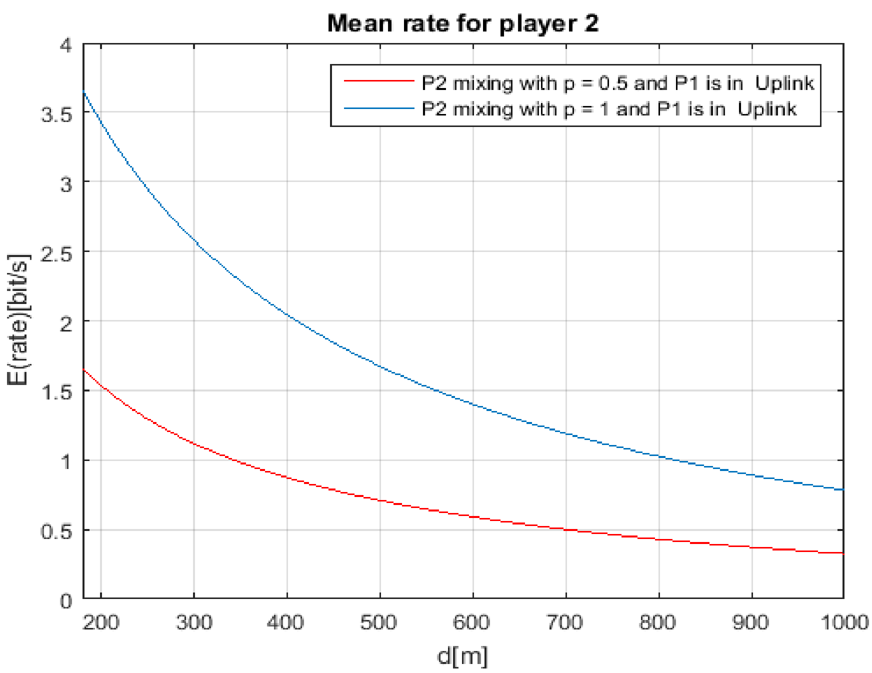

Figure 4.6: Mean rate of player 2 with different mixed strategies when player 1 plays Uplink.

We also analyzed the case in which we evaluated the mean rate for player 1 as a function of different mixed strategies while his opponent was fixed in Uplink. Then we did the same thing for player 2 and the results are reported in Fig. 4.7-4.8.

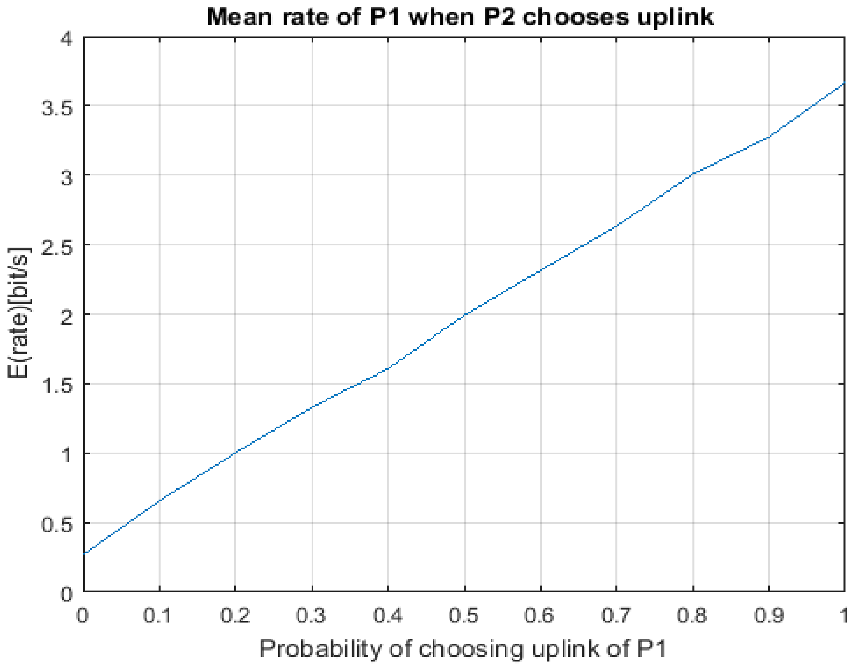

Figure 4.7: Mean rate of player 1 as a function of mixed strategies when player 2 is fixed in Uplink.

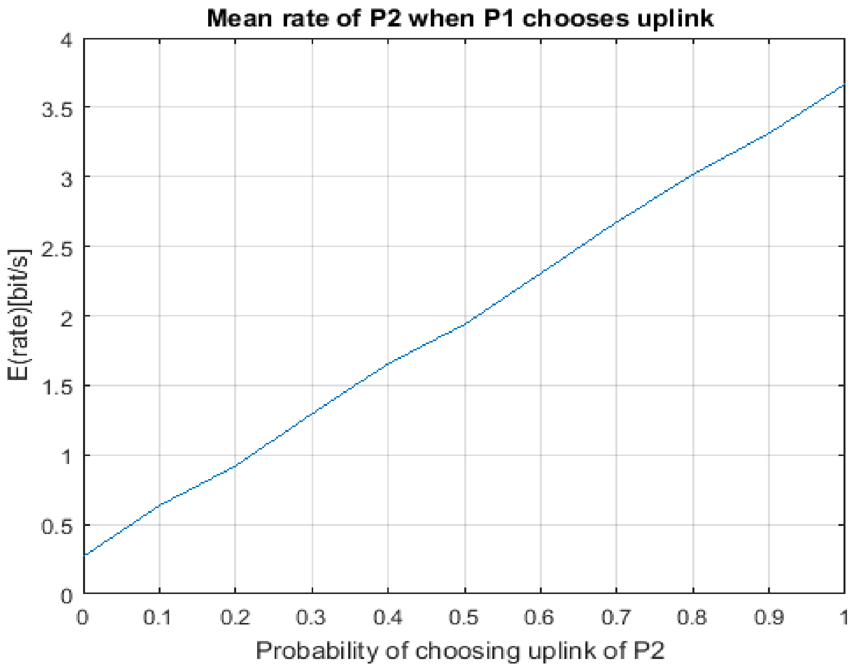

Figure 4.8: Mean rate of player 2 as a function of mixed strategies when player 1 is fixed in Uplink.

It is evident for both cases that the mean rate for both players tends to increase as the probability of playing Uplink increases.

Chapter 5

Conclusions

This thesis examined the problem of sum rate maximization for a most general scenario consisting of N BSs and M UEs, in which al the UEs were associated to their nearest BS. First of all, after a brief introduction about 5G given in Chapter 1, we formulated our problem of sum rate maximization in mathematical terms in Chapter 2. In order to solve it, we discretized the continuous variable X_n to 0 and 1, which means instead of using a fraction of TDD frame to transmit in Uplink or Downlink, we reserved the whole frame just for one type of transmission. Then under some constraints regarding the variable X_n and the maximum power that the BSs and UEs could used, we wrote the Lagrangian function of our maximization problem. We looked up for its solution by exploiting the so-called Multi-start algorithm.

The same problem then has been formulated by exploiting the mathematical tools offered by the game theory, in which BSs acted as players of our game and to maximized their sum rate theu choose optimal mixed strategies based on their knowledge and their degree of freedom. As a first step, we formulated our dynamic TDD assignment game for the case of 2 player as a game of complete information in which everything was of common knowledge to everyone. The hypothesis of everything known by everyone is strong and it's hardly encountered in the reality. But, it allowed us to shift our sum rate maximization game into simple equations like (3.2.9)-(3.2.10) that could be solved very easily to find the optimal mixed strategies for both players. Then we extended our game to the case of 3 players, which led us to more difficult expressions like (3.2.24)-(3.2.26). The concept to find the Nash equilibrium was the same but by analyzing these equations we noticed that there are non linear. From which, it can be concluded that for the n-player case we end up having n equation with degree $n-1$. Which means to obtain the equilibrium we have a system of n equations with each equation having degree $n-1$ and we will end up having multiple solution from which only the ones which belongs to the interval 0-1 should be considered.

As a subsequent step, we considered a more realistic scenario in which the information about the game's parameters were of private knowledge among the players and they could be also of different types. In game terms, everyone was characterized by having multiple characteristics thus different payoffs which information we translate into players having different types. As a first step, everybody knew the joint probability distribution used by Nature to select the players' types and nobody knew which were the players' actual types profile. By exploiting the common knowledge and after learning their types, all the players used the conditional probability to improve their knowledge about their opponents. For the 2-player case the solution is rather simple and can be obtained by solving the equations similar to the 2-player game of completer information as expressed in equations (3.3.19)-(3.3.22). Things become more difficult when we try to solve the same game for the 3-player case. The reason lies in the degree of freedom that every player has and the fact that everything was of private knowledge. We were force to introduce the concept of average game which can be used for any number of players in which every player try to build his system of equations under the hypothesis of knowing an estimate of his opponents' mixed strategies and try to find his own optimal mixed strategy. In this step, nobody assures that the players' mixed strategy's estimate is true, but this is all every player can do as we were playing a game of incomplete information.

We analyzed the dynamic TDD assignment game of complete information and then the game of incomplete information in which we put our players' in a strategic scenarios. Which means all the players were obliged to make their moves simultaneously and nobody was allowed to learn the moves of his opponents. In the final case, we also studied a scenario consisting of players in which everybody was allowed to gather information from his opponents who moved earlier. We called this type of game as dynamic game of complete information.To solve these kind of games we used the so-called **backward induction** method.

It may be worth mentioning that all the cases that we analyzed and the solution methods we developed, everything can be extended to the n-player case. The theorem 1 gives us sufficient condition under which there can be the existence of the Nash equilibrium but its doesn't assure us if it always exists. During the simulations, also ended up having a case in which there was no equilibrium and we found that it exists only when P was comparable to Q or the SINR was reduced to factor 100 at the BSs in Uplink.

Bibliography

[1] P. Sharma, "Evolution of mobile wireless communication networks-1g to 5g as well as future prospective of next generation communication network," *International Journal of Computer Science and Mobile Computing*, vol. 2, no. 8, pp. 47–53, 2013.

[2] J. Rodriguez, "Fundamentals of 5g mobile networks-wiley (2015)," *WILEY*, vol. 1, 2015.

[3] M. R. Bhalla and A. V. Bhalla, "Generations of mobile wireless technology: A survey," *International Journal of Computer Applications*, vol. 5, no. 4, 2010.

[4] B. Raaf, W. Zirwas, K.-J. Friederichs, E. Tiirola, M. Laitila, P. Marsch, and R. Wichman, "Vision for beyond 4g broadband radio systems," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2011 IEEE 22nd International Symposium on.* IEEE, 2011, pp. 2369–2373.

[5] R. B. Myerson, "A course in game theory," *Harvard university press*, 2013.

[6] S. Tadelis, "Game theory: an introduction," 2013.

[7] R. Gibbons, "A primer in game theory," 1992.

[8] A. M. Fouladgar, O. Simeone, O. Sahin, P. Popovski, and S. Shamai, "Joint interference alignment and bi-directional scheduling for mimo two-way multi-link networks," in *Communications (ICC), 2015 IEEE International Conference on.* IEEE, 2015, pp. 4126–4131.

[9] S. Lagen, A. Agustin, and J. Vidal, "Joint user scheduling, precoder design, and transmit direction selection in mimo tdd small cell networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 4, pp. 2434–2449, 2017.

[10] D. Zhu, M. Lei, and A. Goldsmith, "Coordinated resource allocation in centralized radio access networks with dynamic downlink/uplink reconfiguration," in *Global Communications Conference (GLOBECOM), 2013 IEEE.* IEEE, 2013, pp. 3625–3630.

[11] A. A. Dowhuszko, O. Tirkkonen, J. Karjalainen, T. Henttonen, and J. Pirskanen, "A decentralized cooperative uplink/downlink adaptation scheme for tdd small cell networks," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on.* IEEE, 2013, pp. 1682–1687.

[12] M. S. ElBamby, M. Bennis, W. Saad, and M. Latva-Aho, "Dynamic uplink-downlink optimization in tdd-based small cell networks," in *Wireless Communications Systems (ISWCS), 2014 11th International Symposium on.* IEEE, 2014, pp. 939–944.

[13] V. Venkatasubramanian, M. Hesse, P. Marsch, and M. Maternia, "On the performance gain of flexible ul/dl tdd with centralized and decentralized resource allocation in dense 5g deployments," in *Personal, Indoor, and Mobile Radio Communication (PIMRC), 2014 IEEE 25th Annual International Symposium on.* IEEE, 2014, pp. 1840–1845.

[14] M. N. Kulkarni, J. G. Andrews, and A. Ghosh, "Performance of dynamic and static tdd in self-backhauled mmwave cellular networks," *arXiv preprint arXiv:1701.07111*, 2017.

[15] X. Ge, S. Tu, G. Mao, C.-X. Wang, and T. Han, “5g ultra-dense cellular networks,” *IEEE Wireless Communications*, vol. 23, no. 1, pp. 72–79, 2016.

[16] A. Gupta and R. K. Jha, “A survey of 5g network: Architecture and emerging technologies,” *IEEE access*, vol. 3, pp. 1206–1232, 2015.

[17] R. . Schwarz, “Introduction to mimo,” vol. 1.

[18] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, “Millimeter wave mobile communications for 5g cellular: It will work!” *IEEE access*, vol. 1, pp. 335–349, 2013.