UNIVERSIDAD POLITECNICA DE VALENCIA

DEPARTAMENTO DE COMUNICACIONES

Thesis for the degree of PhD

Common Radio Resource Management Strategies for Quality of Service Support in Heterogeneous Wireless Networks

Daniel Calabuig Soler

Valencia, February 2010

Supervised by:
Dr. Narcís Cardona Marcet
Dr. José F. Monserrat del Río

Abstract

Nowadays several technologies coexist in the same area composing a heterogeneous system. Moreover, this fact is expected to become more and more accentuated with all new technologies that are currently being standardized. So far, generally users are who select the technology they are going to connect to, either configuring the user equipment or using different equipments. Nevertheless, this approach does not take the maximum advantage of the available resources. To this aim, a new set of strategies is necessary. These strategies must manage the radio resources of all technologies commonly and jointly, and they must ensure the satisfaction of users Quality of Service (QoS).

Following this idea, this Thesis proposes two new algorithms. The first one is a Joint Dynamic Resource Allocation (JDRA) algorithm capable of allocating resources to users and distributing users among technologies at the same time. The algorithm is formulated as a multi-objective minimization problem that is solved using Hopfield Neural Networks (HNNs). HNNs are interesting because they are supposed to find suboptimal solutions in very short times. Nevertheless, actual implementations of HNNs in computers lose this fast response. For that reason, this Thesis analyses the causes and studies possible improvements.

The second algorithm is a Joint Call Admission Control (JCAC) algorithm that admits and rejects users taking all technologies into account at the same time. The main difference with other proposed algorithms is that they decide on call admission in each technology separately. Thus, a previous mechanism is needed to select which technology users are going to connect to. On the other hand, the technique proposed in this Thesis is capable of deciding on call admission in the whole heterogeneous system. Therefore, users are not attached to any technology prior to admission.

As a result, this Thesis paves the way to an efficient resource management in heterogeneous systems with new ideas and techniques. The functions of the Radio Resource Management (RRM) are divided into the JDRA and JCAC algorithms so they could work together.

ABSTRACT

Resumen

Hoy en día existen varias tecnologías que coexisten en una misma zona formando un sistema heterogéneo. Además, este hecho se espera que se vuelva más acentuado con todas las nuevas tecnologías que se están estandarizando actualmente. Hasta ahora, generalmente son los usuarios los que eligen la tecnología a la que se van a conectar, ya sea configurando sus terminales o usando terminales distintos. Sin embargo, esta solución es incapaz de aprovechar al máximo todos los recursos. Para ello es necesario un nuevo conjunto de estrategias. Estas estrategias deben gestionar los recursos radio eléctricos conjuntamente y asegurar la satisfacción de la calidad de servicio de los usuarios.

Siguiendo esta idea, esta Tesis propone dos nuevos algoritmos. El primero es un algoritmo de asignación dinámica de recursos conjunto (JDRA) capaz de asignar recursos a usuarios y de distribuir usuarios entre tecnologías al mismo tiempo. El algoritmo está formulado en términos de un problema de optimización multi-objetivo que se resuelve usando redes neuronales de Hopfield (HNNs). Las HNNs son interesantes ya que se supone que pueden alcanzar soluciones sub-óptimas en cortos periodos de tiempo. Sin embargo, implementaciones reales de las HNNs en ordenadores pierden esta rápida respuesta. Por ello, en esta Tesis se analizan las causas y se estudian posibles mejoras.

El segundo algoritmo es un algoritmo de control de admisión conjunto (JCAC) que admite y rechaza usuarios teniendo en cuenta todas las tecnologías al mismo tiempo. La principal diferencia con otros algoritmos propuestos es que éstos últimos toman las decisiones de admisión en cada tecnología por separado. Por ello, se necesita de algún mecanismo para seleccionar la tecnología a la que los usuarios se van a conectar. Por el contrario, la técnica propuesta en esta Tesis es capaz de tomar decisiones en todo el sistema heterogéneo. Por lo tanto, los usuarios no se enlazan con ninguna tecnología antes de ser admitidos.

Como resultado, esta Tesis prepara el camino hacia una gestión eficiente de recursos en sistemas heterogéneos con nuevas ideas y técnicas. Las funciones de la gestión de recursos radioeléctricos se dividen entre los algoritmos JDRA y JCAC de forma que puedan trabajar los dos juntos.

RESUMEN

Resum

Hui en dia ni hi ha diverses tecnologies que coexistixen en una mateixa zona formant un sistema heterogeni. A més, este fet s'espera que es torne més accentuat amb totes les noves tecnologies que s'estan estandarditzant actualment. Fins ara, generalment són els usuaris els que elegixen la tecnologia a què es van a connectar, ja siga configurant els seus terminals o usant terminals distints. No obstant, esta solució és incapaç d'aprofitar al màxim tots els recursos. Per a això és necessari un nou conjunt d'estratègies. Estes estratègies han de gestionar els recursos ràdio elèctrics de conjuntament i assegurar la satisfacció de la qualitat de servici dels usuaris.

Seguint esta idea, esta Tesi proposa dos nous algoritmes. El primer és un algoritme d'assignació dinàmica de recursos conjunt (JDRA) capaç d'assignar recursos a usuaris i de distribuir usuaris entre tecnologies al mateix temps. L'algoritme està formulat en termes d'un problema d'optimació multi-objectiu que es resol usant xàrcies neuronals de Hopfield (HNNs). Les HNNs són interessants ja que se suposa que poden aconseguir solucions subòptimes en curts períodes de temps. No obstant, implementacions reals de les HNNs en ordinadors perden esta ràpida resposta. Per això, en esta Tesi s'analitzen les causes i s'estudien possibles millores.

El segon algoritme és un algoritme de control d'admissió conjunt (JCAC) que admet i rebutja usuaris tenint en compte totes les tecnologies al mateix temps. La principal diferència amb altres algoritmes proposats és que estos últims prenen les decisions d'admissió en cada tecnologia per separat. Per això, es necessita algun mecanisme per a seleccionar la tecnologia a què els usuaris es van a connectar. Al contrari, la tècnica proposta en esta Tesi és capaç de prendre decisions en tot el sistema heterogeni. Per tant, els usuaris no s'enllacen amb cap tecnologia abans de ser admesos.

Com resultat, esta Tesi prepara el camí cap a una gestió eficient de recursos en sistemes heterogenis amb noves idees i tècniques. Les funcions de la gestió de recursos radioelèctrics es dividixen entre els algoritmes JDRA i JCAC de forma que puguen treballar tot junts.

Acknowledgements

More than four years have passed since I started this Thesis. Four years is too much time for facing any project all alone. Obviously, this has not been the case. This Thesis would never be what it is without the help of other people.

First of all, I would like to thank my supervisors. Thanks to Professor Narcís Cardona for giving me the opportunity of being part of the Mobile Communications Group (MCG) and the Instituto de Telecomunicaciones y Aplicaciones Multimedia (iTEAM) at the Universidad Politécnica de Valencia (UPV). Without him I would never started this Thesis. I will be always obliged for the support and guidance of Dr. José F. Monserrat, and, what is more important, for believing in me even when I did not. Thanks for all the valuable discussions and comments, and for helping me with all my publications too.

The MCG is a big group with many people I had to thank, but the list would be too long. Nevertheless, I could not forget three Davids that have improve this Thesis with their estimable work. Thanks to David Martín-Sacristán for his knowledge in HSDPA and his comments in some of my publications. Thanks to David Gozálvez for helping me with WLAN and for his friendliness that always makes the routine more bearable. Finally, thanks to Dr. David Gómez-Barquero for his help with the JDRA algorithm in the beginning of this Thesis.

The MCG is not the only group inside the iTEAM. During this four years I had the opportunity to know the people of the Electromagnetic Radiation Group (GRE). Thanks to all of them for their estimable company and for sharing their office with me for almost four years.

I would also like to thank Professor Jie Zhang and all the people of the Centre for Wireless Design (CWiND) at the University of Bedfordshire for their great welcome in Luton. Thanks for making possible those four months I stayed there.

Y aunque lo deje para el final, esta Tesis nunca podría haberse realizado sin el apoyo de mi familia. Gracias por darme la oportunidad de llegar hasta aquí. Ser doctor no es algo que se consiga en cuatro años, en realidad hace falta mucho más tiempo y personas que sepan guiarte en todo momento.

Table of contents

List of Figures

Acronyms

2D	2-Dimensional
3D	3-Dimensional
3GPP	3rd Generation Partnership Project
AP	Access Point
ATM	Asynchronous Transfer Mode
BLER	Block Error Rate
CAC	Call or Connection Admission Control
CDF	Cumulative Density Function
CDMA	Code Division Multiple Access
CLSA	Cross-Layer Scheduling Algorithm
CPC	Common Pilot Channel
CPU	Central Processing Unit
CQI	Channel Quality Indicator
CRRM	Common Radio Resource Management
CS	Coding Scheme
DRA	Dynamic Resource Allocation
EBW	Equivalent Bandwidth
ERC	Equivalent Resource Consumption

FGCP	Fractional Guard Channel Policy
F-HNN	Fast Hopfield Neural Network
FRA	Fixed Resource Allocation
FTP	File Transfer Protocol
GA	Gaussian Approximation
GAN	Generic Access Network
GP-HNN	HNN with Gradient Projections
GPRS	General Packet Radio System
GPS	Generalized Processor Sharing
GPU	Graphics Processing Unit
GSM	Global System for Mobile communications
HCCA	HCF Controlled Channel Access
HNN	Hopfield Neural Network
HSDPA	High Speed Downlink Packet Access
HUR	Histogram Updating Rate
IEEE	Institute of Electrical and Electronics Engineers
IMHA	Interval Mid-point Histogram Approximation
JCAC	Joint Call Admission Control
JDRA	Joint Dynamic Resource Allocation
LA	Link Adaptation
LCHA	Least Conservative Histogram Approximation
LP	Linear Program
LTB	Load Threshold-Based
LUT	Look-Up Table
MAC	Medium Access Control

MBR	Maximum Bit Rate
MCHA	Most Conservative Histogram Approximation
MLWDF	Modified Largest Weighted Delay First
MQP	M-Queens Problem
PDF	Probabilistic Density Function
PER	Packet Error Rate
QoS	Quality of Service
RAN	Radio Access Network
RAT	Radio Access Technology
RRM	Radio Resource Management
r.u.	resource unit
RV	Random Variable
SF	Spreading Factor
S-HNN	Smith HNN
SNIR	Signal to Noise and Interference Ratio
TDMA	Time Division Multiple Access
TM	Transport Mode
TSP	Traveling Salesman Problem
UE	User Equipement
UMA	Unlicensed Mobile Access
UMTS	Universal Mobile Telecommunications System
VoIP	Voice over IP
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WPAN	Wireless Personal Area Network
WWAN	Wireless Wide Area Networks

ACRONYMS

Chapter 1

Introduction

1.1 Background

Mobile wireless systems are in constant evolution due to the continuously evolving requirements and expectations of both users and operators. Users expect high quality communications and full access to digital contents with the same transmission capacity as wired networks, independently of the number of users active in the system. According to this user demand for wireless connectivity, new standards have been designed and launched to the market in the last years to satisfy these increasing requirements. General Packet Radio System (GPRS), Universal Mobile Telecommunications System (UMTS), High Speed Downlink Packet Access (HSDPA), Worldwide Interoperability for Microwave Access (WiMAX), Wireless Local Area Network (WLAN) or Bluetooth are some examples of current standardized technologies. Each Radio Access Technology (RAT) is specially suited for one type of wireless network, ranging from Wireless Wide Area Networks (WWAN) down to Wireless Personal Area Network (WPAN). In addition to the usage scenario, conventional mobile networks were devised to fulfill the specific Quality of Service (QoS) requirements of each service, whereas other technologies paid more attention to system simplicity and flexibility.

Currently it is quite common to have several independent RATs giving coverage to the same area. Moreover, users are who decide upon the technology they get connected to, either configuring the User Equipement (UE) or using different UEs for each technology. Nevertheless, users should not get involved in this type of decisions, or at least not separately, since they do not have a global view of the different RATs. Thus, the future points to a multi-RAT UE capable of getting automatically connected to the most proper RAT. This multi-access wireless system, also referred to as heterogeneous wireless system, could make the most of the individual coverage and instantaneous capacity of each technology taking into account the RAT availability, signal quality and type of service to provide the most appropriate resources for the variety of different users.

The notion of being always best connected, which was first introduced in [3], is an extension for heterogeneous systems of the notion of being always connected. Now, users not only should be able to be connected anywhere and anytime, but also they should be served with the best available connection, which can be only accomplished with the interworking of the different technologies. For that reason, the standardization bodies are doing their best to make the interworking possible. For instance, the 3rd Generation Partnership Project (3GPP) organization not only allows UMTS to interwork with GPRS (two 3GPP RATs) but also establishes the basis for a WLAN interworking (a non-3GPP RAT). In addition, the Institute of Electrical and Electronics Engineers (IEEE) Standards Association is working on the 802.11u standard (scheduled for March 2010), which provides WLAN with the capability of interworking with other external networks. Nowadays RAT interworking is becoming a reality that requires more advanced mechanisms that result in a higher resource usage and quality.

In wireless systems the concept of QoS poses several constraints to networks management to assure an optimum distribution of the scarce radio resources among active users. In this framework, the concept of Radio Resource Management (RRM) encompasses various techniques specially designed to fulfill the negotiated QoS to the end users. Multi-access wireless networks, as distinguished from existing wireless networks, require some kind of overall resource management to select the best RAT, dynamically allocate resources among them, control the congestion and manage handovers. The Common Radio Resource Management (CRRM) concept is widely used to refer to these tasks.

In any Radio Access Network (RAN) users share a set of available resources being the RRM entity who decides on the distribution policy. An additional functional unit, the CRRM entity, is responsible for the interworking of the RANs not only of the same RAT but also of different RATs [4]. The RRM most important functions are: initial RAT and cell selection, Call or Connection Admission Control (CAC), congestion control, power control, scheduling or resource allocation, handover and vertical handover. These functions must be distributed into the RRM and the CRRM entities. The CRRM/RRM interaction degree defines which entity manages each function. The following interaction degrees were defined in [5]:

- Low interaction degree: the RRM entities provide all the functionalities and the CRRM entity only establishes some policies that configure the resource management behavior. Considering this approach, Zhuang *et al.* proposed a functional architecture for QoS management in a hybrid UMTS-WLAN network [6].

- Intermediate interaction degree: the CRRM entity manages the initial RAT selection and vertical handover functions. The local RRM entities provide some RRM measurements, such as the list of candidate cells for the different RATs and cell load measurements, so that the CRRM can take into account the resource availability in each RAT.

- High interaction degree: in this case, the CRRM entity is involved in most of the functionalities, leaving only the power control and scheduling for the RRM entities. Thus, CRRM is involved in each intra-system handover procedure requiring a more frequent measurement exchange. Similarly, joint congestion control mechanisms could be envisaged to avoid overload situations in any of the underlying RANs.

- Very high interaction degree: this approach introduces the joint scheduling in the CRRM entity. The RRM entities only manage the power control functions. This solution requires that the CRRM entity make decisions at a very short time scale in the order of milliseconds, with the possibility of executing frequent RAT changes for a given terminal.

1.2 State of the art

The following sections provide a short analysis of the state of the art on the main RRM functions.

1.2.1 Initial RAT and cell selection

These two functions, together with CAC, must be performed each time a new call or connection arrives to the heterogeneous system, although they should be also executed after call departures in order to accommodate comfortably active calls. Pérez-Romero *et al.* proposed different policies [7] to sort, in order of preference, the list of candidate RATs the user could get connected to. As an example, they proposed selecting GPRS for voice and indoor users and UMTS for web and outdoor users whenever possible. This policy is motivated by the good performance of UMTS for high data rate users as compared with GPRS and by the bad behavior of UMTS for indoor users. Piqueras *et al.* proposed a dynamic pricing model, where the UE decides on the RAT to get connected to [8]. Prices change dynamically as the load of each RAT varies, making new users prefer the RAT with less load. Nevertheless, the main drawback of these two works is that quality expectations of users are not considered. With this aim, Pérez-Romero *et al.* defined a fittingness factor that reflects the degree of adequacy of each RAT to each user [9]. The fittingness factor takes into account two concepts: (a) the capabilities of the RATs and the user terminals, i.e. if a RAT can provide the service that the user is asking for and if the user terminal can get connected to the RAT, and (b) also the suitability of each possible connection in terms of channel quality and bit rate. Although all abovementioned algorithms only consider one cell per RAT, algorithms [8] and [9] can also select the best-suited cell inside the selected RAT. Conversely, [7] needs an additional process for cell selection once the selected RAT is known. Nevertheless, cell selection is a well-known and solved problem in current cellular networks.

Apart from these works, Giupponi *et al.* proposed a RAT and cell selection scheme with a fuzzy neural approach [10]. This algorithm uses some measures of signal strength, resource availability and mobile speed as inputs. The fuzzy neural controller obtains the suitability of selecting each cell and the allocated bandwidth. Nevertheless, all possible situations must be studied to train the system with them. Although it is not actually an algorithm, Houzé *et al.* proposed the definition of a Common Pilot Channel (CPC) to broadcast some RAN-specific information needed by the UE to perform an optimum RAT selection [11]. The CPC should be received anywhere independently of the RANs coverage and, since sent data is location-related, the UEs should be able to know their own position. UEs are who make the decision about RAT selection. Although CPC could seem useful, UEs do not have the global view of the heterogeneous network that the CRRM entity has. Moreover, the management of the CPC presents some additional problems. For instance it is unclear who should be responsible for the CPC maintenance or how home wireless networks data should be broadcasted through the CPC so that connections could be routed through them when users arrive home.

So far, Unlicensed Mobile Access (UMA) is the only mechanism being implemented to dynamically select the best-suited RAT. UMA is the commercial solution of the 3GPP standard called Generic Access Network (GAN) [12, 13]. In the UMA solution, dual UEs can migrate from HSDPA network to a WLAN Access Point (AP) and vice versa. Thus, anytime a UE finds an AP, it tries to get a WLAN connection since this technology is supposed to provide much better throughput capacity than HSDPA. This philosophy is, in some way, similar to the policies proposed in [7].

1.2.2 Call or connection admission control

The aforementioned algorithms [7–13] are focused on selecting the proper RAT and cell for users. Thus, new calls need additional processes performed independently in each RAT to decide on the final admission. CAC algorithms keep the target users' QoS at desired levels by limiting the number of ongoing calls in the system. With this in view, they must control signal quality, handover failure probability and/or some packet-level QoS parameters such as packet delay, delay jitter or packet dropping rate [14]:

- Controlling signal quality becomes essential in interference-limited systems like Code Division Multiple Access (CDMA) systems. In these cases, CAC algorithms measure the interference [15], the Signal to Noise and Interference Ratio (SNIR) [16] or the system load [17] to decide on call admission. New calls are accepted if the measured level is lower (or greater in case of SNIR) than a predefined threshold. Moreover, in Time Division Multiple Access (TDMA) systems, controlling signal quality is recommended, and thus these CAC algorithms, for networks with low frequency reuse factors that increase system capacity but reduce signal quality [14].

- The handover failure probability can be controlled prioritizing handover calls over new calls. The most extended approach to reduce handover calls dropping is the band guard technique proposed by Hong and Rapaport [18]. This method reserves certain amount of resources for handover calls only, whereas the rest can be shared by both, new and handover calls. An enhanced version was proposed by Ramjee *et al.* [19]. This technique, called Fractional Guard Channel Policy (FGCP), accepts new calls with probability β, where β decreases with the number of occupied channels. Conversely, handover calls are always accepted if there is any free channel. FGCP minimizes the blocking probability given a target probability of handover failure.

- Packet-level QoS parameters reflect the quality perceived by the end-user at the application level more accurately than interference, SNIR or load; especially for interactive or real-time services. Zhao *et al.* [20] proposed a CAC algorithm that tries to meet the packet delay and delay jitter constraints of users in wireless Asynchronous Transfer Mode (ATM) networks. Similarly, the algorithm proposed by Koutsakis *et al.* [21] is focused on the packet dropping rate of video calling services. Both algorithms admit new calls only if these packet-level constraints can be guaranteed.

These approaches fit well with the intermediate CRRM/RRM interaction degree. Although they are also valid for high and very high interaction degrees, in these cases it is preferred to perform a Joint Call Admission Control (JCAC) in the CRRM entity where the RAT selection and CAC are executed jointly. Luo *et al.* proposed the use of JCAC to make the best out of the overall resources [22], but Yu and Krishnamurthy were the first to propose an actual JCAC [23]. They used a semi-Markov decision process to formulate the problem and linear programming to solve it. Although this algorithm is only for CDMA and WLAN systems, results show the high benefit of a JCAC. Nevertheless, their proposal have several drawbacks. Firstly, they define different QoS requirements depending on the technology. More specifically, they try to guarantee an average delay and bit rate for users connected to WLAN and a minimum SNIR for users connected to the CDMA network. This definition of the QoS requirements produces an heterogeneous treatment of users.

1.2.3 Resource allocation

Generally, radio resources can be distributed following two main approaches: Fixed Resource Allocation (FRA) and Dynamic Resource Allocation (DRA). The first one is the most extended in the classic circuit-switched networks conceived for voice conversations, which is the case of Global System for Mobile communications (GSM). The second one is the best suited for present and future packet-switched networks, where the bursty nature of new services traffic makes the FRA schemes underutilize the available resources. Nevertheless, the DRA approach requires a smart scheduling algorithm to guarantee the users' satisfaction in terms of QoS. The objective of the DRA algorithm is to select the optimal amount of radio resources to be allocated to each user. Several DRA algorithms for a unique RAT have been already proposed in the literature (see examples in [5, 24–27]).

Some resource allocation techniques allocate resources to those users experiencing best channel quality [5, 24]. This kind of policy can maximize the average system throughput, but at the expense of an unfair distribution that implies relatively bad QoS for users with poor channel quality. A wide range of more sophisticated algorithms are based on the Generalized Processor Sharing (GPS) idea [25], where resources are distributed among users proportionally to some predefined weights. Within this group, Modified Largest Weighted Delay First (MLWDF) [26] and its improvement, Cross-Layer Scheduling Algorithm (CLSA) [27], are noteworthy. Both techniques take into account some weighting coefficients that prioritize users according to their service, current QoS and perceived channel quality. For that reason, these algorithms exhibit better performance than those prioritizing users according to only one of these characteristics. Moreover, MLWDF and CLSA were designed to provide either bit rate or delay-based QoS.

These techniques can be used within low to high CRRM/RRM interaction degrees since the DRA is performed separately in each RAT. However, within a very high interaction degree, a Joint Dynamic Resource Allocation (JDRA), or joint scheduling, is also possible. As well as with JCAC, JDRA increases the efficiency of the available resources. To this date, only few works have dealt seriously with this specific topic, and all of them are derived from this Thesis.

1.2.4 Handover

User mobility may make the best RAT or cell change over time. Therefore, the initial selection might not be optimum in the future. In general, handover (or intra-system handover) is understood as the change process of the cell the UE is connected to, but considering the same RAT. Present cellular system standards include the necessary mechanisms to perform intra-system handovers. A handover occurs when the signal quality of the current cell decreases under the quality perceived from a contiguous cell. Usually a hysteresis margin is used to avoid continuous changes near the cell boundaries. Otherwise, the vertical handover (or inter-system handover) also implies a RAT change. Vertical handovers are similar but including RAT changes. The new RAT can be selected with the help of an initial RAT selection algorithm, just like if a new call was asking for admission. In case of very high interaction degree, JDRA algorithms by themselves are capable of automatically selecting the best RAT for each user and, hence, can perform vertical handovers if necessary. Moreover, the UEs for these scenarios are expected to be capable of being connected to all RATs at the same time, thus RAT changes can be performed almost instantaneously. For the rest of interaction degrees, some vertical handover procedures have been proposed in the literature (see examples in [23, 28–31]).

1.3 Challenges

The problem of managing resources acquires additional complexity in the framework of heterogeneous networks. Focusing on the main objective of CRRM, i.e. increasing the QoS perceived by users, the very high interaction degree can improve the system performance as compared with the rest of interaction degrees. Thus, initial RAT selection, CAC, resource allocation and handovers should be carried out jointly to make the best out of the available technologies and resources. While many initial RAT selection algorithms have been proposed in the literature, so far JCAC and JDRA topics have been somewhat forgotten by the scientific community. This fact could be explained because JCAC and JDRA represent a second step in the process of coupling of several RATs and the first one, that is initial RAT selection, still needs more work and efforts.

Consequently, this Doctoral Thesis has identified the following two main challenges in resource management:

- The design of a JDRA algorithm capable of allocating resources to users and distributing users among RATs at the same time. Therefore, this algorithm should be in charge of the resource allocation and vertical handover functions. Moreover, since it is capable of distributing users, the initial RAT selection function can be also delegated to it.

  The JDRA is a hard optimization problem in which the best bit rate and RAT must be selected for each user. Although many algorithms for DRA have been proposed for resource distribution in single technologies (see examples in [5, 24–27]), these are not effective in a multi-access scenario. Single-RAT algorithms are focused on maximizing the total throughput while providing the QoS that users require. Throughput can be easily maximized if the channel quality of each user is known, since users with best channel quality consume less resources. Therefore, the total throughput is maximized by allocating the maximum bit rate to these users. This simple algorithm for single-RAT becomes useless in a multi-access scenario since the throughput maximization depends on the distribution of users among RATs. The complexity increases when the QoS provision is also considered. There are different approaches to identify the quality users perceive, but all of them use certain parameters of the communication procedure. For instance channel quality, interference, bit rate or delay are some of the most used parameters. The JDRA must ensure certain target level for these parameters to obtain the desired QoS. Transport level parameters, like bit rate or delay, better reflect the actual QoS that end users perceive although link level parameters, like channel quality or interference level, are easier to introduce in the JDRA algorithm. In short, the main challenges of the JDRA algorithms are:

  - Users distribution among RATs.
  - Resource allocation among users.
  - Users QoS fulfilment.

- The design of a JCAC algorithm capable of admitting and rejecting users taking the overall heterogeneous network into account.

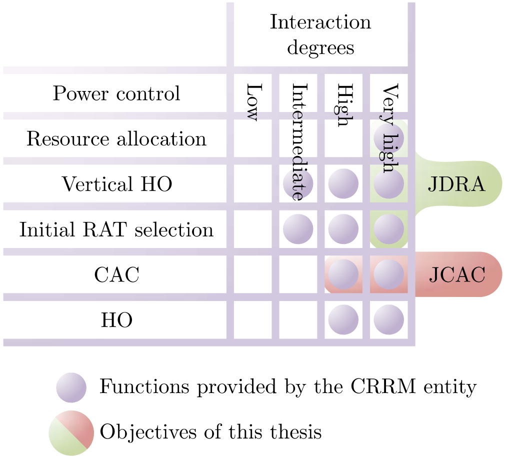

Figure 1.1: Thesis objectives and their relation with the RRM functions.

Initial RAT selection techniques, like those exposed in Section 1.2.1, are followed by CAC algorithms performed independently in each RAT. This fact means that if the selected RAT cannot accept the new call then the call should be rejected although other RAT could serve it or even although the new call could be accepted releasing some resources in the selected RAT by means of vertical handovers. This global vision of the overall heterogeneous network can be only achieved with a JCAC mechanism. Therefore, and similarly to JDRA algorithms, single-RAT CAC algorithms are not effective in a multi-access scenario. JCAC algorithms must consider the availability of resources in all RATs and the RAT changes that could free resources for the new calls. Moreover, QoS support must be also introduced, paying special attention to transport level parameters. Therefore, the JCAC challenges are:

- Consideration of all RATs at the same time.
- Users QoS support.

1.4 Objectives

This Thesis is focused on the development of a new JDRA and a new JCAC techniques. The JDRA technique not only includes the resource allocation in all RATs but also the vertical handover control and initial RAT selection.

In turn, the JCAC technique controls the admission of new calls. Figure 1.1 represents the functions that both techniques provide inside the CRRM entity.

1.4.1 JDRA algorithm

The JDRA problem can be formulated as a multi-objective optimization problem. Many types of algorithms have been proposed in the literature to solve optimization problems, such as genetic algorithms, game theory, linear programming or Hopfield Neural Networks (HNNs). Within this group, HNNs are identified as fast hardware optimizers that could obtain a solution in few microseconds [32]. This fast response is a consequence of the simplicity of each individual neuron and their parallel interworking. Thus, problems that are more complex need more neurons, i.e. more hardware, but maintain the fast response of simpler problems.

HNNs have been widely used in a variety of scientific domains [33–35] and, thanks to their fast response, HNN-based algorithms have been recently proposed to obtain fast and quasi-optimum solutions to the DRA problem. The first study that introduced a HNN-based algorithm in a wireless system was presented by Del Re *et al.* in [36]. The research work carried out by Lázaro and Girma in [37] was built on this algorithm. They proposed the usage of HNNs for the dynamic distribution of frequency channels over the cells of a GSM system together with a guard channel technique for handovers. Ahn and Ramakrishna [32] were the first authors to use HNNs for solving the DRA problem. In the main, their algorithm aimed at maximizing the allocated resources and obtaining a fair distribution among users. García *et al.* [38] applied this philosophy to the distribution of resources in a CDMA system with the objective of satisfying the bit rate expectations of users.

The aforementioned works have shown the utility of HNNs to allocate dynamically resources to users. Thus, starting from the original work of Ahn and Ramakrishna [32], this Thesis aims at proposing a new HNN-based algorithm to distribute jointly resources among users to satisfy their objective QoS. Moreover, this Thesis looks for a feasible implementation of HNNs and, hence, of this algorithm.

1.4.2 JCAC algorithm

In general, any CAC algorithm may be based on a deterministic or a probabilistic bound [39]. A deterministic guaranteed service provides for the worst-case requirements of flows. These requirements are usually computed from parameterized models of traffic sources. For instance, sources may be required to provide peak-rate characterizations of their traffic. The CAC algorithms then check that the sum of all peak rates is less than the link capacity. More sophisticated versions of this approach may consider some tolerable buffer delay. Then this algorithm may be replaced by some leaky bucket approach. Deterministic bound has two main drawbacks. First, usually current traffic is very bursty what makes the link capacity be underutilized. Second, wireless systems are characterized by a very variable channel and hence link capacity. Obviously it is not affordable to ensure the QoS at the cell edge and for peak-rates.

The idea pursued with the probabilistic bound is to ensure QoS only with certain probability. Users do not usually require their peak-rates at the same time and they are not located at the cell edge simultaneously. In fact, the probability of this happening may be negligible. Therefore, with much less resources users can be sufficiently satisfied. The Equivalent Bandwidth (EBW) concept has been successfully applied to ATM networks. EBW may be defined as the maximum aggregate bit rate required by a set of users with certain probability ε. Thus, if the ATM link capacity exactly equals the EBW then users would have enough capacity with probability ε. Nevertheless, EBW lacks a proper modeling of the link variability that characterizes wireless systems.

This Thesis will propose the Equivalent Resource Consumption (ERC), a EBW generalization, as the probabilistic bound for any communication system, including wireless systems. Moreover, the JCAC algorithm will be based on the ERC and hence will inherit all its benefits.

1.5 Thesis outline

Previous section has presented two well-differentiated objectives. For that reason, this Thesis is divided into parts. The work performed towards each objective is described in different parts. Moreover, an additional part includes some final results of both algorithms and the main conclusions. The outline of this Thesis is as follows:

Part I focus on the JDRA algorithm.

> **Chapter 2** describes the HNNs that will be used in the next chapter to implement the JDRA algorithm. This chapter starts defining these networks as they were originally defined by Hopfield. Nevertheless, this definition is not generally used since it requires of an analog circuit. Usually HNNs are implemented in computers hence, becoming discrete in time. After defining these discrete-time HNNs, this chapter presents two techniques for reducing the response time of these networks. Moreover, Section 2.3.5 shows the performance of both

techniques working together, something that has been never studied before.

Chapter 3 presents the design of the JDRA algorithm. To this aim, first of all, the users QoS and any resource quantity are written in terms of bit rate. This transformation makes possible a common treatment of both different types of QoS and technologies. After this first step, this chapter describes the HNN-based JDRA algorithm. Finally, this algorithm is tested within a simulation environment that uses a completely new mobility model to generate hotspots.

Part II focus on the JCAC algorithm.

Chapter 4 defines the ERC as an EBW generalization. The definition uses probabilistic density functions that are not always available. For that reason, this chapter also presents an approximation with histograms and studies the accuracy of this approach. Finally, this chapter shows how the ERC can be used to decide on call admission in one technology.

Chapter 5 uses the ERC to make a common treatment of all technologies, thus, making possible the definition of the JCAC algorithm. The algorithm is introduced in two steps. The first one describes the main idea this algorithm is based on. The second one explains how this idea should be used in real systems. Finally, the ERC-based JCAC algorithm is analyzed by means of computer simulations.

Part III concludes this Thesis.

Chapter 6 joins the two algorithms developed in Parts I and II to test their performance working together.

Chapter 7 states the main conclusions arisen during the development of this Thesis and describes the next steps that should be accomplished in order to continue with the work started here.

1.6 Publications

The work developed during this Thesis made possible the publication of the following journal and conference papers.

Journals

[J1] **D. Calabuig**, J. F. Monserrat, D. Martín-Sacristán and N. Cardona, "Joint dynamic resource allocation for QoS provisioning in multi-access and multi-service wireless systems," Mobile Networks and Applications, in press.

[J2] D. Martín-Sacristán, J. F. Monserrat, J. Cabrejas-Peñuelas, **D. Calabuig**, S. Garrigas and N. Cardona, "On the way towards fourth generation mobile: 3GPP LTE and LTE-Advanced," Journal on Wireless Communications and Networking, in press.

[J3] D. Martín-Sacristán, J. F. Monserrat, J. Cabrejas-Peñuelas, **D. Calabuig**, S. Garrigas and N. Cardona, "3GPP long term evolution: paving the way towards next 4G," Waves, in press.

[J4] J. F. Monserrat, **D. Calabuig**, L. Rubio and N. Cardona, "Hopfield neural-network-based dynamic resource allocation scheme for non-real-time traffic in wireless networks," International Journal of Communication Systems, vol. 22, no. 2, pp. 135 - 158, 2009.

[J5] **D. Calabuig**, J. F. Monserrat, D. Gómez-Barquero and N. Cardona, "A delay-centric dynamic resource allocation algorithm for wireless communication systems based on HNN," IEEE Transactions on Vehicular Technology, vol. 57, no. 6, pp. 3653 - 3665, 2008.

[J6] **D. Calabuig**, J. F. Monserrat, D. Gómez-Barquero and O. Lázaro, "An efficient dynamic resource allocation algorithm for packet-switched communication networks based on Hopfield neural excitation method," Neurocomputing, vol. 71, no. 16 - 18, pp. 3439 - 3446, 2008.

[J7] D. Martín-Sacristán, J. F. Monserrat, **D. Calabuig** and J. Díaz, "Link Adaptation Analysis in 3.5G HSDPA Mobile Communication Systems," Magazine on Computer Science Management, vol. 7, pp. 37 - 47, 2008.

[J8] D. Gozálvez, J. F. Monserrat and **D. Calabuig**, "Policy based channel access mechanism selection for QoS provision in IEEE 802.11e," IEEE Vehicular Technology Magazine, vol. 2, no. 3, pp. 29 - 34, 2007.

[J9] **D. Calabuig**, J. F. Monserrat, D. Gómez-Barquero and O. Lázaro, "User bandwidth usage - driven HNN neuron excitation method for maximum resource utilization within packet - switched communication networks," IEEE Communications Letters, vol. 10, no. 11, pp. 766 - 768, 2006.

[J10] J. F. Monserrat, **D. Calabuig**, D. Gómez-Barquero and N. Cardona, "Scheduling and Dynamic Resource Allocation in UMTS," Magazine on Computer Science Management, vol. 5, pp. 11 - 20, 2006.

Conferences

[C1] D. Martín-Sacristán, J. Cabrejas, **D. Calabuig** and J. F. Monserrat, "MAC layer performance of different channel estimation techniques in UTRAN LTE downlink," IEEE Vehicular Technology Conference, Barcelona, Spring 2009.

[C2] **D. Calabuig**, J. F. Monserrat, D. Martín-Sacristán and N. Cardona, "Joint dynamic resource allocation for coupled heterogeneous wireless networks based on Hopfield neural networks," IEEE Vehicular Technology Conference, Singapore, Spring 2008.

[C3] J. F. Monserrat, R. Fraile. **D. Calabuig** and N. Cardona, "Complete shadowing modeling and its effect on system level performance evaluation," IEEE Vehicular Technology Conference, Singapore, Spring 2008.

[C4] **D. Calabuig**, J. F. Monserrat, D. Martín-Sacristán and N. Cardona, "Joint dynamic resource allocation for coupled heterogeneous wireless networks. A new Hopfield neural network-based approach," Broadband Europe, Antwerp, Fall 2007.

[C5] D. Martín-Sacristán, J. F. Monserrat, **D. Calabuig** and N. Cardona, "HSDPA link adaptation improvement based on node-B CQI processing," IEEE International Symposium on Wireless Communication Systems, Trondheim, Fall 2007.

[C6] **D. Calabuig**, J. F. Monserrat, D. Martín-Sacristán and N. Cardona, "Joint dynamic resource allocation for coupled heterogeneous wireless networks. A new Hopfield neural network-based approach," COST 2100 TD(07)382, Duisburg, Fall 2007.

[C7] M. C. Lucas-Estañ, J. Gozálvez, J. Sánchez-Soriano, M. Pulido and **D. Calabuig**, "Multi-channel radio resource distribution policies in heterogeneous traffic scenarios," IEEE Vehicular Technology Conference, Baltimore, Fall 2007.

[C8] D. Gómez-Barquero, **D. Calabuig**, J. F. Monserrat and N. Cardona, "Hopfield neural network - based approach for joint dynamic resource allocation in heterogeneous wireless networks," IEEE Vehicular Technology Conference, Montral, Fall 2006.

[C9] J. F. Monserrat, D. Gómez-Barquero, **D. Calabuig**, L. Rubio and N. Cardona, "Evaluation of soft handover micro diversity gain on the UMTS system capacity and QoS," IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Helsinki, Fall 2006.

[C10] **D. Calabuig**, J. F. Monserrat, D. Gómez-Barquero and N. Cardona, "Hopfield neural network algorithm for dynamic resource allocation in WCDMA systems," IEEE International Symposium on Wireless Communication Systems, Valencia, Fall 2006.

[C11] D. Gómez-Barquero, **D. Calabuig**, J. F. Monserrat and N. Cardona, "Hopfield neural network algorithm for joint dynamic resource allocation in heterogeneous wireless networks," NEWCOM-ACoRN Joint Workshop, Vienna, Fall 2006.

[C12] **D. Calabuig**, D. Gómez-Barquero, J. Monserrat and N. García, "A dynamic resource allocation algorithm for WCDMA systems with delay constraints based on Hopfield neural networks," Trends in Radio Resource Management, Barcelona. Fall 2005.

Submitted journal papers

[S1] **D. Calabuig**, S. Giménez, J. E. Román and J. F. Monserrat, "Fast Hopfield neural networks using subspace projections," Neurocomputing.

[S2] **D. Calabuig**, J. F. Monserrat and N. Cardona, "Convergence and stability of quantized Hopfield networks operating in a fully parallel mode," Neural Computation.

[S3] **D. Calabuig**, J. F. Monserrat, D. Martín-Sacristán and N. Cardona, "Analytical approach to the required amount of resources to satisfy users quality of service," International Journal of Communication Systems.

[S4] D. Martín-Sacristán, J. F. Monserrat, **D. Calabuig**, "Radio Spectrum Allocation in LTE: Different Alternatives for a Flexible Technology," IEEE Communications Magazine.

Part I

JDRA Algorithm

Chapter 2

Hopfield neural networks

HNNs have gained much relevance as a good tool to solve optimization problems mainly thanks to their fast response time. From Hopfield neuron model, any problem that could be written in terms of a second order Lyapunov function can be solved with a quasi-optimal solution using HNNs. However, HNNs have also acquired many detractors because of the poor quality of the obtained solutions. HNNs are not the best method for solving optimization problems since there are many other techniques that can obtain better solutions (genetic algorithms, simulated annealing, etc.). Consequently, when the response time is of not much significance, other methods are preferred. Nevertheless, in the DRA case it is more interesting to obtain fast solutions although they are not completely optima. For that reason, much of the DRA algorithms proposed in the literature are heuristics with poor performance but low computational cost. Here, HNNs become useful since they conserve the fast response time and their solutions may outperform those of heuristics.

Before describing the JDRA algorithm, this chapter studies the HNNs and their implementability to achieve fast and stable solutions. The Hopfield model will be introduced in Section 2.1 with special mention to the fast convergence and stability. Sections 2.2 and 2.3 will analyze implementability problems of HNNs and study an alternative implementation in digital devices. These two sections aim at making possible the implementation of any HNN.

2.1 Hopfield model

Hopfield modeled the neurons behavior with an analog circuit composed of resistors, operational amplifiers and current generators [40]. His model is shown

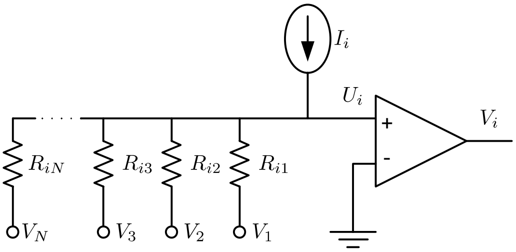

Figure 2.1: Neuron model of a network with N neurons.

schematically in Figure 2.1. Resistors R_{ij} represent the synaptic unions between neurons. These unions can also be written in terms of conductance $T_{ij} = 1/R_{ij}$. U_i and V_i are the neuron input and output respectively and I_i is a bias current. If the Kirchoff laws are applied to the input of neuron i, then:

$$C_i \frac{dU_i}{dt} + \frac{U_i}{R_i} = \sum_{i=1}^{N} T_{ij} V_i + I_i, \tag{2.1}$$

where C_i is the parasitic capacitance at the operational amplifier input and R_i is:

$$\frac{1}{R_i} = \frac{1}{\rho_i} + \sum_{j=1}^{N} \frac{1}{R_{ij}}, \tag{2.2}$$

where ρ_i is the parasitic resistance at the operational amplifier input. The operational transfer function, represented by g_i, i.e. $V_i = g_i(U_i)$, is supposed to be continuous and monotonic increasing. Its output is bounded between the lower and upper saturation levels of the operational amplifier. For the sake of simplicity, those levels are usually supposed to be 0 and 1. The transfer function is generally approximated by either the step, linear or sigmoidal function (see Figure 2.2). These functions are:

$$\text{step function:} \qquad V_i = \begin{cases} 0, & U_i < 0, \\ 0.5, & U_i = 0, \\ 1, & U_i > 0, \end{cases} \tag{2.3}$$

$$\text{linear function:} \qquad V_i = \begin{cases} 0, & U_i < -\frac{0.5}{\alpha_i}, \\ \alpha_i U_i + 0.5, & -\frac{0.5}{\alpha_i} \le U_i \le \frac{0.5}{\alpha_i}, \\ 1, & U_i > \frac{0.5}{\alpha_i}, \end{cases} \tag{2.4}$$

$$\text{sigmoidal function:} \qquad V_i = \frac{1}{1 + e^{-\alpha_i U_i}}, \tag{2.5}$$

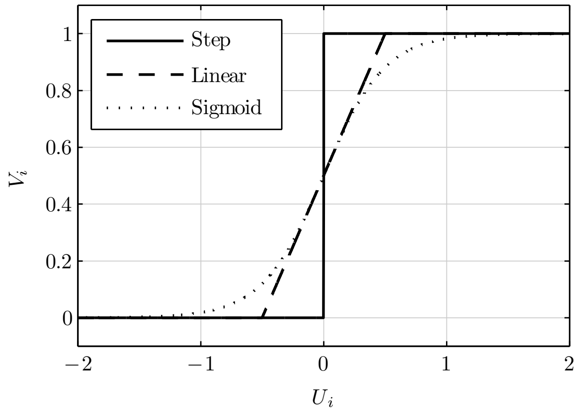

Figure 2.2: Transfer functions with $\alpha_i = 1$ for the linear function and $\alpha_i = 4$ for the sigmoid.

where α_i is the i-th neuron gain. The actual transfer function is more similar to the linear function around $V_i = 0.5$ and more similar to the sigmoid near the saturation levels. Nevertheless, the sigmoid is usually selected as the transfer function whereas the other two are used as approximations of the first one. This fact is due to the non-differentiability of the step and linear functions at some points[1], what makes more difficult to study the convergence and stability of these networks.

Hopfield used the following Lyapunov function, also known as energy function, to prove the stability of these networks [40]:

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} T_{ij}V_iV_j + \sum_{i=1}^{N}\frac{1}{R_i}\int_0^{V_i} g_i^{-1}(V)dV - \sum_{i=1}^{N} I_iV_i. \tag{2.6}$$

Making $T_{ij} = T_{ji}$, the time derivative of the energy function E is:

$$\frac{dE}{dt} = -\sum_{i=1}^{N}\frac{dV_i}{dt}\left(\sum_{j=1}^{N} T_{ij}V_j - \frac{U_i}{R_i} + I_i\right), \tag{2.7}$$

[1] Operational amplifiers are real circuits and hence they have continuous and differentiable transfer functions.

and from (2.1):

$$\frac{dE}{dt} = -\sum_{i=1}^{N} \frac{dV_i}{dt} C_i \frac{dU_i}{dt} = -\sum_{i=1}^{N} C_i \left(\frac{dV_i}{dt}\right)^2 \frac{dU_i}{dV_i}. \tag{2.8}$$

Since for the sigmoid case, g_i is strictly monotonically increasing then $dU_i/dV_i > 0$ and, since $C_i > 0$, then $dE/dt \leq 0$. Thus, E always decreases or remains unchanged. If the latter happens then $dV_i/dt = 0$ and consequently the network has reached equilibrium. This fact, together with the boundedness of E, proves that HNNs always converge to a stable state. The boundedness of E is important because on the contrary E could decrease without limits. With a lower bound, the energy cannot decrease more than that bound and hence at some point necessarily $dE/dt \to 0$. The boundedness can be demostrated proving the boundedness of each term of (2.6) separately. The first and third terms are polynomial, thus they are upper and lower bounded in the interval of interest $[0, 1]$. The second term is:

$$\begin{aligned} \sum_{i=1}^{N} \frac{1}{R_i} \int_0^{V_i} g_i^{-1}(V) dV &= \sum_{i=1}^{N} \frac{1}{R_i} \int_0^{V_i} \frac{1}{\alpha_i} \text{ln} \left(\frac{V}{1-V}\right) dV = \\ &= \sum_{i=1}^{N} \frac{V_i \text{ln}(V_i) + (1-V_i)\text{ln}(1-V_i)}{\alpha_i R_i}. \end{aligned} \tag{2.9}$$

This term has a minimum at $V_i = 0.5$ and two maxima at $V_i = 0$ and $V_i = 1$, see Figure 2.3. Consequently, it is also bounded.

2.1.1 Convergence points

Optimization problems need to be written as an energy function before being solved using HNNs. The most common method to do this is to identify some binary variables in the problem which will be represented by neurons. For example, the M-Queens Problem (MQP) tries to find the distribution of M queens in an $M \times M$ chessboard so that they could not attack each other. The chessboard can be represented by an $M \times M$ matrix of neurons where neuron (i, j) approaches 1 if there is a queen at the i-th row and j-th column and approaches 0 if no queen goes there. Thus, the desired solutions are always at the extremes where $V_{ij} \in \{0, 1\}$.

In general, the desired solutions of a HNN with N neurons are located at the corners of an N-dimensional hypercube. Moreover, at any time t the vector of neuron outputs represent a point inside this hypercube, thus the hypercube is the search space of the solution. From (2.8) it was previously concluded

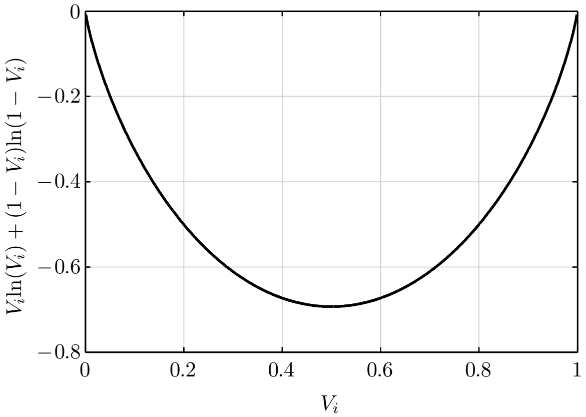

Figure 2.3: Second term of (2.6).

that the energy converges at those points where neuron outputs also converge. Nevertheless, it is not obvious to know if those convergence points are located at the corners. In order to clarify this, it is necessary to look deeper at the time derivative of neuron outputs:

$$\frac{dV_i}{dt} = \frac{\alpha}{e^{\alpha U_i} + 2 + e^{-\alpha U_i}} \frac{dU_i}{dt}. \tag{2.10}$$

If neuron outputs converge then either $dU_i/dt = 0$, $U_i \to \infty$ or $U_i \to -\infty$. In the last two cases, the corresponding neuron output is approaching one extreme since $V_i \to 1$ when $U_i \to \infty$ and $V_i \to 0$ when $U_i \to -\infty$. Nevertheless, if $dU_i/dt = 0$ then the equilibrium may not be located at the extremes, more specifically:

$$\frac{dU_i}{dt} = \frac{1}{C_i}\left(\sum_{j=1}^{N} T_{ij}V_j - \frac{U_i}{R_i} + I_i\right) = -\frac{1}{C_i}\frac{\partial E}{\partial V_i}, \tag{2.11}$$

thus the energy may converge to those points with null gradient, i.e. to critical points. Moreover, it is also possible a mix of these cases where some neurons approach the extremes while others converge due to a null gradient. In that case the equilibrium is not a critical point of the energy function in $\mathbb{R}^N$ but a critical point in the subspace where all neurons approaching to the extremes are fixed. This fact makes impossible to prevent the existence of critical points in the hypercube. Nevertheless, a good design of the energy function can eliminate any minima which are the most problematic critical points. Note that the desired solutions are at the corners thus a good design would make the energy minimum only if $V_i = 0$ or $V_i = 1$ and not between those points.

2.1.2 Energy second term and the extensively used neuron model

An applicability problem of the Hopfield model is the second term of (2.6). This term is hard to add to energy functions that describe real problems. The same Hopfield [33] proposed to eliminate it when he applied his model to the Traveling Salesman Problem (TSP). In fact, if $\alpha_i \to \infty$ then the second term is negligible. Moreover, operational amplifiers have very high gains of the order of 10,000 or 100,000. Thus, making $\alpha_i = \infty$ seems a good approximation.

The main problem of supposing $\alpha_i = \infty$ is that both the linear and sigmoidal functions become the step function. This fact changes the evolution of the neuron outputs. With $\alpha < \infty$, neurons can evolve inside the entire interval $[0, 1]$ but with the step function $V_i \in \{0, 1\}$ only. Thus, the neural network does not approach the minimum step by step but neuron outputs change abruptly changing also abruptly the energy gradient and hence the evolution of neuron inputs U_i. Moreover, neuron outputs do not evolve together but only when inputs cross 0, i.e. if at time t the i-th neuron input is 0, it is probabilistically impossible that another neuron input $j \neq i$ could be also 0, thus two neuron outputs never change simultaneously. Due to this fact, the neural network can only evolve to adjacent hypercube corners, what highly increases the probability of evolving to local minima[2] [41].

Maybe due to the poor outcomes of this model, the step function is not generally used in the literature although the second term of (2.6) is always eliminated. Thus, the energy function results in:

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} T_{ij} V_i V_j - \sum_{i=1}^{N} I_i V_i, \tag{2.12}$$

whereas the transfer function are either the linear or sigmoidal functions. It is worth noting that this model cannot be exactly implemented using the Hopfield neuron model. Nevertheless, the error is negligible and can be assumed if network parameters are well selected (high α_i and R_i). However, it is important to not be mistaken and mix both models, like other authors have previously done [32]. The most common error is to eliminate the integral in the energy

[2] These local minima are different from those caused by null energy gradient mentioned in the last paragraph of Section 2.1.1. These are corners with a minimum energy in a neighborhood *inside* the hypercube. The gradient may not be null.

function but not suppress also the term where it cames from, i.e. the U_i/R_i term in (2.1). This mistake leads to the erroneous dynamics [32]:

$$\frac{dU_i}{dt} = -\frac{1}{C_i}\left(\frac{\partial E}{\partial V_i} + \frac{U_i}{R_i}\right), \tag{2.13}$$

what moves the neural network away from the desired behavior expressed in (2.11).

2.1.3 Implementation

Both the Hopfield model and the approximated model of Section 2.1.2 were defined keeping in mind their implementation with analog devices, i.e. resistors and operational amplifiers. Thus, they are continuous-time and continuous-state models. These analog circuits are somewhat difficult of implementing. First, they can become too large with many neurons and, second, the resistor values may not be exactly the same as R_{ij} what may change the energy function. Moreover, implementing these circuits is too complex for only testing an algorithm. Consequently, HNNs are usually simulated with computers. Nevertheless, computer simulations lose the continuity in both time and neuron states, although, thanks to floating point computation, neuron states can be supposed continuous. Convergence points can be found solving iteratively the differential equation of (2.11) using the Euler's technique:

$$U_i(t + \Delta t) = U_i(t) - \Delta t \frac{1}{C_i}\frac{\partial E}{\partial V_i}, \tag{2.14}$$

where Δt is a time step. This new parameter must be selected very carefully. If Δt is too high then the Euler's technique does not approach the actual differential equation of (2.11). This fact may prevent the network from converging. On the other hand, a low Δt needs many iterations of (2.14) before reaching equilibrium so that HNNs lose their fast response. Consequently, the value of Δt is a tradeoff between performance and response time. Despite the importance of this parameter, there is no technique for selecting it. The designer must guess its value with a try and error method.

2.2 Discrete-time Hopfield neural networks

Since HNNs have been only implemented in computers or other digital devices, it would be possible to define HNNs for such devices instead of thinking on the analog circuit of Figure 2.1. Other authors have defined HNNs this way. This new approach makes HNNs more versatile since they are not limited by their hardware implementation.

Let H be a discrete-time HNN of N neurons uniquely defined by the pair $(\mathbf{T}, \mathbf{i})$ where $\mathbf{T}$ is an $N \times N$ symmetric matrix with elements T_{ij} and $\mathbf{i}$ is an N-dimensional vector with elements I_i. From this pair, an energy function (2.12) is associated with each HNN H. The network state at iteration t is defined by the neuron outputs $V_i(t)$ that are updated by $\Delta_i(t)$. Thus $V_i(t+1) = V_i(t) + \Delta_i(t)$. The updates are:

$$\Delta_i(t) = \beta(t) d_i(t), \tag{2.15}$$

where vector $\mathbf{d}(t)$ with components $d_i(t)$ is the updating direction and $\beta(t)$ is the updating step at time t. In order to prevent from moving out the hypercube, the updating direction and step must satisfy that:

$$-V_i(t) \leq \beta(t) d_i(t) \leq 1 - V_i(t), \ \forall i. \tag{2.16}$$

Comparing (2.14) and (2.15) the updating direction and step can be identify as the gradient and the time step Δt respectively[3]. These discrete-time HNNs are widely used in the literature. Note that in this case Δt is fixed and so it is the updating step.

2.2.1 Optimum neuron outputs update

Most authors use a fixed updating step following the idea of the Euler's technique and Δt explained in Section 2.1.3. Nevertheless, since selecting the best Δt is not an easy task, the definition of (2.15) allows selecting dynamically an updating step for each iteration t. This section finds a variable updating step called optimum directional update. This update can be defined as the neuron states increment that reduces the energy at maximum following certain direction. Thus, if the direction is the vector $\mathbf{d}(t)$, the energy value after updating neuron outputs is:

$$\begin{aligned} E(t+1) = &-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} T_{ij} \left(V_i(t) + \beta(t) d_i(t)\right)\left(V_j(t) + \beta(t) d_j(t)\right) - \\ &- \sum_{i=1}^{N} I_i \left(V_i(t) + \beta(t) d_i(t)\right), \end{aligned} \tag{2.17}$$

The objective is to find the optimum $\beta(t)$, $\beta_o(t)$, so that the neuron increment $\beta_o(t)\mathbf{d}(t)$ is the optimum directional update for the direction $\mathbf{d}(t)$. The

[3] This is not completely true in the hypercube borders due to (2.16)

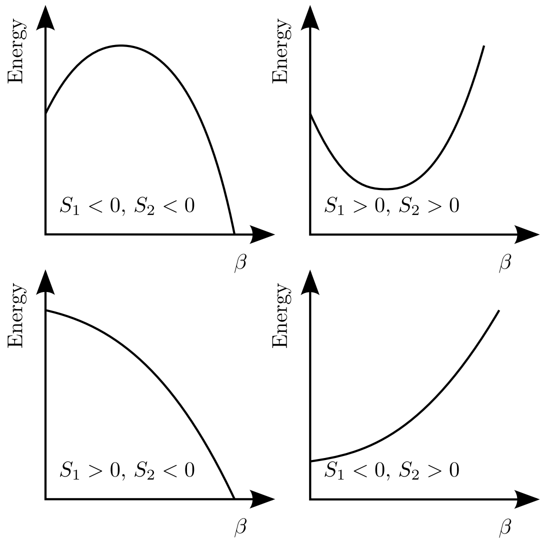

Figure 2.4: Shapes of the energy in function of the signs of S_1 and S_2.

energy of (2.17) is a quadratic function with respect to $\beta(t)$. Thus, it has a critical point which can be either a minimum or a maximum. The variation of the energy through this direction is:

$$\frac{dE(t+1)}{d\beta(t)} = \sum_{i=1}^{N} d_i(t) \frac{\partial E}{\partial V_i}(t) - \sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} \beta(t) d_i(t) d_j(t), \tag{2.18}$$

hence the critical point satisfies that:

$$\frac{dE(t+1)}{d\beta(t)} = 0 \Rightarrow \beta(t) = \frac{S_1(t)}{S_2(t)}, \tag{2.19}$$

$$S_1(t) = -\sum_{i=1}^{N} d_i(t) \frac{\partial E}{\partial V_i}(t), \tag{2.20}$$

$$S_2(t) = -\sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} d_i(t) d_j(t). \tag{2.21}$$

From (2.19) four cases can be identified depending on the signs of $S_1(t)$ and $S_2(t)$ (see Figure 2.4). If $S_2(t) > 0$ then the critical point is a minimum and (2.19) gives the maximum energy decrement. If $S_2(t) < 0$ the critical point is a maximum. In this case $\beta_o(t)$ must point to the opposite direction where the critical point is located and it can be as large as desired. Note from Figure 2.4 that the sign of $\beta_o(t)$ must be the same as the sign of $S_1(t)$.

Additionally, $\beta_o(t)$ must satisfy the constraint (2.16) so that all neurons remain inside the hypercube. Thus:

$$V_i(t) + \beta_o(t)d_i(t) \leq 1, \ \forall i: \ \beta_o(t)d_i(t) > 0, \tag{2.22}$$

$$V_i(t) + \beta_o(t)d_i(t) \geq 0, \ \forall i: \ \beta_o(t)d_i(t) < 0. \tag{2.23}$$

Since $\beta_o(t)$ has the same sign as $S_1(t)$, previous equations are equivalent to:

$$V_i(t) + \beta_o(t)d_i(t) \leq 1, \ \forall i: \ S_1(t)d_i(t) > 0, \tag{2.24}$$

$$V_i(t) + \beta_o(t)d_i(t) \geq 0, \ \forall i: \ S_1(t)d_i(t) < 0. \tag{2.25}$$

Let define $\mathbf{l}(t)$ as a vector of limits with elements $l_i(t)$ where:

$$l_i(t) = \begin{cases} \dfrac{1 - V_i(t)}{|d_i(t)|}, & S_1(t)d_i(t) > 0, \\ -\dfrac{V_i(t)}{|d_i(t)|}, & S_1(t)d_i(t) < 0, \end{cases} \tag{2.26}$$

then $\beta_o(t)$ is:

$$\beta_o(t) = \begin{cases} -\min\limits_i\{|l_i(t)|\}, & S_1(t) < 0, \ S_2(t) < 0, \\ \min\left\{\dfrac{S_1(t)}{S_2(t)}, \min\limits_i\{|l_i(t)|\}\right\}, & S_1(t) > 0, \ S_2(t) > 0, \\ \min\limits_i\{|l_i(t)|\}, & S_1(t) > 0, \ S_2(t) < 0, \\ -\min\left\{-\dfrac{S_1(t)}{S_2(t)}, \min\limits_i\{|l_i(t)|\}\right\}, & S_1(t) < 0, \ S_2(t) > 0. \end{cases} \tag{2.27}$$

A similar idea was proposed by Talaván and Yáñez [42]. They used the direction $d_i(t) = -\partial E/\partial V_i(t)$. Thus, this section is a generalization of the idea of Talaván and Yáñez, necessary for the following sections.

2.3 Energy gradient projections

Usually the solution of a specific problem must strictly satisfy certain condition. Examples of these problems are the Traveling Salesman Problem (TSP) [33] and the M-Queens Problem (MQP) [35]. In all these problems, some neuron subsets can be identified so that the desired solution has only one neuron active in each subset.

For instance, for the case of the HNN defined for the MQP, only one queen can be in each row of the $M \times M$ chessboard. Therefore, any solution of this problem must satisfy that the sum of all neuron outputs of each row is exactly 1. Moreover, only one queen can be in each column. Thus, the sum of all neuron outputs of each column must be exactly 1 too. In general, in many optimization problems valid solutions must satisfy some strict constraints that can be written as linear equations.

Consequently, in real problems a set of linear constraints is incorporated in the energy function adding some additional constraint violation terms [33]. Despite being a common practice, some authors demonstrated that the inclusion of the violation terms results in more likely invalid solutions [43] and even in a change of the network behavior [44]. The underlying problem that causes these convergence problems lies in the fact that the cost terms runs contrary to the constraint terms.

In order to tackle this problem, some authors proposed to confine the HNN to the feasible constraint subspace, hence ensuring the final solution validity [45, 46]. Chu [45] proposed to project the energy gradient, which indicates the direction of movement, modifying the neuron inputs in such a way that the neuron outputs are always in the constraint plane. Although this seminal work was the first one bringing forward the usage of projection-based HNNs, it assumed a continuous-time neural network and no reference was done to more realistic computer implementations, which are inherently discrete in time. The main consequence is that discrete-time implementations continuously separate from the feasible subspace due to large computational errors when neurons are near the extremes. Moreover, the projection matrix was explicitly calculated from the constraints matrix, without considering issues of practical relevance such as computational efficiency and numerical robustness. On the other hand, Smith *et al.* [46] defined an iterative mechanism based on the integration of the projection of the neuron outputs (instead of projecting the energy gradient) together with an annealing technique for escaping local minima by permitting, in a controlled way, increments of the energy function. Comparing this mechanism with [45], both of them use the same calculation method for the projection matrix. However, Smith *et al.* incorporated more effective means to guarantee stability and convergence to feasible solutions, but at the expenses of extremely increasing the computational burden.

This section presents a new computationally efficient subspace projection method. This method will obtain an updating direction so that it will be possible to apply the ideas of previous section to compute the optimum neuron updates. The proposed method performs projections by means of the orthogonalization with respect to an appropriate basis of vectors. This basis is dynamically augmented in order to guarantee that the modified neuron state vector does not exit the space of allowed solutions. This numerical procedure for the projection is computationally efficient.

Concerning the implementation of this Fast Hopfield Neural Network (F-HNN), it is worth highlighting that, although some authors state that the HNN response could be attained in few microseconds [32], this order of magnitude corresponds with the analogue implementation of continuous HNNs. The main reason of the fast response is the parallel interworking of neurons. An implementation on a computer can exploit the increasing potential of parallelism offered by current processor architectures. The proposed F-HNN method retains this inherent capability of parallelization so that fast response times can be expected. Even in a sequential implementation, the method presents advantages.

2.3.1 Projections computation

Let H be a discrete-time HNN with N neurons. Let assume that any feasible solution of this problem satisfies the following M constraints:

$$\mathbf{A}\mathbf{v} = \mathbf{b}, \tag{2.28}$$

where $\mathbf{A}$ is a $M \times N$ matrix and $\mathbf{b}$ is a vector with M elements. This system defines a subset $\mathcal{F}$ of network states inside the hypercube where the M constraints are satisfied, i.e.:

$$\mathcal{F} = \left\{ \mathbf{v} : \mathbf{A}\mathbf{v} = \mathbf{b}, \mathbf{v} \in [0,1]^N \right\}. \tag{2.29}$$

Although the initial state $\mathbf{v}(0)$ belongs to $\mathcal{F}$, the direction of movement (generally minus energy gradient) may move the network state away this subset. Obviously, if the problem is well defined, the stable state belongs to $\mathcal{F}$ and thus the network must return to it. Figure 2.5 represents this idea. The typical path is an illustrative neuron evolution followed by a HNN. Nevertheless, if the stable state is inside the subset $\mathcal{F}$, reaching it could be faster if the search space is reduced to this subset. Alternatively to the typical path, a new path could be followed which entirely belongs to $\mathcal{F}$.

Let define $\overline{\mathcal{F}}$ as the extension of $\mathcal{F}$ to all $\mathbb{R}^N$, i.e.:

$$\overline{\mathcal{F}} = \left\{ \mathbf{v} : \mathbf{A}\mathbf{v} = \mathbf{b} \right\}, \tag{2.30}$$

and let define $\mathcal{F}_0$ as the set parallel to $\mathcal{F}$ that has the coordinate origin, i.e.:

$$\mathcal{F}_0 = \left\{ \mathbf{v} : \mathbf{A}\mathbf{v} = \mathbf{0} \right\}. \tag{2.31}$$

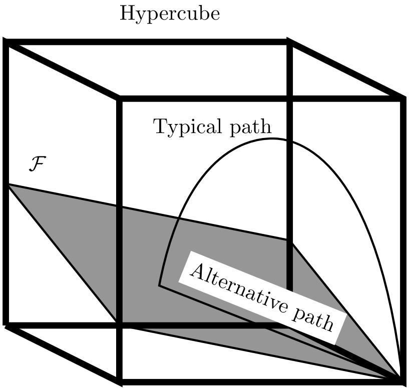

Figure 2.5: Representation of the subset $\mathcal{F}$ inside an hypercube and the typical path followed by a HNN and the alternative path that belongs to $\mathcal{F}$.

Then, if the neuron outputs at some time t, $\mathbf{v}(t)$, belong to $\mathcal{F}$ and the updating direction belongs to $\mathcal{F}_0$ then $\mathbf{v}(t+1) \in \overline{\mathcal{F}}$ independently of the updating step since:

$$\mathbf{A}\mathbf{v}(t+1) = \mathbf{A}\left(\mathbf{v}(t) + \beta(t)\mathbf{d}(t)\right) = \mathbf{A}\mathbf{v}(t) + \beta(t)\mathbf{A}\mathbf{d}(t) = \mathbf{b} + \beta(t)\mathbf{0} = \mathbf{b}. \quad (2.32)$$

Moreover, if the updating step is selected following the explanation of Section 2.2.1 then $\mathbf{v}(t+1) \in \mathcal{F}$, since the neuron outputs will always belong to the unit hypercube. As previously mentioned, other works use the direction of minus energy gradient as the updating direction. Then, the path followed by neuron outputs is something similar to the typical path depicted in Figure 2.5. It is also possible to project the energy gradient into the set $\mathcal{F}_0$ to define a new updating direction. In this case, the alternative path is similar to that of Figure 2.5. Let define $\mathbf{P}$ as the projection matrix that projects any point onto the set $\mathcal{F}_0$, i.e.:

$$\mathbf{A}\mathbf{P}\mathbf{v} = \mathbf{0}, \ \forall \mathbf{v} \in \mathbb{R}^N. \quad (2.33)$$

Then the updating direction of the alternative path is:

$$\mathbf{d}(t) = \mathbf{P}\left(\mathbf{T}\mathbf{v}(t) + \mathbf{i}\right). \quad (2.34)$$

This direction satisfies that $S_1(t) \geq 0$ since the angle between $\mathbf{d}(t)$ and the direction of minus energy gradient is always less or equal than 90° and thus:

$$S_1(t) = \mathbf{d}(t)'\left(\mathbf{T}\mathbf{v}(t) + \mathbf{i}\right) = ||\mathbf{d}(t)||\,||\mathbf{T}\mathbf{v}(t) + \mathbf{i}||\cos\gamma \geq 0, \quad (2.35)$$

where $||\mathbf{x}||$ is the norm of $\mathbf{x}$ and γ is the angle between $\mathbf{d}(t)$ and the direction of minus energy gradient. Consequently $\beta_o(t)$ will be always non-negative.

2.3.2 Projections in the hypercube facets

The main challenge of the projections approach is how to deal with projections in the hypercube facets. Inside the facets, at least one neuron is at one of the extremes 0 or 1. For instance, if neuron i is at the extreme $V_i(t) = 0$ at some time t, that neuron should not be modified if the updating direction is $d_i(t) < 0$. This fact is equivalent to changing the updating direction from $\mathbf{d}(t)$ to $\hat{\mathbf{d}}(t)$, where the components of $\hat{\mathbf{d}}(t)$ are:

$$\hat{d}_j(t) = \begin{cases} d_j(t), & j \neq i, \\ 0, & j = i. \end{cases} \tag{2.36}$$

Since $\mathbf{Ad}(t) = \mathbf{0}$, then, in general, $\mathbf{A}\hat{\mathbf{d}}(t) \neq \mathbf{0}$ due to the change performed in (2.36). This fact means that the next neuron state will not belong to $\overline{\mathcal{F}}$. Obviously, neuron outputs must leave neither the unit hypercube nor the constraints subspace. Both requirements can be accomplished adding new constraints to matrix $\mathbf{A}$. More specifically, (2.36) can be understood as a new linear constraint of the form $\hat{d}_j(t) = 0$. Therefore, it is possible to build a matrix $\mathbf{B}$ with all the new constraints. Then, the projection matrix $\hat{\mathbf{P}}$ is computed from the combination of matrices $\mathbf{A}$ and $\mathbf{B}$, so that:

$$\begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} \hat{\mathbf{P}}\mathbf{v} = \mathbf{0}, \ \mathbf{v} \in \mathbb{R}^N. \tag{2.37}$$

Finally, the updating direction is now:

$$\mathbf{d}(t) = \hat{\mathbf{P}} \left(\mathbf{Tv}(t) + \mathbf{i} \right). \tag{2.38}$$

Continuing with the previous example, this updating direction not only belongs to subset $\mathcal{F}_o$ but, additionally, has its i-th component null.

2.3.3 Practical realization of the projection

Although it has been made so far in other works dealing with projected HNN, explicitly computing the projector matrix, $\mathbf{P}$, for the projection in (2.34) is not practical, because of the computational inefficiency but also due to potential difficulties related to numerical errors. A better approach can be derived from a subspace analysis. The subspace $\mathcal{F}_0$ is the null space of matrix $\mathbf{A}$. It is worth noting that the null space of a matrix is the orthogonal complement to the row space of that matrix (the subspace spanned by its rows). In this setting, $\mathbf{P}$ is the orthogonal projector onto the null space of $\mathbf{A}$. Orthogonal projectors admit a simple representation. Let $\mathbf{Q}$ be an $N \times M$ matrix whose columns constitute an orthonormal basis of the row space of $\mathbf{A}$, that is $\mathbf{Q}'\mathbf{Q} = \mathbf{I}$ and $\text{span}(\mathbf{Q}) = \text{span}(\mathbf{A}')$. Then, the projector can be written as $\mathbf{P} = \mathbf{I} - \mathbf{Q}\mathbf{Q}'$. With this representation, the projection of vector $\mathbf{x}$ can be done with:

$$\mathbf{P}\mathbf{x} = \left(\mathbf{I} - \mathbf{Q}\mathbf{Q}'\right)\mathbf{x} = \mathbf{x} - \mathbf{Q}\mathbf{Q}'\mathbf{x}. \tag{2.39}$$

That is, first the vector is projected onto the subspace spanned by the columns of $\mathbf{Q}$ (the row space of $\mathbf{A}$), then this projection is removed from the original vector. This procedure is extensively used in many areas of numerical linear algebra, and is often referred to as the orthogonalization of a vector with respect to a set of orthonormal vectors.

An interesting property of orthogonal projectors is that, in case of partitioning the columns of $\mathbf{Q}$ in two sets ($\mathbf{Q} = [\mathbf{Q}_1\ \mathbf{Q}_2]$) then the projection can be written as:

$$\mathbf{P}\mathbf{x} = \left(\mathbf{I} - \mathbf{Q}_2\mathbf{Q}_2'\right)\left(\mathbf{I} - \mathbf{Q}_1\mathbf{Q}_1'\right)\mathbf{x}. \tag{2.40}$$

Thus, orthogonalization can be first applied with just a subset of vectors and, then, the final result is obtained after orthogonalizing the previous outcome with respect to the rest. Note that the order is irrelevant because $\left(\mathbf{I} - \mathbf{Q}_2\mathbf{Q}_2'\right)\left(\mathbf{I} - \mathbf{Q}_1\mathbf{Q}_1'\right)$ commute, since $\mathbf{Q}_2'\mathbf{Q}_1 = \mathbf{0}$. In the limit, the orthogonalization can be carried out one vector at a time. This procedure is known as Modified Gram-Schmidt, as opposed to the Classical Gram-Schmidt procedure of (2.39).

In the context of F-HNNs, the explicit computation of the projector $\mathbf{P}$ is replaced by the initial computation of $\mathbf{Q}$, that is, the orthogonalization of the rows of $\mathbf{A}$. This can be done, for instance, by means of Gram-Schmidt procedures for computing the QR matrix decomposition. This initial step has also the advantage that it will eventually detect redundant constraints, since when the result of an orthogonalization is the zero vector it means that the original vector already belonged to the subspace. On the other hand, the property of (2.40) allows dynamically including additional constraints as necessary. This is essential for adding new constraints as explained previously.

2.3.4 Different projection alternatives

Fast HNN (F-HNN)

This is the projection based HNN proposed in this Thesis, which can be summarized as follows:

- Step 1: initialize matrix $\mathbf{A}$ and derive $\mathbf{Q}$. Define a random vector $\mathbf{v}(t)$ for $t = 0$, so that $\mathbf{v}(0) \in [0, 1]^N$ and $\mathbf{A}\mathbf{v}(0) = \mathbf{b}$.

- Step 2: calculate the energy gradient as $\nabla E(t) = -\mathbf{T}\mathbf{v}(t) - \mathbf{i}$.
- Step 3:
  - obtain the updating direction as $\mathbf{d}(t) = -\nabla E + \mathbf{Q}\mathbf{Q}'\nabla E$.
  - check that all neurons will be confined in the hypercube (see procedure described in Section 2.3.2).
  - if all neurons are confined go to Step 5. Otherwise, go to Step 4.
- Step 4: while there is any neuron not confined in the hypercube
  - add new constraints to $\mathbf{A}$ and derive the new columns of $\mathbf{Q}$, $\mathbf{Q}_2$.
  - actualize $\mathbf{d}(t) \leftarrow \mathbf{d}(t) - \mathbf{Q}_2\mathbf{Q}_2'\mathbf{d}(t)$.
  - actualize $\mathbf{Q} \leftarrow [\mathbf{Q}\ \mathbf{Q}_2]$.
- Step 5: calculate $S_1(t)$, $S_2(t)$, $\mathbf{l}(t)$ and $\beta(t)$ following the reasoning described in Section 2.2.1 (equations (2.20), (2.21), (2.26) and (2.27) respectively).
- Step 6: update neuron states as $\mathbf{v}(t+1) = \mathbf{v}(t) + \beta(t)\mathbf{d}(t)$.
- Step 7: $t \leftarrow t+1$. Go to Step 2 until termination criterion is met.

All this procedure is shown in Figure 2.6.

HNN with Gradient Projections (GP-HNN)

As described at the beginning of Section 2.3, Chu [45] proposed the projection of the energy gradient considering a continuous HNN. However, Chu did not take into account all the implications brought by the discrete-time implementation of his proposal.

The GP-HNN method is the discrete counterpart of [45] and can be understood as a simplification of the F-HNN method, since a fixed updating step has been assumed, Δt. Therefore, the GP-HNN procedure is almost the same as F-HNN except that Step 5 is removed and $\beta(t) = \Delta t$. Obviously, Δt must be carefully selected to guarantee the fast convergence to the minimum of the energy.

Smith HNN (S-HNN)

The proposal of Smith *et al.* [46] can be summarized in the following steps:

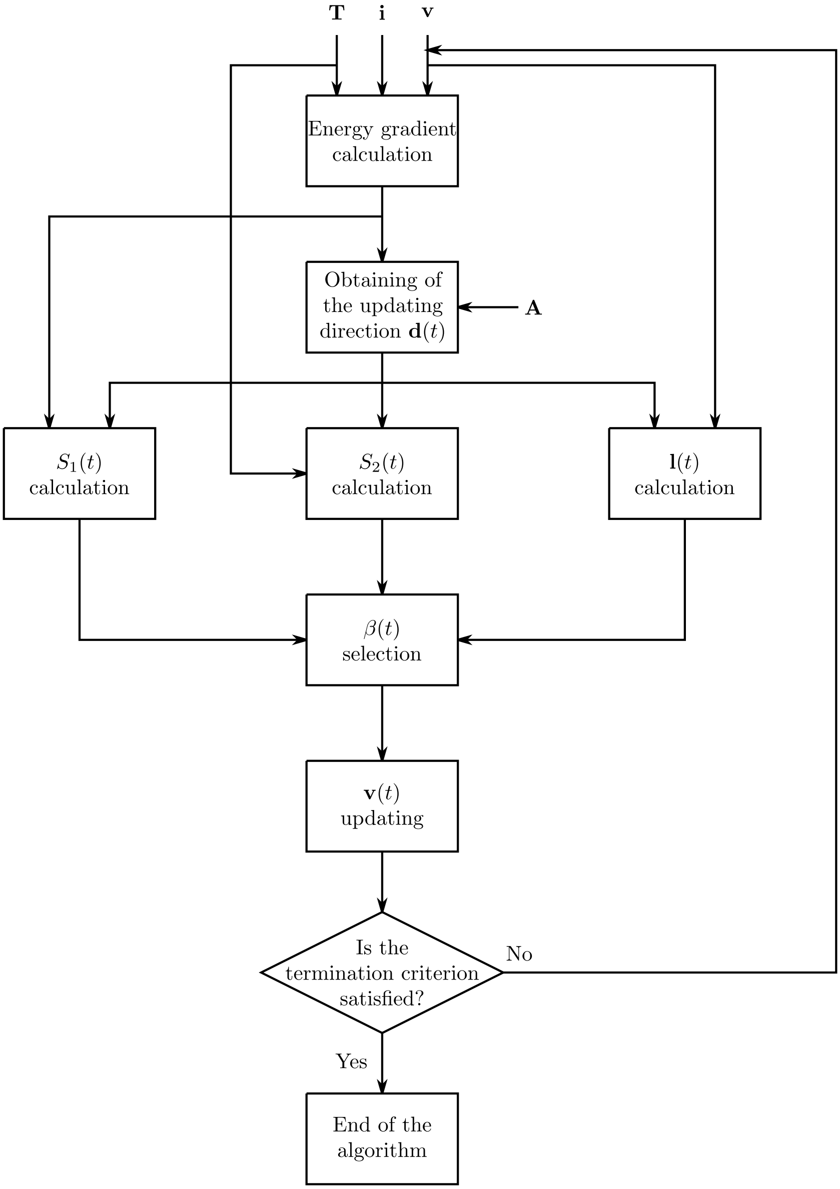
T
i
v
Energy gradient calculation
Obtaining of the updating direction $\mathbf{d}(t)$
A
$S_1(t)$ calculation
$S_2(t)$ calculation
$\mathbf{l}(t)$ calculation
$\beta(t)$ selection
$\mathbf{v}(t)$ updating
Is the termination criterion satisfied?
No
Yes
End of the algorithm

Figure 2.6: Flowchart representation of F-HNN.

- Step 1: initialize matrix $\mathbf{A}$ and derive $\mathbf{Q}$. Define a random vector $\mathbf{v}(t)$ for $t = 0$, so that $\mathbf{v}(0) \in [0,1]^N$. Obtain $\mathbf{s} = \mathbf{A}'(\mathbf{A}\mathbf{A}')^{-1}\mathbf{b}$ and initialize $U = 1$ and $L = 1$.

- Step 2: calculate the energy gradient as $\nabla E(t) = -\mathbf{T}\mathbf{v}(t) - \mathbf{i}$.

- Step 3: update $k(t) = 1 - 2e^{-t/\tau}$, and generate a random $\alpha(t) \in [k(t), 1]$.

- Step 4: calculate $\mathbf{x} = \mathbf{v}(t) - \Delta t \alpha(t) \nabla E$.

- Step 5: perform the projection and clipping procedure of the following steps:

  - find the projection of $\mathbf{x}$ onto the constraint subspace: $\mathbf{x}^p = \mathbf{x} - \mathbf{Q}\mathbf{Q}'\mathbf{x} + \mathbf{s}$.
  - introduce $\mathbf{x}^p$ inside the unit hypercube modifying all its elements following:

$$x_i^p \leftarrow \begin{cases} U, & x_i^p \geq U, \\ L, & x_i^p \leq L, \\ \dfrac{x_i^p - L}{U - L}, & \text{otherwise.} \end{cases} \tag{2.41}$$

  - $\mathbf{x} \leftarrow \mathbf{x}^p$ and $\mathbf{e} = |\mathbf{A}\mathbf{x} - \mathbf{b}|$. If $e_i <$ tol, $\forall i = 1 \cdots N$, go to Step 6. Otherwise, start Step 5 again.

- Step 6: update neuron states as $\mathbf{v}(t+1) = \mathbf{x}$ and actualize $U \leftarrow U + \varepsilon_0$ and $L \leftarrow L + \varepsilon_0$ according to the periodicity described in [46].

- Step 7: $t \leftarrow t + 1$. Repeat from Step 2 until $k(t) = 1$ and $dV_i(t)/dt = 0$ for all i.

Comparing this procedure and the one proposed in [46], it can be noticed that in Step 5 the Gram-Schmidt procedure has been used instead of the direct projection, according to the explanation given in Section 2.3.3. This has been made in order to increase the efficiency of the procedure and allow a fair comparison among the three alternatives. It is also important to highlight that, by means of Step 3 and Step 4, an annealing-like procedure is implemented allowing punctual increments of the energy function. This mechanism was devised by Smith *et al.* to avoid local minima and increase the convergence probability. Step 5 is a projection and clipping procedure that converges to a point inside the unit hypercube and the constraints subspace. Updating U and L as shown in Step 6 makes the clipping more severe with each new iteration, hence forcing the neural network to converge.

This Thesis uses the same constant parameters defined in [46], namely, $\Delta t = 10^{-4}$, $\varepsilon_0 = 10^{-5}$, $\tau = 40$. The value of Δt is the same used in GP-HNN. Additionally, a tolerance value must be defined to detect the convergence of the procedure. For the case of this study $\mathrm{tol} = 10^{-4}$.

2.3.5 Study of the different alternatives using the MQP

This section compares the different approaches using the MQP. As previously mentioned in Section 2.3, for the MQP, neurons are usually organized in matrix form. Nevertheless this does not invalidate the vector notation of previous sections. In fact, the matrix form aims only at making easy writing and understanding the different terms of the energy function. Therefore, it is important to remember that, although some equations of this section use two indices to refer to a specific neuron output, the set of all neurons will be still grouped into the column vector $\mathbf{v}(t)$, just as in previous sections. The energy function used to solve this problem is [35]:

$$\begin{aligned} E\left(\mathbf{v}(t)\right) =& \frac{A}{2}\sum_{i=1}^{Q}\left(\sum_{j=1}^{Q}V_{ij}(t)-1\right)^2 + \frac{A}{2}\sum_{j=1}^{Q}\left(\sum_{i=1}^{Q}V_{ij}(t)-1\right)^2 \\ &+ \frac{B}{2}\sum_{i=1}^{Q}\sum_{j=1}^{Q}V_{ij}(t)\left(\sum_{\substack{1\le i-k\le N\\1\le j-k\le N\\k\neq 0}}^{Q}V_{i-k,j-k}(t)\right) \\ &+ \frac{B}{2}\sum_{i=1}^{Q}\sum_{j=1}^{Q}V_{ij}(t)\left(\sum_{\substack{1\le i-k\le N\\1\le j+k\le N\\k\neq 0}}^{Q}V_{i-k,j+k}(t)\right). \end{aligned} \tag{2.42}$$

The first term aims at allowing only one neuron active per column and, hence, only one queen in each column of the chessboard. Similarly, the second term tries to force only one neuron active in each row. The last two terms are focused on the diagonals of the chessboard. Whereas rows and columns must have exactly one queen each, diagonals can have one or zero queens. These terms are minimized in those situations.

The three approaches described in Section 2.3.4 are studied in this section. The performance of all approaches was tested in terms of number of iterations needed to reach equilibrium, probability of reaching a good solution and computational cost. These performance indicators were obtained using computer

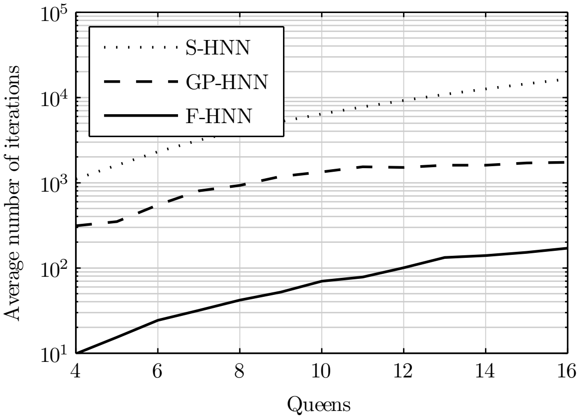

Figure 2.7: Average number of iterations of the three projection-based HNN alternatives.

simulations for different number of queens. More specifically, 5000 different initial states were used for each queens quantity ranging from $Q = 4$ to $Q = 16$. The linear constraints were the sum of rows and columns equal to 1. Thus the first two terms of (2.42) were always equal to 0.

Figure 2.7 shows the average number of iterations until an equilibrium state is reached. The differences between the algorithms are noteworthy. GP-HNN needs a number of iterations 10 times lower than S-HNN, whereas F-HNN needs 10 times less than GP-HNN and 100 times less than S-HNN. The good performance of F-HNN highlights the benefit of using a variable updating step. The high number of iterations of S-HNN is mainly due to the simulated annealing procedure since the system must be slowly "cooled".

The probability of reaching a good solution of the MQP is very similar for all the approaches. Specifically, S-HNN reached a good solution the 52.7% of times, GP-HNN reached them the 56.0% of times and F-HNN did it the 54.0% of times. It is worth noting that S-HNN has the worst behavior in spite of using a simulated annealing procedure. This fact is due to two main causes. First, the simulated annealing procedure is not as good as it could seem initially. The "cooling" procedure needs too many iterations to converge respect to the improvement in the probability of good solutions. Second, the mechanism of projection and clipping used by S-HNN for confining the neuron states into the constraints subspace produces severe instabilities. Although this procedure converges to a point, this point may be very different in two contiguous HNN

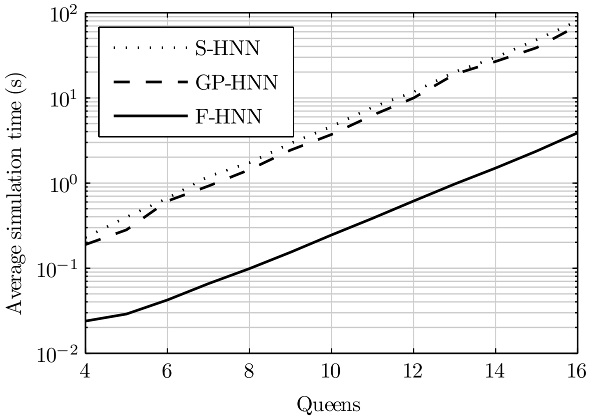

Figure 2.8: Average simulation time of the three projection-based HNN alternatives.

iterations. For that reason the clipping is more and more severe with every new iteration, what forces the neural network to converge. The main problem of this procedure is that the convergence may be forced even if the neuron states are far from a good solution. Another issue is that these probability values are not very high, what reflects the existence of many local minima in the energy function.

Finally, Figure 2.8 depicts some illustrative results of the computational cost of all the approaches. The three techniques were simulated in the same computer (Intel Core 2 Duo processor T7500 working at 2.2 GHz and with 4 GB of physical RAM) using a prototype in Matlab. As it can be observed, S-HNN improves its performance respect to Figure 2.7 and gets very close to GP-HNN. Therefore, although S-HNN needs many more iterations to converge, each iteration can be resolved faster. Nevertheless, this fact does not suffice for S-HNN to be the best approach. F-HNN still has the best behavior reducing more than 10 times the time needed by the other techniques.

2.4 Conclusion

This chapter has introduced the HNNs as they were conceived by Hopfield and has presented a possible implementation. Nowadays, digital devices are preferred instead of the original analog circuit of Hopfield. Nevertheless, digital devices need to sample the time variable. This fact may drastically increase the response time of these networks. The optimum neuron outputs update computed in Section 2.2.1 can reduce the number of iterations, but if this technique is combined with gradient projections the improvement can be notably higher. All this chapter was devoted to show that HNNs can be implemented and, hence, the JDRA algorithm of next chapter too.

This chapter has compared three different projection techniques, showing that the one combined with optimum updates, the F-HNN, is much better than the other two. Nevertheless, this study is not finished yet. F-HNNs should be compared with a HNN implemented with optimum updates only and a HNN with no improvement. Every new method (optimum updates and projections) added to HNNs reduce the number of iterations required to reach the equilibrium but, obviously, at the expenses of an increased computational cost in every iteration. During the development of this Thesis, a preliminary study of these three techniques showed that for the case of the MQP all three had a similar behavior in terms of response time. That is why this study has been left for the next chapter, where these techniques are compared using the JDRA algorithm (see Section 3.6).

Chapter 3

HNN-based JDRA algorithm

Previous chapter has presented the HNNs and several implementation issues. These type of networks can give good solutions in short times, especially if they are implemented over devices capable of working in parallel. This chapter will use these networks to define an energy function to solve the Joint Dynamic Resource Allocation (JDRA) problem efficiently.

Let I be the number of users demanding resources at a specific resource allocation time. All users will be distributed among K RANs that may belong to different radio technologies or not. The resources of all RANs are divided into minimum resource quanta, e.g. time slots in GPRS or 256 Spreading Factors (SFs) in UMTS. Due to this division, for the k-th RAN only one finite set exists, $\mathfrak{R}_k$, with all the feasible resource quantities that may be allocated to one user. For example, in GPRS only 0 up to 8 time slots can be allocated to one user, hence the finite set is $\mathfrak{R}_k = \{0, 1, 2, 3, 4, 5, 6, 7, 8\}$. Let J_k be the number of elements of $\mathfrak{R}_k$. The optimization problem consists in finding the best combination of RAN and quantity of resources that must be allocated to each user in order to satisfy its QoS requirements and system constraints.

3.1 QoS provision

Depending on the type of service, quality requirements can be concreted differently, for instance in terms of maximum packet delay or minimum bit rate. In order to have a common definition for all QoS requirements, this section will introduce the concept of minimum target bit rate. This minimum bit rate has to fulfill the user-specific QoS requirements. Therefore, it must be calculated independently for each user and type of QoS.

3.1.1 QoS based on minimum bit rate

If the i-th user requires a minimum instantaneous bit rate of $\underline{R}_i$, then the minimum target bit rate for that user is:

$$R_{\min,i} = \underline{R}_i. \tag{3.1}$$

Nevertheless, this tight condition is usually relaxed for actual services. For example, data transfer services using File Transfer Protocol (FTP) may require not an instantaneous minimum bit rate but an average one, $\overline{R}_i$. In this case $R_{\min,i}$ can be calculated using a leaky-bucket approach. $\overline{R}_i \Delta t$ tokens are generated each resource allocation time, being Δt the resource allocation period. On the other hand, $R_i(t)\Delta t$ tokens are spent each period, where $R_i(t)$ is the bit rate allocated to the i-th user at time t. Each available token can be understood as a bit that must be transmitted to reach the average minimum bit rate $\overline{R}_i$. The quantity of bits the system owes the i-th user is calculated as follows:

$$b_{\text{owed},i}\left(t+\Delta t\right) = \begin{cases} 0, & b_i(t) = 0, \\ b_{\text{owed},i}\left(t\right) + \left(\overline{R}_i - R_i(t)\right)\Delta t, & \text{otherwise}, \end{cases} \tag{3.2}$$

where $b_i(t)$ is the quantity of bits stored in the buffer of the i-th user at time t. From this definition, it is assumed that $b_{\text{owed},i}$ is reset at the beginning of every data burst. Thus, $\overline{R}_i$ is actually the minimum average bit rate per burst. The objective is that the quantity of owed bits are at most 0 at the burst end. If so, the average bit rate allocated to the user would be greater or equal to $\overline{R}_i$. $R_{\min,i}$ is the minimum bit rate that accomplishes this objective. Thus, if at time t the user is served with $R_{\min,i}$ until the burst end, then the burst would last $b_i(t)/R_{\min,i}$ additional seconds and:

$$b_{\text{owed},i}\left(t + \frac{b_i(t)}{R_{\min,i}}\right) = 0 = b_{\text{owed},i}\left(t\right) + \left(\overline{R}_i - R_{\min,i}\right)\frac{b_i(t)}{R_{\min,i}}. \tag{3.3}$$

Finally, from (3.3):

$$R_{\min,i} = \frac{b_i(t)\overline{R}_i}{b_i(t) - b_{\text{owed},i}(t)}. \tag{3.4}$$

It is worth noting that (3.4) makes sense only if $b_i(t) > b_{\text{owed},i}(t)$. If the quantity of owed bits is greater than those available for transmission then it would not be possible to achieve the objective bit rate $\overline{R}_i$. In that case, the allocated bit rate should be the greatest one in order to approach $\overline{R}_i$ as much as possible. Consequently, $R_{\min,i}$ can be defined as $R_{\min,i} = \infty$ if $b_i(t) \leq b_{\text{owed},i}(t)$.

3.1.2 QoS based on maximum delay

For delay-sensitive services, data packets must be transmitted before certain maximum delay, $D_{\max,i}$ for user i. Following a first-in first-out packet queue policy, although a packet arrives to the buffer, it will not be transmitted until all packets previously generated are sent. Thus, in order to transmit this packet within the $D_{\max,i}$ target seconds, the bit rate must be high enough to transmit also all previous packets, that is, higher or equal to:

$$\frac{\sum_{s=1}^{P_i} \beta_{s,i}}{D_{\max,i}}, \tag{3.5}$$

where P_i is the number of packets in the buffer and $\beta_{s,i}$ is the size in bits of the s-th packet of the i-th user. Therefore, $R_{\min,i}$ can be obtained following:

$$R_{\min,i} = \begin{cases} \max\limits_{p=1\cdots P_i} \dfrac{\sum_{s=1}^{p} \beta_{s,i}}{D_{\max,i} - D_{p,i}}, & D_{\max,i} > \max\limits_{p=1\cdots P_i} D_{p,i}, \\ \infty, & \text{otherwise}, \end{cases} \tag{3.6}$$

where $D_{p,i}$ is the time that the p-th packet of user i has been in the buffer.

3.2 Resources to bit rate mapping

Each RAN may have different amounts of resources available for distribution. Besides, the type of resource can be highly different from one RAT to another. Therefore, it is quite important to calculate the quantity of resources that each user requires by converting $R_{\min,i}$ to amount of resources or vice versa. Nevertheless, this is not a trivial issue since, in wireless systems, the user effective throughput, i.e. the average number of bits that a source can correctly transmit in a time interval, not only depends on the quantity of resources allocated to the user but also on the channel quality or SNIR. With worse channel conditions, for example with more interference or noise, the bit or frame error probability increases and, at some levels, communication can be unfeasible. For that reason, wireless systems have several Transport Modes (TMs) (coding rate and modulation) allowing low error protection and hence high bit rates for high SNIR and high error protection with low bit rates for low SNIR. Each TM implies a fixed nominal bit rate at the Medium Access Control (MAC) level per resource unit (r.u.). For example, the r.u. in GPRS is a time slot and the bit rate at the MAC level is proportional to the number of time slots allocated to each user. The process of selecting the most proper TM, i.e. the TM with the highest effective throughput for a given SNIR, is

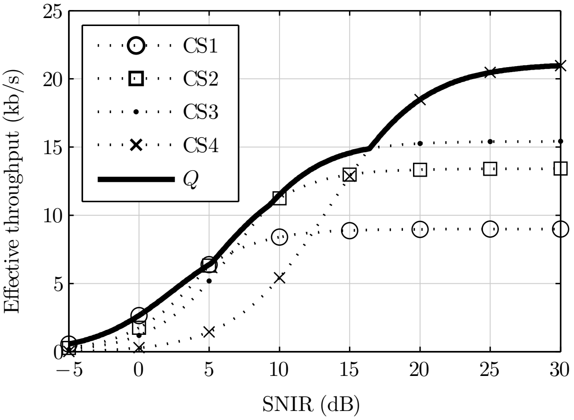

Figure 3.1: Effective throughput for the GPRS CS and the corresponding Q function [1].

known as Link Adaptation (LA). This Thesis assumes perfect LA. Therefore, the highest bit rate for any SNIR is always allocated to the users.

Let define $Q_k(C_{ik})$ as the effective bit rate that the i-th user is capable of achieving with a r.u. of the k-th RAN. C_{ik} is the channel SNIR, which measures the received signal quality. Q_k can be understood as a kind of Look-Up Table (LUT). Supposing a perfect LA, Q_k can be obtained as:

$$Q_k(C_{ik}) = \max_{s=1\cdots S_k} \frac{L_{sk}}{L_{sk}+O_{sk}} Br_{sk}\left(1 - Er_{sk}(C_{ik})\right), \tag{3.7}$$

where S_k is the number of TMs of the k-th RAN and Br_{sk}, Er_{sk}, L_{sk} and O_{sk} are respectively the bit rate per r.u., the error rate, the payload length and the header length of the s-th TM of the k-th RAN. Figure 3.1 shows an example of function Q_k for GPRS.

Once the required $R_{\min,i}$ and the current SNIR perceived by the user are known, the amount of r.u. to be reserved to the i-th user can be easily obtained dividing $R_{\min,i}$ by $Q_k(C_{ik})$.

3.3 HNN for JDRA

Previous works (see [32] and [38]) used 2-Dimensional (2D) HNNs for solving the DRA problem. This Thesis uses a natural evolution of these networks

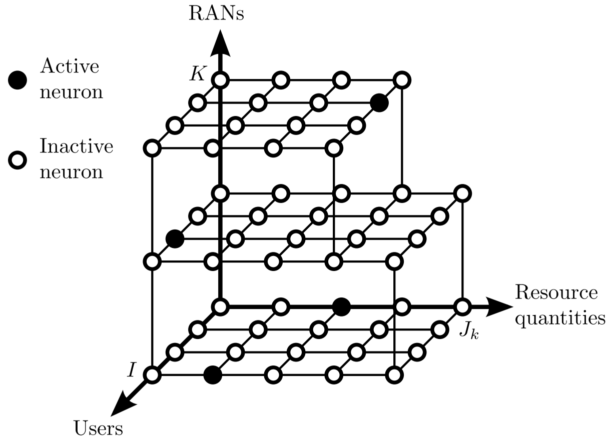

Figure 3.2: HNN and equilibrium example. 4 users ask for resources in 3 RANs. The first two RANs have 5 different resource quantities whereas the last RAN has only 4.

introducing the different RANs in a third dimension. Therefore, neurons are organized in a 3-Dimensional (3D) grid where the fact that the neuron at position (i, j, k) is active represents the allocation of the j-th resource quantity in the k-th RAN, i.e. the j-th element of $\mathfrak{R}_k$, to the user i. At the equilibrium the rest of neurons of user i must be inactive. Figure 3.2 shows an example with 4 active users demanding resources in 3 different RANs.

3.3.1 Energy function

Let define the following objective function that the HNN will minimize:

$$E_1 = -\frac{\mu_1}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} B_{ijk}V_{ijk} - \frac{\mu_2}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} \frac{Q_k(C_{ik})\rho_{jk}}{\max\limits_{lmn}\{Q_n(C_{ln})\rho_{mn}\}} V_{ijk}, \quad (3.8)$$

where V_{ijk} is the output of the neuron located at position (i, j, k), B_{ijk} is the benefit (see Section 3.3.2 for more details) the i-th user perceives in terms of QoS from the allocation of the j-th resource quantity of the k-th RAN, ρ_{jk} is the j-th resource quantity of the k-th RAN and μ_1 and μ_2 weight each term. As explained before, $Q_k(C_{ik})\rho_{jk}$ is the effective bit rate transmitted to user i for a given ρ_{jk}, whereas $\max_{lmn}\{Q_n(C_{ln})\rho_{mn}\}$ is constant in a resource allocation period regardless the value of i, j and k and aims at normalizing the cost of the second term. Minimizing (3.8), the resources allocated to each user will be determined pursuing two objectives, first maximizing the benefit from users' perspective and second maximizing the total throughput of the heterogeneous system. Note that this maximization is possible due to the negative sign of both terms.

Nevertheless, additional constraints have to be taken into account. First and most importantly, RAN resources are finite and, hence, the total amount of allocated resources must be controlled. To this aim a new term has to be added to (3.8):

$$E_2 = E_1 + \frac{\mu_3}{2} \sum_{i=1}^{I} \sum_{k=1}^{K} \sum_{j=1}^{J_k} \xi_{ijk} V_{ijk}, \tag{3.9}$$

where $\xi_{ijk} = 0$ if the k-th RAN has enough resources to allocate the j-th resource quantity to the i-th user given the current load determined by neuron outputs and $\xi_{ijk} = 1$ otherwise (see Section 3.3.3 for further details). In other words, this term penalizes allocations that imply exceeding the maximum available resources in any RAN.

Additionally, some resource quantities may not be allowed to some users. Let define ψ_{ijk} as a permission table where $\psi_{ijk} = 1$ if the j-th resource quantity of the k-th RAN should be prohibited to the i-th user and $\psi_{ijk} = 0$ otherwise. Then, the following objective function takes this effect into account:

$$E_3 = E_2 + \frac{\mu_4}{2} \sum_{i=1}^{I} \sum_{k=1}^{K} \sum_{j=1}^{J_k} \psi_{ijk} V_{ijk}. \tag{3.10}$$

Thanks to this term, the heterogeneous system can define different user priority sets (Gold, Silver and Bronze users) limiting the maximum allowed bit rate according to the user quota. Moreover, when a user starts a vertical handover procedure the connection could migrate from the original RAN to the new serving one. This process takes some non-negligible time. During RAN changes users cannot consume any resource for data transmission. Therefore, the permission table can be modified in accordance with this wasted time so that the user in a vertical handover is not capable of using some of the highest resource quantities. The amount of resource quantities prohibited will depend on the RAN reconfiguration time.

Finally, some additional terms must be introduced to ensure a rapid convergence to correct and stable neuron states. Neuron outputs V_{ijk} must be 0 or 1 at the equilibrium and furthermore, only one neuron must be active, i.e. $\sum_{k=1}^{K} \sum_{j=1}^{J_k} V_{ijk} = 1$, for each user. These two constraints can be introduced in the energy function with the two terms proposed in [32] resulting finally:

$$
\begin{aligned}
E = &-\frac{\mu_1}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} B_{ijk}V_{ijk} - \frac{\mu_2}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} \frac{Q_k(C_{ik})\rho_{jk}}{\max\limits_{lmn}\{Q_n(C_{ln})\rho_{mn}\}}V_{ijk} \\
&+ \frac{\mu_3}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} \xi_{ijk}V_{ijk} + \frac{\mu_4}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} \psi_{ijk}V_{ijk} \\
&+ \frac{\mu_5}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} V_{ijk}\left(1 - V_{ijk}\right) + \frac{\mu_6}{2}\sum_{i=1}^{I}\left(1 - \sum_{k=1}^{K}\sum_{j=1}^{J_k} V_{ijk}\right)^2 .
\end{aligned}
\tag{3.11}
$$

This energy function has been used in this Thesis to solve the JDRA problem. The weighting coefficients μ_1 to μ_6 must be carefully selected. Section 3.4 focuses on their calculation.

3.3.2 Benefit function

Once the QoS is homogenized in terms of minimum bit rate, $R_{\min,i}$, the benefit that users perceive depends on two main factors. Firstly, the higher the bit rate the higher the benefit and secondly, the lower the quality the user is perceiving, i.e. higher delay or lower average bit rate, the higher the achievable benefit in order to introduce some kind of priority between users.

First factor

As a consequence of this first factor, the benefit must bea monotically increasing function. In addition, a great difference in the benefit must exist between the bit rates capable of satisfying the users, i.e. those greater than the minimum target bit rate, and those that are unable to do so. This Thesis proposes the following benefit function to accomplish the requirements of this first factor:

$$
B^*_{ijk} = \frac{S\left(Q_k(C_{ik})\rho_{jk}, s_i, r_i\right) - S(0, s_i, r_i)}{S\left(R_{\max}, s_i, r_i\right) - S(0, s_i, r_i)}, \tag{3.12}
$$

$$
S(x, s, r) = \frac{1}{1 + e^{-s(x+r)}}, \tag{3.13}
$$

$$
s_i = \begin{cases} \dfrac{2\ln(9)}{R_{\min,i}}, & R_{\min,i} \le R_{\max,i}, \\ \dfrac{2\ln(9)R_{\min,i}}{R^2_{\max,i}}, & R_{\min,i} > R_{\max,i}, \end{cases} \tag{3.14}
$$

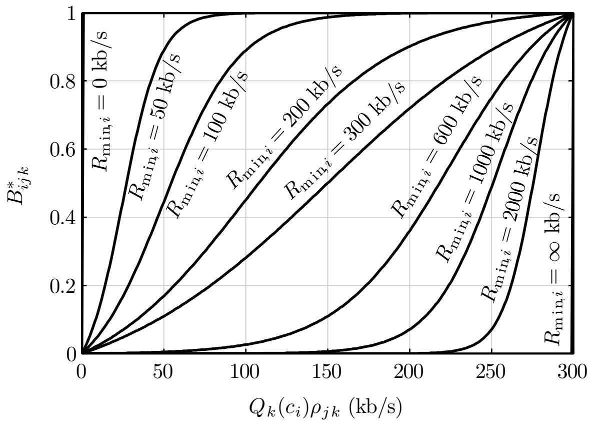

Figure 3.3: $R_{\min,i}$ effect on B^*_{ijk} for $R_{\max} = R_{\max,i} = 300$ kb/s.

$$r_i = \begin{cases} -\dfrac{R_{\min,i}}{2}, & R_{\min,i} \le R_{\max,i}, \\ -R_{\max,i} + \dfrac{R^2_{\max,i}}{2R_{\min,i}}, & R_{\min,i} > R_{\max,i}, \end{cases} \tag{3.15}$$

$$R_{\max} = \max_{ijk} \{Q_k(C_{ik})\rho_{jk}\}\,,\ R_{\max,i} = \max_{jk} \{Q_k(C_{ik})\rho_{jk}\}\,. \tag{3.16}$$

Note that S is the sigmoidal function. With this definition, B^*_{ijk} takes values in the interval $[0, 1]$ for all the resources quantities. Moreover, the s_i and r_i parameters were chosen to increase B^*_{ijk} significantly if $Q_k(C_{ik})\rho_{jk} \ge R_{\min,i}$. Figure 3.3 shows some examples of benefit functions with $R_{\max} = R_{\max,i} = 300$ kb/s and different values of $R_{\min,i}$. The figure shows how the sigmoidal function is scaled over the bit rate axis from a step function centered at 0 kb/s (for $R_{\min,i} = 0$ kb/s) to another step function centered at 300 kb/s (for $R_{\min,i} = \infty$ kb/s). Thus, when the QoS requirements cannot be satisfied and $R_{\min,i} = \infty$, then $B^*_{ijk} = 0$ for $Q_k(C_{ik})\rho_{jk} < R_{\max,i}$ and $B^*_{ijk} = 1$ for $Q_k(C_{ik})\rho_{jk} = R_{\max,i}$. Consequently, the allocation which maximizes the benefit is the maximum resource quantity.

Second factor

The second factor aims at increasing the fairness among users by making that two users with different qualities have different maximum benefits. Thus, users must be weighted inversely to their quality. Taking both factors into account, the benefit is defined in this Thesis as:

$$B_{ijk} = \frac{\min\left\{R_{\max}, R_{\min,i}\right\}}{\min\left\{R_{\max}, \max\limits_{i}\left\{R_{\min,i}\right\}\right\}} B^{*}_{ijk}. \tag{3.17}$$

B_{ijk} preserves the properties of B^{*}_{ijk} and additionally introduces weights for each user favoring those users with higher needs.

3.3.3 Resource saturation control

The saturation control mechanism uses the indicator ξ_{ijk} to know which resource allocations may be supported. The ξ_{ijk} indicator is calculated for each user i assuming that the rest of users, $l \neq i$, maintain the resource allocation of the current neuron outputs. Thus:

$$\xi_{ijk} = \begin{cases} 1, & \rho_{jk} + \sum\limits_{\substack{l=1 \\ l \neq i}}^{I} \sum\limits_{m=1}^{J_k} \rho_{mk} V_{lmk} > \rho_{\max,k}, \\ 0, & \rho_{jk} + \sum\limits_{\substack{l=1 \\ l \neq i}}^{I} \sum\limits_{m=1}^{J_k} \rho_{mk} V_{lmk} \leq \rho_{\max,k}, \end{cases} \tag{3.18}$$

where $\rho_{\max,k}$ is the maximum quantity of resources available in the k-th RAN. If $\xi_{ijk} = 1$ then ρ_{jk} cannot be supported for the i-th user with the current resource distribution. In that case, ξ_{ijk} increases the energy function and, consequently, the HNN tends to decrease V_{ijk}, what finally means not allocating ρ_{jk} to the i-th user.

3.4 Weighting coefficients

In order to obtain the weighting coefficients, the worst cases should be analyzed. For such cases, the chosen weights must ensure the desired behavior of the algorithm. First of all, μ_1 and μ_2 can be selected with certain freedom whereas the remaining weights will depend on these. In order to correctly select μ_1 and μ_2, it is necessary to decide upon the desired algorithm behavior. If QoS satisfaction is more important than throughput maximization, then $\mu_1 > \mu_2$. Furthermore, the greater the difference between these two weights, then the greater the significance of QoS satisfaction for the algorithm.

3.4.1 Fifth term

This term only aims to enhance the convergence speed of the neural network and must not prevent the change in neuron output, from 0 to 1 or vice versa, if the rest of the terms point to this. Let define $(i, j_{\text{high}}, k_{\text{high}})$ and $(i, j_{\text{low}}, k_{\text{low}})$ as two neurons belonging to the same user with bit rates $R_{\text{high}} = Q_{k_{\text{high}}}(C_{ik_{\text{high}}})\rho_{j_{\text{high}}k_{\text{high}}}$ and $R_{\text{low}} = Q_{k_{\text{low}}}(C_{ik_{\text{low}}})\rho_{j_{\text{low}}k_{\text{low}}}$ respectively, $R_{\text{high}} > R_{\text{low}}$, and if neither of these exceeds the maximum resources, the energy gradient of both neurons is:

$$\frac{\partial E}{\partial V_{ij_{\text{high}}k_{\text{high}}}} = -\frac{\mu_1}{2}B_{ij_{\text{high}}k_{\text{high}}} - \frac{\mu_2}{2}\frac{R_{\text{high}}}{R_{\max}} + \frac{\mu_5}{2}\left(1 - 2V_{ij_{\text{high}}k_{\text{high}}}\right) + C, \quad (3.19)$$

$$\frac{\partial E}{\partial V_{ij_{\text{low}}k_{\text{low}}}} = -\frac{\mu_1}{2}B_{ij_{\text{low}}k_{\text{low}}} - \frac{\mu_2}{2}\frac{R_{\text{low}}}{R_{\max}} + \frac{\mu_5}{2}\left(1 - 2V_{ij_{\text{low}}k_{\text{low}}}\right) + C, \quad (3.20)$$

where in C are grouped the rest of terms which are equal to both neurons. The optimum allocation is R_{high} since this maximizes the throughput. In the worst case scenario, both bit rates are equally valid for the QoS satisfaction, i.e. $B_{ij_{\text{high}}k_{\text{high}}} = B_{ij_{\text{low}}k_{\text{low}}}$. Assuming that $V_{ij_{\text{high}}k_{\text{high}}} = 0$ and $V_{ij_{\text{low}}k_{\text{low}}} = 1$, to ensure the correct allocation of R_{high}:

$$\frac{\partial E}{\partial V_{ij_{\text{high}}k_{\text{high}}}} < \frac{\partial E}{\partial V_{ij_{\text{low}}k_{\text{low}}}} \Rightarrow \mu_5 < \frac{\mu_2}{2}\frac{\min\left\{R_{\text{high}} - R_{\text{low}}\right\}}{R_{\max}}. \quad (3.21)$$

3.4.2 Third term

In order to allocate a bit rate not exceeding the maximum resources, at least one neuron must be favored (either increasing faster or decreasing slower) over the neurons exceeding the maximum resources. Supposing that all bit rates are in the permission table of user i, then in the case of the favored neuron $(i, j_{\text{fav}}, k_{\text{fav}})$, the energy gradient would be:

$$\frac{\partial E}{\partial V_{ij_{\text{fav}}k_{\text{fav}}}} = -\frac{\mu_1}{2}B_{ij_{\text{fav}}k_{\text{fav}}} - \frac{\mu_2}{2}\frac{R_{\text{fav}}}{R_{\max}} + \frac{\mu_5}{2}\left(1 - 2V_{ij_{\text{fav}}k_{\text{fav}}}\right) + C, \quad (3.22)$$

where $R_{\text{fav}} = Q_{k_{\text{fav}}}(C_{ik_{\text{fav}}})\rho_{j_{\text{fav}}k_{\text{fav}}}$. On the other hand, the energy gradient of the neurons exceeding the maximum resources $(i, j_{\text{exc}}, k_{\text{exc}})$ would be:

$$\frac{\partial E}{\partial V_{ij_{\text{exc}}k_{\text{exc}}}} = -\frac{\mu_1}{2}B_{ij_{\text{exc}}k_{\text{exc}}} - \frac{\mu_2}{2}\frac{R_{\text{exc}}}{R_{\max}} + \frac{\mu_3}{2} + \frac{\mu_5}{2}\left(1 - 2V_{ij_{\text{exc}}k_{\text{exc}}}\right) + C, \quad (3.23)$$

where $R_{\text{exc}} = Q_{k_{\text{exc}}}(C_{ik_{\text{exc}}})\rho_{j_{\text{exc}}k_{\text{exc}}}$. As such, the condition needed to guarantee the allocation of the correct bit rate is:

$$\frac{\partial E}{\partial V_{ij_{\text{fav}}k_{\text{fav}}}} < \frac{\partial E}{\partial V_{ij_{\text{exc}}k_{\text{exc}}}}. \quad (3.24)$$

The worst case scenario can be found where $B_{ij_{\text{fav}}k_{\text{fav}}} = 0$, $B_{ij_{\text{exc}}k_{\text{exc}}} = 1$, $R_{\text{fav}} = 0$, $R_{\text{exc}} = R_{\max}$, $V_{ij_{\text{fav}}k_{\text{fav}}} = 0$ and $V_{ij_{\text{exc}}k_{\text{exc}}} = 1$. In this case:

$$\mu_3 > \mu_1 + \mu_2 + 2\mu_5. \tag{3.25}$$

3.4.3 Sixth term

Despite the existence of enough resources, users should never have more than one resource quantity allocated, or in terms of the neural network, more than one neuron active. The sixth term is minimum when all the neuron outputs of a user sum one. At these points this term and its derivative are zero. As the first two terms continuously increase the neuron outputs and in the event that neither the third nor the fourth term can reduce them, then all neurons begin to increase their value pushing the outputs away from the desired value for the sum of neurons output. Considering δ as the maximum desired distance from the desired sum value, then equilibrium is achieved when $\left|1 - \sum_{k=1}^{K} \sum_{j=1}^{J_k} V_{ijk}\right| < \delta$. For satisfactory performances, δ should be lower than 1 or even lower than 0.5. With this objective in mind, the following condition needs to be satisfied for the worst case scenario:

$$\left|-\frac{\mu_1}{2} - \frac{\mu_2}{2}\right| < \mu_6\delta \Rightarrow \mu_6 > \frac{\mu_1 + \mu_2}{2\delta}. \tag{3.26}$$

3.4.4 Fourth term

This term must decrease the neuron output if $\psi_{ijk} = 1$, even if the other terms increase it. The worst case is $B_{ijk} = 1$, $Q_k(C_{ik})\rho_{jk} = R_{\max}$ and $\xi_{ijk} = 0$. Here the energy gradient results in:

$$\frac{\partial E}{\partial V_{ijk}} = -\frac{\mu_1}{2} - \frac{\mu_2}{2} + \frac{\mu_4}{2} + \frac{\mu_5}{2}(1 - 2V_{ijk}) - \mu_6\left(1 - \sum_{n=1}^{K}\sum_{m=1}^{J_n} V_{imn}\right) > 0. \tag{3.27}$$

Since $\mu_6 > \mu_5$, the worst case for the neuron outputs is $V_{ijk} = 0$, $\forall j, k$. Finally, μ_4 can be obtained as:

$$\mu_4 > \mu_1 + \mu_2 - \mu_5 + 2\mu_6. \tag{3.28}$$

3.5 Simulation environment

3.5.1 Technologies used in the simulations

The proposed JDRA algorithm was tested in an artificial environment by means of computer simulations. The simulated heterogeneous network comprised two

RATs: HSDPA and 802.11e WLAN. 802.11e WLANs use convolutional codes to protect data from errors. Furthermore, the Packet Error Rate (PER) depends not only on the channel quality but also on the payload length. Assuming a Viterbi decoding at the receiver, the PER of the s-th TM is [47]:

$$\mathrm{PER}_s\left(L_{sk}, C_{ik}\right) = 1 - \left(1 - P_s^u\left(C_{ik}\right)\right)^{L_s}, \tag{3.29}$$

where P_s^u is the bit error probability of the s-th TM. The optimum payload length that maximizes the throughput for each TM is [47]:

$$L_{sk}^*\left(C_{ik}\right) = -\frac{O_{sk}}{2} + \frac{1}{2}\sqrt{O_{sk}^2 - \frac{4O_{sk}}{\ln 1 - P_s^u\left(C_{ik}\right)}}. \tag{3.30}$$

Finally, from (3.7) and (3.30), function Q_k is for WLAN:

$$Q_k\left(C_{ik}\right) = \max_{s=1\cdots S_k} \frac{L_{sk}^*\left(C_{ik}\right)}{L_{sk}^*\left(C_{ik}\right) + O_{sk}} Br_{sk}\left(1 - P_s^u\left(C_{ik}\right)\right)^{L_{sk}^*\left(C_{ik}\right)}. \tag{3.31}$$

The available resources in WLAN are slots of channel occupancy which are collision free thanks to the use of the HCF Controlled Channel Access (HCCA) mechanism.

HSDPA uses turbo codes instead of convolutional codes to protect data from errors. The Block Error Rate (BLER) depends also on the block size and on the channel quality. Nevertheless, each TM has a fixed block size and, hence, the BLER for a specific TM is only a function of the channel quality. HSDPA has a wide range of possible TMs, from which 30 have been defined in the standard as Channel Quality Indicators (CQIs). Only these 30 TMs are used in this Thesis. The BLER of the s-th CQI can be approximated as [48]:

$$Er_{sk}\left(C_{ik}\right) = \left(10^{2^{\frac{C_{ik} - 1.03s + 17.3}{\sqrt{3} - \log\left(C_{ik}\right)}}} + 1\right)^{-\frac{1}{0.7}}. \tag{3.32}$$

Users are supposed to be time multiplexed. Thus, the 15 available codes[1] are always allocated to a unique user each 2 ms. This assumption implies that the actual BLER differs from the one obtained in [48], since BLER is a function of the SNIR per code. If more codes are allocated then more SNIR is needed to maintain the same SNIR per code. Therefore, allocating 15 codes, (3.32) has to be modified to:

$$Er_{sk}\left(C_{ik}\right) = \left(10^{2^{\frac{C_{ik} - 10\log\left(\frac{15}{N_{sk}}\right) - 1.03s + 17.3}{\sqrt{3} - \log\left(C_{ik}\right)}}} + 1\right)^{-\frac{1}{0.7}}, \tag{3.33}$$

[1] Actually, there are 16 codes but one of them is reserved for control

Table 3.1: Number of codes and bit rate of each CQI.

CQI	N_{sk}	Br^*_{sk}(kb/s)	CQI	N_{sk}	Br^*_{sk}(kb/s)	CQI	N_{sk}	Br^*_{sk}(kb/s)
1	1	68.5	11	3	741.5	21	5	3277.0
2	1	86.5	12	3	871.0	22	5	3584.0
3	1	116.5	13	4	1139.5	23	6	4859.5
4	1	158.5	14	4	1291.5	24	7	5709.0
5	1	188.5	15	5	1659.5	25	10	7205.5
6	1	230.5	16	5	1782.5	26	13	8774.0
7	2	325.0	17	5	2094.5	27	15	10877.0
8	2	396.0	18	5	2332.0	28	15	11685.0
9	2	465.5	19	5	2643.5	29	15	12111.0
10	3	631.0	20	5	2943.5	30	15	12779.0

where N_{sk} is the number of codes of the s-th CQI, shown in Table 3.1. Moreover, if all the codes are allocated to a unique user, bit rates of the considered TMs also differ from the standard. Note that the new bit rates are:

$$Br_{sk} = Br^*_{sk}\frac{15}{N_{sk}}, \tag{3.34}$$

where Br^*_{sk} is the s-th CQI bit rate, shown also in Table 3.1. Finally function Q_k is for HSDPA:

$$Q_k\left(C_{ik}\right) = \max_{s=1\cdots 30} Br_{sk}\left(1 - Er_{sk}\left(C_{ik}\right)\right). \tag{3.35}$$

Since the JDRA algorithm allocates all codes to any user every 2 ms, then the quantity of available resources is 500 periods of 2 ms each second, where one resource element is one period of 2ms.

3.5.2 Reference algorithms

The proposed JDRA algorithm was compared with different combinations of RAT selection policies and uni-RAT DRA algorithms. The UMA solution and Maximum Bit Rate (MBR) policy were used for RAT selection. As explained in Section 1.2.1, UMA terminals select WLAN when they are in the coverage area of an AP. With the MBR policy, the user connects to the RAN that could transmit with the highest bit rate given the current channel quality of the user. Once users are distributed among the available RANs using UMA or MBR policy, a DRA algorithm was applied to perform scheduling and allocate resources inside each RAN. As DRA algorithms, MLWDF, CLSA (both were introduced in Section 1.2.3) and the proposed HNN for only one technology were selected. This choice was motivated because MLWDF and CLSA take simultaneously into account channel quality, QoS satisfaction and the type of service of the user. Besides, it was important to make a fair comparison including other algorithms that were also able to cope with a composite of bit rate and delay-based services.

Finally, combining all the possibilities, six different reference algorithms were defined: UMA-MLWDF, UMA-CLSA, UMA-HNN, MBR-MLWDF, MBR-CLSA and MBR-HNN.

3.5.3 Scenario

The simulation scenario comprised 7 cells with the cell under study in the center, i.e. users are moving only inside the cell center whereas the other 6 cells jam the cell center assuming that they constantly consume half the available resources. Each cell had 2 RANs, one HSDPA and another WLAN, being both, the HSDPA base station and the WLAN access point, at the cell center. Users were time multiplexed in both technologies. The studied services were web browsing and FTP data downloading, whose traffic models were extracted from [49]. Two different user classes were defined for each service. The QoS requirements for web users were a maximum delay of 30 s and 60 s for the entire web page, whereas FTP users expect a minimum average bit rate per burst of 150 kb/s or 50 kb/s depending on the service class. All users were randomly assigned to one of these services and classes with equal probability. $R_{\min,i}$ was obtained from (3.4) and (3.6) for FTP and web users respectively. Moreover, MLWDF and CLSA algorithms can use different weights for each service. For these simulations, these weights were extracted from [27], i.e. 1 for web and 0.8 for FTP. Regarding mobility modeling, two different areas were considered, the cell with radius 500 m and a hotspot at the cell center with radius 50 m. Users moved with a random constant speed uniformly distributed between [0,50] km/h. Further details of the mobility model are provided in Section 3.5.4.

The maximum transmitted power was set to 43 dBm for the HSDPA base station and 20 dBm for the WLAN AP. Interfering cells were supposed to transmit half the maximum power being, therefore, half loaded. Noise power at the receiver was set to -102 dBm and -95 dBm for HSDPA and WLAN respectively, as a consequence of the different bandwidth. The path losses of each user were obtained as in [50], i.e. the values expressed in decibels for the i-th user in both technologies were:

$$L_{\mathrm{HSPDA},i} = 137.4 + 35.2 \log\left(d_{\mathrm{HSPDA},i}\right), \tag{3.36}$$

$$L_{\text{WLAN},i} = 135 + 45 \log \left(d_{\text{WLAN},i} \right), \tag{3.37}$$

where $d_{\text{HSPDA},i}$ and $d_{\text{WLAN},i}$ are the distances in km from the i-th user to the base station and the AP. Additionally, the large-scale fading effect is modeled using the Gudmundson approach [51], assuming a standard deviation of 8 dB.

The proposed HNN-based algorithm was run every simulation second. Therefore, RAN changes may only occur every second at most. Besides, users were supposed to spend 0.5 s completing a vertical handover procedure. For the reference algorithms the RAT selection procedures were run also every second, whereas DRA algorithms were run every 0.1 s for computational simplicity reasons. In order to have access to the same set of solutions, the sets of resource quantities were reduced to $\mathfrak{R}_k = \{0, 0.1M_k, 0.2M_k, \cdots, M_k\}$ for both RATs, where M_k is the quantity of available resources in one second.

Each simulation iteration corresponds to 0.1 s. Thus, traffic data generated by users during each 0.1 s is supposed to be generated at the beginning of those 0.1 s. Additionally, users location is updated every 0.1 s too. Note that this time period is enough, no smaller periods are necessary since no fast fading is emulated in the channel. The fast fading is already taken into account with the use of effective throughputs.

3.5.4 Mobility model

Mobility is one of the key characteristic of wireless systems. It produces most of the effects that make quality fluctuate, like signal fading or handovers. Thus, it is very important to emulate user mobility correctly, and to know which are the characteristics of the mobility model used to perform system simulations.

Mobility is not only important in wireless systems but also in other areas like transport, study of migratory birds or even hurricanes [52]. This fact has originated the creation of a huge quantity of different mobility models with different applications. In all them, the mobile entities are usually referred to as mobile nodes or just nodes.

Mobility models

The mobility models used in simulations of wireless networks can be roughly classified into independent or group-based. Independent models characterize the movement of each node independently of the rest of nodes. On the other hand, group models generate some dependence between the movement of certain nodes.

Some independent mobility models are:

- Random walk: Each node moves from its current location to a new location by randomly choosing an arbitrary direction and speed from a given range. Such a move is performed either for a constant time or traveled distance. Then new speed and direction are chosen. At the boundaries nodes bounce off like billiard balls on a pool table. Further description of this model can be found in [53]. This model is the simplest one and many other variations have been proposed to perform a better emulation of nodes mobility.

- Random waypoint [54]: In this model, nodes wait for some random time and then chose a new destination moving towards it with random speed. When the destination has been reached, the process starts again.

- Boundless simulation area mobility model [2]: The model exchanges the planar rectangular simulation area by a boundless torus. This way, nodes that reach one side of the simulation area continue traveling and reappear on the opposite side.

- Smooth random mobility model [55]: This model is basically an extension of the simple random walk model. Here two independent stochastic processes are used to trigger direction and speed changes. The new speeds - for example - are chosen from a weighted distribution of preferred speeds. Upon such a trigger, the speed - or direction - changes are determined by a Poisson process.

- Random Gauss-Markov mobility model [56]: This model enhances the smooth random mobility model. Nodes next location are predicted - or generated - by its past location and velocity. Depending upon the parametrization, this allows modeling along a spectrum from random walk to fluid-flow.

Group mobility models are usually an extension of the above models. An exception to this is the fluid-flow mobility model. This model represents the behavior of all nodes at the same time using flow equations. This approach can be only used if the movement of individual nodes is not relevant. The behavior of the generated traffic is similar to a fluid flowing through a pipe. As a result, the fluid-flow mobility model represents traffic on highways very well [52]. Some other group models are [52]:

- Exponential correlated random mobility model: A motion function creates a group behavior.

- Column mobility model: The set of mobile nodes form a line and move forward in a particular direction.

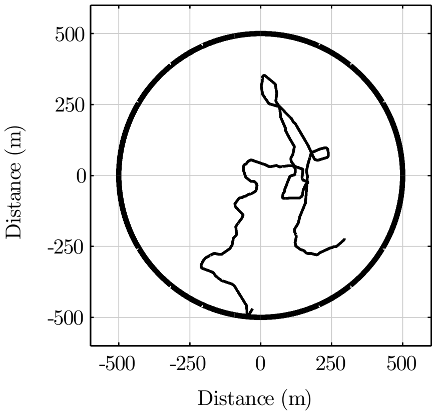

Figure 3.4: Typical path followed by a node. The thick line shows the simulation area border.

- Nomadic community mobility model: A group mobility model where a set of nodes move together from one location to another.
- Pursue mobility model: For each group all members follow a target node moving around the simulation area.
- Reference point group mobility model: The group movement is based upon the path traveled by a logical center. The logical center moves according to an independent mobility model.

The mobility model that is finally selected has severe impact on the results obtained from simulations. This fact can make the same scenario exhibit very different performance depending on the mobility model. For that reason, standardization bodies propose their own models to test the performance of their technologies. Then, different institutions can make reasonable fair studies that could be compared. These models are usually a mix of the previous models and try to take advantage of the main benefit of each model.

For instance, the European Telecommunications Standards Institute (ETSI) proposes a model in [50] that was also used in the MORANS activity of COST 273 [57]. This model is the one used in this Thesis. The model is similar to the random walk. Nodes movement is generated independently. At the initial state, nodes are homogenously distributed among the simulation area and select a random direction between 0 and 2π and a random speed between a minimum and maximum speeds (between 0 and 50 km/h for the case of Section 3.5.3).

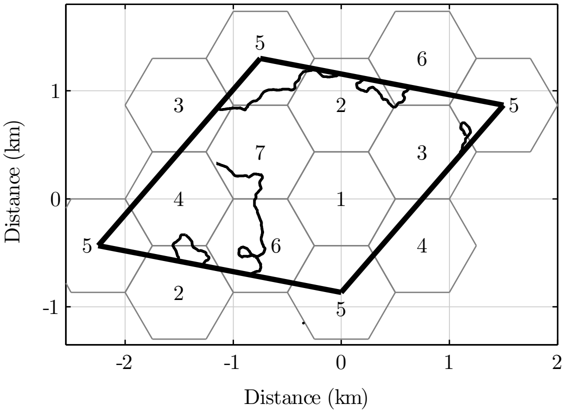

Figure 3.5: Typical path followed by a node in a scenario with seven cells. This figure shows seven cells that are replicated following the idea of the boundless simulation area mobility model of [2].

After this initial state, nodes start moving with the selected direction. In every simulation iteration, all nodes have some chance to change their direction. This chance is calculated so that the mean time spent in the same direction is 5 s. In the scenario presented in Section 3.5.3, the simulation iterations have a length of 0.1 s. Hence the mean number of iterations without changing direction is 50. Therefore, the probability of changing direction must be 1/50, i.e. 0.02. When a node changes its direction, a new one is selected from $-\pi/4$ to $\pi/4$ with respect to the current direction. Figure 3.4 shows an example of the path followed by a node with the model presented in this section. This example is particularized for the scenario explained in Section 3.5.3. Thus, the simulation area is a circle of radius 500 m and nodes bounce when they arrive to the border. Nevertheless, simulations carried out in Chapter 6 comprise seven cells with nodes moving between them freely. In that case, the simulation area will be a quadrilateral and instead of bouncing, nodes reappear in the opposite side following the torus idea of [2]. Figure 3.5 depicts this scenario with a node moving freely all around the cells.

Hotspot modeling

The simulation scenarios used in this Thesis have certain zones with more node (or user) density than the rest of the simulation area. This zones have an elevated load and are known as hotspots. Among all the aforementioned mobility models, only group-based models can somehow emulate hotspots. Nevertheless, most of them do not provide a mechanism to control the size of the hotspot or even the node density. Another important drawback is that if a node is attached to some group, it remains attached to it until the end of the simulation. That means that nodes can not exit the hotspot and come back. It is important to use a model that allows nodes to behave that way, mainly for the simulations performed in the following chapters with the JCAC algorithm. If not, if the hotspot area is a zone with good quality, nodes within it will always receive good quality and will never go to other areas with worse conditions. Moreover, nodes moving throughout the simulation area will always perceive bad quality. This fact will produce a big difference in the quality perceived by different users what may have severe effects in the results and hence induce wrong conclusions.

Another possibility is to create a small simulation area for the hotspot with some predefined size and to generate as many nodes as necessary to have the desired node density. This procedure is widely used (see [57] as an example) but has the same problem explained before, i.e. nodes inside the hotspot will never exit. Other mobility models, like [58] and [59], generate the movement of nodes independently and emulate hotspots. With these models, nodes move away the hotspot and come back independently of the rest of nodes. Nevertheless, hotspot size and node density cannot be controlled. Probably, the most complete model is the one proposed by Hyytiä *et al.* [60]. This model is based on the random waypoint. They proposed to increase the node density inside the hotspot by decreasing the nodes speed when they enter the hotspot or by increasing the waiting time of waypoints in the hotspot. This mechanism allows nodes to enter and exit the hotspot. Moreover it is possible to compute the speed reduction or waiting time increase needed to have certain node density in the hotspot area. The main problem of this model is that it inherits all the drawbacks of the random waypoint model, most of all that nodes tend to concentrate in the center of the simulation area [59]. This fact also affects results and hence can induce wrong conclusions too [61].

So far the hotspot models currently available in the literature do not accomplish the two requirements exposed here at the same time, i.e. nodes should enter and leave the hotspot and the node density should be completely configurable. Therefore, this Thesis proposes a new model in order to satisfy the necessities exposed here. The model is used over other mobility models and allows nodes to move outside hotspots and node density to be defined in all the simulation area.

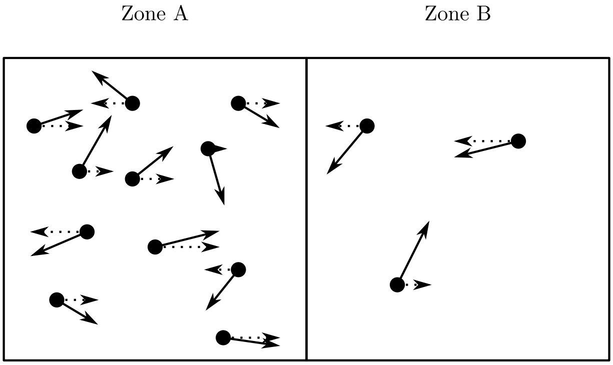

Figure 3.6: Simulation area with two zones with different node density.

Model description

This new approach is based on flow theory, although it is completely different to the fluid-flow mobility model. The idea is to make a normal use of mobility models but assuming that nodes may suddenly bounce off *imaginary* bounds. This forced bouncing prevents nodes from moving outside certain zones and hence, can create different node densities. The mobility models that are used under this idea must create an homogeneous node density if they operate normally. That is the case of the random walk - with or without a boundless simulation area - but not of the random waypoint. Thus, the mobility models proposed by the ETSI in [50] and by the COST 273 in [57] are the preferred option due to its wide usage.

In order to explain the model, let us start with a simpler case before moving into the general case. Therefore, let us focus on the simulation area presented in Figure 3.6. This area is divided into two square zones with a side in common. Zone A has more nodes than zone B. Nodes are moving with random direction and speed. The quantity of nodes that will move from zone A to B in certain time interval Δt depends directly on the velocity component that is perpendicular to the border between zones. Figure 3.6 shows this component with dotted lines. Let us assume that the average speed in zone A is v_A, then the average perpendicular component is:

$$v_A^{\perp} = \frac{1}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} v_A \cos\left(\gamma\right) d\gamma = \frac{2v_A}{\pi}. \tag{3.38}$$

Thus, in average, the quantity of nodes that will move from zone A to B in the time interval Δt are half of those that, at most, are at a distance $2v_A \Delta t/\pi$

from the border. Half of them because the other half is moving to the opposite direction. Note that nodes follow random directions. This quantity is:

$$\Delta n_A = \frac{v_A \Delta t \delta_A L}{\pi}, \tag{3.39}$$

where δ_A is the node density in zone A and L is the length of one side of the squares. Analogously, the quantity of nodes moving from zone B to zone A is:

$$\Delta n_B = \frac{v_B \Delta t \delta_B L}{\pi}. \tag{3.40}$$

If $\Delta n_A = \Delta n_B$ the scenario of Figure 3.6 is in equilibrium. Nodes may change their zone but node densities will not vary, i.e. the quantity of nodes that move from A to B is exactly the same quantity of nodes coming back from B to A. The only parameter available to force the equilibrium is the velocity. Thus, nodes should have different speeds in each zone, depending on the desired densities. If we want velocity and density to be independent, then, a new degree of freedom must be introduced in these equations. Let $\rho_{A\to B}$ and $\rho_{B\to A}$ be the probabilities that any node of zone A and B respectively do not bounce off the border between zones. Then, the previous expressions must be rewritten as:

$$\Delta n_A = \frac{v_A \Delta t \delta_A L \rho_{A\to B}}{\pi}, \tag{3.41}$$

$$\Delta n_B = \frac{v_B \Delta t \delta_B L \rho_{B\to A}}{\pi}. \tag{3.42}$$

Hence, the system is in equilibrium if:

$$\frac{\rho_{A\to B}}{\rho_{B\to A}} = \frac{v_B \delta_B}{v_A \delta_A}. \tag{3.43}$$

Then, if the previous expression is grater than 1, $\rho_{A\to B}$ can be set to 1 and $\rho_{B\to A}$ can be computed from (3.43). On the other hand, if the expression is lower than 1, then $\rho_{B\to A}$ is set to 1 and $\rho_{A\to B}$ is computed from (3.43). This simple example is useful for introducing the idea of the model proposed here. The heterogeneous node distribution is achieved forcing some nodes to bounce, thus maintaining the high node density of certain zones. It is worth noting that this is not a mobility model by itself. This approach just sets some probabilities of bouncing at points where the product of velocity and node density changes. Now, imagine that on the right of zone B there is an additional zone C. The probability that a node in zone A reaches zone C is:

$$\rho_{A\to C} = \rho_{A\to B}\rho_{B\to C}. \tag{3.44}$$

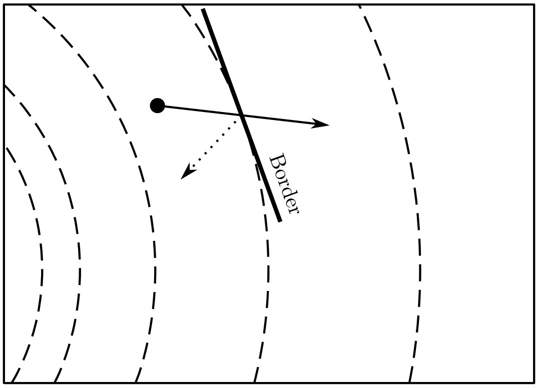

Figure 3.7: Example of a node bouncing. The striped lines are the contour lines of the product velocity-density. The dotted arrow shows the destination after bouncing.

Thus, from 3.43:

$$\frac{\rho_{A\to C}}{\rho_{C\to A}} = \frac{\rho_{A\to B}}{\rho_{B\to A}} \cdot \frac{\rho_{B\to C}}{\rho_{C\to B}} = \frac{v_B\delta_B}{v_A\delta_A} \cdot \frac{v_C\delta_C}{v_B\delta_B} = \frac{v_C\delta_C}{v_A\delta_A}. \tag{3.45}$$

Consequently, the ratio between the probabilities of both directions depends only on the densities and velocities at the zones in the extremes of the path followed by the node. Note that this conclusion is only true if no zone in the path has a null mean velocity or node density. This property can be used to generate the movement of nodes in a simulation area with any density and velocity distributions.

General case

Let $\mathcal{A} \in \mathbb{R}^2$ be the simulation area and $v(r)$ and $\delta(r)$ be the mean velocity and node density respectively at point $r = (x, y)' \in \mathcal{A}$, where x' is the transpose of x. Then, if a node is at point $r_1 = (x_1, y_1)'$ and, following the mobility model, the node should move to $r_2 = (x_2, y_2)'$ in the next simulation iteration, the probability of this happening is:

$$\rho_{r_1\to r_2} = \min\left\{\frac{v(r_2)\delta(r_2)}{v(r_1)\delta(r_1)}, 1\right\}. \tag{3.46}$$

Note that (3.46) is equivalent to the conclusions arisen from (3.45). The main difference is that now zones are infinitesimally small having, thus, one zone at each point of the simulation area. For that reason the mean velocity and density are now represented as functions of the node location.

If the node bounces off, then this may happen at any point between r_1 and r_2. For the sake of simplicity this Thesis assumes that the bouncing always occurs at the middle point, i.e. $r_m = (r_1{+}r_2)/2$. The border that nodes bounce off is tangential to the contour lines of the product velocity-density or, in other words, perpendicular to the gradient of this product. Figure 3.7 depicts how a node bounces with this model.

The gradient can be computed analytically although it is possible to approximate it with intervals, simplifying the calculus. Let us define the points r_a and r_b as $r_a = (x_2, y1)'$ and $r_b = (x_1, y_2)'$. Then, the gradient at r_m can be approximated by the vector:

$$\nabla\left(v(r_m)\delta(r_m)\right) \approx \begin{pmatrix} g_x(r_m) \\ g_y(r_m) \end{pmatrix}, \tag{3.47}$$

where:

$$g_x(r_m) = \frac{v(r_a)\delta(r_a) - v(r_1)\delta(r_1)}{x_2 - x_1}, \tag{3.48}$$

$$g_y(r_m) = \frac{v(r_b)\delta(r_b) - v(r_1)\delta(r_1)}{y_2 - y_1}. \tag{3.49}$$

The fastest way to compute the points nodes will reach after bouncing is by means of a rotation. Using the rotation matrix:

$$R = \frac{1}{\sqrt{g_x^2(r_m) + g_y^2(r_m)}} \begin{pmatrix} g_x(r_m) & g_y(r_m) \\ -g_y(r_m) & g_x(r_m) \end{pmatrix}, \tag{3.50}$$

the gradient will point towards the direction of the abscissa axis. Figure 3.8 shows an example of such rotations. Afterwards, the node movement after bouncing is equivalent to a movement parallel to the ordinate axis. Then, the rotation can be undone with the inverse of R. These three transformations can be concatenated mathematically as the product of the following matrices:

$$T = R^{-1} \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} R, \tag{3.51}$$

$$T = \begin{pmatrix} \dfrac{g_y^2(r_m)}{g_x^2(r_m) + g_y^2(r_m)} & -\dfrac{g_x(r_m)g_y(r_m)}{g_x^2(r_m) + g_y^2(r_m)} \\ -\dfrac{g_x(r_m)g_y(r_m)}{g_x^2(r_m) + g_y^2(r_m)} & \dfrac{g_x^2(r_m)}{g_x^2(r_m) + g_y^2(r_m)} \end{pmatrix}. \tag{3.52}$$

Thus, the point where nodes go after bouncing is:

$$r_3 = r_1 + T(r_2 - r_1). \tag{3.53}$$

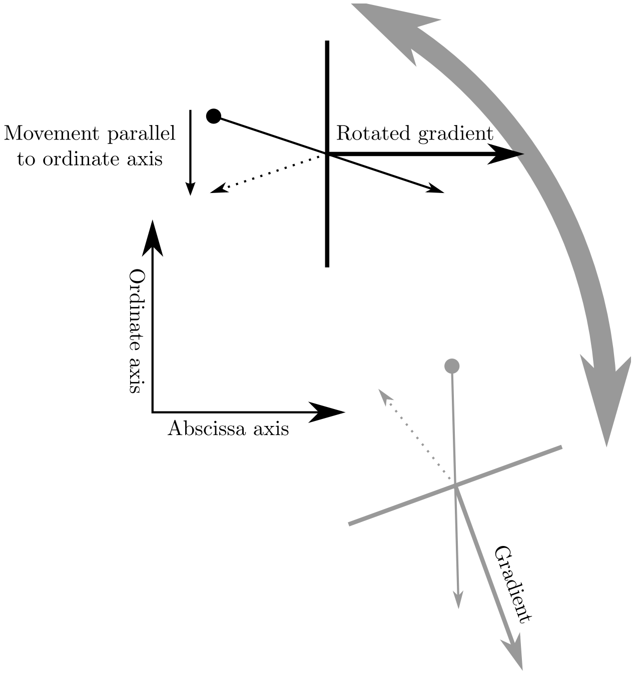

Figure 3.8: Example of the methodology followed to obtain the destination of nodes after bouncing.

Model performance

This section shows the performance of this model for the simulation scenario of Section 3.5.3. The cell has revolution symmetry around the cell center. Its simplicity makes easier to present results since they are only radius dependant. The hotspot has a smooth border between 40 and 50 m where the node density decreases from the maximum in the hotspot to its value outside the hotspot. Mathematically, the objective node density is:

$$\delta(d) = \begin{cases} U_1, & d \leq d_1, \\ \frac{U_1-U_2}{2}\cos\left(\frac{\pi(d-d_1)}{d_2-d_1}\right) + \frac{U_1+U_2}{2}, & d_1 < d < d_2, \\ U_2, & d \geq d_2, \end{cases} \tag{3.54}$$

where $d_1 = 40$ m, $d_2 = 50$ m and d is the distance to the cell center, i.e. $d = ||r|| = \sqrt{x^2+y^2}$. The quantities U_1 and U_2 are respectively the node densities inside the hotspot and in the rest of the cell. For these simulations,

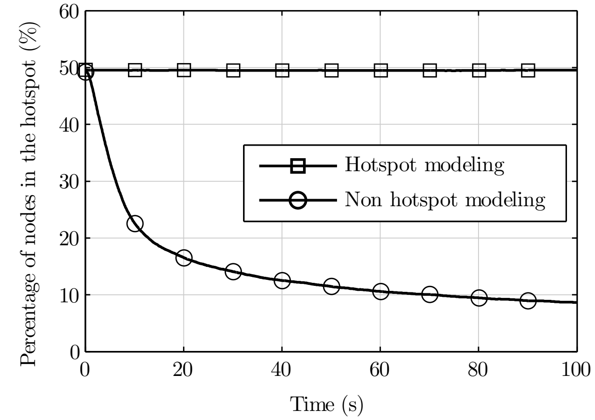

Figure 3.9: Percentage of users in the hotspot as the simulation progresses.

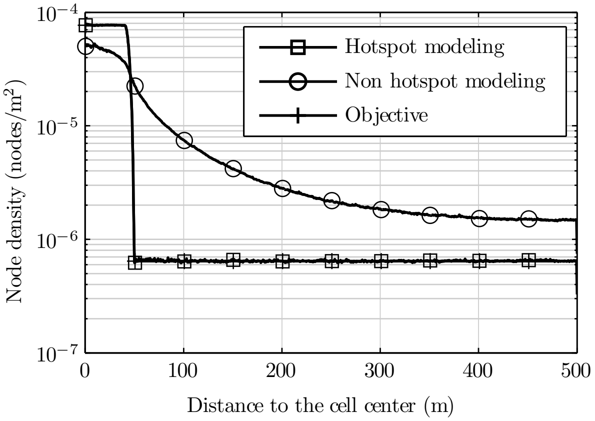

Figure 3.10: Node density normalized by the quantity of users U as a function of the distance to the cell center.

they were computed so that 50% of the nodes were in the hotspot, i.e. $d \leq d_2$, and the rest were outside the hotspot. These quantities are $U_1 = 7.67 \cdot 10^{-5} U$ and $U_2 = 6.43 \cdot 10^{-7} U$, where U is the number of nodes in the simulation area.

Figures 3.9, 3.10 and 3.11 show the performance of this model. Results were obtained for 10^5 nodes moving during 100 simulation seconds (1000 simulation

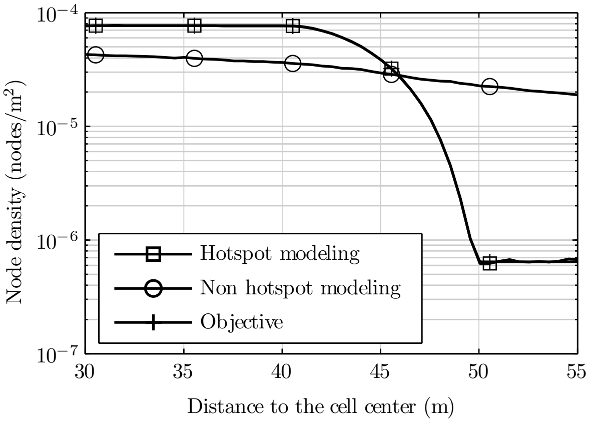

Figure 3.11: Node density normalized by the quantity of users U as a function of the distance of the cell center. Detail of the smooth border.

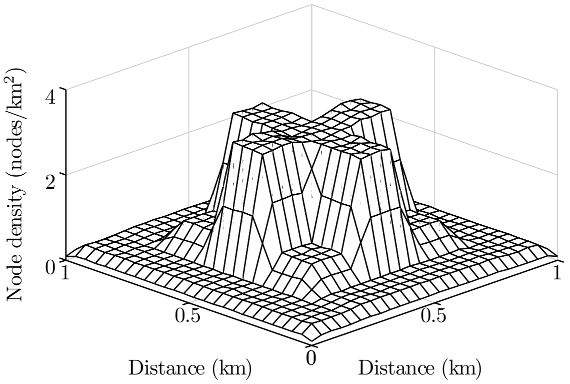

Figure 3.12: Node density normalized by the quantity of users U with the form of a cross in a square simulation area.

iterations). In the initial state, half of the nodes were placed at the hotspot (from 0 to 50 m) and the other half outside the hotspot (from 50 to 500 m). Figure 3.9 represents the variation of this initial distribution as the simulation progresses. The hotspot model presented here is capable of maintaining the

Table 3.2: Weighting coefficients.

μ_1	μ_2	μ_3	μ_4	μ_5	μ_6
1500	500	2500	16000	15	7000

nodes distribution. On the other hand, if nodes move freely with the random walk, the node density tends to be homogeneous, what makes the initial distribution vary. Figure 3.10 depicts the node density averaged over the 100 seconds of simulation. Moreover this figure shows the objective node density too. Obviously, in the initial state, nodes are placed following this objective. It is worth noting that with the hotspot modeling the objective is perfectly matched. Figure 3.11 represents in detail the variation of the node density around the hotspot border. As it can be seen, this model is capable of modeling smooth borders between zones. Obviously, any other node density function could be implemented as the objective. For instance, Figure 3.12 shows a node density with the form of a cross. Moreover, it would be also possible to define a mean node speed dependant of the nodes location, like the case of simulation areas with streets, highways and malls. In this latter case, a node could exit the mall, enter the streets of a city and take the highway, increasing its speed during the process.

3.6 Study of different HNN implementation alternatives

This section continues with the study of the different alternatives for implementing HNNs. This study started in Section 2.3.5 comparing different projection techniques using the MQP. This section will use the simulation scenario described before and the energy function of (3.11) to compare F-HNNs, HNNs with optimum updates and normal HNNs with no improvement. The objective of this section is to show if the reduction of the number of iterations is worth the increase of the computational cost to complete each iteration. For this study, the capacity of the RATs will be reduced to the 20%, i.e. WLAN has only 0.2 seconds to serve users and HSDPA has only 100 periods of 2 ms (instead of 500) every simulated second. This reduction allows the system to be saturated with less users. Therefore, the neural networks have less neurons what increase the response time of the algorithms. The weighting coefficients of the energy function were obtained following the rationale of Section 3.4 and are shown in Table 3.2.

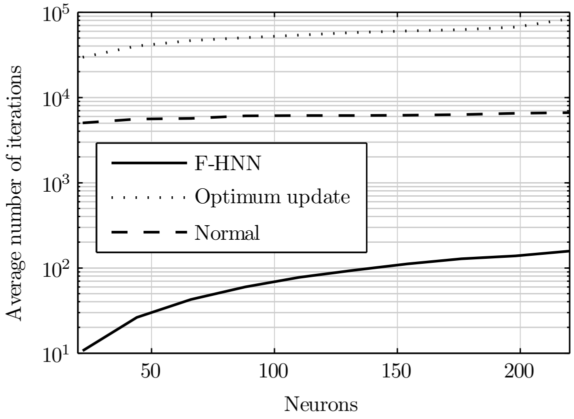

Figure 3.13: Average number of iterations.

Figure 3.13 depicts the average number of iterations needed by the three techniques to reach an equilibrium state. The number of neurons that compose the network depends on the quantity of active users demanding resources, with more users the network has more neurons. The figure shows that the F-HNN needs around 100 times less iterations than a normal HNN. The surprising result is that the HNN with optimum updates needs 10 times more iterations than the normal HNN. The explanation to this lies in the oscillation probability. The optimum updates make the HNN oscillate the 25% of times. That means that these networks cannot reach equilibrium during the 10^5 maximum iterations of the algorithm, hence, increasing the average number of iterations. The oscillation probability of the normal HNN is 0.0074% and exactly 0% for the F-HNN. Such low oscillation probability of the normal HNN does not actually mean that these networks oscillate but that maybe 10^5 iterations were not enough. Nevertheless, the 25% of cases are too many cases for needing more than 10^5 iterations. The main reason for oscillating is that the constant parameters of the HNN are not constant anymore. More specifically, the third term of (3.11) has the parameter ξ_{ijk} which is not constant during all the iterations. This fact makes the linear term of (2.12), i.e. the I_i, being not constant. Then, if at a specific point the optimum updates state that it is possible to perform a big jump in the direction of minus energy gradient, maybe at the destination one or more ξ_{ijk} change and then it is necessary to come back to the original point. The projections performed by the F-HNN solve this problem. The big difference between the normal HNN and the F-HNN is due to the low

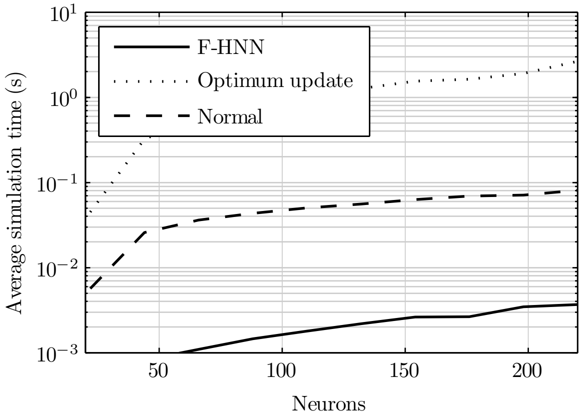

Figure 3.14: Average simulation time.

updating step needed by the former to be stable, in this case $\Delta t = 10^{-5}$. This fact increases the number of iterations.

Figure 3.14 shows the average simulation time needed to reach equilibrium. The normal HNN approaches a bit the F-HNN requiring only 20 times more time. Moreover the HNN with optimum updates needs around 25 times more time than the normal HNN. This fact shows that each iteration of the HNN with optimum updates and the F-HNN has more computational cost than the normal HNN. Nevertheless, recalling the results of Section 2.3.5, S-HNN needed 100 times more iterations but only 10 times more time what implies that S-HNN iterations have 10 times less computational cost than F-HNN although S-HNN also performs projections. Then, the fact that the normal HNN iterations are only 4 times faster than those of the F-HNN seems strange. Since there are no projection, the reader would expect a significantly higher reduction of the iteration time with the normal HNN. This is due to the projection performed in the specific case of the JDRA. The projection plays the role of the last term of (3.11), i.e. all the neuron outputs of each user must sum exactly 1. This projection is very simple because no neuron belongs to two users at the same time. In order to maintain the same sum, the gradient can be projected subtracting the mean of all the components of the same user. An example will help understanding this procedure. Imagine that a user has two neurons with outputs 0.8 and 0.2 which sum exactly 1. At the next iteration the gradient for this neurons is -0.4 and -0.2 respectively. The mean of these components is -0.3. Then neurons are updated following 0.8-(-0.4-(-0.3))=0.9 and 0.2-(-0.2-

(-0.3))=0.1 which still sum 1. Calculating and subtracting the mean is much faster than performing the projection from a projection matrix. Moreover, adding more restrictions for the hypercube facets is as simple as not counting that gradient component for computing the mean and making it equal to zero.

Additionally, there is another reason for the F-HNN to be so fast with the JDRA. Thanks to the projection, the last term of (3.11) is always zero, then it can be eliminated from the energy function. In that case only the fifth term is quadratic. This term produces a matrix **T** with non-zero elements only in the main diagonal. The objective of this term is to force the equilibrium at the extremes where $V_{ijk} = 0$ or $V_{ijk} = 1$. This same effect can be achieved with a null main diagonal in **T**. Therefore, this term can be also eliminated, leaving a linear energy function. In this case the optimum update is much easier to compute since always $S_2(t) = 0$, what means that $\beta_o(t)$ must be as high as possible.

3.7 Results

The results assessment has been divided into two parts. The first part is focused on the improvement achieved by a joint scheduling. Consequently, it compares the proposed HNN-based algorithm with the UMA-HNN and MBR-HNN algorithms. Next, the HNN algorithm is evaluated against the rest of reference algorithms previously defined in Section 3.5.2.

3.7.1 Joint scheduling improvement

For this first study, the number of users per service and class ranged from 20, for the least loaded case, up to 80 for the most loaded case. Moreover, the user density was set in such a way that half of them were located in the hotspot in average. Figure 3.15 shows the non-satisfaction probability, i.e. the probability of not fulfilling the expected QoS, for the three algorithms studied in this section and each service. The improvement of the JDRA is highly noticeable with respect to the MBR policy, whereas the difference with UMA policy is quite negligible. Depending on the service and the quantity of users UMA is sightly better or worse. Figure 3.16 depicts the same non-satisfaction probability but averaged over the four services. This figure shows that the difference between UMA and the JDRA algorithm is even lower when considering all services together. The good performance observed with the UMA policy is due to the fact that the WLAN RAN is much less loaded than HSDPA. Consequently, this policy is always the best since it unloads HSDPA from users as soon as they are in the coverage area of the WLAN AP. Nevertheless, in a different scenario

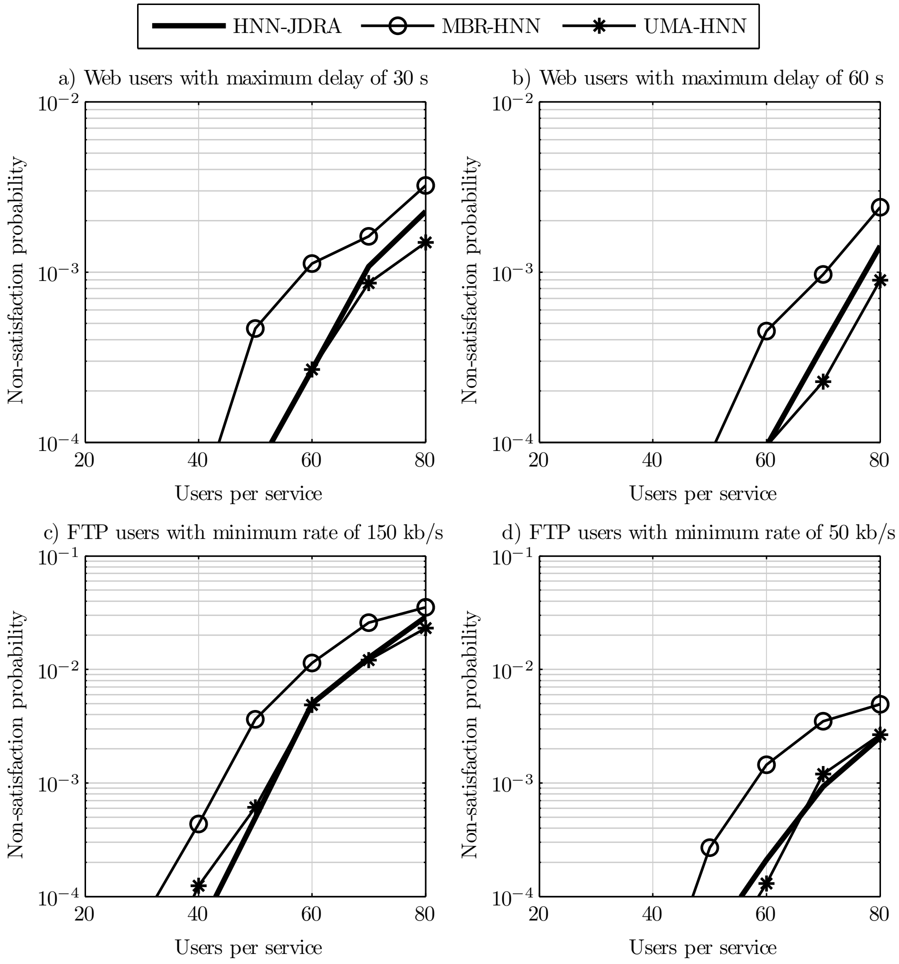

Figure 3.15: Non-satisfaction probability vs. quantity of users for each service.

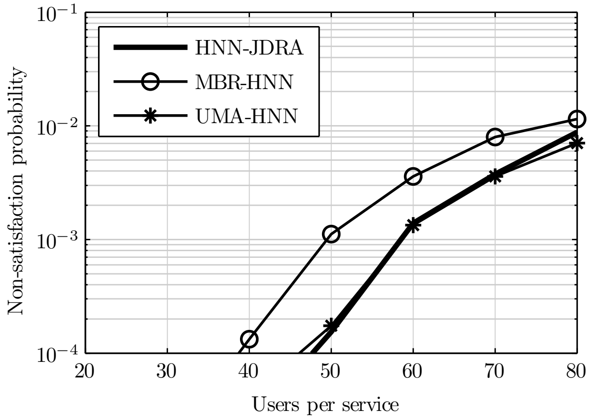

Figure 3.16: Non-satisfaction probability vs. quantity of users for all the services.

the UMA policy may not be the best one. Figures 3.17 and 3.18 represent the same quality indicator but with a fixed quantity of users per service in the system, i.e. 60, and different probabilities of being in the hotspot. It can be observed that the MBR policy outperforms UMA when more than the 80% of users are in the hotspot. Thus, if the WLAN coverage area is highly loaded, the policy of allocating all users to it is not optimum as compared with distributing them among the overlapping RANs. It is worth highlighting that the proposed algorithm is the best one in all cases and, moreover, it extends the good behavior outlined by the UMA policy with low loaded WLAN to more saturated scenarios.

Finally, Figure 3.19 shows the average bit rate allocated to users in different locations of the cell. Figure 3.19a can be understood as a scenario with a low loaded WLAN RAN. In this case, the proposed algorithm and the UMA policy have almost the same performance being both more homogeneous than the MBR policy. This fact means that the dependence of the bit rate allocated to users and their position is stronger for the MBR case, hence this policy is less fair and produces more differences between the QoS perceived by two different users. On the other hand, Figure 3.19b can be understood as a high loaded WLAN RAN case. Now, the UMA policy is the one with the most heterogeneous average bit rate whereas the proposed algorithm is clearly the fairest. This fact can help understanding where the power of the JDRA algorithm is.

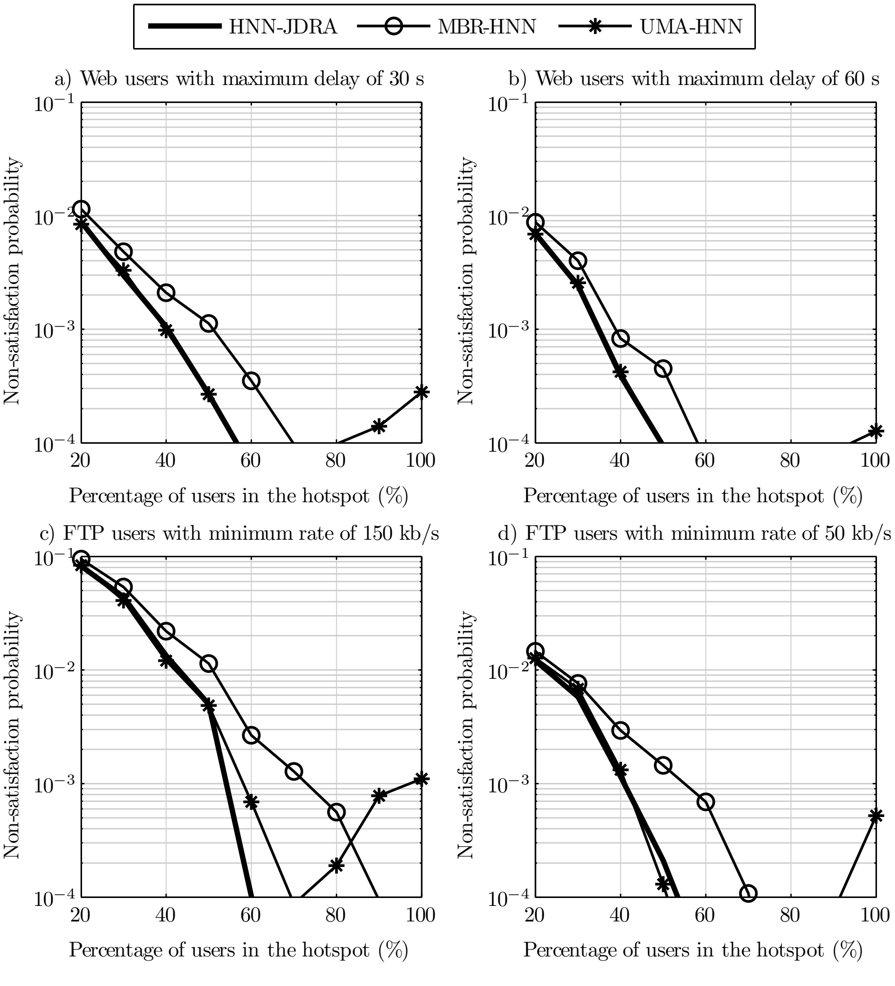

Figure 3.17: Non-satisfaction probability vs. percentage of users in the hotspot for each service.

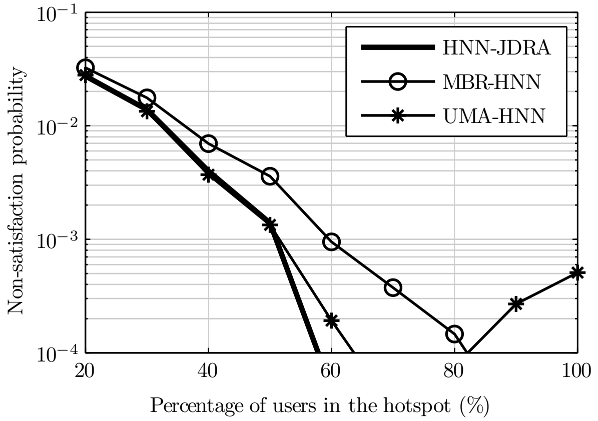

Figure 3.18: Non-satisfaction probability vs. percentage of users in the hotspot for all the services.

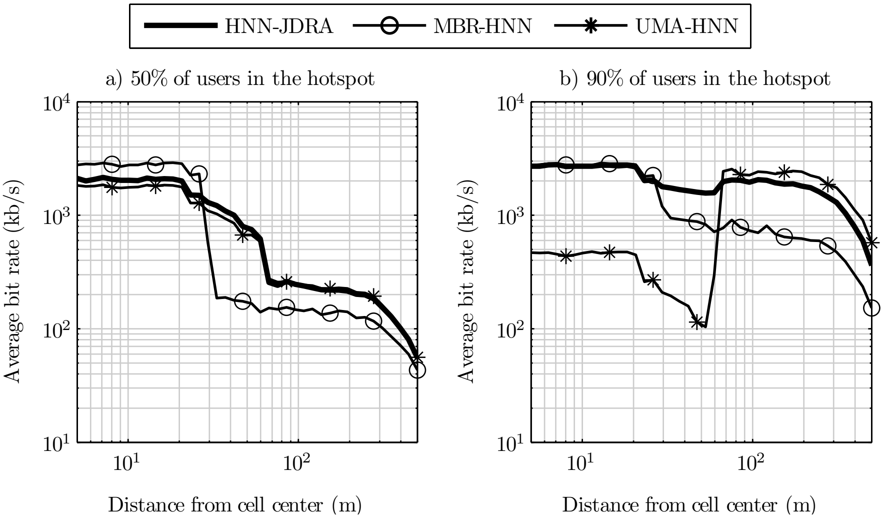

Figure 3.19: Average bit rate allocated to each user depending on the distance from the cell center.

In fact, this algorithm provides around 2 Mb/s in average to all users that are closer than 250 m to the cell center, independently of their exact position.

3.7.2 JDRA algorithm evaluation

Now, the same scenarios (20 to 80 users per service and class equally split up into the hotspot and the entire cell and 60 users per service and class with different users distribution ratios between the hotspot and the entire cell) were simulated with the rest of algorithms. Figures 3.20 and 3.21 show the non-satisfaction probability for an increasing quantity of users. In these graphs, three groups of lines can be identified: those belonging to the CLSA, those of the MLWDF and that of the proposed JDRA algorithm. Within the first two groups, the UMA policy is always slightly better than the MBR but, in general, they are very similar. Regarding CLSA and MLWDF, none of them is the best one in all situations. CLSA is better for high loaded cases whereas MLWDF is better for low loaded cases. This is the main reason for comparing with both algorithms. The proposed HNN-based JDRA is by far the best algorithm for all cases. Figure 3.22 depicts the improvement achieved by the JDRA with respect to the reference algorithms. This improvement is computed from the non-satisfaction probability of the HNN-based JDRA algorithm, $\mathrm{NSP}_{\mathrm{HNN}}$, and the non-satisfaction probability of the reference algorithms, $\mathrm{NSP}_{\mathrm{ref}}$, as:

$$\mathrm{IMP}(\%) = 100\left(1 - \frac{\mathrm{NSP}_{\mathrm{HNN}}}{\mathrm{NSP}_{\mathrm{ref}}}\right). \tag{3.55}$$

In general, the improvement achieved by the JDRA is higher than the 75%, except for the MLWDF in low load situations where these reference algorithms reduce drastically their non-satisfaction probability due to their good performance with delay-based services (see Figure 3.20).

Figures 3.23 and 3.24 show the non-satisfaction probability for 60 users and for different proportions of users located in the hotspot. In these figures, the same three groups can be identified, but with a slight difference. Now the two algorithms that use CLSA (or MLWDF) have a similar performance unless for high loaded hotspot cases. Then, UMA policy is significantly worse than MBR. Something similar was previously pointed out in Figures 3.17 and 3.18. Note that the performance deterioration of users depends on their service. For the UMA-CLSA algorithm web users are more affected than FTP users, whereas for the UMA-MLWDF algorithm FTP users suffer the biggest deterioration. Among all the algorithms, the proposed JDRA arises again as the best one for all situations. Figure 3.25 depicts the improvement achieved by the JDRA algorithm. Now, this improvement is always above the 70%. Moreover, the improvement increases with the quantity of users located in the

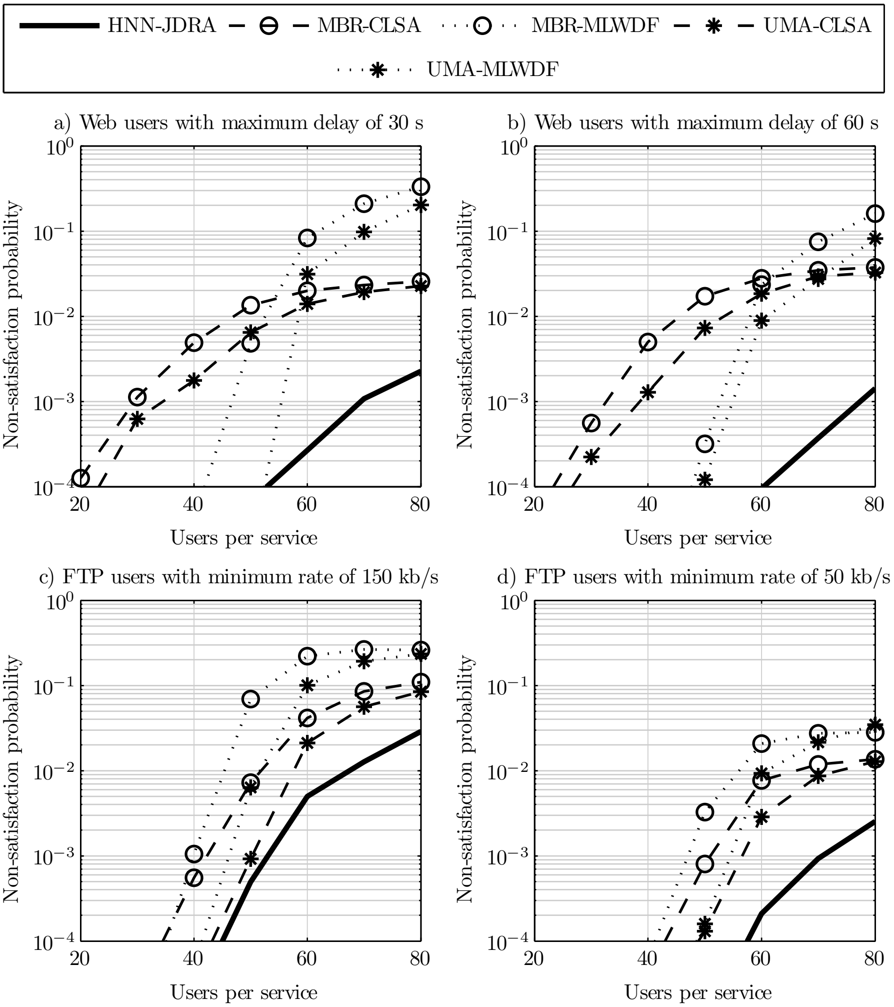

Figure 3.20: Non-satisfaction probability vs. quantity of users for each service.

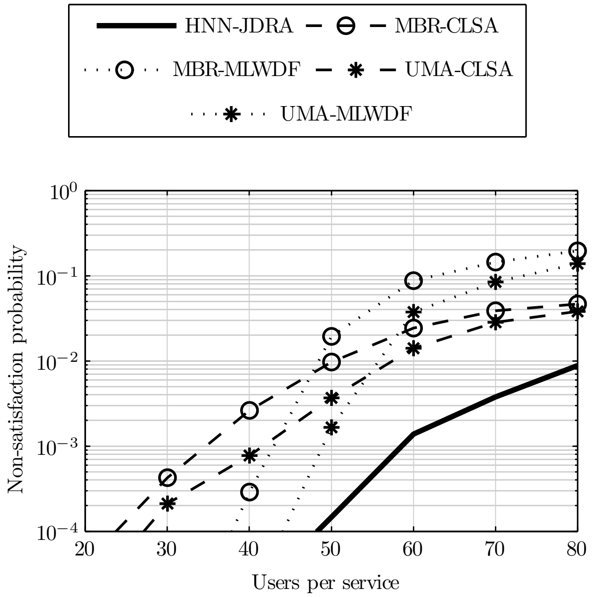

Figure 3.21: Non-satisfaction probability vs. quantity of users for all services.

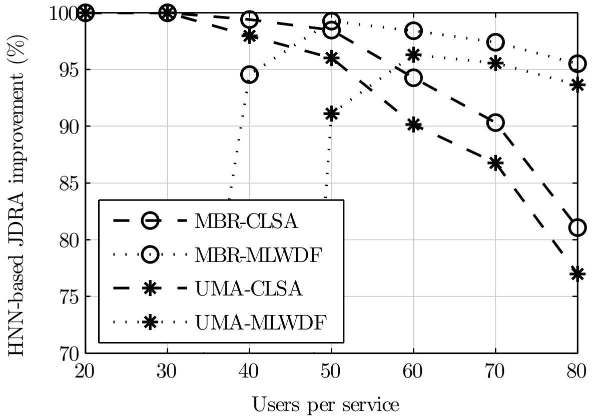

Figure 3.22: Improvement of the HNN-based JDRA respect the reference algorithms.

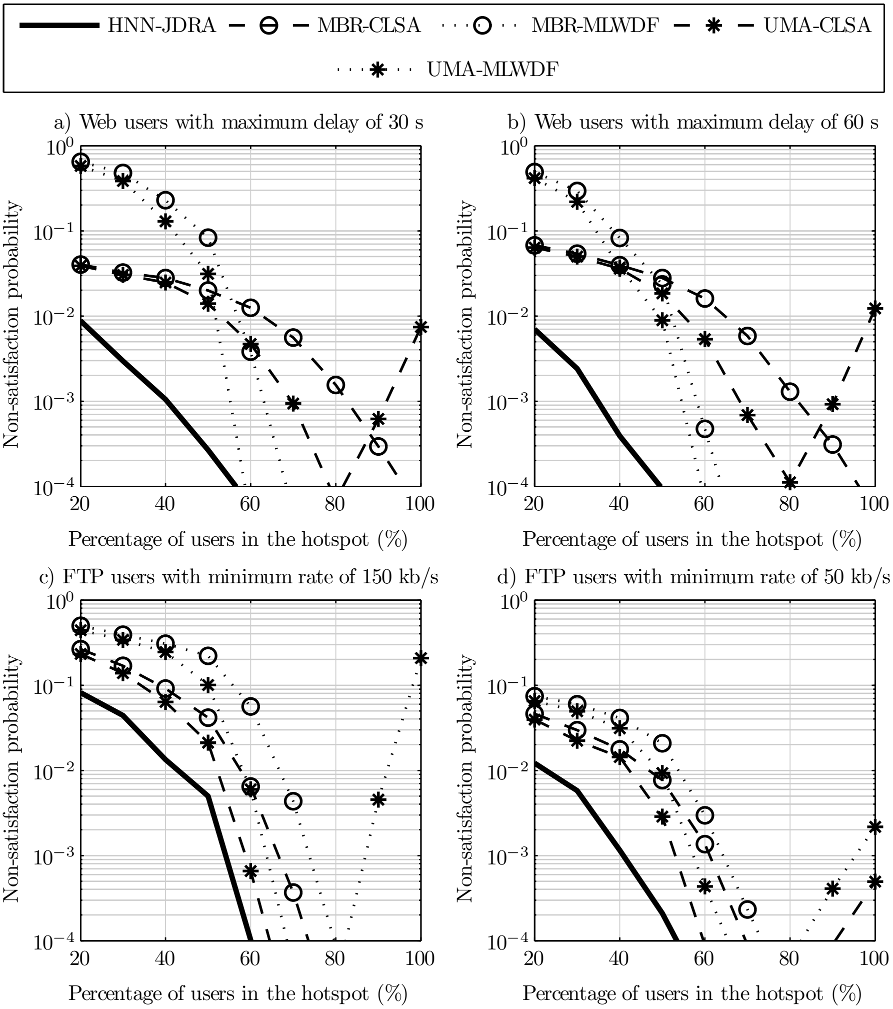

Figure 3.23: Non-satisfaction probability vs. percentage of users in the hotspot for each service.

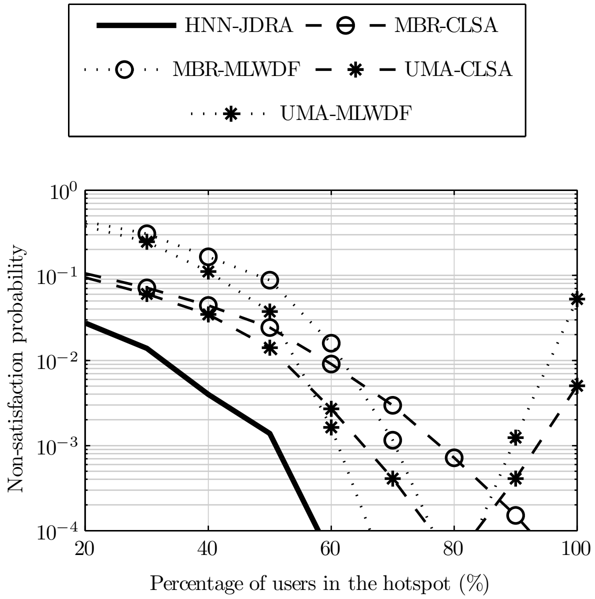

Figure 3.24: Non-satisfaction probability vs. percentage of users in the hotspot for all services.

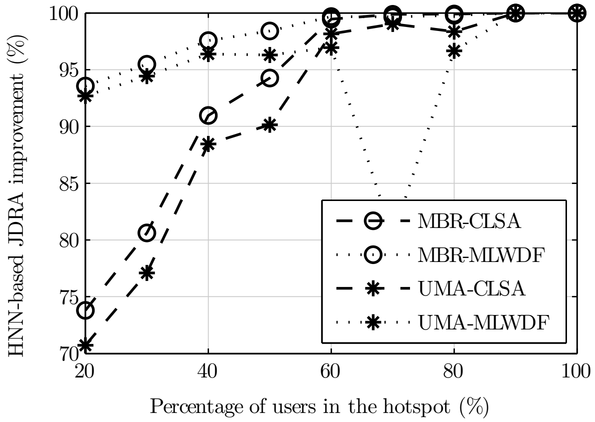

Figure 3.25: Non-satisfaction probability vs. percentage of users in the hotspot for all services.

hotspot, reaching levels of more than the 99%. It is worth noting that the more users are in the hotspot, the more users have coverage of both technologies at the same time. This fact means that any JDRA algorithm could optimize the resource allocation to further levels than in other situations.

3.7.3 Evaluation of the HNN optimality

Previous sections have evaluated the HNN-based JDRA algorithm with other resource allocation techniques. This section will evaluate how optimum the solutions found by HNNs are, or more specifically, by F-HNNs. To this aim, all the techniques studied before were compared with an integer program. There are several methods to solve integer programs as for example branch and bound [62] or cutting plane techniques. The technique used in this Thesis is branch and bound. Integer programs are a special case of Linear Programs (LPs), even more, integer programs are solved by solving many LPs. These techniques give the absolute minimum of any problem expressed as linear equations but with an extremely high computational cost. This is the main reason for not adding this solution method in the previous sections. The required time makes this unfeasible.

The integer program used to solve the JDRA is as follows:

$$\begin{gathered}\text{Minimize } LE,\\ \sum_{k=1}^{K}\sum_{j=1}^{J_k} V_{ijk} = 1,\\ V_{ijk} \in \{0,1\},\end{gathered} \tag{3.56}$$

where:

$$\begin{aligned}LE = &-\frac{\mu_1}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k} B_{ijk}V_{ijk} - \frac{\mu_2}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k}\frac{Q_k(C_{ik})\rho_{jk}}{\max\limits_{lmn}\{Q_n(C_{lk})\rho_{mn}\}}V_{ijk}\\ &+\frac{\mu_3}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k}\xi_{ijk}V_{ijk} + \frac{\mu_4}{2}\sum_{i=1}^{I}\sum_{k=1}^{K}\sum_{j=1}^{J_k}\psi_{ijk}V_{ijk}.\end{aligned} \tag{3.57}$$

Note that LE comprises the first four terms of the energy function defined in (3.11). These terms are linear, and the other two are included in the constraints of (3.56).

The procedure followed for this evaluation was running a simulation calling all algorithms every resource allocation period with the same input data. Then, the energy of the allocation of each algorithm was stored. This way the results

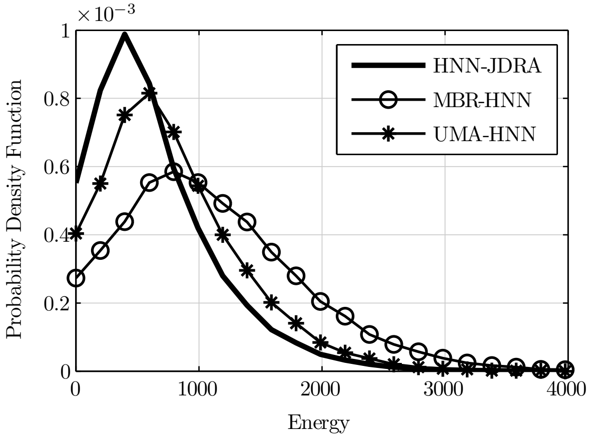

Figure 3.26: PDF of the energy difference with respect to the integer program.

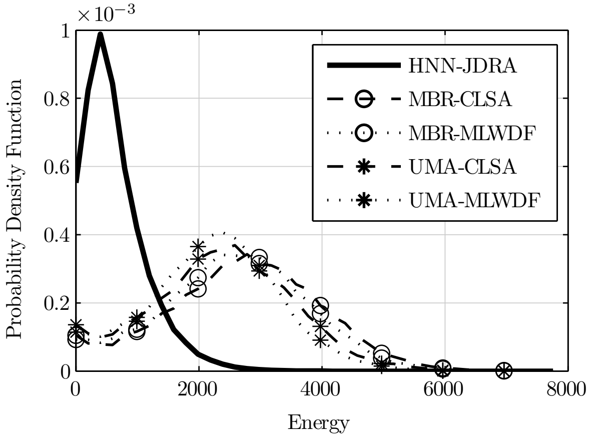

Figure 3.27: PDF of the energy difference with respect to the integer program.

at each instant are completely comparable. For this simulation, the system was loaded with 60 users per service locating half of them in the hotspot.

Figures 3.26 and 3.27 represent the Probabilistic Density Function (PDF) of the distance of the energy obtained with all the algorithms with respect the energy obtained with the linear program. As can be observed, the probability density of the HNN-based JDRA algorithm is closest to zero as compared with the rest of algorithms. These figures give an idea of the optimality of the solutions obtained using HNNs.

3.7.4 Parallel JDRA algorithm implementation

This section compares the performance of the F-HNN used to solve the JDRA problem implemented in parallel and serial hardware. The serial hardware was an Intel Pentium D Central Processing Unit (CPU) with a clock of 3.2 GHz and 2 GB of RAM. The parallel hardware was the Graphics Processing Unit (GPU) Nvidia GeForce GTS 250 with a clock of 738 MHz and 512 MB of RAM. This GPU is composed of 128 cores that can execute the same code using the programming language CUDA of Nvidia. Both of them (the CPU and the GPU) were in the same computer. Thus, this section compares two implementations of the JDRA algorithm. This section will not explain the details of the GPU implementation. This would require an entire chapter since, previously, the reader should understand the GPU architecture and different (and many) types of memory conflicts, common in multicore processing. Just as an initial guide, it is worth highlighting the following differences between CPUs and GPUs:

- CPUs are generally faster running code that cannot be parallelized, since they use higher clock frequencies.

- CPUs memory has less latency. For that reason, GPUs should read and write large amounts of data at once instead of many times reading and writing few data.

- Conversely, GPUs have many types of in-chip memory. A perfect program should read only once from RAM memory and use in-chip memory afterwards. Nevertheless, the best way of managing these types of memory is not trivial, although they have a very significative effect in the GPU performance. The programmer and not the compiler is who must decide how to use this memory. This fact is what makes the use of GPUs so complicated.

- The compiled programs are executed in the CPU. If the GPU is used, the CPU initializes it and copies all the needed data from the CPU RAM to the GPU RAM via the PCI Express port. After copying all, the CPU starts the program in the GPU. When the computation ends, the CPU must read the results from the GPU through the same port. This procedure requires an additional time that is not necessary if the program

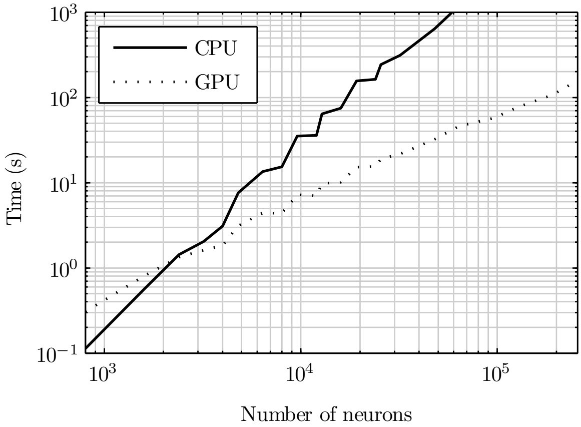

Figure 3.28: Average time spent to reach equilibrium as a function of the number of neurons.

is completely executed in the CPU. Therefore, using the GPU may not be always the best option. An advantage is that the CPU is freed during the time the GPU is working and, hence, it can be used for other computations.

- CPUs are well prepared for computing with double precision arithmetic, whereas GPUs work better with single precision.

Up to 5 different simulations were carried out with different initial states (user positions, buffer states, etc.) for all the 48 scenarios studied in this section. These scenarios are different combinations of number of users (10, 50 and 100), bit rates (16, 32, 64 and 128) and RATs (5, 10, 15 and 20). The simplest scenario requires only 800 neurons whereas the most complex one requires 256000 neurons. Results are shown as a function of the number of neurons.

Figure 3.28 depicts the average time that both implementations require to reach equilibrium. As it can be observed, only for those situations with more than 2500 neurons the GPU is faster than the CPU. Moreover, compared with the CPU, the more neurons the network has, the faster the GPU is. This is due to the number of iterations, that also increases with the number of neurons, as shown in Figure 3.29. With many iterations, the GPU is used for longer time, which makes the time spent in sending data to the GPU more and more worthwhile. A curious difference of the CPU and GPU implementations is that

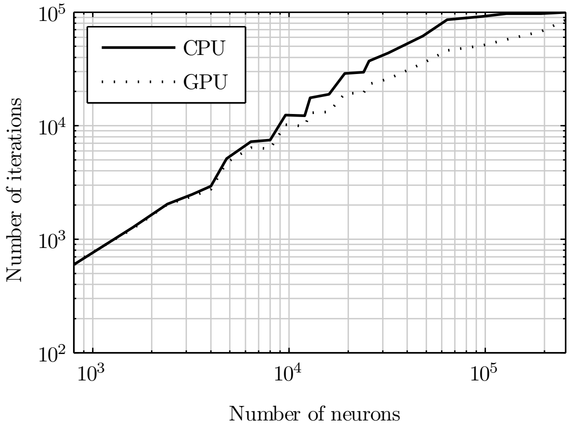

Figure 3.29: Average number of iterations required to reach equilibrium as a function of the number of neurons.

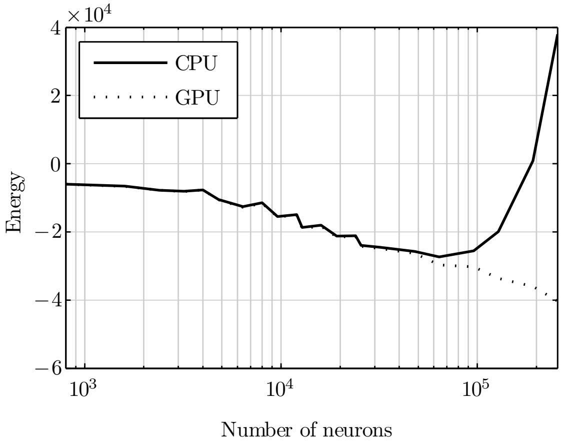

Figure 3.30: Average energy at equilibrium as a function of the number of neurons.

the latter needs sightly less iterations to reach equilibrium. This fact is due to the single precision arithmetic, which makes neurons reach the extremes faster. On the contrary, with double precision, an extra iteration may be required for each neuron. For many neurons, the CPU reaches the maximum number of iterations, 10^5. This fact makes the final results of the CPU significantly bad. Figure 3.30 show the average energy. In this figure, it is possible to see how the average energy increases for the CPU when it reaches the maximum number of iterations. With less neurons, the solutions obtained with the CPU and GPU have the same energy, what reflects the fact that they reach identical solutions.

To sum up, the GPU is generally faster and reduces the execution time more than one order of magnitude (and almost two) when sufficient iterations are needed. Nevertheless, the additional computational cost for copying data to the GPU makes the CPU more suitable for F-HNNs with few neurons.

3.8 Conclusion

This chapter has presented the JDRA algorithm proposed in this Thesis. This algorithm distributes users of diverse types of service and classes among RANs of different technologies and the RANs resources among users.

F-HNNs have been used to solve this complex problem. The neuron parallel interconnection of these networks have made the definition of the algorithm easier as shown in the rationale followed in Section 3.3.1. Moreover and most importantly, F-HNNs have very fast response times what makes feasible a real-time functioning of the algorithm.

The JDRA algorithm has shown a significant reduction of the non-satisfaction probability of users as compared with other RAT selection techniques. Moreover, it approaches the UMA policy when optimum. The proposed algorithm is also better than other DRA techniques proposed in the literature unless for low loaded networks. Despite the sub-optimum nature of HNN solutions, in some cases the MLWDF algorithm is near optimum, what justifies the slight differences. Nevertheless, the region where MLWDF outperforms the proposed algorithm is below a threshold of non-satisfaction probability of 0.05%. Therefore, it is preferred the use of the HNN-based JDRA algorithm instead of MLWDF since it reduces the non-satisfaction probability from 13% to 0.9% for other cases.

Moreover this chapter has presented a hotspot modeling capable of creating any general node density in a simulation area. The model is based on sudden bounces that depend on the desired node density and average node speed. Moreover, Section 3.5.4 has presented an easy way of obtaining the destination of nodes after bouncing. Results show the accuracy of the model that perfectly matches the desired node density. Even smooth borders are precisely emulated allowing the definition of complex densities in the simulation area. This model allows a fair study of any wireless system, since all nodes will have the same movement pattern and will not be stuck in specific zones.

Finally, the JDRA algorithm has been implemented in a GPU with 128 cores. The results show the potential benefit of a parallel implementation with F-HNNs.

Part II

JCAC Algorithm

Chapter 4

Equivalent resource consumption

4.1 Equivalent bandwidth

The EBW concept was conceived as a step forward to assist the CAC algorithm with the decision making on call admission. The EBW concept was introduced by three contemporaneous works [63–65], although the underlying principle was described earlier by Hui [66]. From these works, EBW became an important element of CAC with several applications reflected in the numerous articles found in the literature (see, e.g. [67–74]). This popularity is due to its capacity to simplify CAC for bursty-traffic services in wired ATM networks. In these networks traffic generated by different sources may temporarily coincide, requiring more bandwidth than the server is able to provide. In that case, the ATM cells not transmitted are stored in the server buffer. If the bandwidth requirements decrease, the buffer could be emptied but, otherwise, the buffer could be filled up, hence discarding new incoming cells. The EBW was originally defined as the service rate which ensures a certain cell loss rate given a finite-buffer server:

$$\Pr\{(\text{Aggregate traffic generation rate} - \text{EBW})\tau > \text{Buffer size}\} = \varepsilon, \qquad (4.1)$$

where τ is a certain time interval and ε is the desired loss rate. The application of the EBW concept to CAC is that simple: if the EBW obtained from (4.1) of all the existing calls plus the new one is greater than the available bandwidth, the incoming call is rejected. To calculate the EBW, the only unknown variable in (4.1) is the aggregate traffic generation rate, i.e. the amount of traffic generated by all active users per time unit. Several approaches have been proposed to model this aggregate traffic generation rate, such as binomial distribution [63, 67, 68], fluid-flow approximation [63], Gaussian distribution [69] and large deviation approximation [65, 66]. Each model has its own drawbacks, as highlighted in the second chapter of [39]. For instance, binomial distribution defined in [68] underestimates the required bandwidth [70], whereas the same approach in [63] and [67], as well as fluid-flow approximations [63], Gaussian distributions [63] and large deviation approximations [71] overestimates it. On the other hand, for small buffer sizes Poisson distribution is the best approximation [70]. Finally, it is appropriate to say that all these approaches can be complemented with real measurements in order to dynamically determine the parameters of the different probabilistic distributions [72, 73]. Thanks to this measurement-based adjustment, EBW becomes more adaptable and feasible.

4.1.1 EBW generalization

The main works concerning EBW have been presented in Section 4.1. All these contributions are characterized by the fact that they try to find out a required service bit rate. Next, they compare this service rate with the fixed maximum available bit rate of the system in order to decide on call admission. On the contrary, in wireless systems the maximum bit rate is not fixed and depends on the signal quality perceived by the user. Nevertheless, wireless communication systems have a maximum quantity of resources for distribution among users. In turn, each bit rate needs certain amount of resources to make the transmission feasible. This quantity is not only conditional on the actual bit rate, but also on other factors, such as user location, interference conditions, etc. Obviously, a greater bit rate implies demanding more resources. The type of shared resources varies according to the specific technology under consideration, e.g., time slots in GSM, transmit power and spreading codes in UMTS or the time of channel occupancy in WLAN. Evans and Everitt [74] extended the probabilistic idea of the EBW to ensure a certain SNIR level in downlink CDMA systems. This work is the first and most relevant approach that obtained the ERC of users in wireless systems, specifically in CDMA systems. The ERC concept defined in [74] is the quantity of resources that ensures a target SNIR with a given probability. Therefore, ERC can be understood as a generalization of the original EBW concept of (4.1) since it represents a certain quantity of resources rather than a specific bit rate. Nevertheless, this work approximates the ERC value by means of Gaussian distributions without studying their performance. Moreover, this approach is not general since it is useless for non-CDMA systems.

4.1.2 Objective of this chapter

This chapter introduces a general ERC definition to clearly determine the quantity of resources needed to guarantee the desired quality levels. This new ERC calculation results in a more efficient CAC in multi-service wireless systems. Besides, the definition is valid for any general wireless system and any QoS criterion, not only for CDMA systems and SNIR level guaranty like in [74]. In order to facilitate the implementation and increase the flexibility of the ERC calculation, this Thesis analyzes the consequences of using histograms obtained from real measurements of the system random variables, like the experienced channel quality, instead of using ideal and continuous probabilistic distributions. Special attention is paid to the accuracy of this approach and moreover an idea of its computational cost is also given. Finally, the performance of the measurement-based ERC calculation is compared with the Gaussian approximation, deriving the cases in which each method is preferred.

As it has been mentioned, the ERC concept is useful to help on call admission in a unique RAT. After introducing and studying the ERC in this chapter, Chapter 5 presents the ERC-based JCAC proposed in this Thesis.

4.2 ERC definition

The general definition of EBW has been shown in (4.1). Analogously, ERC can be defined as:

$$\Pr\{\text{Aggregate resources needs} - \text{ERC} > 0\} = \varepsilon. \tag{4.2}$$

The aggregate resource needs is the random variable of the sum of all resources that users require to satisfy their QoS. It is worth noting that, in general, the amount of resources that several users need is less than the sum of the resources needed by each user separately. This fact is mainly due to the bursty nature of data traffic and channel quality fluctuations. Both quantities are equal for static users and completely constant traffic generation. Therefore, if the aggregate resource needs was calculated by summing all ERCs separately obtained for each user, it would result in an underestimation of system capacity. Therefore, it is necessary to compute an equivalent ERC value for all users.

Let $\rho_i(t)$ be the resources that the i-th user needs at time t and $f_{\rho_i}(v)$ its PDF. Then it is possible to calculate the aggregate resource needs, $\rho(t)$, and its PDF, $f_\rho(v)$, as follows:

$$\rho(t) = \rho_1(t) + \rho_2(t) + \cdots + \rho_U(t), \tag{4.3}$$

$$f_\rho(v) = f_{\rho_1}(v) * f_{\rho_2}(v) * \cdots * f_{\rho_U}(v), \tag{4.4}$$

where U is the number of users and $f * g$ is the convolution of f and g. Note that (4.4) assumes that the different $\rho_i(t)$ are independent. Hence, the ERC is:

$$\mathrm{ERC} = F_{\rho}^{-1}(1 - \varepsilon), \tag{4.5}$$

where F_X^{-1} is the inverse of the Cumulative Density Function (CDF) of X. Therefore, in order to calculate the ERC, the PDF of every $\rho_i(t)$ must be calculated first.

4.2.1 ERC calculation

Network operators usually define QoS metrics in terms of guaranteed or minimum bit rate. In fact, although quality needs can be concreted in a different way (for instance in terms of maximum packet delay or dropping rate) in order to have a common definition of QoS an equivalent minimum target bit rate can be calculated as shown in Section 3.1.

The ERC aims at determining the quantity of resources the users need, but this quantity depends on the maximum bit rate per r.u. that each user can reach, what in turn depends on the SNIR the user perceives (see Section 4.2.1).

The quantity of resources the i-th user needs at some time t is:

$$\rho_i(t) = \frac{R_{\mathrm{min,i}}(t)}{R_i(t)}, \tag{4.6}$$

where $R_{\mathrm{min,i}}(t)$ is the minimum target bit rate (or just target bit rate) for the i-th user at time t and $R_i(t)$ is the maximum bit rate per r.u. that the user can reach at time t. $R_{\mathrm{min,i}}$ depends directly on the type of service of the i-th user. Thus, $R_{\mathrm{min,i}}$ can either be constant for some services or have active and inactive periods (or ON and OFF periods) where data may be processed by the user. In the most general case, $R_{\mathrm{min},i}$ can have any value at any time. Moreover, different users may have different $R_{\mathrm{min,i}}$ in multi-service scenarios. Since this latter case is more general, users are not supposed to have the same statistics in the following rationale.

The maximum bit rate per r.u. of each user can be derived following (3.7). As shown in Section , function $Q(C)$ is the maximum bit rate per r.u. with a SNIR of C. The subindex k in (3.7) made reference to the different RANs or RATs. Since this chapter is not aware of the interaction of different RATs, the subindex disappears, that is, the techniques proposed here are for only one RAT. It will have sense again in Chapter 5. Thus, the bit rate per r.u. of the i-th user is:

$$R_i\left(t\right) = Q\left(C_i\left(t\right)\right), \tag{4.7}$$

where $C_i(t)$ is the SNIR perceived by the i-th user at time t. If the mathematical expression of the SNIR PDF is known, what can be obtained from users' measurement reports, then the following theorem gives the necessary tools to calculate $f_{R_i}(r)$ from $f_{C_i}(c)$.

Theorem 4.1 (extended from theorem 3, section 2.5 of [75])**.** *Let X be a random variable with PDF f_X. Let $g(x)$ be a continuous function differentiable for all x unless for a set D of measure zero. Let either $g'(x) > 0$, $\forall x \notin D$, or $g'(x) < 0$, $\forall x \notin D$. Then $Y = g(x)$ is also a random variable with PDF:*

$$f_Y(y) = \begin{cases} f_X\left(g^{-1}(y)\right)\left|\frac{d}{dy}g^{-1}(y)\right|, & \alpha < y < \beta,\ g^{-1}(y) \notin D, \\ 0, & \text{otherwise,} \end{cases} \tag{4.8}$$

where $\alpha = \min\{g(-\infty),\ g(+\infty)\}$ and $\beta = \max\{g(-\infty),\ g(+\infty)\}$.

Proof. Since g is continuous, it is measurable. Moreover, since random variables are measurable functions over the sample space and the composition of measurable functions is also measurable, function $Y = g(X)$ is also a random variable.

First, let define G as the set of all the images of g, i.e. $g : \mathbb{R} \to G \subseteq \mathbb{R}$. Let also define G^* as the set of images of g where g is differentiable, i.e. $G^* = G \setminus g(D)$. The proof is divided into three steps: points of set G^*, points of set $G \setminus G^*$ and points of set $\mathbb{R} \setminus G$.

Points of set G^*. Let focus on the case that $g'(x) > 0$. Let F_X and F_Y be the CDFs of X and Y respectively. Then, for all $y = g(x)$:

$$F_Y(y) = \Pr\{Y \leq y\} = \Pr\{g(X) \leq g(x)\}. \tag{4.9}$$

Since g is continuous and $g'(x) > 0$, g is strictly increasing and invertible. Therefore:

$$F_Y(y) = \Pr\{g(X) \leq g(x)\} = \Pr\{X \leq x\} = F_X(x) = F_X\left(g^{-1}(y)\right), \tag{4.10}$$

$$f_Y(y) = \frac{d}{dy}F_Y(y) = \frac{d}{dy}F_X\left(g^{-1}(y)\right) = f_X\left(g^{-1}(y)\right)\frac{d}{dy}g^{-1}(y). \tag{4.11}$$

Considering the case that $g'(x) < 0$ and following the same rationale:

$$f_Y(y) = -f_X\left(g^{-1}(y)\right)\frac{d}{dy}g^{-1}(y). \tag{4.12}$$

Therefore, since $\frac{d}{dy}g^{-1}(y) < 0$ when g is a strictly decreasing function, for both cases:

$$f_Y(y) = f_X\left(g^{-1}(y)\right)\left|\frac{d}{dy}g^{-1}(y)\right|. \tag{4.13}$$

Points of set $G \setminus G^*$. The right side of (4.13) is not defined at these points. For that reason g has been defined equal to zero in this set (note that the definition could be any other finite value). The previous step of the proof has demonstrated that f_Y is the PDF of Y in the set G^*. Now, the proof focuses on demonstrating that f_Y defined as (4.13) satisfies the definition of any PDF of Y in the set G. f_Y is a valid PDF of Y if and only if [75]:

$$\Pr\{a \leq Y \leq b\} = \int_a^b f_Y(y)\, dy. \tag{4.14}$$

Let define the interval $[a, b] \subseteq G$ in such a way that at least one point $y \in [a, b]$ is in the set $G \setminus G^*$. Since D is of measure zero, any subset of it is also of measure zero. Therefore, the result of the integral (4.14) is independent of the values of f_Y in the set $G \setminus G^*$ since it is of measure zero [76].

Points of set $\mathbb{R} \setminus G$. The demonstration for these points is quite obvious. Since the set $\mathbb{R} \setminus G$ is the complementary of G, the probability that Y falls in it is zero. Therefore:

$$\int\limits_{y \in \mathbb{R} \setminus G} f_Y(y)\, dy = 0. \tag{4.15}$$

Since by definition $f_Y(y) \geq 0$ for all y, then $f_Y(y) = 0$ for all $y \in \mathbb{R} \setminus G$. □

Then, from Theorem 4.1:

$$f_{R_i}(r) = \begin{cases} f_{C_i}\left(Q^{-1}(r)\right) \left|\frac{d}{dr} Q^{-1}(r)\right|, & 0 < r < Q(\infty),\ Q^{-1}(r) \notin D, \\ 0, & \text{otherwise.} \end{cases} \tag{4.16}$$

Note that the points of D are the points of change between different TMs, see Figure 3.1. Recalling (4.6), $f_{\rho_i}(\upsilon)$ can be computed from the following theorem.

Theorem 4.2 (theorem 7, section 4.4 of [75])**.** *Let X and Y be two independent random variables with PDFs f_X and f_Y respectively. Let $Z = XY$ and $W = X/Y$. Then the PDFs of Z and W are:*

$$f_Z(z) = \int_{-\infty}^{\infty} f_X(x)\, f_Y\left(\frac{z}{x}\right) \frac{1}{|x|} dx, \tag{4.17}$$

$$f_W(w) = \int_{-\infty}^{\infty} f_X(xw)\, f_Y(x)\, |x|\, dx. \tag{4.18}$$

Thus, if $f_{R_{\min,i}}(r)$ is the PDF of $R_{\min,i}$ (what again can be calculated from user activity) and assuming that $R_{\min,i}$ and R_i are independent:

$$f_{\rho_i}(\upsilon) = \int\limits_0^{\infty} f_{R_{\min,i}}(r\upsilon)\, f_{R_i}(r)\, rdr, \tag{4.19}$$

since $R_i(t) \geq 0$ for all t. Once all f_{ρ_i} are calculated, (4.4) and (4.5) give the exact ERC of all users.

4.3 Measurement-based ERC calculation method

Some mathematical tools and the analytical obtaining of the ERC have been presented in the previous section. Nevertheless, the needed mathematical expressions of the PDFs of the channel SNIR and target bit rate are not always easy to obtain. Therefore, in this section PDFs are replaced with histograms performed over a set of measures of the main variables C_i and $R_{\min,i}$. In addition to approximate PDFs with histograms, this section derives the error of such approximations. Finally, the ERC value obtained from these histograms is also studied.

4.3.1 Histogram definition

Given a specific Random Variable (RV), X, and a set of intervals $I_X = \{I_X^m,\ m = 0 \cdots N_X - 1\}$ being I_X^m the m-th interval and N_X the number of intervals, so that the following conditions are satisfied:

$$I_X^m \cap I_X^k = \emptyset,\ \forall m \neq k, \tag{4.20}$$

$$\Pr\left\{X \notin \bigcup_m I_X^m\right\} = 0, \tag{4.21}$$

the histogram of X, H_X, is a vector whose elements satisfy:

$$H_X(m) = \mathrm{fr}\left\{\mathrm{X} \in \mathrm{I_X^m}\right\}, \tag{4.22}$$

where function $\mathrm{fr}\{\mathrm{x}\}$ is the relative frequency of event x in an experiment.

Therefore, and since (4.20) and (4.21) are satisfied:

$$\sum_m H_X(m) = 1. \tag{4.23}$$

When the number of measurements of an experiment tends to infinity, the relative frequency tends to the probability and:

$$H_X(m) = \Pr\left\{X \in I_X^m\right\}. \tag{4.24}$$

4.3.2 Histograms as PDF estimate

The PDFs of RVs can be approximated with histograms. A histogram sample, $H_X(m)$, is the probability that the RV lies in the interval I_X^m (4.24). However, how this probability is distributed along the interval is something ignored. Nevertheless, it can be supposed or approximated in some way. A first approach may suppose that $H_X(m)$ is homogeneously distributed along the interval I_X^m. Despite this could seem to be a good approach, for the sake of simplicity $H_X(m)$ is usually supposed to be concentrated in a unique point, generally the middle point of I_X^m. Let define $C\left(I_X^m\right)$ as a closed interval with the same limits of I_X^m, i.e $C\left(I_X^m\right)$ includes both limit points, and let define the set of interval representatives $P\left(I_X\right) = \left\{p_X^m : p_X^m \in C\left(I_X^m\right),\ m = 0 \cdots N_X - 1\right\}$. Note that p_X^m is a specific value of variable X included in $C\left(I_X^m\right)$ that represents the interval I_X^m. The representative can be any point of the interval, although among all the possibilities the following three sets are of main relevance:

$$\widetilde{P}\left(I_X\right) = \left\{ \widetilde{p}_X^m : \widetilde{p}_X^m = \frac{\min\limits_{x \in C\left(I_X^m\right)} x + \max\limits_{x \in C\left(I_X^m\right)} x}{2},\ m = 0 \cdots N_X - 1 \right\}, \qquad (4.25)$$

$$\overline{P}\left(I_X\right) = \left\{ \overline{p}_X^m : \overline{p}_X^m = \max_{x \in C\left(I_X^m\right)} x,\ m = 0 \cdots N_X - 1 \right\}, \qquad (4.26)$$

$$\underline{P}\left(I_X\right) = \left\{ \underline{p}_X^m : \underline{p}_X^m = \min_{x \in C\left(I_X^m\right)} x,\ m = 0 \cdots N_X - 1 \right\}. \qquad (4.27)$$

Note that the representatives of $\widetilde{P}\left(I_X\right)$ are the middle points of intervals and the representatives of $\overline{P}\left(I_X\right)$ and $\underline{P}\left(I_X\right)$ are respectively the upper and lower bounds of intervals. $\overline{P}\left(I_X\right)$ and $\underline{P}\left(I_X\right)$ are necessary to obtain the accuracy of this ERC calculation method.

Finally, the PDF and CDF of a RV X can be approximated using the set of representatives $P\left(I_X\right)$ as:

$$\widehat{f}_X\left(x, P\left(I_X\right)\right) = \sum_{m=0}^{N_X-1} \delta\left(x - p_X^m\right) H_X\left(m\right), \qquad (4.28)$$

$$\widehat{F}_X\left(x, P\left(I_X\right)\right) = \int\limits_{-\infty}^{x} \sum_{m=0}^{N_X-1} \delta\left(s - p_X^m\right) H_X\left(m\right) ds = \sum_{\substack{m \\ p_X^m \leq x}} H_X\left(m\right), \qquad (4.29)$$

where $\delta(x)$ is the Dirac delta of x.

Next theorem proves why the sets of intervals $\overline{P}\left(I_X\right)$ and $\underline{P}\left(I_X\right)$ determine the accuracy of the measurement-based calculation of ERC.

Theorem 4.3. *Let X be a RV with histogram $H_X(m)$ over the set of intervals I_X. Then the CDF of X is bounded by:*

$$\widehat{F}_X\left(x, \overline{P}\left(I_X\right)\right) \leq F_X\left(x\right) \leq \widehat{F}_X\left(x, \underline{P}\left(I_X\right)\right). \tag{4.30}$$

Proof. Without loss of generality, let assume that any set I_X is ordered thus $\cdots < p_X^{m-1} < p_X^m < p_X^{m+1} < \cdots$. Then:

$$\widehat{F}_X\left(x, P\left(I_X\right)\right) = \widehat{F}_X\left(p_X^m, P\left(I_X\right)\right), \text{ for all } p_X^m \leq x < p_X^{m+1}. \tag{4.31}$$

Moreover:

$$F_X\left(\overline{p}_X^n\right) = \sum_{\substack{m \\ \overline{p}_X^m \leq \overline{p}_X^n}} H\left(m\right) = \widehat{F}_X\left(\overline{p}_X^n, \overline{P}\left(I_X\right)\right), \tag{4.32}$$

$$\begin{aligned} F_X\left(\underline{p}_X^n\right) &= \sum_{\substack{m \\ \overline{p}_X^m \leq \underline{p}_X^n}} H\left(m\right) = \sum_{\substack{m \\ \underline{p}_X^m < \underline{p}_X^n}} H\left(m\right) = \\ &= \sum_{\substack{m \\ \underline{p}_X^m \leq \underline{p}_X^{n-1}}} H\left(m\right) = \widehat{F}_X\left(\underline{p}_X^{n-1}, \underline{P}\left(I_X\right)\right). \end{aligned} \tag{4.33}$$

Since F_X is increasing:

$$F_X\left(x\right) \geq \widehat{F}_X\left(\overline{p}_X^m, \overline{P}\left(I_X\right)\right), \text{ for all } x \geq \overline{p}_X^m, \tag{4.34}$$

$$F_X\left(x\right) \leq \widehat{F}_X\left(\underline{p}_X^m, \underline{P}\left(I_X\right)\right), \text{ for all } x < \underline{p}_X^{m+1}. \tag{4.35}$$

Finally, from (4.31):

$$\widehat{F}_X\left(x, \overline{P}\left(I_X\right)\right) \leq F_X\left(x\right) \leq \widehat{F}_X\left(x, \underline{P}\left(I_X\right)\right). \tag{4.36}$$

□

Moreover, the CDF approximation calculated using any other set of representatives is also between these two bounds. This last conclusion is somewhat trivial to define a specific theorem but it is of main relevance together with Theorem 4.3. First, because they demonstrate that these bounds enclose the actual CDF and any of its approximations. Second, because it is also proved that $\widehat{F}_X\left(x, \underline{P}\left(I_X\right)\right) - \widehat{F}_X\left(x, \overline{P}\left(I_X\right)\right)$ is the maximum error made by any CDF approximation, provided a certain set of intervals.

4.3.3 $f_{\rho_i}(v)$ computation with histograms

Using the PDF approximation of (4.28) and Theorems 4.1 and 4.2, $f_{\rho_i}(v)$ can be estimated hence obtaining the ERC value. Nevertheless, as demonstrated before, the histogram approximation of the PDF entails a bounded error in the calculation. This section studies how this error is transmitted from the histograms of C_i and $R_{\min,i}$ (obtained from real measures) to the approximated PDF of ρ_i. This study is divided into two steps, first it is analyzed how R_i is affected by C_i and, then, how ρ_i is affected by R_i and $R_{\min,i}$.

Assuming that $0 < r < Q(\infty)$ and from the representatives set $P(I_{C_i})$ the approximated PDF of R_i is:

$$\begin{aligned} \widehat{f}_{R_i}\left(r, P\left(I_{R_i}\right)\right) &= \widehat{f}_{C_i}\left(Q^{-1}(r), P\left(I_{C_i}\right)\right)\left|\frac{d}{dr}Q^{-1}(r)\right| = \\ &= \left|\frac{d}{dr}Q^{-1}(r)\right| \sum_{m=0}^{N_{C_i}-1} \delta\left(Q^{-1}(r) - p_{C_i}^m\right) H_{C_i}(m) = \\ &= \sum_{m=0}^{N_{C_i}-1} \delta\left(r - Q\left(p_{C_i}^m\right)\right) H_{C_i}(m) = \\ &= \widehat{f}_{R_i}\left(r, Q\left(P\left(I_{C_i}\right)\right)\right). \end{aligned} \tag{4.37}$$

This equation allows drawing very relevant conclusions. First, the new set of representatives $P(I_{R_i})$ is composed of the transformations of the elements of $P(I_{C_i})$ with function Q. Moreover, the histogram values do not change, i.e. $H_{R_i}(m) = H_{C_i}(m)$, and the number of samples is the same, i.e. $N_{R_i} = N_{C_i}$. Therefore the histogram of R_i can be easily derived from the histogram of C_i by just calculating the new intervals with function Q and then assigning them the same probability values.

Following a similar rationale, the approximated PDF of ρ_i is (see Appendix 7.2 for further details):

$$\begin{aligned} \widehat{f}_{\rho_i}\left(v, P\left(I_{\rho_i}\right)\right) = & \sum_{n=0}^{N_{R_{\min,i}}-1} \sum_{m=0}^{N_{R_i}-1} \delta\left(v - \frac{p_{R_{\min,i}}^n}{p_{R_i}^m}\right) H_{R_{\min,i}}(n) \cdot \\ & \cdot H_{R_i}(m) = \sum_{k=0}^{N_{\rho_i}-1} \delta\left(v - p_{\rho_i}^k\right) H_{\rho_i}(k). \end{aligned} \tag{4.38}$$

A direct comparison of last equality allows identifying the features of the histogram of ρ_i. The number of histogram intervals increases notably, specifically $N_{\rho_i} = N_{R_{\min,i}} N_{R_i}$, since for every k of $H_{\rho_i}(k)$ there is a unique pair

(n, m) for $H_{R_{\min,i}}(n)$ and $H_{R_i}(m)$. The values of the histogram, $H_{\rho_i}(k)$, are obtained by simply multiplying the corresponding samples of the histograms of $R_{\min,i}$ and R_i, $H_{R_{\min,i}}(n)$ and $H_{R_i}(m)$. Besides, the new set of representatives is calculated dividing both representatives. Thus, the sets of interval extremes are now:

$$\overline{P}(I_{\rho_i}) = \left\{ \overline{p}^m_{\rho_i} = \frac{\overline{p}_{R_{\min,i}}^{\left\lfloor \frac{m}{N_{R_i}} \right\rfloor}}{\underline{p}_{R_i}^{m-\left\lfloor \frac{m}{N_{R_i}} \right\rfloor N_{R_i}}},\ m = 0 \cdots N_{R_{\min,i}} N_{R_i} - 1 \right\}, \tag{4.39}$$

$$\underline{P}(I_{\rho_i}) = \left\{ \underline{p}^m_{\rho_i} = \frac{\underline{p}_{R_{\min,i}}^{\left\lfloor \frac{m}{N_{R_i}} \right\rfloor}}{\overline{p}_{R_i}^{m-\left\lfloor \frac{m}{N_{R_i}} \right\rfloor N_{R_i}}},\ m = 0 \cdots N_{R_{\min,i}} N_{R_i} - 1 \right\}, \tag{4.40}$$

where $\lfloor x \rfloor$ is the greatest integer lower or equal to x.

4.3.4 Convolution of histograms

Previous section has shown how to obtain the PDF of the amount of resources required by a single user. Nevertheless, for the ERC value it is necessary to compute the aggregate resource needs. This can be accomplished with the convolution of all the needed resources, as shown in (4.4). This section studies the convolution of the histograms of ρ_i and which are the bounds of the CDF of the aggregate resource needs. Finally, the ERC value is extracted from the approximated CDF.

Convolutions can be performed one after another, adding one user to the aggregate resources with each new convolution. For that reason and for the sake of simplicity, this section assumes that only two users are in the system. Thus, the approximated $f_\rho(v)$ is (see Appendix 7.2 for further details):

$$\begin{aligned}\widehat{f_\rho}(v, P(I_\rho)) &= \sum_{n=0}^{N_{\rho_1}-1} \sum_{m=0}^{N_{\rho_2}-1} \delta\left(v - p^n_{\rho_1} - p^m_{\rho_2}\right) H_{\rho_1}(n) H_{\rho_2}(m) = \\ &= \sum_{k=0}^{N_\rho - 1} \delta\left(x - p^k_\rho\right) H_\rho(k).\end{aligned} \tag{4.41}$$

Like in (4.38), the number of samples of the histogram increases. In this case, the number of samples increases with each user aggregation, being even possible to overflow the available memory. Considering, for example, U users and that all histograms of ρ_i have N samples, the histogram H_ρ has a total of N^U samples. Even with few users, the number of samples could be unmanageable. However, the number of samples can be reduced selecting the sets of representatives with the following closed form:

$$P\left(I_{\rho_i}\right)=\left\{p_{\rho_i}^m=m\Delta p+\upsilon_i,\ m=0\cdots N_{\rho_i}-1\right\}, \tag{4.42}$$

where Δp is an increment common for all representatives sets and υ_i is the lowest representative for the i-th user. If intervals are of the same size and equidistant then $\widetilde{P}\left(I_{\rho_i}\right)$, $\overline{P}\left(I_{\rho_i}\right)$ and $\underline{P}\left(I_{\rho_i}\right)$ satisfy this condition. Then, with these types of sets:

$$\widehat{f}_\rho\left(\upsilon,P\left(I_\rho\right)\right)=\sum_{n=0}^{N_{\rho_1}-1}\sum_{m=0}^{N_{\rho_2}-1}\delta\left(\upsilon-(n+m)\,\Delta p-\upsilon_1-\upsilon_2\right)H_{\rho_1}(n)\,H_{\rho_2}(m). \tag{4.43}$$

Thus, for all n and m so that $k=n+m$, the corresponding product of histogram samples has the same Dirac delta, i.e. $\delta(\upsilon-k\Delta p-\upsilon_1-\upsilon_2)$, and hence can be stored in the same sample. The new histogram has only $N_{\rho_1}+N_{\rho_2}-1$ samples, which, continuing with the same example, reduces the previous N^U samples to only $U(N-1)+1$. Note that the histogram samples of the aggregate resource needs can be computed as the discrete convolution, hence:

$$\widehat{f}_\rho\left(\upsilon,P\left(I_\rho\right)\right)=\sum_{m=0}^{N_\rho-1}\delta\left(\upsilon-p_\rho^m\right)H_\rho(m), \tag{4.44}$$

$$N_\rho=N_{\rho_1}+N_{\rho_2}-1, \tag{4.45}$$

$$p_\rho^m=m\Delta p+\upsilon_1+\upsilon_2, \tag{4.46}$$

$$H_\rho(m)=\sum_{k=\max\left(0,m-N_{\rho_2}+1\right)}^{\min\left(N_{\rho_1}-1,m\right)}H_{\rho_1}(k)\,H_{\rho_2}(m-k)\,. \tag{4.47}$$

The ERC value can be approximated by:

$$\mathrm{ERC}\simeq\widehat{F}_\rho^{-1}\left(1-\varepsilon,P\left(I_\rho\right)\right)=p_\rho^{m_0}, \tag{4.48}$$

$$m_0=\min\left\{m:\sum_{n=0}^{m}H_\rho(n)\geq1-\varepsilon\right\}. \tag{4.49}$$

Due to the approximation of the PDF, the measurement-based ERC calculation method is not exact. However, the actual ERC value lies in a certain interval that can be also calculated. The upper bound of ERC is determined using $\overline{P}(I_\rho)$ in (4.48), whereas the lower one comes from $\underline{P}(I_\rho)$. Like in the case of the CDF error, these bounds give the maximum error made in the ERC computation.

4.3.5 ERC computation accuracy

This section studies the characteristics of the ERC computation error and also how to reduce it. With U users, the sets of interval extremes are:

$$\overline{P}(I_\rho) = \left\{ \overline{p}_\rho^m = m\Delta p + \sum_{i=1}^{U} \overline{v}_i,\ m = 0 \cdots \sum_{i=1}^{U} N_{\rho_i} - U + 1 \right\}, \tag{4.50}$$

$$\underline{P}(I_\rho) = \left\{ \underline{p}_\rho^m = m\Delta p + \sum_{i=1}^{U} \underline{v}_i,\ m = 0 \cdots \sum_{i=1}^{U} N_{\rho_i} - U + 1 \right\}, \tag{4.51}$$

where $\overline{v}_i$ and $\underline{v}_i$ are the minimum elements of $\overline{P}(I_{\rho_i})$ and $\underline{P}(I_{\rho_i})$ respectively. Thus, the maximum error is:

$$\text{error} = \overline{p}_\rho^{m_0} - \underline{p}_\rho^{m_0} = \sum_{i=1}^{U} (\overline{v}_i - \underline{v}_i). \tag{4.52}$$

Note that $\overline{v}_i - \underline{v}_i$ is the intervals length for the i-th user. Consequently, although the ERC value computed from histograms has certain error, this error can be reduced using shorter intervals. Nevertheless, if intervals are shortened generally more intervals would be necessary to keep on satisfying (4.21). Moreover, more intervals mean more memory to store histograms and more computational burden (in terms of number of operations) in the calculation of the ERC.

4.4 Gaussian approximation

Previous section has described a feasible approach to obtain the upper and lower bounds of the ERC of several users. Nevertheless, computational cost of an accurate version (with enough intervals) makes this approach be too slow for giving real-time solutions. This problem has two possible solutions: computing the ERC values offline or approximating the aggregate resource needs by means of one of the methods described in the introduction.

Regarding the first option, users generally belong to one service of a finite set of service classes. Therefore, the PDFs of $R_{\mathrm{min,i}}$ depend only on the specific service class. Moreover, their channel quality statistics are very similar to a finite set of patterns. Each of these patterns comes from different mobility models - static, pedestrian or vehicular - and from different locations - cell center or edge. Thanks to these facts, users can be allocated to some a priori $R_{\mathrm{min,i}}$ and C_i PDFs. Consequently, it is possible to compute all possible ERC values offline (different combinations of users), store them in a database and access to the right one on demand.

Regarding the approximation of sources, in the EBW framework this problem was studied for the traffic generation rate. The Gaussian approximation is the most extended method since, from the central limit theorem, the PDF of any sum of independent identically distributed RVs tends to the Gaussian distribution. This section studies this last approach.

Let S be the number of different mixes of service classes and mobility patterns and $f_{\rho_s}(\upsilon)$ the pdf of the resources needed by a user of the s-th mix. Moreover, let μ_s and σ_s be its mean and standard deviation, respectively. If U_s users of the s-th mix are in the system, the PDF of their resource needs can be approximated by:

$$\underbrace{f_{\rho_s}(\upsilon) * f_{\rho_s}(\upsilon) * \cdots * f_{\rho_s}(\upsilon)}_{U_s} \cong \mathcal{N}\left(\upsilon, U_s\mu_s, U_s\sigma_s^2\right), \tag{4.53}$$

$$\mathcal{N}\left(\upsilon, \mu, \sigma^2\right) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\upsilon-\mu)^2}{2\sigma^2}}. \tag{4.54}$$

The overall $f_\rho(\upsilon)$ can be computed as the convolution of all the normal distributions obtained for the different service and mobility mixes. Such PDF is also normally distributed with mean the sum of means and variance the sum of variances. Thus:

$$f_\rho(\upsilon) \cong \mathcal{N}\left(\upsilon, \sum_{s=1}^{S} U_s\mu_s, \sum_{s=1}^{S} U_s\sigma_s^2\right), \tag{4.55}$$

$$F_\rho(\upsilon) \cong \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{\upsilon - \sum\limits_{s=1}^{S} U_s\mu_s}{\sqrt{2\sum\limits_{s=1}^{S} U_s\sigma_s^2}}\right)\right). \tag{4.56}$$

Finally, the ERC can be approximated as:

$$\mathrm{ERC} \cong \mathrm{erf}^{-1}(1-2\varepsilon)\sqrt{2\sum_{s=1}^{S} U_s\sigma_s^2} + \sum_{s=1}^{S} U_s\mu_s, \tag{4.57}$$

where $\mathrm{erf}(x)$ is the error function of x.

4.5 Example of ERC calculation

This section provides some results of the ERC calculation methods explained in the previous sections. The different approaches to compute the ERC value were applied in a simulated GPRS scenario considering downlink. GPRS has 4 Coding Schemes (CSs) with different error protections. Each CS has the following theoretic bit rates per slot: 9.05 kb/s for the CS1, 13.4 kb/s for the CS2, 15.6 kb/s for the CS3 and 21.4 kb/s for the CS4 [1]. Nevertheless, the transmitted bits are not error free and, thus, the effective throughput of each CS depends on the perceived SNIR. In [1], some curves of effective throughput vs. SNIR were obtained by simulating users moving at a constant speed of 50 km/h. Function Q can be obtained from these curves as shown in Figure 3.1. This function has 3 non-differentiability points which correspond with the 3 points of change between the 4 CSs. Therefore, in case of an analytical obtaining of the ERC, the set D of Theorem 4.1 is only composed by these three points.

The simulation scenario consisted of a cell with radius 0.5 km. Similarly to Chapter 3, the path loss of each user was obtained as in [50]. Considering a carrier frequency of 1800 MHz and a base station antenna height of 15 meters, the formula becomes:

$$L_{p,i} = 127.2 + 37.6 \log_{10}(d_i), \qquad (4.58)$$

where d_i is the distance in km between the i-th user and the center cell. For the sake of coherence with [1], users were moving at 50 km/h and thus all of them followed the same mobility pattern. The thermal noise power was set to -102 dBm. Provided a cluster size of 4, only the six nearest co-channel cells were considered as interferers. Within this scenario, a first simulation measured 10^5 users moving in the cell during 100 seconds. The SNIR histogram was obtained from these measures.

All users were supposed to belong to the same service class, an ON/OFF traffic pattern with a fixed target bit rate of $R_{\min} = 10$ kb/s for the ON periods. This service tries to emulate typical Voice over IP (VoIP) with voice activity detectors. The ON/OFF model was implemented with a two-state discrete-time Markov chain. The mean ON and OFF periods were 1.2 s and 1.8 s respectively, just as in [77, 78]. Using this traffic model, the $R_{\mathrm{min,i}}$ PDF is of the form:

$$f_{R_{\mathrm{min,i}}}(r) = 0.6\delta(r) + 0.4\delta(r - 10000). \qquad (4.59)$$

The studied approaches were: Least Conservative Histogram Approximation (LCHA), Interval Mid-point Histogram Approximation (IMHA), Most Conservative Histogram Approximation (MCHA) and Gaussian Approximation (GA). First three approaches (LCHA, IMHA and MCHA) use sets $\underline{P}(I_X)$ (4.27),

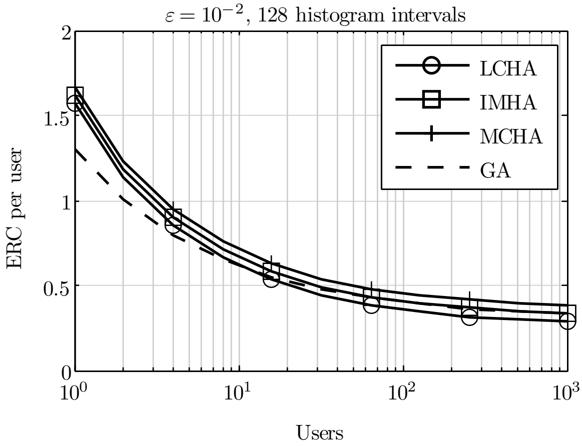

Figure 4.1: ERC value per user.

$\widetilde{P}(I_X)$ (4.25) and $\overline{P}(I_X)$ (4.26) as intervals representatives respectively. Besides, GA follows the approach explained in Section 4.4. Results shown in this section can be grouped into two main types: ERC values and non-satisfaction probabilities. The ERC values were obtained for each calculation method considering some specific number of users in the system and using the SNIR histogram obtained from the first simulation. The non-satisfaction probabilities results were averaged over 10^4 additional simulations. In each simulation, all users moved in the cell during 100 seconds demanding resources. Hence, the non-satisfaction probability was computed as the normalized number of times in which users demanded more resources than the ERC value.

The performance of the 4 approaches was studied versus three different variables: number of users, QoS requirement ε and number of histogram intervals.

4.5.1 Performance vs. number of users

Figure 4.1 represents the ERC per user obtained with the 4 approaches for $\varepsilon = 10^{-2}$ and 128 histogram intervals. The most remarkable conclusion is that this value decreases with the number of users since the aggregate resource needs are less than the sum of each individual resources needs. This fact was pointed out at the beginning of Section 4.2. The accuracy of each approach can be checked in Figure 4.2. LCHA is always over the objective QoS requirement whereas MCHA is always below, as expected. These approaches are the two extremes of the histogram approximation. With more users, more discrete

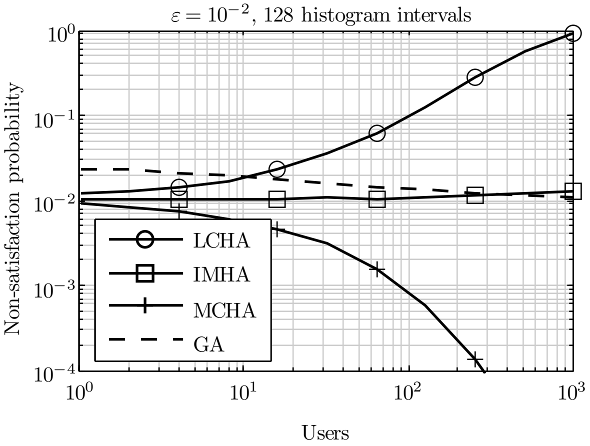

Figure 4.2: Non-satisfaction probability with the ERC values of Figure 4.1.

convolutions are calculated and, hence, both approaches are less accurate. GA behavior is completely different. This approach makes more error with few users but improves its accuracy as the number of users increase. In fact, the error tends to zero due to the central limit theorem. The initial error of this approach depends on the shape of the SNIR PDF. If the SNIR had a Gaussian-like PDF, the error would be much lower than in other cases. Consequently, the error could be very significant if the PDF is not Gaussian-like and the number of users is not high enough. IMHA is generally the best approach although GA outperforms it with more than 400 users. IMHA remains close to the QoS objective for the first 100 or 200 users. With more users, this approach is slightly separated from the objective due to the less accuracy of the histogram approximation. Therefore, the best solution is using IMHA method for few users and GA for many users. The threshold between both approaches depends on the desired accuracy and computational capabilities of the hardware.

4.5.2 Performance vs. QoS requirement

Figures 4.3 and 4.4 show the ERC values and non-satisfaction probabilities respectively for 16 users, 128 histogram intervals and a QoS requirement ranging from 10^{-3} to 10^{-1}. IMHA continues being very close to the QoS target independently of its value. LCHA is also over the objective and MCHA is below. Again, GA has a different behavior. Although the absolute error of the non-satisfaction probability is always around 0.006 (a maximum of 0.008 and

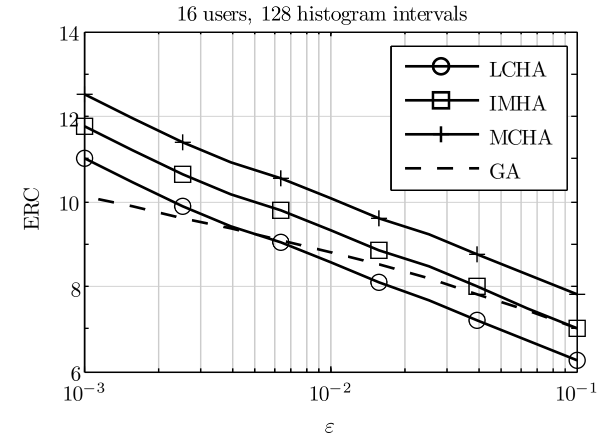

Figure 4.3: ERC value.

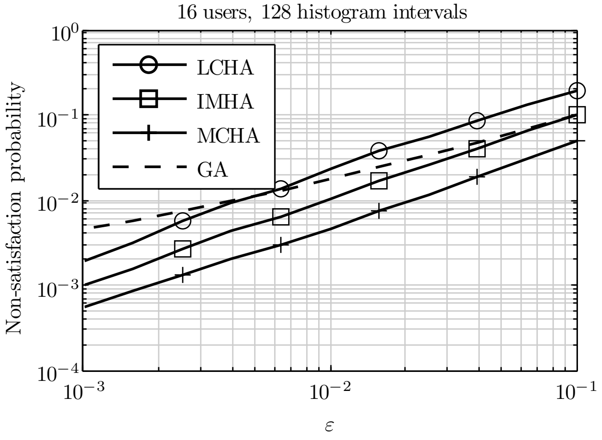

Figure 4.4: Non-satisfaction probability with the ERC values of Figure 4.3.

a minimum of 0.004) the relative error is notably reduced with higher ε: from 3.56 with $\varepsilon = 10^{-3}$ to 0.026 with $\varepsilon = 10^{-1}$. Thus, GA not only improves its accuracy with more users but also with more relaxed QoS requirements. Hence, ε value is another variable to take into account when selecting the most proper approach, IMHA or GA.

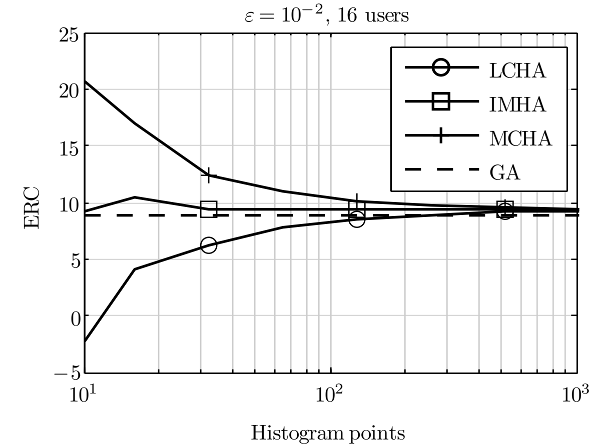

Figure 4.5: ERC value.

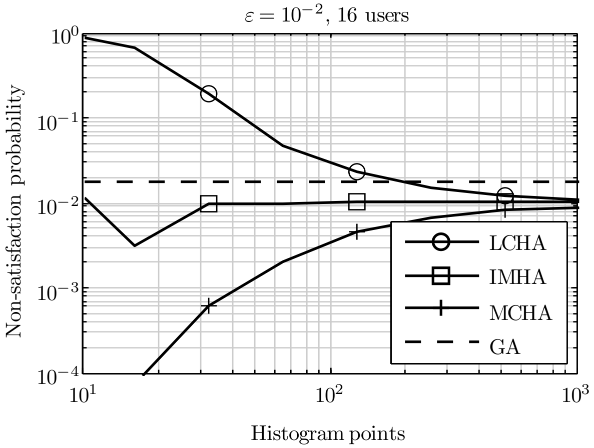

Figure 4.6: Non-satisfaction probability with the ERC values of Figure 4.5.

4.5.3 Performance vs. number of histogram intervals

Figures 4.5 and 4.6 depict the results of this section for $\varepsilon = 10^{-2}$ and 16 users. The number of histogram intervals ranges from 10 to 1000. Histogram approximation extremes (LCHA and MCHA) get closer with more intervals. Figure 4.6 also shows how both extremes get closer to the QoS requirement.

Therefore, error is reduced and not only for LCHA and MCHA but also for any approach using histograms, including IMHA. The maximum relative error of the non-satisfaction probability for any histogram-based approach falls from 99 with 10 intervals to less than 0.12 with 1000 intervals.

The number of histogram intervals must be carefully selected. As demonstrated, error can be significantly reduced but many intervals may consume too many computer resources. The main advantage of GA is precisely its simplicity and reduced computational burden, making possible a fast calculation of the ERC value and ensuring enough accuracy at least with a high number of users.

4.6 ERC application to CAC

So far, this chapter has addressed the calculation of the ERC. This is by far the most complex part of a CAC based on ERC in a unique RAT. Once this complicated part is solved, new calls are accepted only if the ERC of all calls (ongoing and new ones) is less or equal to the maximum quantity of resources available in the system. This simple algorithm was compared with a Load Threshold-Based (LTB) CAC algorithm. Despite Section 1.2.2 has introduced other CAC techniques for signal quality control (based on either interference or SNIR level), this approach is the most common in current wireless networks. Regarding handover failure, it has not been taken into account in this analysis for the sake of simplicity. Note that band guard techniques can be applied to ERC-based CAC reducing the quantity of system resources for incoming calls. Finally, it is worth noting that QoS satisfaction is implicitly addressed in the ERC definition. This section provides some results of this evaluation to stress the good behavior of an ERC-based CAC.

The reference algorithm admits new calls only if the average amount of resources that ongoing calls consume is less or equal to the threshold. The simulated scenario was the same described in the previous section but with three different services in order to assess the performance of ERC in a multi-service scenario. Two services generate ON/OFF traffic with $R_{\min} = 10$ kb/s and $R_{\min} = 20$ kb/s respectively during ON periods. Both have an average duration of ON and OFF periods of 1.2 s and 1.8 s. The last service has a constant $R_{\min} = 5$ kb/s without OFF periods. All users were randomly allocated to any of these three services with equal probability.

The results of this section were obtained with 10^4 different seeds. For each seed, several simulations were conducted with an increasing number of users ranging from 1 to $U_{\max}$. $U_{\max}$ is sufficiently large so that this quantity of users can never be satisfied. After performing all simulations for a specific seed, the

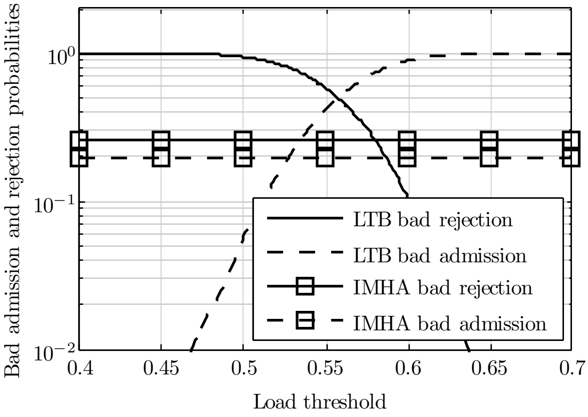

Figure 4.7: Bad admission and rejection probabilities for a LTB CAC and the ERC-IMHA based CAC.

quantity of users U_{opt} can be calculated so that:

$$\Pr\left\{\sum_{i=1}^{U_{\text{opt}}} \rho_i(t) > \rho_{\max}\right\} \leq \varepsilon, \tag{4.60}$$

$$\Pr\left\{\sum_{i=1}^{U_{\text{opt}}+1} \rho_i(t) > \rho_{\max}\right\} > \varepsilon, \tag{4.61}$$

where $\rho_{\max}$ is the maximum quantity of resources in the system. For this section $\rho_{\max} = 16$ GPRS time slots and $\varepsilon = 10^{-2}$. Thus, the first U_{opt} users satisfy the QoS requirements whereas the first $U_{\text{opt}} + 1$ users do not. Note that the probabilities of (4.60) and (4.61) were obtained a posteriori, i.e. after simulation, hence they are the actual percentage of times in which the QoS is not satisfied.

According to U_{opt} definition, in an optimal CAC algorithm the $U_{\text{opt}}+1$ user should not be accepted but the U_{opt} user should be. A bad admission is done if a CAC algorithm decides to admit the $U_{\text{opt}} + 1$ user while a bad rejection is done if a CAC algorithm decides to not accept the U_{opt} user. After simulations, the different CAC algorithms were tested obtaining their bad admission and bad rejection probabilities.

Figure 4.7 depicts the bad admission and bad rejection probabilities as a function of the load threshold using the LTB CAC. This figure includes these

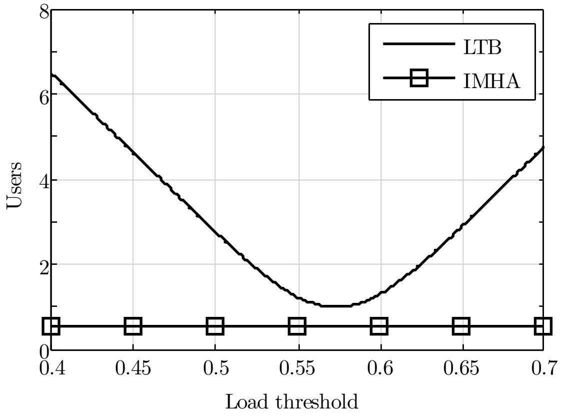

Figure 4.8: Average deviation from U_{opt}.

probabilities for the CAC based on the ERC-IMHA technique. The main advantage of the LTB CAC algorithm is that bad admission probability can be adjusted as desired modifying the load threshold, although there is not a priori knowledge of the optimum threshold. Nevertheless, for low bad admission probabilities the bad rejection probability is especially high, thus wasting lots of resources. On the other hand, the IMHA technique obtains a good bad admission probability but with a notably low bad rejection probability in comparison with the LTB CAC technique. Specifically, the IMHA technique shows a bad admission probability of 19.5% and a bad rejection probability of 26.3%. However, the LTB CAC with a load threshold of 52.7% has also a bad admission probability of 19.5% but with the subsequent bad rejection probability of 80.4%.

The fact that the LTB CAC wastes lots of resources can be seen in Figure 4.8. At the load threshold of 52.7%, the LTB CAC algorithm deviates an average of 1.82 users from the optimum quantity of users to be admitted, i.e. U_{opt}. Since the bad rejection probability is 80.4%, with this threshold the number of admitted users is generally below U_{opt}. This fact shows that almost two users more could be admitted, in average. On the other hand, the IMHA technique has an average deviation of only 0.52 users. Moreover, this value is lower than the lowest value obtained with the LTB approach.

4.7 Conclusion

This chapter has proposed three different approaches to the ERC calculation of traffic sources with QoS constraints in a wireless system: analytical, measurement-based and with Gaussian approximation.

The analytical approach is exact but in most cases unfeasible due to the unavailability of the actual PDFs of system variables. However, this method must be formulated to derive the other two feasible methods.

The measurement-based approach makes the ERC calculation possible but with a high computational burden. Nevertheless, the histograms and even the ERC calculation can be performed offline and, thus, the additional cost is not so critical. Moreover, the exactness of the obtained ERC depends directly on the histograms accuracy. Therefore, increasing the number of measurements of the histogram, i.e. input data, and the number of bins involves a more accurate ERC calculation. Hence, it is possible to obtain an ERC as accurate as desired, what is not possible with the simplified models for the aggregate traffic generation rate proposed in the literature (see Section 4.1).

The last approach uses the Gaussian approximation which is computationally fast and accurate for many users. Therefore, an optimum ERC calculation implementation should include both measurement-based and Gaussian approximation approaches, switching the method according to the number of users. This approach ensures good accuracy and low computational cost in all situations.

Finally, ERC helps the CAC algorithms to perform a fair call admission control. Section 4.6 has shown the joint reduced bad admission and rejection probabilities that can be achieved with an ERC-based CAC. Moreover, the ERC calculation can be easily particularized for each user, scenario and technology with the only knowledge of the corresponding histograms which can be directly obtained from Operation and Maintenance Centers using call tracing tools. This ability of the ERC to work in an heterogeneous scenario will be used in the next chapter to define a JCAC algorithm that uses the ERC to decide on call admission.

Chapter 5

ERC-based JCAC algorithm

5.1 Main concept

Section 4.6 has shown how easy the decision making on call admission in a unique RAT is after calculating the ERC. Basically, after computing the ERC, the following quantity can be obtained:

$$\gamma = \frac{\rho_{\max}}{\mathrm{ERC}}, \tag{5.1}$$

where $\rho_{\max}$ is the maximum quantity of resources of the RAT. Then, the new users can be accepted only if $\gamma \geq 1$. Nevertheless, the quantity γ gives much more information about the capacity of the RAT for serving these users. For example, if $\gamma = 2$ then the RAT could serve twice the load generated by these users. Note that this fact does not mean that twice the users could be served since users can have different traffic and mobility patterns and, although all them had the same patterns, twice the users do not generate twice the load[1]. Continuing with the example, if $\gamma = 0.5$ then the RAT could serve only half the total load. In this case, if there is another RAT that could serve the other half, all users could be admitted in this heterogenous network. This is the basic idea of the JCAC algorithm proposed in this Thesis.

Let assume that there are a total of K RATs, then the following quantities:

$$\gamma_k = \frac{\rho_{\max,k}}{\mathrm{ERC}_k},\ k = 1 \cdots K, \tag{5.2}$$

[1] This fact was already explained in Section 4.2 and in Figure 4.1

can be computed for each one. The quantities $\rho_{\max,k}$ and ERC_k are respectively the maximum quantity of resources of the k-th RAT and the ERC of all the users for the k-th RAT. Thus, all users can be served with this heterogeneous network if:

$$\sum_{k=1}^{K} \gamma_k \geq 1. \tag{5.3}$$

It is worth noting that this approach does not give information about how users should be distributed among RATs or how resources should be allocated to users. These two functions are already provided by the JDRA algorithm. Thus, this technique just says if it is possible to satisfy the QoS of all users at least with the desired probability.

5.2 Description of the algorithm

In order to correctly apply the main concept of previous section, all RATs must have the same coverage area. Nevertheless, this must not (or at least should not) be assumed since, in general, this will not be true. The main problem here is that the ERC has not been defined for taking non-coverage zones into account. That is to say, if the area of interest has some non-coverage zone, then users would require infinite resources in that zone. If the zone is big enough, the probability that users require infinite resources may be greater than the objective ε. Thus, the JCAC algorithm must not include non-coverage zones in the ERC calculus.

To this aim, the area of interest will be divided into different zones. These zones are the intersections of the coverage areas of each RAT. For instance, with two RATs there will be at most 3 zones, one with coverage of the first RAT only, another one with coverage of the second RAT only and a third one with coverage of both RATs. Similarly, with three RATs there are 7 zones at most. In general, with K RATs there are $2^K - 1$ zones at most.

Figure 5.1 depicts an example where 3 RATs (A, B and C) conform 4 zones (1, 2, 3, and 4). Obviously, computing the ERC of RAT C in zones 1 and 2 is not possible since that RAT has no coverage. The basic idea is to apply the main concept of previous section in each zone separately. In other words, the load generated by users in each zone must be divided only into the RATs that have coverage. Thus, for instance, the load generated in zone 1 must be served only by RAT B. That means that it is necessary to be able to compute the ERC only in that zone and to ensure that RAT B has enough resources to support that ERC. Similarly, the ERC for RATs A and B should be computed only for zone 2, then the main concept could be applied in that zone for A and B. Only if the load generated can be served in all zones, users can be satisfied

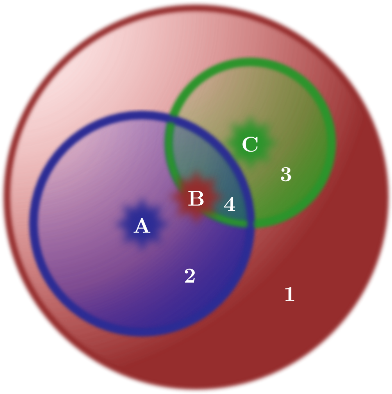

Figure 5.1: Example of 3 RATs creating 4 zones.

in the heterogeneous network. Additionally, more constraints should be added to ensure that the sum of all the ERC values in all zones do not exceed the maximum quantity of resources of each RAT. This section explains how to perform the admission control jointly in all RATs following this idea.

Let $f_{C_{izk}}(c)$ be the PDF of the SNIR perceived by the i-th user when he is in the z-th zone and from the k-th RAT, then it is possible to compute the PDF of the maximum bit rate per r.u. that the i-th user can reach in the z-th zone and in the k-th RAT, i.e. $f_{R_{izk}}(r)$. This calculus is similar to that explained in (4.16). Moreover and similarly to (4.19), the PDF of the resources needed by the i-th user in the k-th RAT when he is in the z-th zone is:

$$f^*_{\rho_{izk}}(\upsilon) = \int_0^\infty f_{R_{\min,i}}(r\upsilon)\, f_{R_{izk}}(r)\, rdr. \tag{5.4}$$

It is worth noting that the PDF of the resources obtained in (5.4) does not take into account the portion of time user i is outside zone z. That is why $f^*_{\rho_{izk}}(\upsilon)$ has been defined as the PDF of resources when the i-th user *is* in the z-th zone. This fact is due to how $f_{C_{izk}}(c)$ (or its corresponding histogram) is obtained. The SNIR levels to compute it are measured only when the i-th user *is* in the z-th zone. However, if P_{iz} is the probability that the i-th user is located in the z-th zone, the PDF of the resources needed by the i-th user in the k-th RAT and the z-th zone can be computed as:

$$f_{\rho_{izk}}(\upsilon) = (1 - P_{iz})\, \delta(\upsilon) + P_{iz} f^*_{\rho_{izk}}(\upsilon). \tag{5.5}$$

Table 5.1: Example of quantities γ_{zk} for the example of Figure 5.1.

	1	2	3	4
A	-	2	-	0.8
B	1.5	1	2	0.8
C	-	-	3	0.8

Obviously, if the i-th user is not in the z-th zone, no resources originated in this zone are demanded from him. This is the reason for adding a Dirac delta centered in 0 in (5.5). This fact does not mean that the user is not demanding any resources if he is outside the z-th zone, but these resources should not be added to this PDF but to the PDF of the corresponding zone and RAT.

Once all $f_{\rho_{izk}}(\upsilon)$ for some z and k have been computed, the ERC of all users in that zone and RAT can be also derived. The ERC obtaining from the PDFs of the needed resources has been already discussed in Chapter 4. All methods and results of that chapter apply also here. Thus, the PDF of the aggregate resources needs and the ERC of all users in the z-th zone and the k-th RAT are respectively:

$$f_{\rho_{zk}}(\upsilon) = f_{\rho_{1zk}}(\upsilon) * f_{\rho_{2zk}}(\upsilon) * \cdots * f_{\rho_{Uzk}}(\upsilon), \tag{5.6}$$

$$\mathrm{ERC}_{zk} = F^{-1}_{\rho_{zk}}(1-\varepsilon), \tag{5.7}$$

where U is the number of users and ε is the target non-satisfaction probability of the QoS requirements. Analogously to (5.2), from the ERC values, the following quantities can be obtained:

$$\gamma_{zk} = \frac{\rho_{\max,k}}{\mathrm{ERC}_{zk}}. \tag{5.8}$$

These quantities represent the percentage of the load generated by the users that each RAT is capable of serving using all resources in each individual zone. Obviously, RATs can not use all their resources in all zones at the same time and, moreover, the 100% of the load will be divided into all of them.

In order to better explain the following steps, let us continue with the example of Figure 5.1. Table 5.1 shows a possible set of values γ_{zk} for this example. According to the value of $\gamma_{1\mathrm{B}}$ the load generated in zone 1 can be served with the RAT B alone. Obviously, this is a necessary condition since no other RAT has coverage in zone 1. Moreover, after reserving the required resources for this zone, RAT B still has 1/3 of its resources free for the rest of zones. From now on there are several possibilities, for instance RAT B could spend all its remaining resources in zone 2 to free as much resources as possible from RAT A so it could use them in zone 4. On the other hand, RAT A could serve all the load in zone 2 and leave the resources of RAT B for zones 3 and 4. A third option is a mix of the previous two. It is not easy to know the best solution a priori or, what is more important, if all users could be satisfied in all zones with any of these options. For this example, let us assume that RAT A serves all the load of zone 2 and RAT C all the load of zone 3. Then, RAT A still has 1/2 of its resources for zone 4 and RAT C 2/3. Using their remaining resources, RAT A can serve the 40% of the load of zone 4, RAT B the 26.7% and RAT C the 53.3%. All together sum more than the 100% and consequently, users in zone 4 can be satisfied too. Hence, the heterogeneous network of Figure 5.1 can satisfy a set of users that generate a load so that the γ_{zk} values are those of Table 5.1. The procedure followed to conclude this has been finding a distribution of load between RATs in all zones in such a way that no RAT needed more resources than their maximum available. Although this distribution could be understood as the future resource allocation, it is not. The JDRA algorithm is in charge of that. This solution only states that it is possible to satisfy all users with the desired probability. The rest of this section explains how to conclude if a good solution exists or not to any general heterogeneous network and γ_{zk} values. If so, and only if so, all users could be served by the system.

Let define p_{zk} as the percentage of load that the k-th RAT could serve at the z-th zone. Then, the following constraint ensures that all users will be satisfied in each zone:

$$\sum_{k=1}^{K} p_{zk} \geq 1. \tag{5.9}$$

Obviously, every p_{zk} must also satisfy:

$$p_{zk} \geq 0, \tag{5.10}$$

$$p_{zk} \leq 1, \tag{5.11}$$

$$p_{zk} = 0, \text{ if the } k\text{-th RAT has no coverage in the } z\text{-th zone.} \tag{5.12}$$

Additionally, all RATs have a maximum quantity of resources that must be divided into all zones. That means that the sum of all the resource quantities consumed in each zone must not exceed the maximum, i.e.:

$$\sum_{z=1}^{Z} p_{zk} \mathrm{ERC}_{zk} \leq \rho_{\max,k}, \tag{5.13}$$

where Z is the number of zones. Thus, from (5.8):

$$\sum_{z=1}^{Z} \frac{p_{zk}}{\gamma_{zk}} \leq 1. \tag{5.14}$$

Let define the vector $\mathbf{p}$ as an ZK-dimensional vector whose components are the variables p_{zk}. Then, constraints (5.9), (5.10), (5.11), (5.12) and (5.14) can be written in matrix form so that:

$$\begin{gathered} \text{(5.11) and (5.14): } \mathbf{Ap} \leq 1, \\ \text{(5.9): } \mathbf{Bp} \geq 1, \\ \text{(5.12): } \mathbf{Cp} = 0, \\ \text{(5.10): } \mathbf{p} \geq 0, \end{gathered} \tag{5.15}$$

where matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ have non-negative elements. Any ZK-dimensional point $\mathbf{p}$ that satisfies (5.15) is a possible distribution of load among RATs and zones. If such a point exists then all users can be satisfied in the heterogeneous network, if not then the new users should not be accepted. Note that the point here is not to find the best $\mathbf{p}$ that satisfies (5.15) but to know if such a point exists or, what is the same thing, if the polytope defined by (5.15) is empty or not.

5.3 Feasibility and infeasibility of optimization problems

In optimization problems, especially in linear programming, it is of main relevance to know if the problem is feasible, i.e. if it has a solution, or not. There are several algorithms based on the *simplex* algorithm (see [79] as an example) that are able to find the optimum solution of any LP. However, they must be initialized with a feasible solution, i.e. a point that satisfies all constraints. The technique commonly used to find a feasible solution is to define an auxiliary LP that has an easy-to-find feasible solution and whose optimum may be a feasible solution of the original LP. Moreover, if the solution is not feasible, then the original LP is unfeasible. With this idea in mind, let define the following LP:

$$\begin{gathered} \text{Minimize } p_0, \\ \mathbf{Ap} \leq 1, \\ \mathbf{Bp} + p_0 \geq 1, \\ \mathbf{Cp} = 0, \\ \mathbf{p} \geq 0, \\ p_0 \geq 0. \end{gathered} \tag{5.16}$$

A feasible solution of this problem is very easy to find. Note that making $\mathbf{p} = 0$ and $p_0 = 1$ all constraints of (5.16) are satisfied. This feasible solution can be used as the initial point of the *simplex* algorithm to find the optimum solution of (5.16). In the optimum solution, if the value of the function to minimize is 0 then $p_0 = 0$ (note that the cost function cannot take any lower value since $p_0 \geq 0$). In this case, the constraints of (5.16) become the constraints of (5.15) and the value of $\mathbf{p}$ in that solution is a feasible point of (5.15). If in the optimum solution $p_0 > 0$ then the polytope defined by (5.15) is empty and there is no point $\mathbf{p}$ that satisfies these constraints.

5.4 Simulation and results

The JCAC algorithm described in this chapter was tested in the same simulation scenario presented in Section 3.5. Therefore, each cell consists of two RATs, one HSDPA and another WLAN with both transmitters in the cell center. Users move following the mobility model described in Section 3.5.4. Moreover, for the ERC computation, histograms of 128 intervals and the IMHA approach were used.

For these simulations, users belong to any of three different services with the same probability. All of them generate ON/OFF traffic with $R_{\min} = 10$ kb/s, $R_{\min} = 100$ kb/s and $R_{\min} = 1000$ kb/s respectively during ON periods. Moreover, they have an average duration of ON and OFF periods of 1.2 s and 1.8 s. It is worth noting that these services were defined as such for computational reasons as it will be explained next. Firstly, note that Section 4.6 already showed that the ERC can be applied to systems with several services, thus proving this again is not an objective of this section. Secondly, the figures that will be shown in this chapter were generated modifying certain parameters (like ε) that notably increase the capacity of the heterogeneous system for some values. For instance, for $\varepsilon = 10^{-3}$ 7 users with $R_{\min} = 10$ kb/s during the ON periods could be admitted in the system if the 90% are in the hotspot, although up to 740 could be admitted for $\varepsilon = 10^{-1}$. Simulating 740 users has an extremely high computational cost in terms of memory and time. For that reason, the other two services were defined to introduce more load. Following with the previous example, only 28 users with $R_{\min} = 1000$ kb/s could be admitted for $\varepsilon = 10^{-1}$.

Using this scenario a first simulation was conducted to obtain the needed PDFs. 10^5 users were generated and moved during 100 simulation seconds. As expected, two different zones were detected during this first simulation, a zone with only coverage of HSDPA and another with coverage of both HSDPA and WLAN. The probabilities of being in each zone were also computed from the measurements of users. With all these data, it was possible to compute the PDFs of the resources needed by one user of each service. These PDFs were stored for their later use in the final simulations.

After the first simulation, 400 simulations were run with different seeds. For each simulation, a list of 150 users were generated. Each simulation comprises

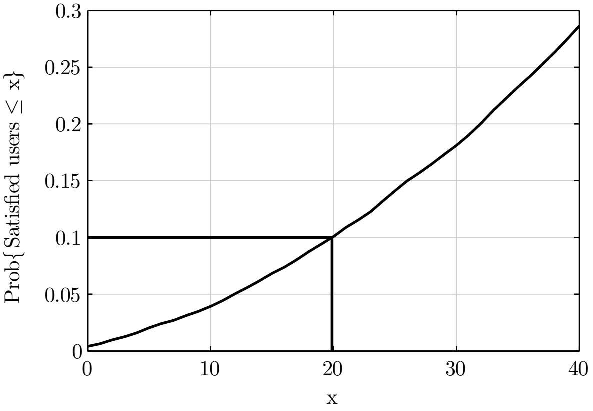

Figure 5.2: CDF of the users that can be satisfied at each iteration of a specific simulation.

two parts. In the first part, the ERC-based JCAC algorithm proposed in this chapter is applied to know how many users can be admitted. This procedure starts with the first user in the list. If this user can be admitted, then the second user is included in the JCAC decision process. If the two first users can be admitted, then the third user is added too. This procedure continues until no more users can be included. Note that this can happen even in the first step, i.e. if the first user in the list cannot be admitted, then 0 users would be in the system for this simulation following the JCAC algorithm.

During the second part, all the 150 users were moved and were generating traffic for one simulation hour. Note that even though the JCAC algorithm stated that less than these 150 users could be admitted, all of them were used in the simulation. This made possible to compute a perfect JCAC a posteriori, that is, after the simulation is possible to compute the maximum number of users that could have been admitted to obtain a non-satisfaction probability below the desired limit of ε. Let define U_{ERC} and U_{perfect} as the number of users admitted by the ERC-based JCAC algorithm and by the perfect JCAC respectively.

5.4.1 Perfect JCAC

The quantity U_{perfect} is not easy to find. The mechanism followed in this Thesis requires some calculus at each simulation iteration. At each iteration, users are demanding a specific bit rate and are perceiving certain specific channel quality. These data can be used to obtain the exact number of users that could be served at each iteration. The procedure followed to obtain this quantity is similar to that explained for the ERC-based JCAC, i.e. starting with the first user in the list, the rest of users are added until the non-satisfaction probability is exceeded. After computing this quantity during the whole simulation, it is possible to draw a CDF. U_{perfect} is the lowest integer value with a CDF above ε.

Figure 5.2 shows an example of this procedure. Imagine that the maximum non-satisfaction probability is set to $\varepsilon = 0.1$ and the CDF for 19 users is 0.0939. This means that in the 9.39% of the iterations in that simulation the users that could be satisfied were the first 19 or less. If the system admits the first 20 users then 9.39% is exactly the probability that all of them could not be satisfied. This value is below the target hence the system can serve 20 users for $\varepsilon = 0.1$. The CDF for 20 users is 0.1007, that is above the objective, hence 21 users cannot be served for $\varepsilon = 0.1$.

The quantity of users that can be satisfied at each iteration was obtained using integer programming. Let us consider only the first I users in the list. The next step is to know if it is possible to serve all users with the resources that they require using all RATs. These users are at certain positions and perceive certain channel qualities in form of specific SNIR levels for each RAT. These levels can be used together with the Q_k functions of each RAT k to obtain the bit rate per r.u. that each user will perceive from each RAT. An example of these functions were shown in Section 3.5.1. Let R_{ik} be the bit rate per r.u. that the i-th user perceives from the k-th RAT and let $R_{\min,i}$ be the bit rate that the i-th user is demanding. Then, the quantity of resources that the i-th user needs from the k-th RAT is $\rho_{ik} = R_{\min,i}/R_{ik}$. Moreover, the variable a_{ik} will be used to identify if the i-th user will be served by the k-th RAT, if $a_{ik} = 1$, or not, if $a_{ik} = 0$. Therefore, all RATs will not exceed their maximum number of resources if:

$$\sum_{i=1}^{I} a_{ik}\rho_{ik} \leq \rho_{\max,k}, \; k = 1, \cdots, K. \tag{5.17}$$

Moreover, all users must be served by only one RAT, thus:

$$\sum_{k=1}^{K} a_{ik} = 1, \tag{5.18}$$

(5.17), (5.18) and the additional conditions $a_{ik} \leq 1$ and $a_{ik} \geq 0$ define a polytope in $\mathbb{R}^{IK}$. If that polytope is not empty and there is al least one

IK-dimensional point with integer components in it, then it is possible to find the values a_{ik}, where all of them are 0 or 1, that satisfy (5.17) and (5.18). Therefore, the problem has a solution and all users can be served with the quantity of resources that they require. This Thesis uses integer programs to find one of these solutions if exists. It is worth noting that no cost function has been defined for the integer program. This is because the objective is not to find the best solution but just to know if a solution exists.

If the heterogeneous system can serve the first I users at this simulation iteration, then the process is repeated for the first $I+1$ users, and so on until no more users could be added. At that moment, the maximum quantity of users that can be satisfied is stored for computing the CDF of Figure 5.2 and the simulation starts next iteration.

5.4.2 Results

This section compares the perfect JCAC and the ERC-based JCAC algorithms. The study was divided into two scenarios. In the first scenario, the quantity of users in the hotspot was 50% whereas in the second scenario it was 90%. The percentage of users in the hotspot may have severe consequences for the algorithm. If 50% of users are in the hotspot, the other 50% is mostly in the zone with coverage of only HSDPA, and moreover, far from the cell center. Therefore, this zone will get saturated earlier than the hotspot. As a consequence, the JCAC algorithm could be simplified to a CAC algorithm in HSDPA. With 90% of users in the hotspot, the scenario is completely different. In this case, the hotspot will get saturated earlier, and it will be necessary to perform some kind of JCAC since that zone has coverage of both HSDPA and WLAN. The JCAC algorithm should have good performance in both cases.

Figure 5.3 shows the results for the first scenario. Figure 5.3a depicts the average quantity of admitted users in each simulation for different values of ε. The differences between the perfect and the ERC-based JCAC algorithms are not negligible. The perfect JCAC admits from almost 1 user more for $\varepsilon = 10^{-3}$ up to 3.3 users more for $\varepsilon = 10^{-1}$. The first conclusion that could be drawn from this figure is that the ERC-based JCAC is admitting less users than those that could be served and, consequently, they are perceiving an extremely high quality. Nevertheless, Figure 5.3b seems to show the opposite. The non-satisfaction probability of the ERC-based JCAC is very close to the objective ε, whereas for the perfect JCAC it is surprisingly lower.

This fact can be explained using an example. Imagine that three simulations are performed with different seeds and with only one service. Imagine that the ERC-based JCAC states that up to 7 users can be admitted to serve them with a non-satisfaction probability of exactly 10%. In the first simulation, 20 users are

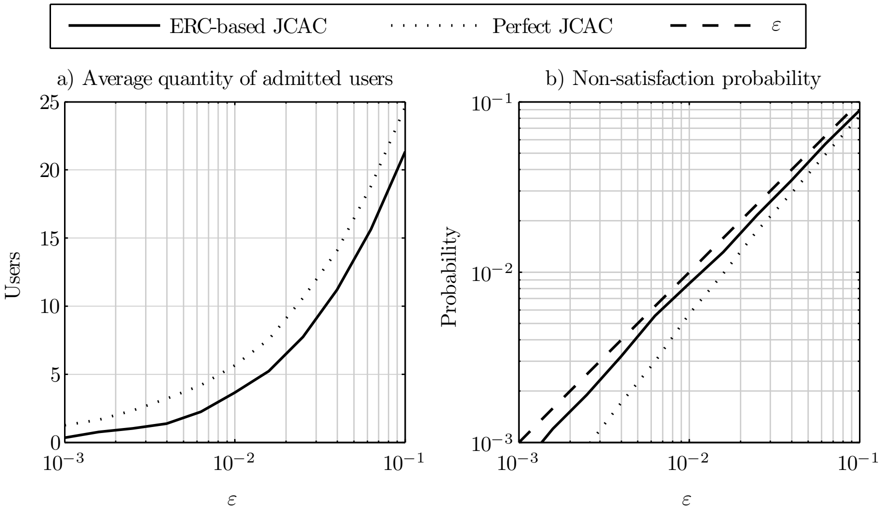

Figure 5.3: 50% of users in the hotspot.

generated and located at different positions and with different velocities. After the simulation, using the perfect JCAC, only 4 users should be admitted for that specific seed. The first 4 users are perceiving a non-satisfaction probability of exactly 10% whereas for the first 7 users it is of exactly 20%. The other two simulations are very similar and the perfect JCAC states that 10 users can be admitted in both cases. An explanation to this is that in the first simulation the first 4 users might be at the cell edge whereas in the second and third simulations users might be nearer the cell center. The first 7 users alone perceive a non-satisfaction probability of 5%. The perfect JCAC serves users with the desired non-satisfaction probability in all simulations and the average number of admitted users is 8, one more than the ERC-based JCAC. The ERC-based JCAC do not satisfy the first 7 users in the first simulation but the average non-satisfaction probability is still the objective of 10%. This example shows that is possible to admit less users and have the same average non-satisfaction probability.

The main difference between the perfect and the ERC-based JCAC is that the former has a perfect knowledge of what will happen. The ERC-based JCAC works only with the data known a priori, i.e. SNIR and traffic PDFs of previous users in the system and the service of each user. The ERC-based JCAC is hence a causal algorithm whereas the perfect JCAC is not.

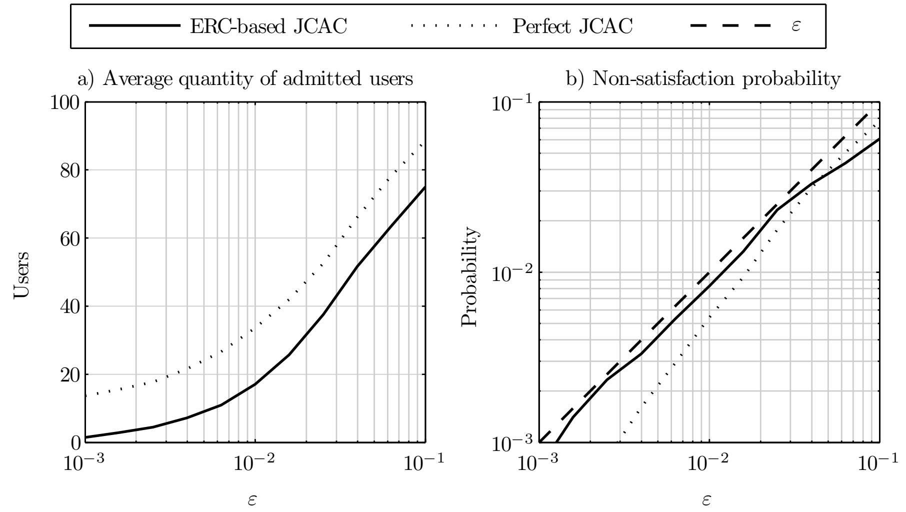

Figure 5.4: 90% of users in the hotspot.

Figure 5.3b shows that the non-satisfaction probabilities of both algorithms are below the objective. This can be understood remembering how the quantity of admitted users is selected. In both cases, this quantity is the maximum number of users whose non-satisfaction probability is *lower* or equal the objective. This fact makes that generally there exists a resource surplus that reduces the non-satisfaction probability to a certain extent. This reduction is more significant for the perfect JCAC since the fact that the ERC-based JCAC exceeds the objective non-satisfaction probability for some seeds allows this algorithm to be closer to the objective.

Figure 5.4 shows the results for the second scenario. The average quantity of admitted users of Figure 5.4a is in general higher than that of Figure 5.3a, as could be expected. Nevertheless, the difference between the perfect and the ERC-based JCAC is notably greater, of around 14 users. The explanation of why both algorithms are different is the same stated before, although now the variability of the admitted users depending on their random location is notably higher. This increased variability is the reason for the greater difference. Before, maybe 50 users could be admitted in a very good case (all users relatively near the cell center). Nevertheless, now maybe 100 users can be admitted in a case with the same level of goodness. Although in both examples users are relatively near the cell center respect to the rest of cases, if the 90% are in the hotspot then they are still nearer. Locating users nearer the cell center implies increasing the system capacity and thus higher variability.

However, in general, the ERC-based JCAC is still closer to the objective than the perfect JCAC, as shown in Figure 5.4b. This fact shows that the algorithm is still working well. Nevertheless, for $\varepsilon > 4 \cdot 10^2$ the ERC-based JCAC performance starts getting worse. These levels of ε correspond to 50 or more admitted users. Thus, the ERC is starting to work badly due to the accumulated inaccuracies of such a number of convolutions, as concluded in Section 4.5.1. This fact could be solved using the Gaussian Approximation.

5.5 Conclusion

This chapter has presented the ERC-based JCAC algorithm proposed in this Thesis. The admission decision is very easy to perform after computing the ERC values. Nevertheless, if the different RATs have different coverage areas, the problem gets more complicated and it is necessary to enunciate it in the form of a LP. If the LP has a feasible solution, and only if so, all users can be satisfied with the heterogeneous network. Thus, the JCAC problem is reduced to the problem of knowing if a LP has a solution or not. This chapter has presented the most common method for testing the feasibility of optimization problems.

Results show that the algorithm approaches the objective QoS. Therefore, it is possible to decide on call admission without wondering which RAT users should connect to. This algorithm is the first one of a new set of algorithms that manage all the heterogeneous network as a whole.

Part III

Results and Conclusions

Chapter 6

JDRA and JCAC operation performance

This chapter is devoted to assess the performance of the HNN-based JDRA and the ERC-based JCAC algorithms working together. More than an evaluation of the JDRA algorithm, this chapter will perform an assessment, by means of simulations, of a realistic implementation of the JCAC algorithm in a realistic scenario with an scheduling algorithm, the HNN-based JDRA.

6.1 New aspects of the system simulator

The scenario used in the simulations was very similar to the scenarios described in previous chapters. The mobility model was that presented in Section 3.5.4 with 7 cells and wrap-around (see Figure 3.5). All cells are completely covered by an HSDPA base station in the cell center co-located with a WLAN access point. Users belong to any of the 4 different services described in Section 3.5.3, i.e. web with maximum delays of 30 s and 60 s and FTP with minimum bit rate of 150 kb/s and 50 kb/s.

Regarding the number of users, there was a stack of 50 users per service. At the beginning of the simulation none of these users were in the system, but at each simulation iteration (every second) all users had a probability of 10% of attempting to access to the system. This fact makes that the users arrival rate depends on the quantity of users already connected to the system. Every time a user attempts to access, a new location, velocity and direction of movement was randomly generated. Moreover, the mean duration of calls was 100 s.

The JCAC and JDRA algorithms were performed at each cell independently, where each cell includes both technologies. Users that ask for admission can be divided into new or handover users. The JCAC algorithm uses the band guard technique to prioritize handover users over new users. The band guard is defined as a percentage of resources that are reserved for handover users only. Moreover, the handover process starts when the signal quality of the neighbor cell exceeds the current signal quality plus certain hysteresis level.

The realistic implementation of the JCAC was approached with two new contributions to the system simulator:

- Realistic histogram computation: During Chapters 4 and 5 the simulations were divided into two phases. In the first phase a lot of users were generated and moved for certain time period to obtain enough measures to compute the histograms. After that, these histograms were fixed and used in the second phase to test the ERC performance. Nevertheless, the reality is that all these measures will not be available a priori, hence histograms have to be computed in real time and continuously updated. This chapter proposes a procedure to perform this real time computation. The Histogram Updating Rate (HUR) is used to configure the updating process, so that histograms are updated following:

  $$H_X(m,t+\Delta t) = \mathrm{HUR}\, H_X^*(m,t) + (1-\mathrm{HUR})\, H_X(m,t), \tag{6.1}$$

  where $H_X(m,t)$ is the m-th sample of the histogram of the RV X at time t, $H_X^*(m,t)$ is this sample computed only with the measurements from t to $t+\Delta t$ and Δt is the updating period, equal to the simulation iteration period in this chapter. In the first iteration, $t=0$, there is no measure and hence no histogram. Since the system is empty, all users attempting to access could be admitted without putting the system in a dangerous saturation state. After the first iteration, the updating process could operate normally. Nevertheless, the effect of the HUR could make that the histograms computed in the first iteration with few measures update very slowly. In order to obtain good histograms as fast as possible the quantity of measures are also stored. Then, in the first iterations histograms are updated following:

  $$H_X(m,t+\Delta t) = \frac{H_X^*(m,t)M_X^*(t) + H_X(m,t)M_X(t)}{M_X^*(t)+M_X(t)}, \tag{6.2}$$

  $$M_X(t+\Delta t) = M_X^*(t) + M_X(t), \tag{6.3}$$

  where $M_X^*(t)$ is the number of measures used to compute the histogram $H_X^*(m,t)$ and $M_X(0)=0$. When $M_X^*(t)/(M_X^*(t)+M_X(t)) < \mathrm{HUR}$ the

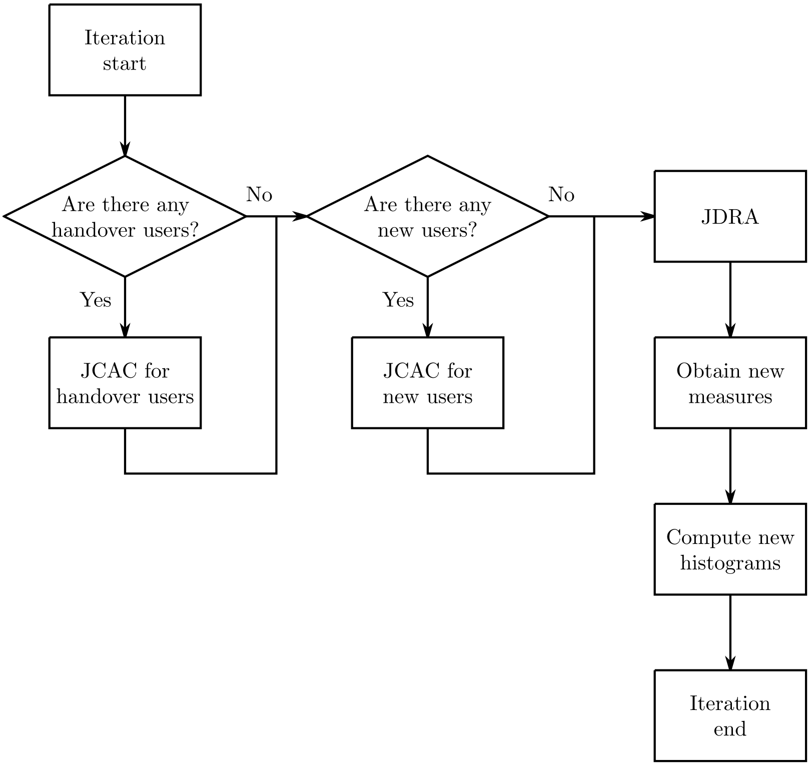

Figure 6.1: Diagram block of each resource allocation period or simulation iteration.

updating process starts operating normally using (6.1) instead of (6.2) and (6.3).

- Realistic services: In previous chapters the ERC was tested with services that require instantaneously a certain bit rate. This bit rate can change over time but if the system does not allocate it, the user is automatically not satisfied. These services are different from those defined in Chapter 3. In that chapter, users do not demand an instantaneous bit rate but some maximum delay or *average* bit rate. The $R_{\text{min},i}$ computed in Section 3.1 for each user i can be understood as an instantaneous required bit rate. Nevertheless, although users are not served with this bit rate in some moment, they could be still satisfied using higher bit rates subsequently. In fact, if a user perceives less than $R_{\text{min},i}$ at some moment,

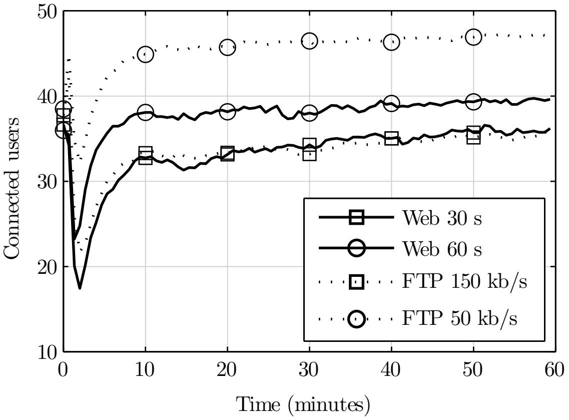

Figure 6.2: Quantity of users connected to the system during the simulations.

> $R_{\min,i}$ will increase in the next iteration hence denoting the fact that the user might not be satisfied. This effect reflects another noteworthy difference between these types of services: the required bit rate of services from Chapter 3 depends on previous allocations whereas for services from Chapters 4 and 5 it does not. The independence from previous allocations is necessary in order to obtain good histograms during the first phase of simulations. If this independence does not exist then histograms must be continuously updated in order to use good histograms that reflect the current users demands.

Figure 6.1 shows a summary of the process executed each iteration. This process should be also followed every resource allocation period in a real implementation.

6.2 Simulation results

The results shown in this section were obtained for 10 different seeds, with a target QoS non-satisfaction probability of $\varepsilon = 0.1$, with histograms of 128 intervals and for one simulated hour.

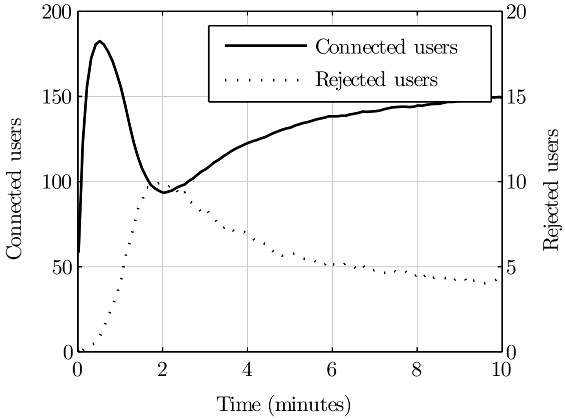

Figure 6.3: Detail of the quantity of users connected to the system at the beginning of the simulations.

6.2.1 System dynamics

Figure 6.2 depicts the evolution of the quantity of users connected to the system for each service. This figure was obtained with the 90% of users in the hotspot, an hysteresis level of 3 dB for handovers, a HUR of 0.001 and a band guard of 50%. The main conclusion drawn is that the quantity of users has a transient state at the beginning of the simulation and after that it reaches a stable state with minimum variations due to the dynamism of the scenario. The transient state is shown with more detail in Figure 6.3. During the first iterations the system is almost empty and users are perceiving very good quality. This fact is reflected in low $R_{\min,i}$ and hence histograms of required resources $f_{\rho,i}$ with very low demands. This makes that, in most cases, the JCAC does not start rejecting users after the first 10 iterations. The number of connected users continues increasing until the bad quality they perceive starts being reflected in the histograms. Nevertheless, since the bad quality must counteract the good quality stored during the first iterations, this change in the behavior of the JCAC comes too late and users start perceiving extremely low quality since the system is saturated. This provokes a fast reaction. The system rejects most new (and even handover) users reducing the quantity of users in the system. After this warm up, the system tends to stabilize and enters in its normal operation. This normal operation is reached after the first 10 minutes.

The value of the HUR affects considerably this transient state. If the HUR is increased then the peak is reached faster and the system also counteracts earlier.

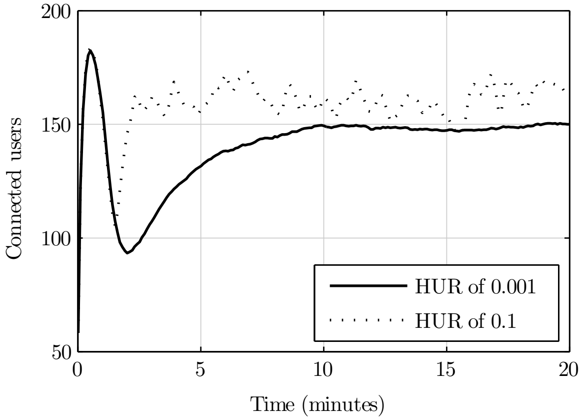

Figure 6.4: Quantity of users connected to the system for different HURs.

Nevertheless, a too high HUR makes histograms not be averaged over enough time and causes oscillations in the quantity of users. Figure 6.4 compares a HUR of 0.001 with a HUR of 0.1. The oscillations with a HUR of 0.1 are considerable. Moreover, it seems that the peak of a HUR of 0.001 and the first peak of a HUR of 0.1 are practically identical. This is because, during the first minute, there are so few measures that both approaches are updating histograms using (6.2) and (6.3), hence both have the same behavior. However, the HUR of 0.1 starts earlier to increase after the first peak, showing that this HUR reacts faster but with oscillations.

6.2.2 Performance vs. hysteresis level

This section studies the effect of the hysteresis level for handovers. The QoS perceived by users is depicted in Figure 6.5. The non-satisfaction probability of the users requirements (maximum delay or average bit rate) increases with the hysteresis level. The system is not capable of maintaining a constant probability. This is due to the JDRA and not to the JCAC. With higher hysteresis levels, users in the cell edge reach worse SNIR levels. This fact increases the amount of resources that users need to fulfil their QoS requirements. Consequently, the JCAC admits less users (see Figure 6.6). Therefore, it should be possible to maintain the same non-satisfaction probability. The problem is that the JDRA not only aims at satisfying users but also at minimizing transmit power and maximizing total throughput. Thus, according to the energy (or

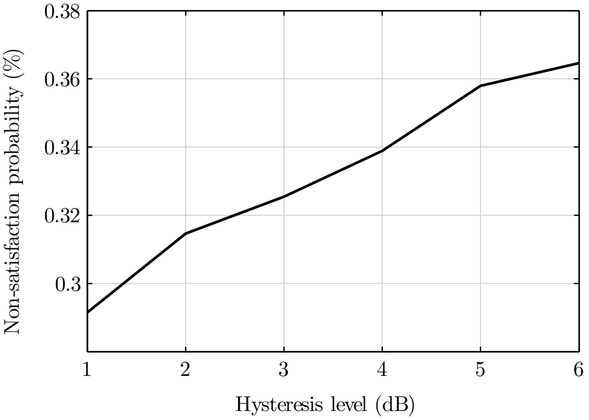

Figure 6.5: Non-satisfaction probability of the users QoS requirements.

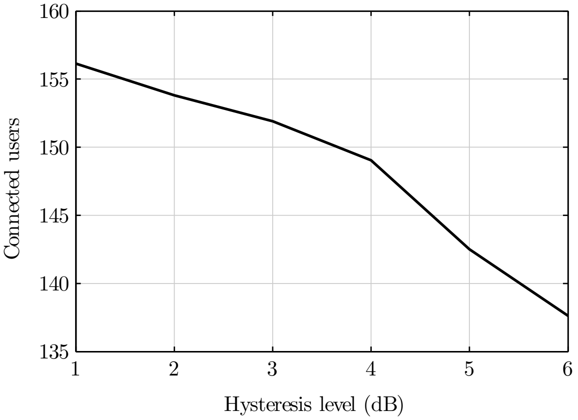

Figure 6.6: Average quantity of users connected to the system.

cost) function defined in Chapter 3, any resource quantity spent in the cell edge does not reduce the cost as much as the same quantity spent in the cell center. If the energy function was defined focusing only on users satisfaction and trying to serve all users with the same quality, users in the cell edge would monopolize the system resources. Obviously, this is not desirable and hence

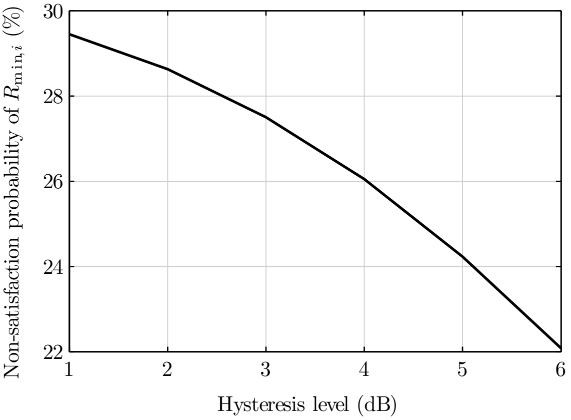

Figure 6.7: Probability of serving users with a bit rate lower than their $R_{\mathrm{min},i}$.

a good deployment is necessary to prevent extremely bad qualities in the cell edge.

Another important aspect of Figure 6.5 is that the non-satisfaction probability is significantly lower than the objective of 10%. The cause of it is that the JCAC does not take this probability into account but the required $R_{\mathrm{min},i}$ of each user. Figure 6.7 depicts the non-satisfaction probability of $R_{\mathrm{min},i}$, i.e. the probability of serving users with a bit rate lower than $R_{\mathrm{min},i}$. This probability is between 2 and 3 times higher than the objective of 10% and between 60 and 100 times higher than the non-satisfaction probability. This difference remarks two distinct philosophies between the JCAC and the JDRA. The users that are being admitted by the JCAC are those that could be served with their $R_{\mathrm{min},i}$ in the 90% of times. This is achieved by giving more priority to users in the cell center. Nevertheless, the JDRA tries to not marginalize cell edge users to a certain extent. This is the cause of the high values of Figure 6.7. Nevertheless, when the hysteresis level increases, cell edge users start to demand too much resources. The JDRA realizes that it is not optimum to try to satisfy them (or even that it is not possible) and starts focusing more on users in the cell center. Hence, the probability decreases since its behavior is more similar to the philosophy of the JCAC.

Another question may rise from Figure 6.7. If users are perceiving so bad quality of their $R_{\mathrm{min},i}$ satisfaction, this must be reflected in the histograms. Then the JCAC should react and admit less users to reduce the non-satisfaction probability. Nevertheless, this is not happening. The JCAC is not reacting,

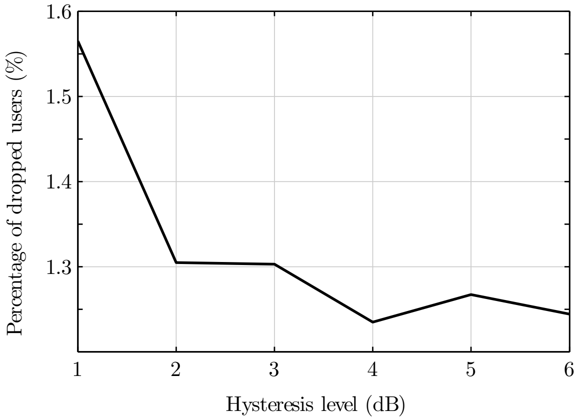

Figure 6.8: Percentage of dropped users.

at least, as much as in Chapter 5. There are three main differences between the simulations performed in Chapter 5 and this chapter. Thus, one or several of them cause the high non-satisfaction probability of Figure 6.7. These differences are:

- Histograms are computed in real time and, in general, with less measures than in Chapter 5. This can be partially solved with lower HURs. Next section studies the effect of the HUR.
- $R_{\min,i}$ depends on previous resource allocations. The ERC value presented in Chapter 4 assumed that the bit rate required by users was independent of previous demands and channel SNIR. Nevertheless, this is not true in this chapter.
- There is a scheduler, the HNN-based JDRA algorithm, who is making the final decision of how allocating resources to users. This scheduler and the JCAC algorithm have different philosophies.

These three points should be analyzed before trying to give a solution to the high non-satisfaction probability. Nevertheless, this chapter is only devoted to show that both algorithms proposed in this Thesis can work together and only the first point will be briefly analyzed in the next section. The rest of this study is proposed as future work.

Finally, handovers not only increase control traffic but also with every new handover there is a new chance of dropping users. Figure 6.8 shows the per-

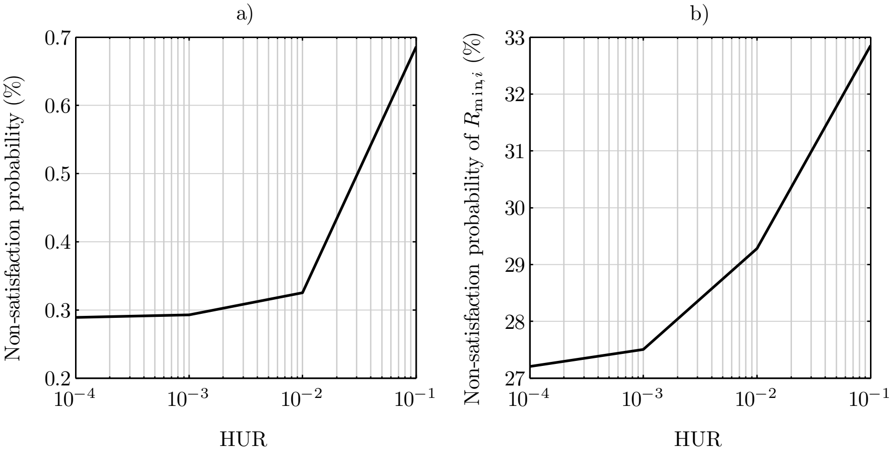

Figure 6.9: Non-satisfaction probability of the users QoS requirements a) and their $R_{\min,i}$ b).

centage of users that are dropped from the system. The percentage decreases for higher hysteresis levels although from 2 dB there is no significant reduction.

6.2.3 Performance vs. histogram updating rate

Section 6.2.1 has given a first insight on the effect of the HUR value. This section will give more details regarding non-satisfaction probability.

Figure 6.9 shows the non-satisfaction probability of the users QoS requirements and the probability of providing a bit rate lower than $R_{\min,i}$. In both cases, the probability increases with the HUR. Fast histogram updates that provoke oscillations of the number of connected users, also reduce the quality perceived by users. Moreover, the periodic reduction of connected users is mainly achieved by dropping users and not by normal call terminations. Figure 6.10 stresses this behavior showing that for HUR greater than 0.001 the quantity of dropped users increase significantly.

Recalling the three points stated in previous section, the non-satisfaction probability of $R_{\min,i}$ decreases very slowly for HURs bellow 10^{-3}. The probability at those points is still too high. This fact means that, although the HUR affects the non-satisfaction probability, it must be any other factor which provoke this high non-satisfaction.

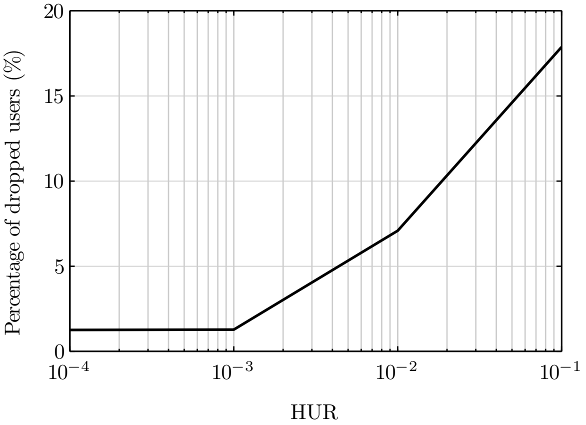

Figure 6.10: Percentage of dropped users.

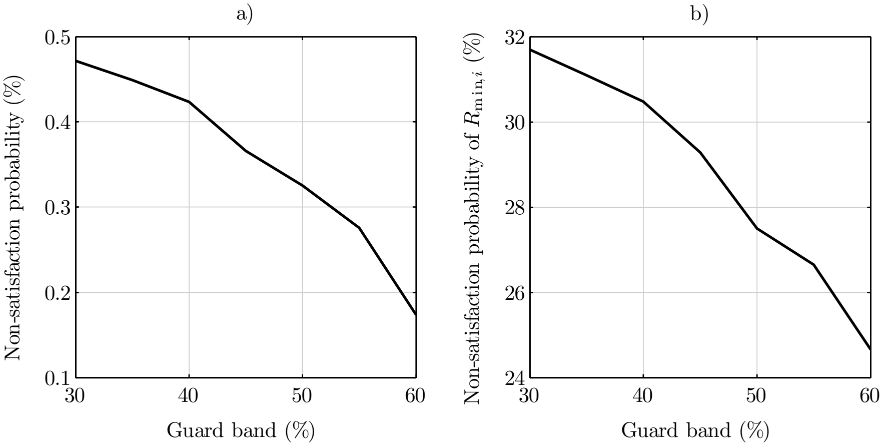

Figure 6.11: Non-satisfaction probability of the users QoS requirements a) and their $R_{\min,i}$ b).

6.2.4 Performance vs. guard band

This section presents the effect of the guard band value in the system performance. Figure 6.11 depicts the non-satisfaction probabilities. Greater guard bands load the system with less traffic what frees more resources. These free

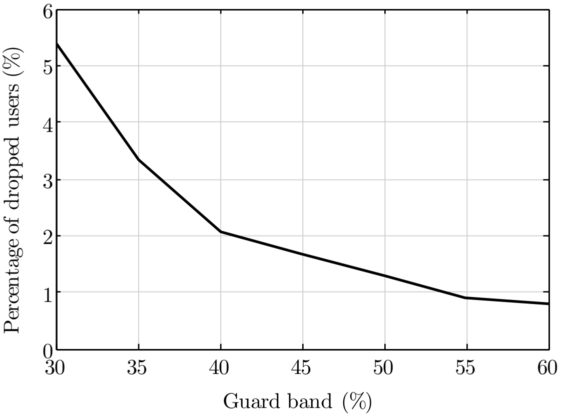

Figure 6.12: Percentage of dropped users.

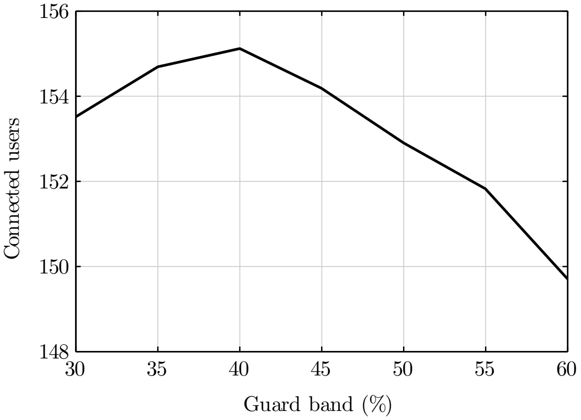

Figure 6.13: Average quantity of users connected to the system.

resources are used to serve users with better quality. This fact is reflected in the reduction of the non-satisfaction probabilities.

As expected, Figure 6.12 shows that greater guard bands reduce significantly the quantity of dropped users. Obviously, this reduction is caused by those freed resources. A priori and in order to free them, it could seem that less users must be connected to the system. Figure 6.13 shows that this is not com-

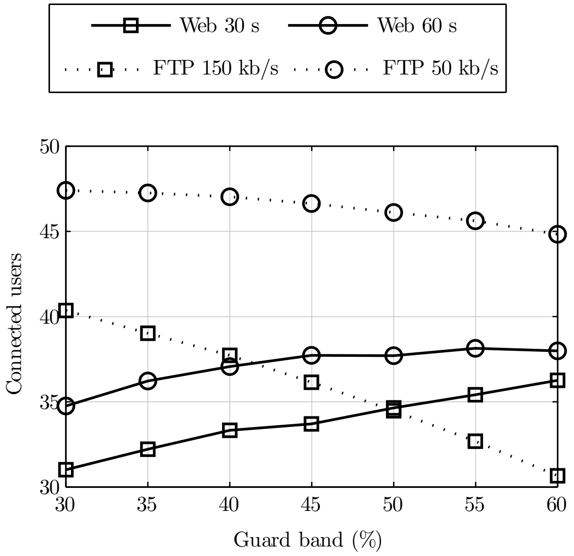

Figure 6.14: Average quantity of users per service connected to the system.

pletely true. There is a maximum for a guard band of 40%. The explanation to this can be found in Figure 6.14. The FTP service with an average bit rate of 150 kb/s is the one introducing more load to the system. The number of users of this service is highly reduced with the guard band value. Nevertheless and despite the increment of the guard band, the resources freed by these users can be consumed by web users that have lower requirements. At the end, the total load is reduced (this is reflected in the reduction of the quantity of dropped users) although at some points the quantity of users in the system increases.

Finally, it is remarkable the high guard band values shown in these figures. A guard band of 50% is necessary in order to drop around 1% of the users. Reserving the 50% of the resources for handover users could seem excessive, but it is worth noting that actually less resources are being reserved. The way the JCAC algorithm works produces this effect. Although the JCAC algorithm admits only those users that can be served with certain percentage of the available resources, all of them will be served with the 100%. This means that they will perceive better quality than that expected during the admission decision. This fact reduces the requirements of users, what is reflected in the histograms of their required bit rates. Consequently, these histograms show that users require lower bit rates than those they would require in a full loaded system. Thus, if certain quantity of users could be admitted using the 50% of the resources, they *could not* be satisfied if they are actually served with the 50% of the resources. For that reason, although the guard band is the 50%, the admitted users are consuming more resources.

6.3 Conclusion

In this chapter both algorithms proposed in this Thesis have been implemented in the same simulator. Results show that they can work together and that they provide QoS to users, controllable to a certain extent. Nevertheless, results also exhibit that further work is needed in order to homogenize them. The objectives of both algorithms are not exactly the same and this makes that the requirements are not completely satisfied.

Despite these drawbacks, this chapter has shown that it is possible to control the QoS provided to users in an heterogeneous network and multi-service scenario. Hence, these algorithms give a new viewpoint for developing new mechanisms to jointly manage resources in future networks.

Chapter 7

Conclusions and further work

Currently there are many different technologies that coexist in the same area. This situation is expected to be accentuated in the future. This Thesis has presented two novel mechanisms to jointly manage all technologies with the main objective of ensuring certain QoS to users, namely a JDRA algorithm and a JCAC algorithm.

For the JDRA algorithm, this Thesis has performed a deep analysis of HNNs as the best candidates to find the best resources distribution. This analysis ended with a new technique that joins projections and variable updating steps and is capable of significantly reducing the number of iterations. Using HNNs, the JDRA problem has been solved using the energy function proposed in Section 3.3.1. This technique is the first algorithm that distributes resources among users and users among technologies at the same time.

Additionally, for the JCAC algorithm, this Thesis has generalized the equivalent bandwidth concept to wireless heterogeneous networks. Results have showed that it is possible to admit or reject users in all the technologies and independently of the technology each user is going to get connected to. This novel idea is different from the rest of algorithms found in the literature where users must be admitted to a specific technology. The main drawback of this extensively used approach is that admitting a user in a specific technology does not mean that this user will be able to receive the negotiated quality after a vertical handover to another technology. Nevertheless, the JCAC algorithm proposed in this Thesis only admits users that the entire heterogeneous system is able to satisfy under any circumstances.

7.1 Conclusions

The following points summarize the main conclusions of this Thesis:

- It is possible to allocate resources to users and distribute users among RATs at the same time. The JDRA algorithm increases notably the QoS provided to users making the most of the available technologies. With this approach, users are not stuck in a technology until they run out of coverage. The JDRA algorithm has a global view of all the technologies and users and may force a vertical handover if it considers it interesting. This property gives this approach the main advantage with respect to the rest of techniques. This complex problem can be formulated as a multi-objective function to be minimized.

- HNNs have been widely used to find suboptimum solutions of multi-objective optimization problems in very short times. Nevertheless, the analog circuit of traditional HNNs are not easy of implementing. This fact has made that HNNs were usually emulated in computers although these emulations lose the fast convergence of these networks. This Thesis has shown that it is possible to reduce the number of iterations required in a computer to find a suboptimum solution. The proposed F-HNNs make a joint usage of projections and optimum updates and they can reduce the convergence by a factor of 20 as shown in Section 3.6. This reduction opens up a new horizon in the real-time implementation of the JDRA algorithm and, at the same time, of other applications of HNNs that require fast responses.

- The main reason of the fast response of the original neuron model of Hopfield is the parallel interworking of neurons. F-HNNs were designed in this Thesis to be implemented in parallel if possible. Section 3.7.4 studied a possible implementation in a GPU with 128 cores. Results showed that this parallel implementation can reduce the response time of the same F-HNN implemented in a serial mode in a CPU. The main drawback of GPUs is that they require an extra time to copy all data from the CPU memory to the one of the GPU. This extra time can make the GPU slower than the CPU if the neural network is not too big.

- The mobility models proposed by other authors do not satisfy the requirements of this Thesis. The JCAC algorithm needs that all users move around the entire simulation area in order to follow the same pattern and be characterized by the same SNIR histogram. If not, several SNIR histograms should be obtained for each movement pattern and users should be classified to assign one histogram to each one. This procedure of classification will be proposed in the next section as a future improvement. However, this Thesis has assumed that all users follow the same movement pattern and, hence, they can be characterize with only one SNIR histogram. For this reason, this Thesis had no other choice but to propose its own mobility model. More than a model by itself, the proposal is an improvement of other models that generate a uniform node density in the simulation area. Then, the model proposed here, guarantees certain objective non-uniform node density and mean node speed by means of allowing nodes to suddenly bounce off imaginary borders. The model is capable of generating any node density and mean speed, as the case of hotspots or streets and highways with different speeds.

- The EBW concept has been successfully applied to ATM systems to decide on call admission. This Thesis realized that this concept can be helpful to *translate* or turn all the different types of resources of different technologies into a common measurement, i.e. bit rate. To this aim, this Thesis has generalized EBW with the concept of ERC. The ERC can be applied to wireless and mobile systems whereas EBW cannot. This new concept computes the amount of resources that a set of users require to satisfy their QoS with certain predefined probability. Similarly to EBW within ATM systems, the call admission in wireless system is very easy after computing the ERC. Only if the system has more or the same resources than those users require, new calls can be admitted. Section 4.6 showed the good behavior of these CAC mechanisms.

- This Thesis has used the ERC concept to design a JCAC algorithm in heterogeneous systems. The algorithm takes all technologies into account at the same time and is capable of knowing if the heterogeneous system can satisfy the QoS of users with the required probability. The algorithm admits and rejects users without wondering the technology they should get connected to. The function of distributing users among technologies is already provided by the JDRA algorithm. Therefore, there is no sense in duplicating this functionality since both algorithms could reach contradictory conclusions. This approach is completely new. All previous works have always first selected the RAT of new users admitting or rejecting them in the same procedure. This Thesis has proved that it is possible to decide on call admission in the whole heterogeneous system regardless the technologies users will get connected to in the future.

- The JDRA and JCAC algorithms of this Thesis have been design following sightly different philosophies. This fact is reflected on the results of

Chapter 6. The JDRA algorithm tries to satisfy the required bit rate, $R_{\min,i}$, of all users independently of their location and SNIR level. This fact makes that the bit rates perceived by users in function of the distance from the cell center are allocated more homogeneously with the JDRA algorithm. This was shown in Figure 3.19. On the other hand, the JCAC algorithm states if it is possible to satisfy all users with certain probability. In this case, it is much better to start satisfying users with good channel quality that consume less resources. Then, more users could be satisfied although those near the cell edge will never have resources. This different policy makes that the JCAC objective is not totally satisfied.

7.2 Further work

This chapter concludes this Thesis but, obviously, the work developed here is not finished. This Thesis has opened a new branch in the management of resources in heterogeneous networks that will continue growing up in the near future. Among the most relevant aspects that should be studied, this Thesis highlights the following points:

- The third point of the conclusions stated that the main drawback of GPUs is the extra time required for data transfer. Therefore, it would be interesting to study the GPUs in a new hardware (not graphics cards) that does not require the copy phase. The problem is that this hardware would not be as cheap as graphics cards because of economy of scale. Another option would be the use of multicore CPUs. A new research line is opened in this sense, although is out of the scope of this Thesis.

- All simulations carried out in those sections that used the ERC; i.e. those of Chapter 4, Chapter 5 with the JCAC algorithm and Chapter 6; assumed that all users follow the same movement pattern. Nevertheless, it is possible to refine the ERC value approximating to the perfect JCAC of Section 5.4.1 with a better knowledge of users. For instance, certain area may have vehicular and pedestrian users that do not have the same SNIR histogram. Moreover, the chance that one of this users changes his movement pattern (to pedestrian to vehicular or vice versa) may be negligible. In this case, the ERC value will be more accurate if one SNIR histogram is computed for each movement pattern. The definition of the ERC allows this approach. The bad point is that the type of movement pattern is not one of the data transferred and negotiated during the connection phase. Therefore, the system should guess with a classification process. This procedure should be studied in the future to check if it is worthy, since the classifier errors may cause a non negligible loss in performance.

- Chapter 6 showed that the non-satisfaction probability was between 2 and 3 times greater than the JCAC objective. Section 6.2.2 stated 3 possible causes, i.e. the HUR value, the fact that the required bit rates depend on previous allocations and the fact that the JDRA algorithm does not follow exactly the same philosophy. Section 6.2.3 showed that the HUR value affects the non-satisfaction probability but also that this is not the unique cause of such high probability. Therefore, it is interesting to study the other 2 possible causes to know what is changing the JCAC performance.

- If the different philosophy of the JDRA and JCAC is changing the desired behavior of these algorithms (especially the JCAC), they must be coordinated in some way. A possible solution is to define an ERC that assumes that all users will perceive the same bit rate. The JDRA philosophy is between the one of the ERC of this Thesis and the one of this new ERC. This philosophy could be characterized by a parameter a depending on how homogeneously bit rates are allocated to users. This parameter can be understood as a measure of this homogeneity. Thus, $a = 0$ if bit rates are allocated following the philosophy of the ERC defined in Chapter 4 and $a = 1$ if they are completely homogeneous like those of the new ERC stated previously. Any value between 0 and 1 states the degree of homogeneity of the algorithm. Then, the ERC could be computed as ERC=(1-a)(ERC of Chapter 4)+a(new ERC). This approach assumes the linearity of the ERC value with respect to the homogeneity parameter a, what may not be true. Nevertheless, this approach may solve the problems of different philosophies.

Appendices

A.1 f_{ρ_i} approximation

$$\widehat{f}_{\rho_i}\left(\upsilon, P\left(I_{\rho_i}\right)\right) = \int\limits_{-\infty}^{\infty} \widehat{f}_{R_{\min,i}}\left(r\upsilon, P\left(I_{R_{\min,i}}\right)\right) \widehat{f}_{R_i}\left(r, P\left(I_{R_i}\right)\right) |r|\, dr =$$

$$= \int\limits_{-\infty}^{\infty} \left[\sum_{n=0}^{N_{R_{\min,i}}-1} \delta\left(r\upsilon - p^n_{R_{\min,i}}\right) H_{R_{\min,i}}\left(n\right)\right] \left[\sum_{m=0}^{N_{R_i}-1} \delta\left(r - p^m_{R_i}\right) H_{R_i}\left(m\right)\right] |r|\, dr =$$

$$= \int\limits_{-\infty}^{\infty} \left[\sum_{n=0}^{N_{R_{\min,i}}-1} \delta\left(\upsilon - \frac{p^n_{R_{\min,i}}}{r}\right) H_{R_{\min,i}}\left(n\right)\right] \left[\sum_{m=0}^{N_{R_i}-1} \delta\left(r - p^m_{R_i}\right) H_{R_i}\left(m\right)\right] dr =$$

$$= \int\limits_{-\infty}^{\infty} \sum_{n=0}^{N_{R_{\min,i}}-1} \sum_{m=0}^{N_{R_i}-1} \delta\left(\upsilon - \frac{p^n_{R_{\min,i}}}{r}\right) \delta\left(r - p^m_{R_i}\right) H_{R_{\min,i}}\left(n\right) H_{R_i}\left(m\right) dr =$$

$$= \sum_{n=0}^{N_{R_{\min,i}}-1} \sum_{m=0}^{N_{R_i}-1} \delta\left(\upsilon - \frac{p^n_{R_{\min,i}}}{p^m_{R_i}}\right) H_{R_{\min,i}}\left(n\right) H_{R_i}\left(m\right) =$$

$$= \sum_{k=0}^{N_{\rho_i}-1} \delta\left(\upsilon - p^k_{\rho_i}\right) H_{\rho_i}\left(k\right).$$

A.2 f_ρ approximation

$$
\begin{aligned}
&\widehat{f}_\rho\left(\upsilon, P\left(I_\rho\right)\right) = \widehat{f}_{\rho_1}\left(\upsilon, P\left(I_{\rho_1}\right)\right) * \widehat{f}_{\rho_2}\left(\upsilon, P\left(I_{\rho_2}\right)\right) = \\
&= \int\limits_{-\infty}^{\infty} \left[\sum_{n=0}^{N_{\rho_1}-1} \delta\left(x - p_{\rho_1}^n\right) H_{\rho_1}\left(n\right)\right] \left[\sum_{m=0}^{N_{\rho_2}-1} \delta\left(\upsilon - x - p_{\rho_2}^m\right) H_{\rho_2}\left(m\right)\right] dx = \\
&= \int\limits_{-\infty}^{\infty} \sum_{n=0}^{N_{\rho_1}-1} \sum_{m=0}^{N_{\rho_2}-1} \delta\left(x - p_{\rho_1}^n\right) \delta\left(\upsilon - x - p_{\rho_2}^m\right) H_{\rho_1}\left(n\right) H_{\rho_2}\left(m\right) dx = \\
&= \sum_{n=0}^{N_{\rho_1}-1} \sum_{m=0}^{N_{\rho_2}-1} \delta\left(\upsilon - p_{\rho_1}^n - p_{\rho_2}^m\right) H_{\rho_1}\left(n\right) H_{\rho_2}\left(m\right) = \\
&= \sum_{k=0}^{N_\rho-1} \delta\left(x - p_\rho^k\right) H_\rho\left(k\right).
\end{aligned}
$$

References

[1] J. Gozálvez and J. Dunlop, “Link level modelling techniques for analysing the configuration of link adaptation algorithms in mobile radio networks,” in *Proceedings of the European Wireless*, Barcelona, Spain, Spring 2004.

[2] Z. J. Haas, “A new routing protocol for reconfigurable wireless networks,” in *IEEE International Conference on Universal Personal Communications*, San Diego, United States of America, Fall 1997.

[3] E. Gustafsson and A. Jonsson, “Always best connected,” *IEEE Wireless Communications*, vol. 10, no. 1, pp. 49–55, 2003.

[4] 3GPP, “Improvement of RRM across RNS and RNS/BSS (post rel-5),” 3GPP, TR 25.891 v0.3.0, 2003.

[5] J. Pérez-Romero, O. Sallent, R. Agustí, and M. A. Díaz-Guerra, *Radio resource management strategies in UMTS.* J. Wiley & Sons, 2005.

[6] W. Zhuang, Y.-S. Gan, K.-J. Loh, and K.-C. Chua, “Policy-based QoS-management architecture in an integrated UMTS and WLAN environment,” *IEEE Communications Magazine*, vol. 41, no. 11, pp. 118–125, 2003.

[7] J. Pérez-Romero, O. Sallent, and R. Agustí, “Policy-based initial RAT selection algorithms in heterogeneous networks,” in *International Workshop on Mobile and Wireless Communications Network (MWCN)*, Marrakesh, Morocco, Fall 2005.

[8] R. Piqueras, J. Pérez-Romero, and R. A. O. Sallent, “Dynamic pricing for decentralised RAT selection in heterogeneous scenarios,” in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Helsinki, Finland, Fall 2006.

[9] J. Pérez-Romero, O. Sallent, and R. Agustí, “A novel metric for context-aware RAT selection in wireless multi-access systems,” in *IEEE International Conference on Communications (ICC)*, Glasgow, United Kingdom, Spring 2007.

[10] L. Giupponi, R. Agustí, J. Pérez-Romero, and O. Sallent, “A framework for JRRM with resource reservation and multiservice provisioning in heterogeneous networks,” *Mobile Networks and Applications*, vol. 11, no. 6, pp. 825–846, 2006.

[11] P. Houzé, S. B. Jemaa, and P. Cordier, “Common pilot channel for network selection,” in *IEEE Vehicular Technology Conference (VTC)*, Melbourne, Australia, Spring 2006.

[12] 3GPP, “Generic access network (GAN); stage 2,” 3GPP, TS 43.318 V8.3.0, 2008.

[13] ——, “Generic access network (GAN); mobile GAN interface layer 3 specification,” 3GPP, TS 44.318 V8.4.0, 2008.

[14] M. H. Ahmed, “Call admission control in wireless networks: a comprehensive survey,” *IEEE Communications Surveys & Tutorials*, vol. 1, no. 1, pp. 49–68, 2005.

[15] Z. Dziong, M. Jia, and P. Mermelstein, “Adaptive traffic admission for integrated services in CDMA wireless-access networks,” *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 9, pp. 1737–1747, 1996.

[16] Z. Liu and M. E. Zarki, “Sir-based call admission control for DS-CDMA cellular systems,” *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 4, pp. 638–644, 1994.

[17] T. Liu and J.Silvester, “Joint admission/congestion control for wireless CDMA systems supporting integrated services,” *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 6, pp. 845–857, 1998.

[18] D. Hong and S. Rapaport, “Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and non-prioritized handoff procedures,” *IEEE Transactions on Vehicular Technology*, vol. 35, no. 3, pp. 77–92, 1986.

[19] R. Ramjee, R. Ragarajan, and D. Towsley, “On optimal call admission control in cellular networks,” *Wireless Networks*, vol. 3, no. 1, pp. 29–41, 1997.

[20] D. Zhao, X. Shen, and J. Mark, "Efficient call admission control for heterogeneous services in wireless mobile ATM networks," *IEEE Communications Magazine*, vol. 38, no. 10, pp. 72–78, 2000.

[21] P. Koutsakis, M. Paterakis, and P. Psychis, "Call admission control and traffic policing mechanisms for the wireless transmission of layered videoconference traffic from MPEG-4 and H.263 video coders," in *International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Lisbon, Portugal, Fall 2002.

[22] L. Luo, R. Mukerjee, M. Dillinger, E. Mohyeldin, and E. Schulz, "Investigation of radio resource scheduling in WLANs coupled with 3g cellular network," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 108–115, 2003.

[23] F. Yu and V. Krishnamurthy, "Optimal joint session admission control in integrated WLAN and CDMA cellular networks with vertical handoff," *IEEE Transactions on Mobile Computing*, vol. 6, no. 1, pp. 126–139, 2007.

[24] C. Mihailescu, X. Lagrange, and P. Godlewski, "Performance evaluation of a dynamic resource allocation algorithm for UMTS-TDD systems," in *IEEE Vehicular Technology Conference (VTC)*, Tokyo, Japan, Spring 2000.

[25] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single-node case," *IEE/ACM Transactions on Networking*, vol. 1, no. 3, pp. 344–357, 1993.

[26] M. Andrews, K. Kumaran, K. Ramanan, A. Stolyar, P. Whiting, and R. Vijayakumar, "Providing quality of service over a shared wireless link," *IEEE Communications Magazine*, vol. 39, no. 2, pp. 150–154, 2001.

[27] Q. Liu, X. Wang, and G. B. Giannakis, "A cross-layer scheduling algorithm with qos support in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 3, pp. 839–847, 2006.

[28] S. Z. Ahmad, M. S. Akbar, and M. A. Qadir, "A cross-layer vertical handover decision model for heterogeneous wireless networks," in *IEEE International Conference on Innovations in Information Technology*, Dubai, United Arab Emirates, Fall 2007.

[29] R. Corvaja, "QoS analysis in overlay bluetooth-WiFi networks with profile-based vertical handover," *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1679–1690, 2006.

[30] L. Ma, F. Yu, V. C. M. Leung, and T. Randhawa, "A new method to support UMTS/WLAN vertical handover using SCTP," *IEEE Wireless Communications*, vol. 11, no. 4, pp. 44–51, 2004.

[31] G. Nyberg, C. Ahlund, and T. Rojmyr, "Semo: A policy-based system for handovers in heterogeneous networks," in *IEEE International Conference on Wireless and Mobile Communications (ICWMC)*, Bucharest, Romania, Spring 2006.

[32] C. W. Ahn and R. S. Ramakrishna, "QoS provisioning dynamic connection-admission control for multimedia wireless networks using hopfield neural networks," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 1, pp. 106–117, 2004.

[33] J. Hopfield and D. Tank, "'neural' computation of decisions in optimization problems," *Biological Cybernetics*, vol. 52, no. 3, pp. 141–152, 1985.

[34] K. C. Tan, H. Tang, and S. S. Ge, "On parameter settings of hopfield networks applied to traveling salesman problems," *IEEE Transactions on Circuits and Systems*, vol. 52, no. 5, pp. 994–1002, 2005.

[35] T.-N. Le and C.-K. Pham, "A new N-parallel updating method of the hopfield type neural network for N-queens problem," in *IEEE International Joint Conference on Neural Networks (IJCNN)*, Montreal, Canada, Spring 2005.

[36] E. D. Re, R. Fantacci, and L. Ronga, "A dynamic channel allocation technique based on hopfield neural networks," *IEEE Transactions on Vehicular Technology*, vol. 45, no. 1, pp. 26–32, 1996.

[37] O. Lázaro and D. Girma, "A hopfield neural-network-based dynamic channel allocation with handoff channel reservation control," *IEEE Transactions on Vehicular Technology*, vol. 49, no. 5, pp. 1578–1587, 2000.

[38] N. García, R. Agustí, and J. Pérez-Romero, "A user-centric approach for dynamic resource allocation in CDMA systems based on hopfield neural networks," in *IST Mobile & Wireless Communications Summit*, Dresden, Germany, Spring 2005.

[39] H. J. Chao and X. Guo, *Quality of Service Control in High-Speed Networks*. J. Wiley & Sons, 2002.

[40] J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proceedings of the National*

Academy of Sciences of the United States of America, vol. 81, no. 10, pp. 3088–3092, 1984.

[41] G. Joya, M. A. Atencia, and F. Sandoval, “Hopfield neural networks for optimization: study of the different dynamics,” *Neurocomputing*, vol. 43, no. 1, pp. 219–237, 2002.

[42] P. M. Talaván and J. Yáñez, “A continuous hopfield network equilibrium points algorithm,” *Computers and Operations Research*, vol. 32, no. 8, pp. 2179–2196, 2005.

[43] G. V. Wilson and G. S. Pawley, “On the stability of the travelling salesman problem algorithm of hopfield and tank,” *Biological Cybernetics*, vol. 58, no. 1, pp. 63–70, 1988.

[44] S. Aiyer, M. Niranjan, and F. Fallside, “A theoretical investigation into the performance of the hopfield model,” *IEEE Transactions on Neural Networks*, vol. 1, no. 2, pp. 204–215, 1990.

[45] P. Chu, “A neural network for solving optimization problems with linear equality constraints,” in *IEEE International Joint Conference on Neural Networks*, Baltimore, United States of America, Spring 1992.

[46] K. Smith, M. Palaniswami, and M. Krishnamoorthy, “Neural techniques for combinatorial optimization with applications,” *IEEE Transaction on Neural Networks*, vol. 9, no. 6, pp. 1301–1318, 1998.

[47] S. Choudhury and J. D. Gibson, “Payload length and rate adaptation for throughput optimization in wireless LANs,” in *IEEE Vehicular Technology Conference*, Melbourne, Australia, Spring 2006.

[48] F. Brouwer, I. de Bruin, J. C. Silva, N. Souto, F. Cercas, and A. Correia, “Usage of link-level performance indicators for HSDPA network-level simulations in E-UMTS,” in *IEEE International Symposium Spread Spectrum Techniques and Applications*, Sydney, Australia, Fall 2004.

[49] 3GPP2, “HTTP and FTP traffic model for 1xEV-DV simulations,” 3GPP2, TS TSGC5.

[50] ETSI, “Universal mobile telecommunications system (UMTS); selection procedures for the choice of radio transmission technologies of the UMTS,” ETSI, TR 101.112, 1998.

[51] M. Gudmundson, “Correlation model for shadow fading in mobile radio systems,” *Electronic Letters*, vol. 27, no. 23, p. 21452146, 1991.

[52] C. Schindelhauer, "Mobility in wireless networks," in *International Conference on Current Trends in Theory and Practice of Computer Science*, Merin, Czech Republic, Spring 2006.

[53] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications & Mobile Computing*, vol. 2, no. 5, pp. 483–502, 2002.

[54] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *ACM/IEEE International Conference on Mobile Computing and Networking*, Dallas, United States of America, Fall 1998.

[55] C. Bettstetter, "Smooth is better than sharp: a random mobility model for simulation of wireless networks," in *ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, Rome, Italy, Spring 2001.

[56] B. Liang and Z. J. Haas, "Predictive distance-based mobility management for pcs networks," in *IEEE Annual Joint Conference Computer and Communications Societies (INFOCOM)*, New York, United States of America, Spring 1999.

[57] J. Monserrat, L. M. González, R. Fraile, and N. Cardona, "Morans layer 2: Traffic information synthetic scenario," COST 273, TD (04)071, 2004.

[58] V. Borrel, M. D. de Amorim, and S. Fdida, "A preferential attachment gathering mobility model," *IEEE Communications Letters*, pp. 900–902, 2005.

[59] S. Lim, C. Yu, and C. R. Das, "Clustered mobility model for scale-free wireless networks," in *IEEE Conference on Local Computer Networks (LCN)*, Tampa, United States of America, Fall 2006.

[60] E. Hyytiä, P. Lassila, and J. Virtamo, "A markovian waypoint mobility model with application to hotspot modeling," in *IEEE International Conference on Communications (ICC)*, Istambul, Turkey, Spring 2006.

[61] C. Bettstetter and C. Wagner, "The spatial node distribution of the random waypoint mobility model," in *German Workshop Mobile Ad Hoc Networks (WMAN)*, Ulm, Germany, Spring 2002.

[62] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol. 28, no. 3, pp. 497–520, 1960.

[63] R. Guérin, H. Ahmadi, and M. Naghshineh, “Equivalent capacity and its application to bandwidth allocation in high-speed networks,” *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 7, pp. 968–980, 1991.

[64] R. Gibbens and P. Hunt, “Effective bandwidths for the multi-type UAS channel,” *Queuing Systems*, vol. 9, no. 1, pp. 17–28, 1991.

[65] K. Kelly, “Effective bandwidths at multi-class queues,” *Queuing Systems*, vol. 9, no. 1, pp. 5–16, 1991.

[66] J. Y. Hui, “Resource allocation for broadband networks,” *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, pp. 1598–1608, 1988.

[67] H. Saito and K. Shiomoto, “Dynamic call admission control in ATM networks,” *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 7, pp. 982–989, 1991.

[68] H. Zhang and E. W. Knightly, “Providing end-to-end statistical performance guarantee with bounding interval dependent stochastic models,” in *ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*, Nashville, USA, Spring 1994.

[69] S. Abe and T. Soumiya, “A traffic control method for service quality assurance in an ATM network,” *IEEE Journal on Selected Areas in Communications*, vol. 12, no. 2, pp. 322–331, 1994.

[70] C. Rasmussen, J. Sorensen, K. S. Kvols, and S. B. Jacobsen, “Source independent call acceptance procedures in atm networks,” *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 3, pp. 351–358, 1991.

[71] G. L. Choudhury, D. M. Lucantoni, and W. Whitt, “Squeezing the most out of ATM,” *IEEE Transactions on Communications*, vol. 44, no. 2, pp. 203–217, 1996.

[72] G. de Veciana, G. Kesidis, and J. Walrand, “Resource management in wide-area ATM networks using effective bandwidths,” *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 6, pp. 1081–1090, 1995.

[73] Z. Dziong, M. Juda, and L. G. Mason, “A framework for bandwidth management in ATM networks-aggregate equivalent bandwidth estimation approach,” *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 134–147, 1995.

[74] J. S. Evans and D. Everitt, “Effective bandwidth-based admission control for multiservice CDMA cellular networks,” *IEEE Transactions on Vehicular Technology*, vol. 48, no. 1, pp. 36–46, 1999.

[75] V. K. Rohatgi, *An introduction to probability theory and mathematical statistics.* J. Wiley & Sons, 1976.

[76] W. Rudin, *Principles of mathematical analysis.* McGraw-Hill, 1976.

[77] P. T. Brady, “A technique for investigating on-off patterns of speech,” *The Bell System Tech. Journal*, pp. 1–22, 1965.

[78] ——, “Statistical analysis of on-off patterns in 16 conversations,” *The Bell System Tech. Journal*, pp. 77–91, 1968.

[79] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” *Combinatorica*, vol. 4, no. 4, pp. 373–395, 1984.