Numerical Optimisation Methods for Power Consumption in Multi-Hop Mobile Phone Networks

by

Carlos Sosa Paz

A thesis submitted to
The University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Mathematics
The University of Birmingham
February 2010

Abstract

In recent years the importance of multi hop wireless networks has been growing mainly due to key factors such as: no backbone infrastructure or cost of installation are required and the network can be rapidly deployed and configured. A main problem in this kind of networks is to establish an efficient use of the power involved in the communication between network devices. In this dissertation, we present two different mathematical models which represent a multi hop wireless network: The first model considers the joint routing, scheduling and power control for TDMA/CDMA multi hop wireless network systems; minimising the used power to send messages through a multi hop wireless network. In this model we consider the scheduling of the message transmission. The scheduling is essential since it coordinates the transmission between devices in order to reduce the interference caused by the neighborhood devices and the background noise. Interference plays an important role in this kind of networks since the quality of service depends on it. We use two quality indicators: The Signal to Noise Ratio (SNR) and the Signal to Interference Noise Ratio (SINR). The second model considers the joint routing, power control for CDMA multi hop wireless network systems model in which we include two different filters: the Single User Matched Filter (SUMF) and the Minimum Mean Squared Error (MMSE), without considering the scheduling problem. The nature of the set of constraints inherent to both mathematical models is non-convex, therefore we have a non convex optimisation problem. Since the problem is non convex, the obtained solutions are only local minimisers. We present and prove two theorems which, in general terms, state that at a local minimiser, the capacity over the links is fully exploited. We present different sets of experiments and numerical solution for both mathematical models.

Acknowledgements

This thesis is the end of my hardest journey so far in my academic life. It would not be possible without the support of many people around me.

First of all, I want to thank my supervisor Dr. Jan J. Rückmann for his guidance and support. For his instruction in mathematics, en especial cuando me decías “Carlos ..., pero hay algo que no me gusta”, frase que generaba, genera y espero generará un desafío. Muchas Gracias.

At the same time, I want to thank Prof. Dr. Jörg Fliege, (Director of CORMSIS, The University of Southampton) for giving me the opportunity to start my research under his guidance and for his stimulating suggestions that helped me during my research and writing of this thesis. Here I would like to express my gratitude to Dr. Armin Dekorsy (Qualcomm) for his guidance, and to all the professors of the School of Mathematics, University of Birmingham for their support.

I want to express my sincere thanks to the Instituto Politécnico Nacional and the Consejo Nacional de Ciencia y Tecnología from México for sponsoring my study.

I want to express all my gratitude to Adibah Shuib, Waleed Abu-Elela and Abdulhadi Aminu for their time and help. Thanks to Jim Perks, Vincent Warltire, Ellie Boston, Matthew Hamilton, and Christine Turner for the proof reading of this thesis.

I would like to express my deepest gratefulness to María Mato (Vecina) , Anaximandro Gómez (Chafatista) and Fabián Peña (Panucho), true friends, for all their support.

I would like to dedicate this research work to my beloved wife Tomoko Okada, whose patience, support and love helped me to complete this thesis. Thank you, Amor.

Finally, I would like to acknowledge my parents, Javier Sosa and Bertha Paz, my brother Gabriel and sister Sandra who are always there for me.

Contents

1 Introduction **3**

1.1 Problem Description and Motivation 3

1.2 Related Work 9

1.3 Contributions of the Dissertation 11

1.4 The Structure of the Thesis 13

2 Wireless Network System **15**

2.1 Wireless Network System 16

2.2 Layered Communication Architecture 19

2.3 Physical Layer 20

2.3.1 Signal to Noise Ratio and Signal to Interference Noise Ratio . 22

2.3.2 Mathematical Models of SNR and SINR 23

2.3.3 Filters 25

2.4 Access Layer 27

2.4.1 Interference 31

2.5 Network Layer 32

2.6 Ad Hoc Wireless Network 34

2.6.1 Ad Hoc Wireless Challenges 37
2.6.2 Cross Layer 39
2.7 Summary 40

3 Non-Linear Programming 43

3.1 Non-Linear Programming 44
3.2 Graph Theory 46
3.3 Network Flow 49
3.4 A Modelling Language for Mathematical Programming AMPL 50
3.5 Solvers 53
3.6 Summary 56

4 First Mathematical Modelling Approach 57

4.1 Mathematical Model 58
4.1.1 Design Variables 61
4.1.2 Parameters 62
4.1.3 Set of Constraints 63
4.1.3.1 Upper Bounds for the Decision Variables 63
4.1.3.2 Constraints of the System 64
4.1.3.3 Non-Negativity Constraints 65
4.1.4 The Non-Linear Programming Problem 66
4.2 Theoretical Contributions 67

4.2.1 First Theorem 68
4.2.2 Second Theorem 71
4.3 Methodology for the Numerical Solutions 74
4.4 Numerical Results and Sensitivity Analysis 82
4.5 Summary 108

5 Second Mathematical Modelling Approach 110

5.1 Mathematical Model 111
5.1.1 Design Variables 113
5.1.2 Interference Function 113
5.1.3 Parameters 115
5.1.4 Set of Constraints 115
5.1.4.1 Upper bounds for our variables 115
5.1.4.2 Constraints of the system 116
5.1.4.3 Non Negativity Constraints 117
5.1.5 The Non-Linear Programming Problem 118
5.2 Methodology for the Numerical Solution 118
5.3 Numerical Results 119
5.4 Summary 129

6 Conclusions and Future Work 130

6.1 Summary and Conclusions 130
6.2 Future Outlook 133

Bibliography **141**

Bibliography 141

A AMPL Model File and Data file **151**

B Model file **156**

C Matlab file **163**

List of Figures

1.1 The radio frequency spectrum. Borrowed from [16] 5

2.1 Wireless System. Borrowed from [75]. 17

2.2 Infrastructure based. Borrowed from [75]. 18

2.3 Infrastructure-less wireless network. 19

2.4 Frequency Division. Borrowed from [35]. 28

2.5 Time Division. Borrowed from [35]. 29

2.6 Code Division. Borrowed from [35]. 30

2.7 Flooding protocol. Borrowed from [3] 33

2.8 Node's Radio Range. Borrowed from [11]. 36

3.1 Examples of coloured graphs. 48

4.1 Six nodes network graph. 77

4.2 Coloured network graph. 78

4.3 Time slot t_1 *blue* coloured. 79

4.4 Time slot *red* coloured. 80

4.5 Two time slots with different colour associated. 81

4.6 Time Coloured Network Topology. 84

4.7 Message routing. 86

4.8 Minimum transmission power VS. size of message for starting time slot t_3. 90

4.9 Transmission power VS. size of message at different starting time slots. 91

4.10 Eight node graph with two messages. 93

4.11 Message m_1 flowing through the network. 95

4.12 Message m_2 flowing through the network. 96

4.13 Minimum transmission power vs. size of message for starting time slot t_3. 99

4.14 Transmission power vs. the combination of two messages sizes at different starting time slots. 99

4.15 Five messages network topology. 101

4.16 Total power used to send five messages. 104

4.17 The flows of the five messages. 105

4.18 Five messages network topology. 107

4.19 Back-haul network with 15 nodes graph. 108

5.1 Spider network. 120

5.2 Six nodes graph with two messages. 123

5.3 Eighth node graph with two messages. 126

5.4 Message m_1 flowing through the network. 127

5.5 Message m_2 flowing through the network. 128

6.1 Message m_1 flowing through the network. 137

6.2 Message m_2 flowing through the network. 138

List of Tables

4.1 Node colour assignation. 78

4.2 Time slots colour assignation. 83

4.3 List of parameters. 85

4.5 Transmission power p_e used at the edge e. 87

4.6 Size of the piece of message m_1 sent through edge e. 87

4.7 Size of the piece of message m_1 sent through edge e (buffer edges). . . 88

4.8 Colour assignation for $v \in \widehat{V}$. 93

4.9 Time slot assignation table for two messages. 94

4.11 Transmission powers and sizes of pieces of messages m_1 and m_2. 97

4.12 Colour assignation for $v \in \widetilde{V}$ 101

4.13 Time slot assignation table for five messages. 102

4.15 Transmission power and sizes of pieces for messages $m_i, i = 1, \ldots 5$. . 103

4.16 Colour assignation for $v \in \widetilde{V}$ 106

4.17 Time slot assignation table. 107

5.1 Parameter values for the first experiment. 121

5.2 Transmission powers p_e for SUMF filter. 121

5.3 Transmission powers p_e for MMSE filter. 122

5.4 Set of parameters for the second experiment. 123

5.5 Transmission powers p_e for SUMF filter. 124

5.6 Transmission powers p_e for MMSE filter. 124

5.7 Set of parameters for the third experiment. 126

5.8 Transmission powers p_e for SUMF filter. 127

5.9 Transmission powers p_e for MMSE. 128

6.1 List of parameters. 135

6.3 Transmission powers and sizes of pieces message m_1 and m_2. 139

List of Symbols

Real numbers:	
$\mathbb{R}_+$	The set of positive real numbers.
$\mathbb{R}_*$	The set of non negative real numbers.
$\mathbb{Z}$	The set of integer numbers.
$\varrho_u \in \mathbb{R}_+$	The transmission range of the node u.
$\varsigma_{u,v} \in \mathbb{R}_+$	The distance between any two devices u and v.
$\eta \in \mathbb{R}_+$	The background noise.
$\gamma_e^* \in \mathbb{R}_+$	The minimum acceptable level of $SINR$ for the edge e.
$\tau \in \mathbb{R}_+$	The duration of the time slot t.
$B \in \mathbb{R}_+$	The bandwidth
$\|x\|$	The absolute value for a $x \in \mathbb{R}$.
$a\%b$	The modulo operation finds the remainder of division of $a \in \mathbb{Z}$ by $b \in \mathbb{Z}$.
$\lceil x \rceil$	Ceiling function. Given $x \in \mathbb{R}$ and $n \in \mathbb{Z}$, the ceiling function is defined as follows: $\lceil x \rceil = min \{n \in \mathbb{Z} \,\vert n \geq x\}$
$\lfloor x \rfloor$	Floor function. Given $x \in \mathbb{R}$ and $n \in \mathbb{Z}$, the floor function is defined as follows: $\lfloor x \rfloor = max \{n \in \mathbb{Z} \,\vert n \leq x\}$
Sets:	
$V \subset \mathbb{Z}$	The set of nodes (devices).
$E \subset \mathbb{Z}$	The set of edges (links).
$\|E\|$	The cardinality of the set E.
Vectors:	
$\\|\mathbf{x}\\|$	The euclidean norm of the vector $\mathbf{x} \in \mathbb{R}^n$.
Special:	
$G = (V, E)$	A directed graph.

List of Abbreviations

Abbreviation:	
AMPL	A Mathematical Programming Language
AWGN	Additive White Gaussian Noise
CDMA	Code Division Multiple Access
FDMA	Frequency Division Multiple Access
Ipopt	Interior Point Optimiser
ISO	International Organisation for Standardisation
MMSE	Minimum Mean Squared Filter
OSI	Layer Open System Inter-connectivity
PAN	Personal Area Network
PDA	Personal Digital Assistants
QoS	Quality of Service
SDMA	Space Division Multiple Access
SINR	Signal to Interference plus Noise Ratio
SNR	Signal to Noise Ratio
SUMF	Single User Matched Filter
TDMA	Time Division Multiple Access
WAN	Wireless Area Network
WLAN	Wireless Local Area Network
WWAN	Wireless Wide Area Network

Chapter 1

Introduction

1.1 Problem Description and Motivation

One of the main necessities of humankind is communication. For the last three decades, our methods of communication have been changing, especially, due to technological developments. Nowadays it is possible, with some special telecommunication devices, to be virtually "present" in a job conference, a lecture or a special family event. Moreover, for some people the need to be in constant communication with others has become a permanent necessity. The advances in technology give us all the opportunity to be in touch almost anytime and anywhere.

The principal elements involved in communication are the *source (sender)* and *sink (receiver)* of the information and the physical transmission *medium* that carries the information. The union of all these elements will be called communication system. A "long" distance communication system will be referred here as a telecommunication system.

The communication could be considered as personal communication or impersonal communication. Personal communication means that there is no communication device involved while for the impersonal communication it is necessary to use a communication device.

The communication between the source and the sink in a telecommunication system can be done by a special device such as mobile phone, laptop, desktop, telephone, (etc.)

In a telecommunication system the physical transmission medium can be air, cable, water, etc. The information sent through a telecommunication system must be transformed into a suitable form for the physical transmission medium. The set of telecommunications systems can be divided into two main groups: wired and wireless telecommunication systems.

The wired telecommunication system usually called wired Ethernet technology provides a better data rate performance than any other telecommunication wireless system [44]. Despite the fact that wired systems provide a much better rate performance than a wireless telecommunication system, wireless telecommunication systems is becoming more popular due to the lack of a wire plugged into the devices i.e. devices have free mobility (mobile device) in the network system.

One fundamental problem for wireless network systems is the optimal use of the radio frequency spectrum. The radio frequency spectrum is a scarce resource that must be allocated for all wireless systems such as: AM and FM radio stations broadcast, television, amateur radio, wireless networks, etc. [58]. The radio frequency spectrum is defined from frequencies starting from around 3 kHz [kilohertz] to 300 GHz [gigahertz], illustrated in Figure 1.1. In the wireless network system, the available spectrum is shared in a number of ways by various devices, which is referred to as a multiple access scheme. There are basically four main schemes: Frequency Division Multiple Access (FDMA), Space Division Multiple Access (SDMA), Code Division

Multiple Access (CDMA), Time Division Multiple Access (TDMA) or a combination of any of them. In CDMA, each device is assigned a unique code which is used to modulate the device signal. These access schemes allow concurrent multiple transmissions using the whole available spectrum. These multiple transmissions can be modelled as a multi-commodity network flow problem [22].

Over wireless telecommunication systems, the information is converted into radio or electromagnetic waves which occur on the frequency spectrum. The radio or electromagnetic waves have very important characteristics: they can travel long distances through the air or space, they can go through walls, etc.

One of the areas of wireless network systems that has caught the attention of researchers in recent years is Mesh networks.

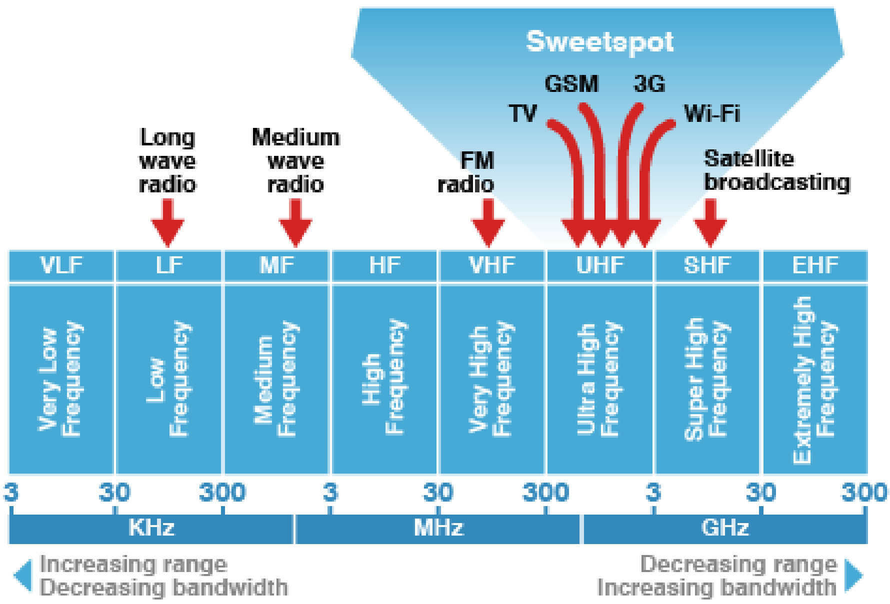

Figure 1.1: The radio frequency spectrum. Borrowed from [16]

A Mesh wireless network system [19, 89, 101] is characterised by the presence of a base station or in its absence by a network known as "*ad hoc*". A base station or access point has overall management of the wireless network system, controlling and coordinating all communications through the network. In contrast, an “ad hoc” wireless network system is a self-organising and distributed controlled network, i.e. devices are in charge of the management of the network. In the absence of a base station each device is considered as a transceiver, that is, devices are able to send and receive information. Some application of the mesh wireless network system are: Sensor wireless network system [99], Mobile ad hoc wireless network system, Multi-hop wireless network system [87].

An ad hoc multi-hop wireless network system is a collection of wireless mobile devices dynamically forming a temporary network without the use of any predefined infrastructure or centralised network administration. Due to the fact that there is no network infrastructure, the connection between devices is limited by its own transmission range (the maximum distance for which two devices can establish communication with each other). Consequently, multiple "hops" between devices might be needed for one message to go from the source to the sink through a wireless network system. Therefore, it is necessary to develop efficient routing protocols [27, 77]. Due to the “hops” involved in the route through which a message goes through to a receiver, it is necessary to take into consideration the power consumption which each device needs to forward a message. Adequate power consumption must be taken into consideration since each device has a limited battery capacity. Efficient power consumption is a major design goal for ad hoc wireless networks.

In order to have successful communication between two persons, it is necessary that both of them follow the same semiotic rules. These rules are also known as protocols. The need for protocols also applies to any wired or wireless network devices. These rules determine where information is kept and found, and how it is transmitted. The organisation of these functions is called *network architecture* and includes software and hardware. The development of transmission equipment by several companies with specific architecture, which varied from one company to another, generated communication barriers between equipments. It was not until 1978 that the International Organisation for Standardisation (ISO) proposed to develop a communication model known as *Layer Open System Interconnection* (OSI) [91]. In this model the protocol functions are divided into a hierarchical structure made up of seven layers, levels or strata, where each layer is responsible for a set of predefined functions. This layered model allows a better control of the protocol functions, manages the activity in the network and ensures inter-connectivity between different communication systems. Nevertheless, since each layer optimises its own goal separately and communication between layers is constrained to consecutive layers due to its hierarchical scheme, the network performance is limited [49, 73]. For these reasons, the OSI model does not work well in ad hoc wireless network systems. By contrast, in ad hoc wireless network system the administration of the communication is intertwined with different layers forming a cross layer model. The cross layer model jointly optimises the frequencies and the power control which are allocated in the transmission of a message through the wireless network system [82, 90]. This enables the physical, medium access, network and transportation layers to interact and promote energy efficient consumption in the wireless network system. Cross-layer design emphasises the performance of the wireless network by enabling different layers of communication to share information or to coordinate their actions in order to jointly optimise the wireless network performance. In Chapter 2 a brief description of the cross layer design used in ad hoc wireless networks is given, as is for each layer.

One of the main objectives in an ad hoc wireless network system is to minimise power consumption, since the devices are usually powered by batteries with a finite electrical charge. For certain applications over a wireless network system the *quality of service* (QoS) is still more important than power consumption. Among these are delay-sensitivity real time applications such as voice and video. In these applications delaying this transmission will affect the quality of the received data, i.e. the delay will cause pauses in the playback of the transmission. The QoS is measured by reference to various metrics like: minimum delay, robustness, high reliability and levels of high throughput [56]. For example, in the case of a natural disaster, the ad hoc wireless network requires a minimum delay in the network in order to maximise the number of survivors. In many applications, the QoS requires a cross layer design [82].

In summary, we have four main problems that are crucial for the ad hoc multi-hop wireless network system in order to minimise the total power requirements of the network while preserving the quality requirements:

- The routing problem of finding a path for a message from its sender to its destination.
- The power control problem of allocating the power in each device to send a message at a given time slot.
- The scheduling problem of determining these time slots when there is communication between devices.
- The optimal use of the radio spectrum.

Even though the technical problems are challenging, the corresponding mathematical modelling and optimisation problems are crucial. Since the corresponding optimisation problems are non-linear and non-convex, there is no guarantee that an optimal solution will be found and at best, it is only possible to find local minimisers. Literature on these subjects report advances [33, 39, 47, 93, 94, 96, 97].

In order to solve the mathematical problem, we will build upon some of these advances and develop computational programs. We present numerical solutions for different network topologies (depending on the number of nodes and links as well as number and size of messages).

1.2 Related Work

So far, many separate efforts have been made in order to solve each one of these problems. For instance, routing problems based on single or multi-commodity flow assignment have been studied by Gallager, Stern and more recently by Grover [43, 48, 78], the power control has been studied by Yates, Foschini and Grandhi [41, 47, 93, 94] and the scheduling problem has been studied by Makarevitch [65]. Although they have addressed the uses of routing, power control and scheduling, none of them has solved all three problems in a single solution. Ephremides and ElBatt [39] have considered the joint scheduling and power control in wireless systems problem by alternating two phases: First, they solve the scheduling problem to obtain a power assignation and in the second phase they solve the power control by determinate admissible powers, if there are any. If not, the process is sent back to the first phase to get a different schedule and power assignation without considering the routing of the message. Ephremides and Li present a similar idea in [100]. They prioritise the links that have large queues and buffer status in the first phase. Then, in the second phase, they assign power values by a distributed power control and continue this iterative algorithm until a solution, if one exist, is found. In both cases they are not considering the combination of the routing, scheduling and power control as a total system. Therefore the possible solutions found could be suboptimal.

An interesting approach is presented by Johanson, Xiao and Boyd [92]. They formulate a simultaneous routing and resource allocation, assuming link capacity and resource allocations for TDMA/FDMA systems. They separate the problem via dual decomposition, i.e. the problem is separated into two problems: data routing and resource allocation. Both problems are coordinated by the dual problem. In this article the authors do not consider the CDMA system which is presented in [55]. They only consider the joint optimisation of routing and power control. They propose a heuristic link-removal procedure consisting in the following two steps: First they solve the simultaneous routing and resource allocation problem removing those links with zero capacity and then they repeat this process until no links are left to be removed and it converges to a feasible solution. Cruz and Santana [33] study the problem of joint routing, scheduling and power control for wireless multi-hop networks. The algorithm that they present computes an optimal link scheduling and power control policy that minimises the total average power transmission in the network. Their algorithm present is efficient, but it is only designed for single path routing.

Kodialam and Bhatia in [25] present an interesting approach to the solution of the joint optimisation problem of routing, scheduling and power control for a TDMA-CDMA network. Nevertheless, they imposed restrictions on the model so that a device cannot be used in multiple communications at the same time and also the powers are calculated for given data rates.

1.3 Contributions of the Dissertation

As we mentioned before, the main characteristics of an ad hoc wireless network system is the lack of infrastructure. Therefore, devices need to perform the network set-up, management and control between themselves. Each device must act as a router and data forwarder.

In a wireless network system with an infrastructure dependent upon a base station, devices use their resources for data communication only, while the infrastructure runs centralised algorithms in order to determine the "optimal" network behaviour. On the contrary, in an ad hoc wireless network system each device must support network information and management activities in addition to data communication. Therefore, the optimal use of resources in each device is a critical issue.

In addition, the design of an ad hoc wireless network system is focused on the optimal use of the power consumption in the network. The power consumption in a device is an aspect that relies on the design of different layers. Medium access scheme, routing and data communication layers contribute to energy consumption. Enabling these layers to interact can promote efficient energy consumption within the network.

The main goal of this work is to solve the routing, scheduling and power control problem for TDMA/CDMA systems and routing, filtering and power control problem for CDMA systems for wireless networks. We also consider, for both systems, the single and multi-commodity problems, since in CDMA systems, it is possible to send one or more messages concurrently through an ad hoc wireless network, whether a message is a voice call, a multimedia stream or information data.

We propose two mathematical models. In both models the objective function is to minimise the total transmission power required to send one or more messages through an ad hoc wireless network system. They also share the same general set of constraints which describes the formal aspects of routing, power control, quality of service and link capacity. In both models, the coupling constraints between link capacity and power allocation are non-linear non-convex functions, so the formulation of the problem is a non-linear non-convex optimisation problem.

The first model for CDMA/TDMA systems is presented in Chapter 4. In this model we considered the joint routing, scheduling and power control optimisation problem without violating the access layer requirements and quality indicators. For the purpose of this work we will use two quality indicators: the Signal to Noise Ratio (SNR) [61] and the Signal to Interference Noise Ratio (SINR) [44] which will be described in Subsection 2.3.1.

The joint CDMA/TDMA scheme provides a higher throughput than a pure TDMA scheme because it enables multiple concurrent transmissions within the same time slot. In order to integrate the joint CDMA/TDMA scheme into our model we assign a colour mapping to each predefined time slot and device (refer to Section 3.2).

As we mention above, we are facing four different problems: The scheduling due to communication between devices, routing of multiple messages from a source device to a sink device, the allocating of power for each link and the optimal use of the radio spectrum; therefore, we need a power control policy. For the scheduling problem we will consider a pre-defined coloured network graph and a time-slot colour assignation i.e. each time slot has a pre-defined colour.

The second model presented in Chapter 5 considers only a CDMA system, i.e. we have multiple concurrent transmissions. The optimisation problem considers joint routing and power control, without violating the quality metrics. In this model we use the signal to interference noise ratio and include in this quality indicator one of two filters: the Single User Match Filter and Minimum Mean Squared Error filter [50] which will be described in Subsection 2.3.3.

To study these problems analytically, we rely upon previous works, done by Yates [93], Ulkus and Yates [96], Huang [95], Yates, Ulkus and Yener [97]. Then, to solve them numerically, we designed computer programs and experiments for different network topologies and different initial conditions (size and number of messages, number of time slots, bandwidth, etc).

We propose and prove two theorems, obtaining one remarkable result: in both models, at a local minimiser, the capacity of a link is maximised, assuring a good quality of service, i.e. minimum SINR or SNR targets are satisfied.

In this work we are not considering the mobility of devices.

1.4 The Structure of the Thesis

In Chapter 2, we give an introduction to wireless network systems. We present a description of the layers of the OSI model and the cross layer model which is considered for an ad hoc communication wireless network system in this work. At the same time, we give an introduction to very important concepts such as quality indicators, power control and filters. In Chapter 3 we give a brief introduction to non-linear programming, graph and network flow theory. In this chapter we give a brief description of the software and the solvers that will be used in order to solve the two mathematical problems presented in Chapters 4 and 5. In Chapter 4 we present a mathematical model which considers the joint routing, scheduling and power control optimisation problem in TDMA/CDMA ad hoc wireless network systems. In this chapter we present one of the most important parts of this thesis: the two theorems which support our mathematical models. We also present computational solutions to our non-linear programming problem, the results of the experiments for different initial conditions and results for sensitivity analysis. In Chapter 5 we present a different approach to solve the joint routing and power control optimisation problem by introducing two different filters which are described in Chapter 2. In this chapter we present the numerical results of experiments obtained with the application of the computational program for this mathematical model. In Chapter 6 we present our conclusions and suggested ideas for future research. At the end of this thesis there are two appendices for the model and data files used to solve the optimisation problems presented in Chapter 4 and Chapter 5 and a third appendix for the Matlab code corresponding to the generation of one parameter used in Chapter 5.

Chapter 2

Wireless Network System

In this chapter we give a brief description of the different wireless network systems. All communication systems regardless of their nature (wired or wireless) have protocols which describe the communication rules among devices. These protocols are organised in layers that have different functions. We introduce the concepts that are going to be used in the mathematical model presented in Chapters 4 and 5 including a brief description of the characteristics of each layer that is considered. The concepts defined are: signal to noise ratio, signal to interference noise ratio, the medium access control scheme, quality of service, interference, reception filters and data flow protocols. It is important to mention that in this thesis some concepts are used in various ways depending on the particular context, for example: a channel can be considered as a transmission medium such as: optical fibre, coaxial cable, twisted pair cable, water, air or as the radio frequency band. We will also consider a user as a device and in the mathematical concepts both (user, device) will be considered as nodes.

2.1 Wireless Network System

In this section we present a brief introduction to a typical wireless network system and the different groups of wireless network systems.

The term wireless network system refers to a technology which allows two or more devices to transmit information between them without network cabling. These devices can be either mobile or fixed such as: mobile phones, computers (laptops or desktops) or personal digital assistants (PDAs), etc.

Digital wireless communication is not a recent idea. By 1901, the engineer Gugliemo Marconi [66] had demonstrated that wireless transmission was possible. Marconi's demonstration opened the door to a new way of communication. Wireless communication systems have had a deep impact on our daily life. Nowadays it is possible to have many different services on our mobile phones such as using the Internet, reading our emails, receiving local TV or radio station signals.

The wireless network system, as far as is known [44], can be divided by their coverage area into three main groups; Short range/PAN (Blue-tooth), Mesh wireless network (WAN) and Satellite systems. Blue-tooth was developed to interconnect communication devices and accessories either between one other or to a computer using short-range and low-power wireless radio waves. The Mesh wireless network [17] can be divided into two groups: Wireless Local Area Network (WLAN) and Wireless Wide Area Network (WWAN). A WLAN supports high-speed data transmission among wireless devices such as laptops and desktop computers within a small region like a small building or a university campus through radio frequencies. A WWAN provides wireless data communication to high mobility users over a large area. A Satellite system is one which provides the widest coverage. This system requires a great amount of power to reach all of the satellites. Satellite systems are mainly used to broadcast video and audio over a large geographic region. A representation of these wireless systems is shown in Figure 2.1.

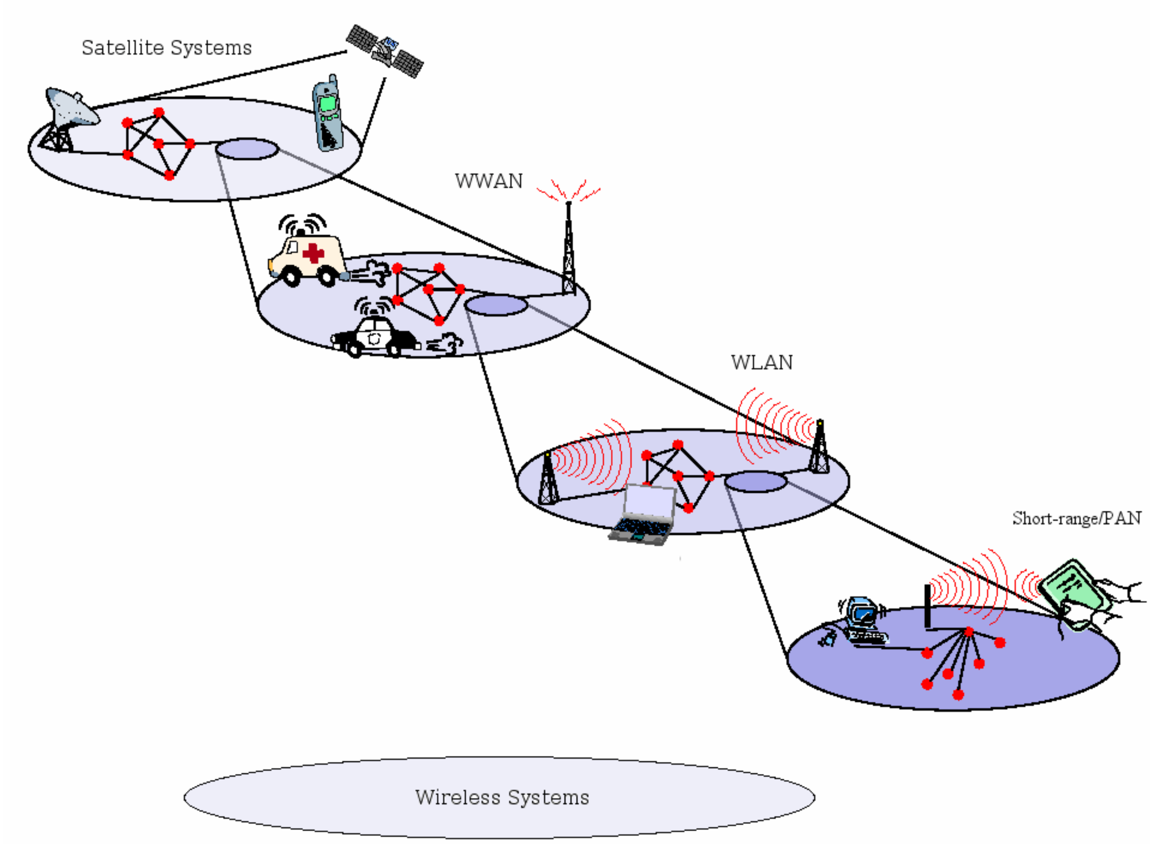

Figure 2.1: Wireless System. Borrowed from [75].

A mesh wireless network can be implemented in two ways: by the presence of a predefined infrastructure or, in its absence, by what is called infrastructure-less.

Infrastructure-based wireless networks have central or base stations. The networks are controlled by base stations. Often the base stations are interconnected to facilitate this control. The devices, such as mobile phones, personal digital assistance (PDAs), try to log on to the wireless network through the closest base station to them. To transmit a message along this network, the sender device will send the information to the base station (a single hop) and then the base station is in charge of forwarding the message to the receiver device (another single hop), as we can see in Figure 2.2.

The communication between devices can be either point-to-point or broadcasted [44]. The point-to-point communication establishes a one-to-one communication between a sender device and the intended receiver device, while for the broadcasted communication the sender device broadcast the signal to multiple receiver devices. In point-to-point communication, it is required that the sender device and the receiver device have a direct alignment. This alignment it is not required in broadcast communication.

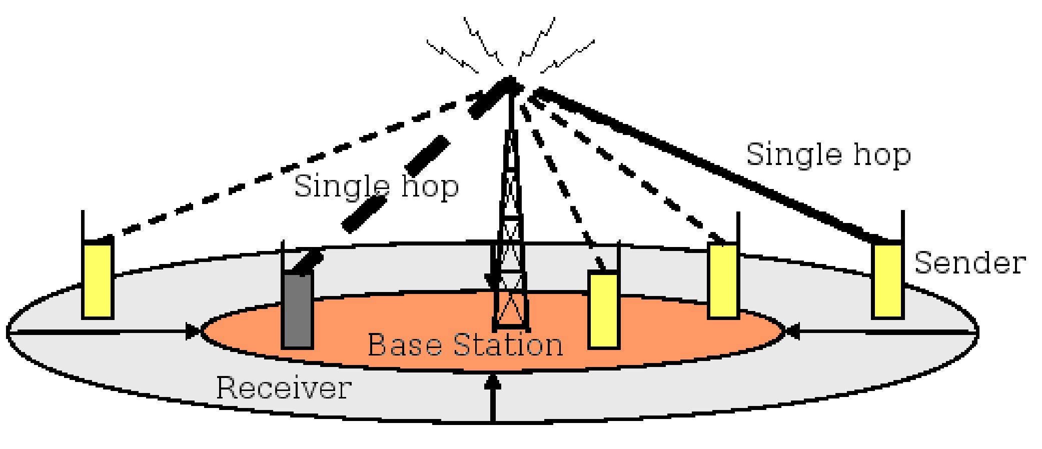

Figure 2.2: Infrastructure based. Borrowed from [75].

If the infrastructure wireless system has no central station, the devices dynamically create a wireless network among themselves. This is known as an infrastructure-less wireless network. Since the infrastructure-less wireless networks lack a central base station, the devices take control of the networking tasks. In this kind of a network, if the communication path is broken or blocked, the devices have the ability to "self-heal" by auto-reconfiguration in order to guarantee the network's connectivity. To transmit a message along this network the message will "hop" from device to device until the destination is reached. An example of this kind of network is shown in Figure 2.3.

Figure 2.3: Infrastructure-less wireless network.

Conventionally, wired network design follows a layered communication architecture. Each layer handles specific functions in the network. In the following sections we present the layer communication architecture.

2.2 Layered Communication Architecture

This section is a small introduction to the subsequent Sections 2.3, 2.4, 2.5.

The design of the equipment for a wireless network system is complex. It involves the hardware and the software for each of the devices that make up the equipment.

In the early 1970, various manufactures developed better communication technology. Since they competed against one another, each of them had their own design. This led to a lack of communication between devices of different manufacturers.

For this reason ISO presented the idea of a unique protocol for communication. In 1984 the OSI model was finally released. This model divides the functions of network communication into seven layers. The concept of a layer is intertwined in the architecture of the equipment and relies both on the hardware and the software used and determines where specific information is found. Each layer is independent of the other layers and assigned specific functions. The protocol defines the communication between layers in the device with the corresponding layers of other devices. The layers are ordered according to the nature of their functions. Usually the first three layers are referred to as lower layers and the rest as upper layers.

The set of protocols associated with all layers is known as the protocol stack of the network. In Sections 2.3, 2.4, 2.5 a brief description of the layers considered in our mathematical models is given. It is not the aim of this thesis to describe in detail a network layer design; for more details, we refer to [44, 51, 83].

2.3 Physical Layer

In this section we present the characteristics of the physical layer and definition of the signal to noise ratio and the signal to interference plus noise ratio. The first layer of the communication architecture is the physical layer. This layer deals basically with transmitting bits such as binary digits over a device-to-device wireless link. Sometimes the physical layer is also referred to as *a link layer.* The physical layer has two main basic functions: transmission and reception of information data through the transmission channel. The basic functions of the transmission component are: taking the information bits from the source device and modifying them into a suitable form for the transmission channel and the mapping of information bits into energy. In this case channel refers to the physical medium used to transfer the information from the sender device to the receiver device. A possible communication channel for wireless networks include air, water and vacuum, while for a wired network it could be coaxial cable, fibre optic cable or others. In wireless network systems, the transmitted signal suffers deterioration due to the distance between a source device and receiver device, interference and noise while traversing the wireless medium. In general, the noise can be defined as an unwanted electrical signal interfering with the desired signal . This unwanted signal comes from a wide variety of sources such as: electronic devices, atmospheric disturbances, extraterrestrial radiation, etc. The transmitted signal loses energy as it is travelling from the source device to the receiver device. The further away the source and the receiver are from each other, the greater will be the energy loss and the attenuation of the signal also known as path-loss.

The design of the physical layer is extremely crucial since it has an important impact in the upper layers protocols. But, there are also few aspects of the physical layer design that do not affect the upper layers protocols [91].

Since any two devices can transmit information directly between themselves, the transmission power must be sufficiently high to achieve an acceptable signal quality at the receiver device. However, high transmission power from a link can produce a high interference in the neighbouring links within the network thereby degrading the performance of the link and sometimes breaking the communication between devices. The link performance of the network is driven either by the Signal to Noise Ratio (SNR) or Signal to Interference plus Noise Ratio (SINR) which are defined in the following subsection.

2.3.1 Signal to Noise Ratio and Signal to Interference Noise Ratio

We consider in this thesis CDMA/TDMA systems (refer to Section 2.4). We are proposing to combine two multiple access schemes, TDMA and CDMA, so that it is possible to include the best features of both. CDMA systems are based on code division while TDMA are based on time division. In a telecommunication system where both schemes are present it is possible to have concurrent transmissions and at the same time yielding a better utilisation of the radio spectrum. The radio spectrum is a scarce resource and the use of the time transmission is also important. Unfortunately in concurrent transmissions, the signal sent through one link is highly sensitive to the activity of other neighbourhood links and the background noise which, causes interference in the received signal. Therefore it is necessary to measure the quality of the transmission of information over each link. We will refer to this measure as link quality indicator. Two well known quality indicators for communication links are the SNR and the SINR.

The SNR is defined as the ratio of the received signal power to the background noise, while the SINR is computed as the ratio of the received signal power to the interference caused by other concurrent links and the background noise. In general background noise is considered a random process. It can be caused by atmospheric disturbances, by the surrounding devices or any other unwanted signal. If the precise statistics of the background noise is known, then it is possible to fit a probability density function. Generally this probability distribution follows a Gaussian process and in most cases it will be an Additive White Gaussian Noise (AWGN) (refer to [61]). Both SNR and SINR at each link must exceed a predefined lower boundary level or minimal threshold known as target. The links for which the SNR or SINR are below this level are not considered good enough for transmitting information due to their poor quality. The flexibility in link connectivity that results from varying the power has major implications for routing since lowering the power used by devices reduces the interference in the network system. This will in turn allow more links to be considered in the routing of a message.

2.3.2 Mathematical Models of SNR and SINR

The terminology and mathematical notation presented in this subsection will be used throughout the whole thesis. We will use the following mathematical notation: capital and boldface letters for real matrices e.g. $\mathbf{W}$; lowercase and boldface letters for real vectors e.g. $\mathbf{p}$, capital italics letters for sets e.g. E.

Let us define the structural relationship in a wireless network system which consists of a subset of possible one hop communication links among the devices. Each link is determined by the transmission range of the device. The network is represented by a directed graph.

Consider a wireless network system consisting of r nodes which communicate with each other via a wireless link. Let V be the set of nodes $V = \{v_1, \ldots, v_r\}$. Let us define the transmission range of a device (node) u by $\varrho_u \in \mathbb{R}_+$ and the distance between any two nodes u and v by $\varsigma_{u,v} \in \mathbb{R}_+$. Both the distance and transmission range are measured in the same units. The electrical energy needed to send information from one device to another is known as transmission power and is measured in Watt. The network topology is defined as follows:

Definition 2.1: *Network topology*

The network topology is the set of links:

$$E = \{e = (u, v) \in V \times V \mid u \neq v, \varsigma_{u,v} \leq \varrho_u\}$$

In other words, there exists a link $e = (u, v) \in E$ from sender node $u \in V$ to receiver node $v \in V$ if and only if $u \neq v$ and node v is within the transmission range of node u. If every node can communicate with every other node, then we have a full mesh.

Therefore, the directed graph G associated to a wireless network topology is defined as $G = (V, E)$. Each link $e \in E$ has a variable transmission power associated with the sender node, which is denoted by the variable $p_e \in \mathbb{R}_*$.

In the absence of interfering users (TDMA scheme access) the interference plus noise ratio is defined as follows (refer to [85]) :

Given $\eta \in \mathbb{R}_+$ as the background noise, the signal to noise ratio SNR_e for link $e \in E$ is given by:

$$SNR_e = \frac{p_e}{\eta^2}. \tag{2.1}$$

When we consider interfering users and simultaneous transmissions (CDMA scheme access) the signal to interference noise ratio $SINR_e$ for link $e \in E$ is given by (refer to [85]):

$$SINR_e = \frac{p_e}{\sum\limits_{e \neq i,\, i \in E} \rho_i p_i + \eta^2} \tag{2.2}$$

where ρ_i weights the effect of interferences which depends on the orthogonality between codes used. For the purpose of this thesis we set $\rho_i = 1, i \in E$ (refer to [37, 54]).

To guarantee a good QoS over the link $e \in E$ we must ensure that

$$SNR_e \geq \gamma^*$$

or

$$SINR_e \geq \gamma^*.$$

Here, $\gamma^* \in \mathbb{R}_*$ is called the ratio target, which is the minimum acceptable level of SNR_e or $SINR_e$ for the link $e \in E$. We can rewrite (2.1) and (2.2) as:

$$\gamma^* \leq \frac{p_e}{\eta^2}, \tag{2.3}$$

$$\gamma^* \leq \frac{p_e}{\sum\limits_{e \neq i,\, i \in E} p_i + \eta^2} \tag{2.4}$$

respectively.

The inequalities (2.3) and (2.4) will be constraints in our mathematical models presented in Chapters 4 and 5.

2.3.3 Filters

Signal processing is another technique used to reduce the interference in wireless network systems. This signal processing is also known as filtering or equalisation of the signal and is done at the receiver device. It is not an aim in this thesis to fully describe filtering. For more details we refer to [21, 80, 85]. For the purpose of this thesis we will present two different linear standard filters: the Single User Matched Filter (SUMF) and the Minimum Mean Squared Error filter (MMSE) which are included in the set of constraints of the mathematical model in Chapter 5.

The filtering of the signal helps to mitigate the interference while ensuring specific

SINR requirements.

Let us define the interference function for SUMF filter for the link $e \in E$ as:

$$I_e^{SUMF}(\mathbf{p}) : \; = \; \gamma_e \left(\sum_{\substack{k \neq e \\ k \in E}} |r_{e,k}|^2 p_k + \eta^2 \right) \tag{2.5}$$

where γ_e is the $SINR_e$ for the link $e \in E$, η is the background noise and $r_{e,k} \in \mathbb{R}_*$ is the interrelation between the received signal by the e^{th} user and the interference caused by k^{th} user after the signal has been filtered. This generates a normalised cross-correlation matrix $\mathbf{R}$ of size $|E| \times |E|$ with entries $R_{e,k}$. Since $\mathbf{R}$ is a normalised matrix, we assume that $R_{e,e} = 1$.

For the MMSE filter (refer to [85, 86]), we will define the $|E| \times |E|$ filter matrix $\mathbf{W}$ as:

$$\mathbf{W} := W(\mathbf{p}) := \left(\mathbf{R} + \eta^2 \mathbf{A}\right)^{-1} \tag{2.6}$$

where $\mathbf{R}$ is a normalised cross-correlation matrix, the matrix

$$\mathbf{A} = diag\left\{ {}^1/_{p_{e_1}}, {}^1/_{p_{e_2}}, \ldots, {}^1/_{p_{e_{|E|}}} \right\}$$

is of diagonal structure and η is the background noise. The properties of (2.6) are being widely studied by Verdu [63, 85].

The impact on noise is given by

$$\beta_e = \left(\mathbf{W R W^T}\right)_{e,e} \tag{2.7}$$

where $\mathbf{W}^T$ denotes the transposed matrix of $\mathbf{W}$.

Let us define the interference function for *MMSE* filter for the link $e \in E$ as:

$$I_e^{MMSE}(\mathbf{p}) := \frac{\gamma_e}{\left|(\mathbf{WR})_{e,e}\right|^2}\left(\sum_{\substack{k\neq e\\ k\in E}}\left|(\mathbf{WR})_{e,k}\right|^2 \cdot p_k + \beta_e\eta^2\right) \tag{2.8}$$

here $\left|(\mathbf{WR})_{e,k}\right|$ denotes the absolute value for the $(e,k)^{th}$ entry of the matrix $(\mathbf{WR})$. From (2.8) and (2.7) it follows that if the matrix $\mathbf{W}$ is equal to the identity matrix $\mathbf{I}$, then we have the SUMF filter (2.5) since the matrix $\mathbf{R}$ is a normalised cross-correlation matrix.

There are some articles related to minimisation of the transmitted power while ensuring a specific SINR requirements over a wireless network system [28, 38, 98]. These filters are going to be part of the set of constraints in the mathematical model presented in Chapter 5.

The second layer considered is the access layer responsible for administering the available radio spectrum and correct reception of the information.

2.4 Access Layer

In this section we introduce some concepts such as: multiple access techniques sharing the radio spectrum. We will use these concepts in the mathematical models presented in Chapters 4 and 5.

The access layer administers the way different devices share the available radio spectrum and ensures successful reception of the transmitted information. It is also in charge of the retransmission of a data if it has been received with errors. The access of a device to the wireless transmission channel is done either by multiple access or random access.

The radio spectrum is a scarce resource that must be allocated for different applications in wireless communication networks [52]. Therefore, it is necessary to divide up the available resources into channels, so as to assign each channel to a different device. The assignation of a dedicated channel to a single device is often called *multiple access*. The most common multiple access methods of dividing up the spectrum is the *frequency division*.

Frequency Division Multiple Access (FDMA) : *"Refers to sharing the spectrum by assigning specific frequency channels to specific user"* [51]. Here the devices can transmit information simultaneously over time. For example: a room with a group of people talking all at the same time, but at a different tone. Figure 2.4, shows a graphical representation of FDMA, where the coordinates axes are: $t = time$, $f = frequency$ and $c = code$. Where the available spectrum is divided according to the access method.

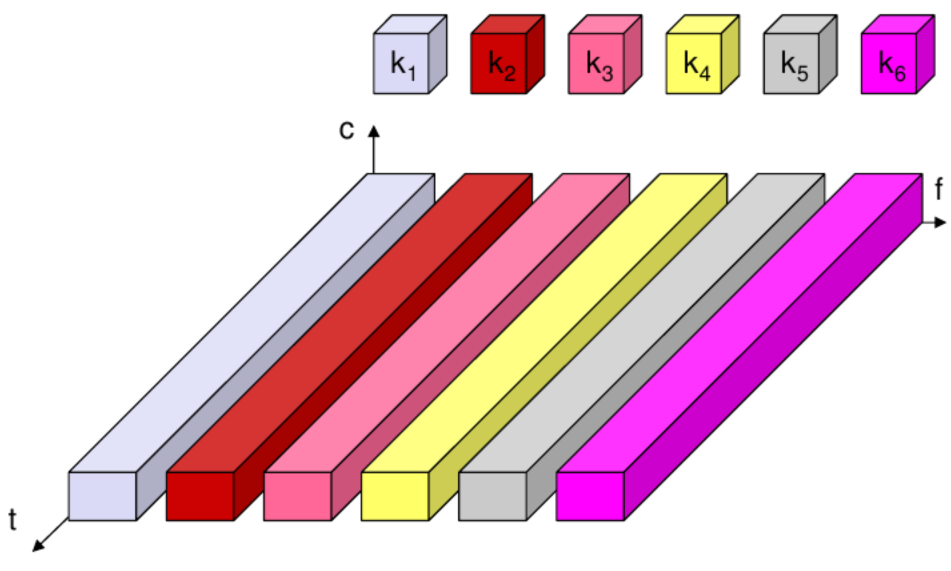

Figure 2.4: Frequency Division. Borrowed from [35].

Time Division Multiple Access (TDMA): *"Refers to allowing all users access to all* *of the available spectrum, but users are assigned specific time intervals during which they can access it*" [51]. Each channel will use the entire bandwidth in a specific time slot. B*andwidth* is a rank of frequencies assigned for a particular application of the electromagnetic spectrum and it is a limited resource. The time is assumed to be partitioned in *time slots* of duration $\tau \in \mathbb{R}_+$. Each device is assigned (cyclically) repeating time slots which implies that the transmission is not continuous for any user. Therefore, the information must be stored in the device area usually called device buffer (this concept is very important since it is part of the set of constraints (4.1, 4.4, 4.6, 4.7)). For example: the same group of people, but this time only one is allowed to talk at a time and the rest will be quiet.

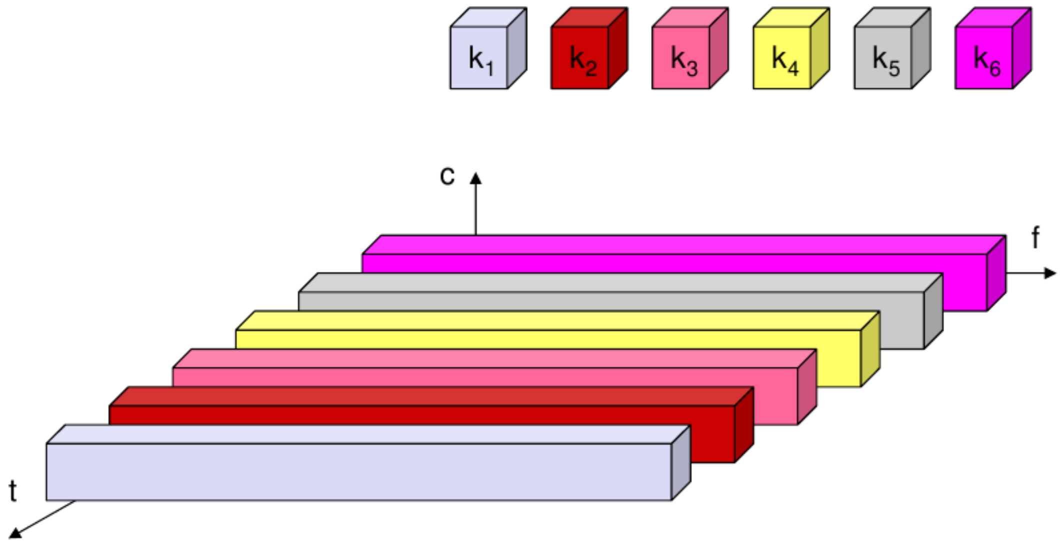

Figure 2.5: Time Division. Borrowed from [35].

Code Division Multiple Access (CDMA): "*is a form of spread spectrum modulation in which users are allowed to use the available spectrum, but their signal must be spread ("encrypted") with a specific code to distinguish it from other signals*" [51]. Here each channel uses the entire frequency all the time. For example: the same group of people all talking at the same time, but in different languages.

Access layer is in charge of assigning channels to devices in the wireless network system. In order to accept a new device in the wireless network system, the methods FDMA, TDMA, and CDMA provide protection for those existing SINR link targets when a new node enters in the system.

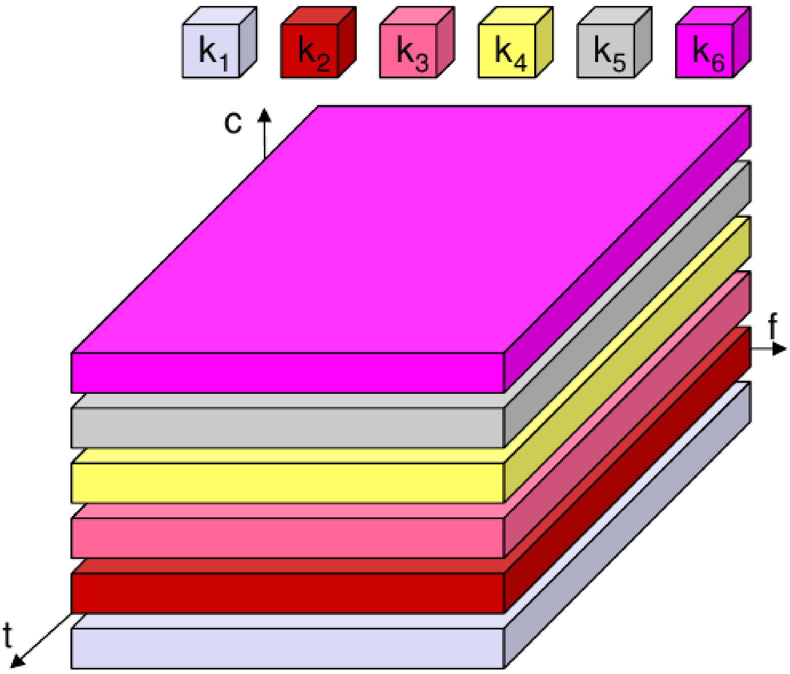

Figure 2.6: Code Division. Borrowed from [35].

These access methods allow multiple users to access portions of the available bandwidth. The data rate available to each device is another important consideration, especially for data and multimedia services. This sometimes is wrongly referred to as the “bandwidth”, but in fact the device data rate is not a simple function of the available bandwidth, since it is influenced by important elements as: the modulation of the scheme employed, the correction and detection schemes in use and the channel quality [44]. The capacity c of a channel is given by Shannon’s [26, 81] well known formula (measured in bits per second):

$$c = B \log_2 (1 + SR) \tag{2.9}$$

where SR represents either the $SINR$ or SNR and $B \in \mathbb{R}_+$ is the bandwidth. The equation (2.9) is generally used as an upper bound on the data rates and will be a constraint in the mathematical models in Chapters 4 and 5.

2.4.1 Interference

In telecommunication systems, interference is everything which modifies, disturbs or alters a signal transmitted over a channel from source device to receiver device. Some examples are: Adjacent channel interference, co-channel interference, inter-symbol interference, electromagnetic interference and others [18]. The definition of the interference function may vary from system to system. In Chapter 5 we give the definition of an interference function used in the mathematical model.

An "adequate" communication between devices in a wireless network system is subject to the neighbourhood interference. In order to reduce the interference between devices, there are two different techniques: power control and filters over the received signal.

In this section we have introduced concepts which will be used in the different mathematical models in this thesis. We will continue with the network layer which is responsible for the routing of a message.

2.5 Network Layer

In this section we present the network layer, which is responsible for routing the message from the source device all the way to the destination device. Reaching the destination device may require making several hops to intermediate devices along the network.

The topology of the network changes all the time, either by the admission of a new device into the system or by the mobility of the devices. When a new device is accepted in the system, it has to identify which devices are within its transmission range. This process is known as *neighbour discovery* [44]. During this process all the devices in the system must apply power control in order to meet the SINR target γ^* for all the links in the system. The mobility of the devices can result in the disappearance of established routes. These continuous changes in the network topology will modify the message routing. Therefore, it is necessary to design efficient routing protocols. There exist three main categories of routing protocols: *flooding, proactive* and *reactive routing* [44].

The flooding protocols, [44] main task is to broadcast the message from sender device to those devices that are within its range. Then the receiver devices broadcast the message in the same way, until the message arrives at its destiny device. This is not an efficient solution because there is an excessive use of vital resources like battery and bandwidth. On the other hand, it is an efficient solution in terms of network mobility. In Figure 2.7 the source node broadcasts a message to all those devices that are within “one hop” range distance; then these devices forward the message in the same way until the message reaches its sink.

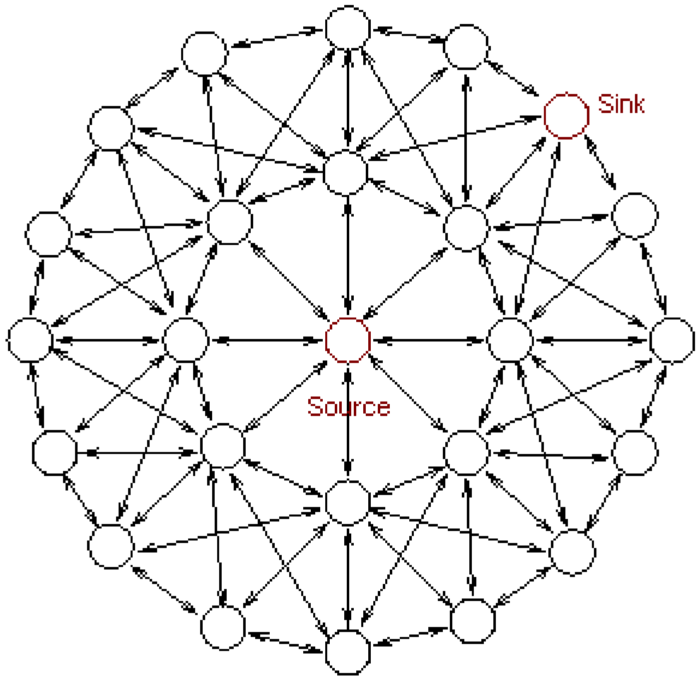

Figure 2.7: Flooding protocol. Borrowed from [3]

Proactive routing protocols [44] try to maintain routes to all devices in a wireless network system at all times by broadcasting routing updates in the system. Proactive protocols are subdivided into three categories: *centralised, source-driven* and *distributed* [44].

In reactive routing protocols, routes are created only when a device needs to send information through a wireless network system. Therefore, it is not necessary to maintain link connectivity all the time.

For the purpose of this thesis, we consider all known routes that are present in the topology of a wireless network system.

The optimal route that a message must follow is dictated by routing protocols. This optimisation is based on different criteria such as delay or minimum congestion in the network. In any criterion, it is necessary to take into consideration not only the routing of the message, but also the capacity of the link in order to allocate the power and bandwidth. Thus, there exists a significant interaction between the physical, access and network layers [100]. This interaction is also used in this thesis through the set of constraints ((4.7), (4.14), (5.7) and (5.8)) in the mathematical models in Chapters 4 and 5. This interaction is present in the *ad hoc wireless networks* which will be described in the following section.

2.6 Ad Hoc Wireless Network

In this section we give a short introduction to ad hoc wireless networks and the cross layer protocol design. For more detailed information about this section, refer to [31, 45, 76, 82, 86].

An ad hoc wireless network is a collection of wireless devices, dynamically forming a temporary network without the use of any pre-existing network infrastructure or base station. Therefore, the devices must handle the control and networking task by themselves. Each device in the network can communicate directly to each other through a wireless link, therefore each device is a *transceiver*. A transceiver is a device that can receive and transmit radio signals. Therefore, each device can either transmit or receive information (message) from one or more devices. For instance, let us consider the following event. A big earthquake has devastated a city, destroying among other things, the communication systems, both wire and wireless system (phone lines, cellular networks, etc.). Emergency services such as police, fire, paramedic, along with volunteers, are working to help the people. One of the main problems that they may face is the communication between them since the city's communication systems are down. An ad hoc networks system can be used in this kind of situation providing the communication between all the different rescue teams.

Due to the fact that there is not a network infrastructure, the connection between devices is limited by its own transmission range. Consequently, multiple "hops" might be needed for one device to exchange data with another device across the network. Therefore, the communication between device u and device v is established if, and only, if the power of the radio signal received by device v is “inside” within the transmission range of device u. For example, in Figure 2.8, devices B, E and D are within the range of device C. If the receiver device is within the source's transmission range, the message goes directly to the receiver in a single “hop”, otherwise the message goes through intermediate devices from the source device to the receiver device in a “multi-hop” process. For example, from Figure 2.8, the source device A is out of the range of receiver F. Therefore, in order to send the message from device A to device F it is necessary to make a “multi-hop” from device A to B, from B to C, from C to E and finally from E to F. Ad hoc wireless networks will likely be utilised more frequently in the future due to the following reasons:

- The cost of installation and maintenance of the network is minimal.
- It can be quickly deployed and reconfigured.

However, there are some disadvantages to this kind of network, such as the routing of the message and the device's power consumption.

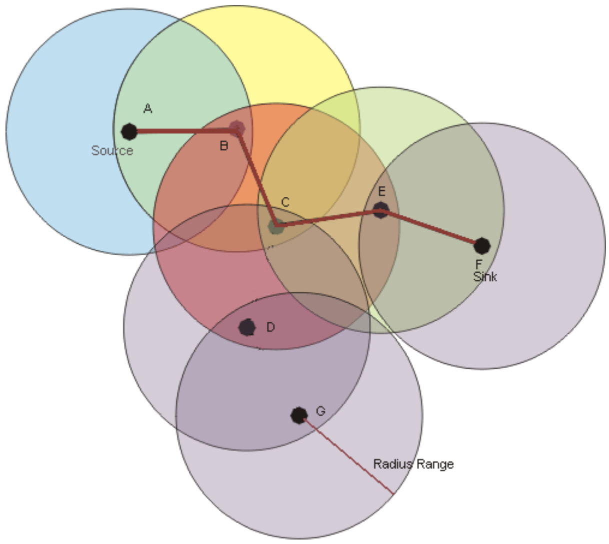

Figure 2.8: Node's Radio Range. Borrowed from [11].

As mentioned above, the transmission in a wireless network system is through radio waves. Radio waves have some important properties;

- They can be generated easily.
- They can travel long distances.
- They go through walls or buildings easily.
- They are omnidirectional (meaning they travel in all directions).

However, the radio waves are attenuated by the media that the signal must travel through such as air, wood, concrete, rain. Radio waves are *frequency* dependent. The term frequency means the number of occurrences (waves) per second and is measured in Hertz (Hz). Although at low frequencies, radio waves pass through obstacles well, the quality of transmission power is drastically reduced in proportion to the distance from the source. On the other hand, at high frequencies radio waves are good for long distances, but they are absorbed by rain and can not go through obstacles easily so they tend to be bounced off. At all frequencies, radio waves are susceptible to noise and interference from other electrical devices.

2.6.1 Ad Hoc Wireless Challenges

Although technology for ad hoc networks exists, it has not been fully implemented in our society. This is due to the challenges to be faced in the practical implementation. The main challenges are as follows:

Nodes are often powered by batteries and have limited transmission range. The device's transmission range is power dependent. Since power can be increased or reduced dynamically, the transmission range will change accordingly. With sufficient transmission power any device can transmit a signal directly to any other device. The transmission power associated to a single device affects all other devices. If the transmission power of each device is fixed, then the SINR of the communication link between any two devices will decrease as the distance between the devices increases. Therefore, power control across all devices in the network is recommended. Power control is one of the main challenges for the ad hoc wireless networks. As we mentioned before the power control is strongly related to the SINR. At the same time the SINR determines the communication link's performance. Therefore, a link with a very low SINR is not considered good enough to transmit information, due to its extremely poor quality. Moreover, the SINR is not only affected by the distance, but also by the interference caused by the power used by other devices and the background noise. Due to the lack of infrastructure, the connectivity in the network is constantly changing due to devices entering or leaving the system and their continuous mobility. The connectivity on the network is controlled by adapting the transmission power of existing devices. In general the device's battery power is reduced as each operation is performed until the battery's power is exhausted. Therefore, controlling the device's power not only provides an SINR required, but will also extend battery life.

Another important challenge is message routing. As mentioned above, all devices contribute to the management of the network. Therefore, any device entering, leaving or moving within the network will affect not only the corresponding SINR, but also the routing of the message along the network. Unlike a single-hop, this causes an excessive interference with surrounding devices affecting the SINR. On the other hand, sending the message through intermediate devices (multi-hop) reduces the transmission power, therefore obtaining a better SINR. There are few routing protocols that take into consideration the power control algorithms such as Power-Aware Multi Access Protocol with Signalling ad hoc Networks (PAMAS) [77] or Power-Aware Routing Optimisation Protocol (PARO) [46]. In some ad hoc network scenarios, the network can include hundreds or thousands of devices, so the implemented protocol must be able to operate efficiently in the presence of a large number of devices. It is also possible that each device sends a message through the network; therefore we could have as many messages as devices. This is known as a multi-commodity network flow problem. We refer to Section 3.3 or [67, 70].

To support real-time applications, such as video transmission, high data rates and low delay rates are required. Transmitting high data rates presents a big problem due to the link quality which diminishes rapidly as the number of hops increases. Since real-time applications are delay sensitive, they require a good QoS assured by fixed bit rate reception. In addition, it is difficult to provide a good QoS in an ad hoc wireless network due to its dynamic nature. As mentioned above, to guarantee a good QoS each device needs a dedicated channel.

In summary: in order to guarantee a good performance of a network, there are some requirements that ad hoc wireless networks must satisfy such as: the SINR and QoS constraints, both of them being subject to power control, routing and bandwidth.

In an ad hoc wireless network the OSI layer model (refer to Section 2.2) does not apply due to the changing nature of the system, so it is necessary to define a new layer model in which this need is reflected.

2.6.2 Cross Layer

As mentioned above, in the traditional layer OSI model the set of tasks of each layer is independent from each other. Each layer interacts directly only with the layer beneath it and provides facilities to be used by the next layer. In contrast, the functions of the layers of the ad hoc wireless network are mutually dependent. For example, the power control is managed by the physical and by the access layer and is also present in the routing protocol. Therefore, the traditional layers and protocols stack used in wire and wireless networks cannot be used in ad hoc wireless networks. An ad hoc wireless network has requirements to be jointly fulfilled such as power control, routing, high data transmission rates and QoS metrics. To meet these requirements, it is necessary to design a cross-layer protocol that supports adaptivity and optimisation across multiple layers of the protocol stack. Moreover, good protocol designs for independent layers are not efficient for dependent layers. For example in a cross layer design, if the interference or the transmission power in the system changes, then all layers of the protocol stack will be affected and all layers must respond by changing channel conditions: at the physical layer the SINR; at the access layer the transmission power and the interference; at network layer the scheduling and routing. Therefore, the relations between layers must be designed, exploited and jointly optimised. As an additional advantage, the cross-layer design can be implemented in wire networks.

The nature of the wireless links determines the features of the cross layer design and affects all levels of the network stack. One of these features is the ability of all layers to respond to changes in the channel condition. Therefore, it is necessary that corresponding coupling conditions between protocols of different layers are satisfied [31, 45].

For example, at the physical layer, signal power can be dynamically adjusted to respond to interference changes; at the access layer, adaptive scheduling can be implemented based on the current level of interference and the link quality; at the network layer, routing can be implemented in response to the levels of interference; and in all layers the QoS metrics requirements are met adjusting dynamically the levels of interference. All layers react to the impact of the levels of interference by the number of devices in the network. Therefore, for efficient design, the adaptation at each layer should not be independently developed, but it should be designed in an integrated way.

Due to the characteristics of the cross layer design, the interest of the research community has increased recently since it presents advantages over the traditional network layer design [20, 53, 71, 72].

2.7 Summary

Due to the nature of ad hoc wireless networks, devices must manage several performance aspects, such as power control, access control, routing schemes, security, etc., in contrast with a wired or wireless network, infrastructure based, where these aspects are managed by a central station. These particular aspects cut across the traditional layer stack. For instance, routing decisions and medium access have a mayor impact on the power consumption, considering both of them jointly can provide efficient power consumption. Another important aspect in an ad hoc wireless network is the mobility of the devices since it causes changes over the physical layer due to change in the levels of interference, or over the access layer since the strength of a link is strongly related to the interference as well.

We propose in this thesis a cross layer approach which will solve the joint routing, link scheduling and power control problem for ad hoc wireless network with the aim to minimise the total consumed power in this kind of network. It will be possible to send real time applications such as browsing, interactive video or voice calling and also to send one or more messages through the ad hoc wireless network. Therefore, we will address the multi-commodity network flow problem (refer to Section 3.3). Since these applications are generally large sized, messages will be split into parts of different sizes. For higher traffic loads, our approach supports higher data rates through concurrent scheduling links since we are considering TDMA/CDMA network access.

We consider three sets of decision variables: transmit power, size of buffer and size of the parts of a message. Transmission power is constrained by the maximum transmission power that each device has ((4.2) and (4.3)). The size of the buffer has a maximum upper bound given by the device characteristics (4.4).

The mechanisms of this approach consist in the coupling of power consumption variables and the size of the part of a message through Shannon's equation, see (2.9). The right hand side of this equation will be an upper bound for a part of a message while the “strength” of the link is guaranteed (4.14) or (4.16).

The feasible sets described in Chapters 4 and 5 are non-convex since the inequalities (4.16) and (5.8), respectively, are non-convex. The objective function is increasing with respect to the power consumption. Therefore, we have non-convex programming optimisation problems. In these models we consider that the routing, power control and scheduling joint optimisation guarantee QoS given metrics.

Our model does not consider mobility and distances between nodes; nevertheless they will be considered for a future research project.

In the next chapter we give a brief introduction to the non-linear programming problem, some aspects of network flow and graph theory and the mathematical language tools that are used in this thesis in order to solve the mathematical models presented in Chapters 4 and 5.

Chapter 3

Non-Linear Programming

In this chapter we present a brief description of several aspects of non-linear programming optimisation problems, graph theory as well as network flows. We have mentioned in the preceding chapters of this thesis that one of the main objectives of the control of ad hoc wireless networks is the optimal use of the device's battery, since it is limited. To this end, it is necessary to consider the cross layer protocol design. This protocol considers the joint optimisation of different layer protocols and is the basis for the mathematical models presented in Chapters 4 and 5. In these mathematical models the main goal is the minimisation of the total power consumed in an ad hoc wireless network subject to several constraints. These constraints are linear or non-linear real valued functions. In order to solve the corresponding non-linear programming problem, we will use a set of computational algorithms called "solvers". The mathematical model must be "translated" into a suitable input for the solvers. Therefore, we present a brief introduction to A Mathematical Programming Language (AMPL) which is a high level programming language for describing and solving large and complex optimisation problems.

3.1 Non-Linear Programming

Optimisation problems are often classified according to a special structure of their constraints and objective function. Consider the mathematical programming model (L), consisting of the optimisation of a real valued function f subject to a set S. The set S is represented by real valued equality and inequality constraints. If we assume that at least one function, either the function f or any of the constraints, is non-linear, then the general non-linear mathematical programming problem (L) can be stated as follows:

$$\textit{minimize} \qquad f(\mathbf{x})$$

subject to

$$\mathbf{x} \in S$$

where

$$S = \left\{ \mathbf{x} \in \mathbb{R}^n \;\middle|\; \begin{array}{ll} g_i(\mathbf{x}) = 0, & i \in N_{\mathcal{O}} = \{1, \ldots, k\} \\ h_j(\mathbf{x}) \leq 0, & j \in N_I = \{k+1, \ldots k+m\} \\ & \mathbf{x} \in X. \end{array} \right\}$$

Here, $f, g_1, \ldots, g_k,\ h_{k+1}, \ldots, h_{k+m}$ are twice continuously differentiable real valued functions defined on the *n-dimensional* space $\mathbb{R}^n$; X is a subset of $\mathbb{R}^n$; $\mathbf{x}$ is an $n-$dimensional vector of variables $x_1, \ldots, x_n$ and $N_{\mathcal{O}} = \{1, \ldots, k\}$ and $N_I = \{k+1, \ldots k+m\}$ are index sets. The aim is to determine a solution $\bar{\mathbf{x}}$ of (L) which satisfies the k equalities $g_i(\bar{\mathbf{x}}) = 0, i \in N_{\mathcal{O}}$ and the m inequalities $h_j(\bar{\mathbf{x}}) \leq 0,\ j \in N_I$

and that minimises (locally or globally) the function f over S. There is no fundamental difference between minimisation and maximisation problems. We can maximise f by minimising $-f$.

We call $f : \mathbb{R}^n \to \mathbb{R}$ the objective function and the functions $g_i(\mathbf{x})$, $i \in N_{\mathcal{O}}$ and $h_j(\mathbf{x})$, $j \in N_I$ are referred to as constraints. The description of the subset $X \subset \mathbb{R}^n$ includes in many cases lower and upper bounds for the variables.

A point $\mathbf{x} \in S$ is called a *feasible point.* The set of all feasible points forms the *feasible region* or *feasible set.* An inequality constraint $h_l(\mathbf{x}) \leq 0$ is said to be *active* at a feasible point $\bar{\mathbf{x}}$ if $h_l(\bar{\mathbf{x}}) = 0$, and *inactive* if $h_l(\bar{\mathbf{x}}) < 0$. All equality constraints $g_i(\mathbf{x}) = 0$ are regarded as active at any feasible point. An *active set* at a feasible point $\bar{\mathbf{x}}$ is defined as the set of those indices belonging to the constraints that are active at $\bar{\mathbf{x}}$.

For any non-linear programming problem, we distinguish two kinds of solution points: *local minimiser* and *global minimiser.*

Definition 3.1: *Given a point* $\mathbf{x} \in \mathbb{R}^n$ *and a scalar* $\varepsilon > 0$*, the ball* $B_\varepsilon(\mathbf{x}) := \{\mathbf{y} \in \mathbb{R}^n \mid \|\mathbf{y} - \mathbf{x}\| < \varepsilon\}$ *is called an* $\varepsilon -$ *neighbourhood of* $\mathbf{x}$*.*

A feasible point $\mathbf{x}^*$ *of* (L) *is said to be a local minimiser of* (L) *if there is an* $\varepsilon -$ *neighbourhood* $B_\varepsilon(\mathbf{x}^*)$ *of* $\mathbf{x}^*$ *such that* $f(\mathbf{x}) \geq f(\mathbf{x}^*)$ *for each* $\mathbf{x} \in S \cap B_\varepsilon(\mathbf{x}^*)$*. Similarly, if* $\mathbf{x}^* \in S$ *and if* $f(\mathbf{x}) > f(\mathbf{x}^*)$ *for all* $\mathbf{x} \in S \cap B_\varepsilon(\mathbf{x}^*)$, $\mathbf{x} \neq \mathbf{x}^*$ *for some* $\varepsilon > 0$*, then* $\mathbf{x}^*$ *is called a strict local minimiser of* (L)*.*

Definition 3.2: *A point* $\mathbf{x}^* \in S$ *is said to be a global minimiser of* (L) *if* $f(\mathbf{x}) \geq f(\mathbf{x}^*)$ *for all* $\mathbf{x} \in S$*. If* $f(\mathbf{x}) > f(\mathbf{x}^*)$ *for all* $\mathbf{x} \in S$, $\mathbf{x} \neq \mathbf{x}^*$ *then* $\mathbf{x}^*$*is said to be strict global minimiser of* (L)*.*

Definition 3.3: *A set $C \subset \mathbb{R}^n$ is said to be convex if, for any elements $\mathbf{y}^1$ and $\mathbf{y}^2$ of C and every real number α, $0 \leq \alpha \leq 1$, we have $\alpha \mathbf{y}^1 + (1-\alpha)\,\mathbf{y}^2 \in C$. In other words, if $\mathbf{y}^1$ and $\mathbf{y}^2$ belong to C, then the line segment connecting $\mathbf{y}^1$ and $\mathbf{y}^2$ is also in C.*

Definition 3.4: *A function $\ell : C \subset \mathbb{R}^n \rightarrow \mathbb{R}$ is convex on a convex set C if it satisfies*

$$\ell\left(\alpha \mathbf{y}^1 + (1-\alpha)\,\mathbf{y}^2\right) \leq \alpha \ell\left(\mathbf{y}^1\right) + (1-\alpha)\,\ell\left(\mathbf{y}^2\right)$$

for all $0 \leq \alpha \leq 1$ and for all $\mathbf{y}^1, \mathbf{y}^2 \in C$. In other words, the line segment connecting the points $(\mathbf{y}^1, \ell(\mathbf{y}^1))$ and $(\mathbf{y}^2, \ell(\mathbf{y}^2))$ lies on or above the graph of the function ℓ.

When f and each constraint $h_j(\mathbf{x})$, $j \in N_I$ are convex and the constraints $g_i(\mathbf{x})$, $i \in N_{\mathcal{O}}$ are linear, then the problem (L) is referred to as a convex programming problem and in this case each local minimiser is also a global minimiser. For a non-convex programming problem, we can only expect to obtain local minimisers. The necessary local optimality conditions for (L) are the well known *Karush-Kuhn-Tucker (KKT)* conditions [24].

For more information about non-linear programming theory we refer to [23, 69].

The routing and scheduling set of constraints that are presented in our mathematical model are mainly based on graph theory since we consider wireless networks as graphs. Therefore, we present here a brief introduction to graph theory.

3.2 Graph Theory

In this section we recall the concepts of graphs, directed and undirected graphs as well as edge and node colouring mappings.

A wireless network is a set consisting of devices and wireless links arranged so that a message can go from the source node to the sink node over multiple links and through multiple devices. Therefore, the *geometric structure* of a wireless network can be described by a graph, "which is a configuration consisting of points (nodes) in a space interconnected by a system of curves (edges)" [74]. The main components or elements of a graph are the set of nodes and the set of edges. In general, a wireless network can be described by a graph.

Graph theory is the study of *graphs* and covers a large area of topics. It is not our intention to cover all of its subjects in this thesis. For more information, refer to [36]. Graph theory, provides simple, accessible and powerful tools for modelling and solving problems that are related to a discrete arrangements of objects. These problems range from graph colouring, critical-path analysis, communication systems, optimal routing and network flows, organic-chemical identification, games and puzzles, social group structures, etc. (refer to [68]). In this work we will consider two of them: graph colouring and network flows.

There exist two main groups of graphs; undirected graphs usually called just graphs and directed graphs or digraphs. An undirected graph G is an ordered pair $G = (V, A)$ where V is a nonempty set of nodes and A is a set of (unordered) pairs of distinct nodes, called edges. While a digraph is an ordered pair $G = (V, E)$, where V is a nonempty set of nodes, $E \subseteq \{e = (u, v) \in V \times V \mid u \neq v\}$ is a set of ordered pairs of distinct nodes called edges. For instance, the pair of ordered nodes (u, v) represents the edge e from the node u to the node v. Moreover, if the edge $e = (u, v)$ exists, then the nodes u and v are called adjacent. An ad hoc wireless network can be represented by a digraph .

A graph colouring is an assignation of colours to elements of a graph or digraph (nodes and edges). This assignation of colours is subject to a set of constraints.

Definition 3.5: *Node Colouring.*

Let $C = \{\alpha_1, \ldots, \alpha_k\}$ be the set of k available colours. A node-colouring *of a digraph $G = (V, E)$ is a mapping $Co_v : V \to C$ such that $Co_v(v) \neq Co_v(u)$ with $(u, v) \in E$ [36].*

Definition 3.6: *Edge Colouring*

An edge-colouring of a digraph $G = (V, E)$ is a mapping

$$Co_e : (u, v) \in E \to Co_e(u, v) := Co_v(u),$$

where Co_v is a corresponding node-colouring.

Some examples of node and edge coloured graphs are shown in Figure 3.1.

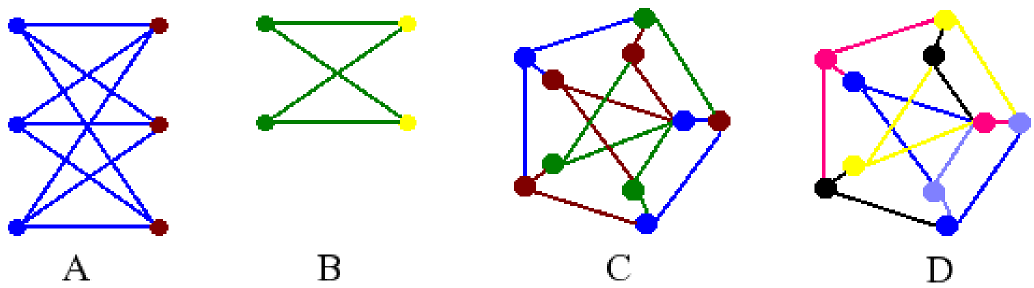

Figure 3.1: Examples of coloured graphs.

Since we will also consider the problem of sending more than one message through an ad hoc wireless network, we give a brief explanation of the idea of network flow and single and multi-commodity flow problems.

3.3 Network Flow

As we mentioned before, network flow is a research area of graph theory. If we assign to each edge of a digraph the flow of a (hypothetical) substance such as: oil, water, gas, data, etc; then the structure of the graph becomes a useful model known as a network flow problem. Network flow problems are focused on determining the network connectivity, the transportation or any other concept concerned with the movement of commodities, information or people. One important characteristic of a network is that the flow must be conserved i.e. the incoming flow into a node must be equal to the outgoing flow. The flow that can go through is limited since links have a limited "capacity". One of the most important constraints in network flow problems is the link capacity, since the maximum flow over the network depends upon it. If there is a single homogeneous substance (e.g. message or commodity) to be sent along the network, we refer to this problem as a single-commodity problem. But, if more than one commodity going along the network exists, it is considered as a multi-commodity network flow problem. For more details refer to [67, 70].

The single or multi-commodity network flow problem is embedded as part of the set of constraints of our mathematical models presented in Chapters 4 and 5.

Generally, the solution of a mathematical problem requires the use of appropriate software. Therefore, in the next section we present a modelling language.

3.4 A Modelling Language for Mathematical Programming AMPL

In this section we introduce a modelling language that is used for "translating" the mathematical model into a suitable set of computer-compatible instructions.

Consider an ad hoc wireless network consisting of 6 devices and 8 edges and the mathematical model presented in Chapter 4. This mathematical model has 1438 variables and 1172 constraints and, therefore, a corresponding software is needed for its solution.

In the late 1940s, the first computational method for solving linear programming problems was proposed [34]. Since then many other computational methods for solving linear, non-linear, integer programming, quadratic problems, etc. have been developed. The "translation" of the mathematical problem into an "understandable" code for the computational routines was developed primarily in languages like FORTRAN or C. [12]

Nowadays, it is possible to find different computer languages for describing production, scheduling, distribution and many other kind of problems generally known as large scale optimisation or mathematical programming such as AMPL [42], GAMS [4], LINDO [5] among others. The aim of these languages is to provide an interface between the mathematical model and the different computational routines or solvers, i.e. these languages will "translate" the mathematical model into a set of understandable sentences for the solver.

For this work, AMPL seems to be the best option due to the fact that it can interact with a large number of different non-linear solvers. For more details about AMPL refer to [42].

AMPL is a comprehensive, powerful and flexible algebraic modelling language for linear, non-linear and integer programming problems. It was developed at Bell Laboratories [42]. By using AMPL's algebraic notation, even very large and complex models can be written in a concise and understandable form.

Here are some features of AMPL software: [1]

- Allows users to express models concisely in familiar, flexible algebraic notation.
- Uses indexing and sets, whole classes of constraints or objective terms can be described in relatively few generalised model statements.
- Supports a wide variety of natural algebraic expressions.
- Allows separation of model and data.
- Is highly flexible. For example, does not impose any particular data or model entry order.
- Allows models to be easily modified and maintained:
  - Sets, variables and constraints can be readily added, changed or deleted.
  - Separate data files can be readily updated, or imported from outside data sources.
- Minimises modelling errors with extensive error checking:
  - Checks model syntax and consistency.
  - Closely checks validity of subscripts.
- Covers most optimisation model types like:

  - Linear programming problems.
  - Network problems.
  - Mixed integer programming problems.
  - Quadratic programming problems.
  - General non-linear programming problems.

- Provides powerful, flexible display options:
  - Permits interactive data and solution browsing.
  - Allows the creation of customised solution reporting.

We will consider the AMPL model for the following linear programming problem.

Example 3.1: The Company ACME makes chairs and tables. The chairs are sold for £10 per unit, while the tables are sold for £15 per unit. The company has one factory and can produce either a chair or a table at the same time. The number of chairs that can be produced per hour is 40, and the number of tables that can be produced per hour is 30. The company knows that they cannot sell more than 480 tables and 800 chairs per week. Assume that each week has 40 working hours and that tables and chairs cannot be stored until next week. The company wants to determine the number of tables and chairs to be produced so that the total sale is maximised. Let x_1 and x_2 represent the number of chairs and tables, respectively, to be produced in one week. Then, we have the following mathematical model which represents this problem:

$max\ 10x_1 + 15x_2$

$s.t.\ :$

$\frac{1}{40}x_1 + \frac{1}{30}x_2 \leq 40$

$0 \leq x_1 \leq 800$

$0 \leq x_2 \leq 480$

The "translation" of the LP mathematical model into an AMPL model is:

var x_1*; # number of chairs*

var x_2*; # number of tables*

maximise sale: $10 * x_1 + 15 * x_2$;

subject to time: $(1/40) * x_1 + (1/30) * x_2 <= 40$*;*

subject to chair_limit: $0 <= x_1 <= 800$*;*

subject to table_limit:$0 <= x_2 <= 480$;

For more information about AMPL's language notation, refer to [42]:

3.5 Solvers

Most algorithms designed to solve large optimisation problems are based on an iterative process. Such an algorithm may be thought of as a method which generates a series of points, each point dependent upon the points preceding it. It is desirable for the sequence generated by the algorithm to converge to a global or local minimiser. Indeed, many algorithms for non-linear programming problems approximate only local minimisers, specially if the non-linear programming problem is non-convex. Therefore, many algorithms are designed to find a local minimiser i.e. a feasible point at which the objective function value is as good or better than at any "nearby" feasible point.

Many algorithms make use of different tools and concepts from calculus [62, 69] such as: gradient, Hessian, Jacobian; and from linear algebra [60, 79] like: matrix theory, vector spaces, etc.

Some "free of charge" solvers exist on the Internet such as Ipopt[88], Conopt[2], Filter[40], Lancelot[32], Minos[6], Mosek[8], Pennon[59], Loqo[84] and more. For the mathematical models in Chapter 4 and 5, we used Knitro since we obtained a non-restricted copy of Knitro thanks to Professor Robert Fourer, professor of the Northwester University U.S.A, which gave us the opportunity to use AMPL and Knitro for free for a short period of time. We will give a brief description of Knitro and Ipopt since both are used in Chapter 5. We tried to solve the mathematical model presented in Chapter 5 with almost all the solvers mentioned before, but only Knitro and IPOPT were able to find a local minimiser. Furthermore, Knitro and IPOPT found slightly different solutions to our models.

Knitro [30] is based on an interior point algorithm [29, 57, 64] for non-linear programming problems that combines line search and trust region steps. Knitro is a software package for finding solutions to continuous, smooth optimisation problems, with or without constraints. Knitro is designed for finding local minimisers, but multi-start heuristics are provided for trying to locate global minimisers. Although Knitro is designed for solving large-scale general non-linear problems, it is efficient at solving all of the following classes of smooth optimisation problems:

- Bound constrained.
- Equality constrained, both linear and non-linear.
- General non-linear constrained problems, both convex and non-convex.

As we will see in the mathematical models presented in Chapters 4 and 5, Knitro is a good option to be considered as a solver in this context since it considers the classes of smooth optimisation problems and it is possible to use AMPL with Knitro.

Ipopt [88] (Interior Point Optimiser, pronounced “I-P-Opt”) is an open source software package for large scale non-linear programming problems. Ipopt implements an interior point line search filter method [29] that aims to find a local minimiser of non-linear programming problems. It provides an option which activates heuristics that may speed up the infeasibility determination if it is expected that there is a chance for the problem to become unfeasible. Ipopt is written in FORTRAN and C++ with an open-source license and includes the source code. This means, it is available on the Internet free of charge.

The main disadvantage of AMPL is that the free available version found on the Internet [7] is restricted to 300 variables or 300 constraints for any optimisation problem. At the beginning of Section 3.4, we mentioned that for a “small” network topology we have a “large” non-linear programming problem. Therefore, the free AMPL version is not adequate for our problems. Fortunately, the Optimisation Technology Centre (OTC) [13] provides the option to solve, among others, non-linear programming problems free of charge: “The main purpose of OTC is to do numerical optimisation research, Internet and distributed computing, problem-solving environments, and the study of optimisation in a wide range of applications.” [13]

The OTC has four different projects: the Network Enabled Optimisation System (NEOS) Server [10], NEOS Guide [9], OTC software [15] and OTC research [14]. NEOS Server is capable of solving optimisation problems automatically over the Internet. NEOS provides an easy access to all optimisation solvers available with

NEOS server and can work either with GAMS [4] or AMPL [42] modelling languages as the interface for the non-linear programming problem.

Since our optimisation programming problem has a large number of constraints and variables, we found that NEOS is an appropriate way to solve the optimisation problem considering that NEOS accepts AMPL as a modelling language.

3.6 Summary

In this chapter, we have presented the basic mathematical concepts that are used in the modelling of an ad hoc wireless network system in Chapter 4 and Chapter 5. Furthermore, we have given a brief introduction to non-linear programming problems, the basic concepts of AMPL and the solvers that are going to be used to solve our non-linear programming problems.

Chapter 4

First Mathematical Modelling Approach

In this chapter, we state the first of two mathematical models to be presented in this thesis that represents an ad hoc wireless network. First, we describe an ad hoc wireless network by a directed graph and edge-colouring scheduling. Second, we define the set of decision variables, the objective function and the constraints, which in turn represent the characteristic of the ad hoc wireless network. The so obtained non-linear programming problem will be denoted by Θ. Third, we present two theorems where we prove that if there exists a local minimiser for problem Θ, then the set of constraints (4.14) or (4.16) is active. Fourth, the methodology used to solve the mathematical model is presented. Fifth, the solutions for different ad hoc network topologies and different parameters are calculated. We recall Chapters 2 and 3 where the corresponding technical properties and mathematical concepts are discussed, respectively.

This mathematical model has to meet the following requirements:

- Optimal use of battery: The nodes are powered by batteries, whose electrical charges have a finite life. Therefore, it is important to make an efficient use of batteries.

- Guarantee a minimum level of SNR ratio target for each link in the network.
- Guarantee a QoS that provides a good performance level for real time applications such as video or audio.

In the mathematical model, we will consider joint routing, scheduling and power control for TDMA/CDMA systems.

4.1 Mathematical Model

In this section we present the set of variables, the set of constraints and the objective function for the mathematical model of an ad hoc wireless network for CDMA/TDMA systems.

Let us examine an ad hoc wireless network consisting of r devices. These devices can communicate with each other via a wireless link. Without loss of generality, consider a message originated at a particular source device and destined to arrive at a particular sink device. The rest of the devices are considered as intermediate devices. The intermediate devices will be source devices or sink devices according to their function (sending or receiving mode) at a given time. The link connectivity between devices is given in Definition 2.1. The message is sent from the origin source device through the network until it reaches its destination via an unknown path. The path is set by the connectivity and link transmission power and capacity. In this scheme there are three possibilities:

- The whole message is sent as a single piece.

- The message remains in the buffer (refer to Section 2.4) of the device either because the device is in receiving mode or the capacity of the out going links are full, until it can be sent, (TDMA property).
- The message is split into smaller pieces, depending on the links capacity. Then, each piece could be considered as a smaller message, which can be sent as a single piece, remain in the buffer or be split into smaller pieces.

Independently of how it is sent, the complete message will arrive at its destination integrated as a whole.

The behaviour of the message going through the network will be mathematically described by a set of constraints, decision variables and lower and upper bounds. Each device will be assigned a node and each link an edge.

Let us consider an ad hoc wireless network consisting of r devices. Therefore, the set of nodes will be $V = \{v_1, \ldots, v_r\}$ and the set of links E as defined in Definition 2.1. The ad hoc wireless network, then can be represented by a digraph which is an ordered pair $G = (V, E)$ defined by the set of nodes V and the set of edges E. For the set of edges E, we define two subsets E^+ and E^- as follows:

For any node, $v \in V$, let us denote by

$$E^+ (v) := \{e \in E | \exists u \in V : e = (v, u)\}$$

the set of edges that leaves the node v, and by

$$E^- (v) := \{e \in E | \exists u \in V : e = (u, v)\}$$

the set of edges that enters the node v, i.e. $E^+ (v)$ is the set of outgoing edges from node v, and $E^{-}(v)$ is the set of incoming edges to node v.

One can either send a message from the source node to the sink node by a single hop, (in case that there is a direct edge from source node to sink node), or by consecutive multi-hops. Also, since an ad hoc wireless network consists of several devices, it is possible for more than one message to go through the network (CDMA) at the same time. Let us define the set of messages $M = \{m_1, \dots, m_L\}$, where L is the number of messages to be sent along the network and let $z_{m_i} \in \mathbb{R}_+,\ i = 1, \dots, L$ be the size of the message m_i. For each message $m_i \in M$, $i = 1, \dots, L$ there exist a source node $s_{m_i} \in V$ and a sink node $d_{m_i} \in V$ with $s_{m_i} \neq d_{m_i}$ exist. Since there are different senders and receivers in the network, we face a multi-commodity flow problem (refer to Section 3.3).

As mentioned in Chapter 1, we consider the use of TDMA as the access method system for the ad hoc wireless network. The time interval under consideration is equally divided into transmission slots. Define the set of consecutive time slots as $T = \left\{t_1, \dots, t_{t^{Max}}\right\}$, where $t_i \in \mathbb{R}_+,\ i \in 1, \dots, t^{Max}$ is a time interval whose end point is the starting point t_{i+1}.

Given a set of colours $C = \{\alpha_1, \dots, \alpha_k\}$, where k is the number of available colours, assign to each node $v \in V$ a colour according to Definition 3.5 and define

$$V_{\alpha_i} = \{v \in V : \ Co_v(v) = \alpha_i\}$$

for $\alpha_i \in C$. In particular k is chosen in such a way that

$$V = \bigcup_{i=1}^{k} V_{\alpha_i}.$$

Furthermore, define the edge-colouring mapping according to Definition 3.6 and the time interval colouring mapping as

$$Co_t : T \to C$$

by

$$Co_t : t_i \in T \mapsto Co_t\left(t_i\right) = \alpha_j \quad and\, j = i\%k$$

(i.e. $Co_t\left(t_i\right) = \alpha_i,\ Co_t\left(t_{k+i}\right) = \alpha_i,\ i = 1, \ldots, k$ etc.).

4.1.1 Design Variables

To model the ad hoc wireless network, we define now the decision variables involved and their units of measurement, and the set of constraints. Unless specified otherwise, the indices e, m, t, v vary always in the sets E, M, T, V, respectively, in the remaining of this thesis.

As we mentioned in Section 4.1, a message m to be sent through an ad hoc wireless network may travel as a complete message or it may be split. The size of the part of that message m going through the edge e in time slot t is denoted by the decision variable $c_{e,m,t} \in \mathbb{R}_*$, (measured in bits).

The size of that part of the message m that is stored in the buffer of node v, at the beginning of the time slot t is denoted by $b_{v,m,t} \in \mathbb{R}_*$, (measured in Kbits). Moreover, at the beginning of the first time slot t_1 we have all messages m_i are stored in the buffer of the source nodes s_{m_i}. Nodes that are not source nodes have an empty buffer at the beginning of t_1. This is described by the following set of equations.

$$b_{v,m_i,t_1} = \begin{cases} z_{m_i} & if\ v = s_{m_i} \\ 0 & else \end{cases} \tag{4.1}$$

The transmission power allocated at the edge e at the time slot t is denoted by $p_{e,t} \in \mathbb{R}_*$, (measured in watts or dBm).

4.1.2 Parameters

In this subsection we present the parameters and their units of measurement.

The size of the buffer at node v is defined by $B_v^{Max} \in \mathbb{R}_+$(measured in Kbits).

We consider two maximum transmission powers: The first is the maximum transmission power available for all edges defined by $P^{Max} \in \mathbb{R}_+$ (measured in watts or dBm). The second is the maximum transmission power for each node v defined by $P_v^{Max} \in \mathbb{R}_+$ (measured in watts or dBm).

Each message m is sent through the ad hoc wireless network from the source node $s_m \in V$ to the sink $d_m \in V$ and stored at the sink buffer. The starting time slot $t_{j(m)} \in T$ is the specific time slot at which the message is released (made available) to the user and $R_{m_i} > 0$ is the average rate at which the message is released (measured Kbits/time slot), from the starting time slot $t_{j(m)} \in T$ onwards until the complete message is received by the user.

The maximum number of time slots t^{Max} needed for the complete reception of all messages is given by:

$$t^{Max} = \max \left\{ t_{j(m_i)} + \left\lceil \frac{z_{m_i}}{R_{m_i}} \right\rceil |m_i \in M, \right\}.$$

We use two parameter values in the set of constraints (4.14) and (4.16). First, $B \in \mathbb{R}_+$ is the bandwidth of the communication channel or the width of the range of frequencies that an electronic signal uses on a given transmission medium usually measured in hertz [18] described in Section 2.4. Second, $\eta \in \mathbb{R}_+$ is the background noise energy described in Subsection 2.3.1, (measured in watts or dBm).

4.1.3 Set of Constraints

In this subsection we present the upper bounds for the decision variables and the set of constraints for the mathematical model.

4.1.3.1 Upper Bounds for the Decision Variables

The transmission power $p_{e,t}$ cannot exceed the maximum power P^{Max},

$$p_{e,t} \leq P^{Max} \qquad \forall t \in T. \tag{4.2}$$

For any node v and for any time slot t, the sum of all transmission powers $p_{e,t}$ where e is varying in $E^+(v)$ cannot exceed the maximum power P_v^{Max}:

$$\sum_{e \in E^+(v)} p_{e,t} \leq P_v^{Max} \qquad \forall t \in T. \tag{4.3}$$

The sum of the sizes of those parts of the messages stored at the beginning of the time slot t in the buffer of v cannot exceed the size of this buffer:

$$\sum_{m \in M} b_{v,m,t} \leq B_v^{Max} \qquad \forall t \in T. \tag{4.4}$$

The capacity of a channel (edge) is given by Shannon's [26, 81] well-known formula given in (2.9) which implies that the sum of all sizes of the parts of the messages $c_{e,m,t}$ cannot exceed the maximum capacity of the channel

$$\sum_{m \in M} c_{e,m,t} \leq B \log_2 (1 + SR_t) \qquad \forall t \in T. \tag{4.5}$$

4.1.3.2 Constraints of the System

One of the main characteristics of TDMA system is that nodes are allowed to send information through their outgoing edges at a given time slot. As we mentioned in Section 2.4, this process is repeated cyclically and it is considered in this mathematical model as the scheduling process for the ad hoc wireless network. The following constraint describes the scheduling process by assigning the value zero to the decision variable $c_{e,m,t}$ when the edge colouring $Co_e(e)$ for the edge e is different from time colouring $Co_t(t)$:

$$c_{e,m,t} = 0 \qquad (m \in M, Co_e(e) \neq Co_t(t)). \tag{4.6}$$

In other words, $c_{e,m,t} \neq 0$ implies that the edge e has the same colour as the time slot t.

The relationship between the size b_{v,m,t_l} and $b_{v,m,t_{l+1}}$ $\left(l = 1, \ldots t^{Max} - 1\right)$ of those parts of message m stored in the buffer of v at the beginning of time slots t_l and t_{l+1}

respectively and the corresponding ingoing and outgoing message in v is represented by:

$$\begin{aligned} b_{v,m,t_{l+1}} - b_{v,m,t_l} &= \\ b_{v,m,t_1} + \sum_{e\in E^-(v)} \sum_{i=1}^{l} c_{e,m,t_i} - \sum_{e\in E^+(v)} \sum_{i=1}^{l} c_{e,m,t_i} &- \\ - \left[b_{v,m,t_1} + \sum_{e\in E^-(v)} \sum_{i=1}^{l-1} c_{e,m,t_i} - \sum_{e\in E^+(v)} \sum_{i=1}^{l-1} c_{e,m,t_i} \right] & \\ = \sum_{e\in E^-(v)} c_{e,m,t_l} - \sum_{e\in E^+(v)} c_{e,m,t_l} & \end{aligned} \tag{4.7}$$

At the sink node d_m, the message m will be stored in its buffer and will be released from it to the user at a constant reduction rate R_m, from the starting time slot $t_{j(m)} \in T$ onwards until the complete message is released. This is described by the following set of constraints

$$\sum_{i=1}^{t_{j(m)}+\iota} \sum_{e\in E^-(d_m)} c_{e,m,t_i} \;\geq\; (\iota+1)\, R_m \;\; (\iota = 0, \ldots, \lfloor Z_m/R_m \rfloor - 1)\,. \tag{4.8}$$

4.1.3.3 Non-Negativity Constraints

The following constraints refer to the non-negativity of the variables.

$$p_{e,t} \geq 0 \tag{4.9}$$

$$c_{e,m,t} \geq 0 \tag{4.10}$$

$$b_{v,m,t} \geq 0. \tag{4.11}$$

4.1.4 The Non-Linear Programming Problem

The objective function is to minimise the total power consumption in the system, with respect to all variables $p_{e,t}, c_{e,m,t}, b_{v,m,t}$; it is defined as follows:

$$f\left(p_{e,t}, c_{e,m,t}, b_{v,m,t}\right) = \sum_{e \in E} \sum_{t \in T} p_{e,t}. \tag{4.12}$$

Note that this function is monotonically increasing with respect to $\left(p_{e,t}\right)_{e \in E, t \in T}$.

As a matter of simplicity, we define the vectors $\mathbf{p} \in \mathbb{R}^{|E|*|T|}$, $\mathbf{c} \in \mathbb{R}^{|E|*|M|*|T|}$, $\mathbf{b} \in \mathbb{R}^{|V|*|M|*|T|}$ whose components are $p_{e,t},\ c_{e,m,t},\ b_{v,m,t}$:

$$\mathbf{p} = \left(p_{e,t}\right)_{e \in E, t \in T}$$
$$\mathbf{c} = \left(c_{e,m,t}\right)_{e \in E, m \in M, t \in T}$$
$$\mathbf{b} = \left(b_{v,m,t}\right)_{v \in V, m \in M, t \in T}.$$

Furthermore, $\mathbf{p} \geq \mathbf{0}$ means that $p_{e,t} \geq 0$ for all $e \in E, t \in T$. For any two vectors $\mathbf{p}$ and $\mathbf{p}'$ we say that $\mathbf{p} \geq \mathbf{p}'$ if and only if $p_{e,t} \geq p'_{e,t}$ for all $e \in E, t \in T$. The strict inequality $>$ is used in an analogous manner.

Now, we can define the non-linear programming problem Θ:

$$\text{minimise } f\left(\mathbf{p}, \mathbf{c}, \mathbf{b}\right) = \sum_{e \in E} \sum_{t \in T} p_{e,t}$$

subject to (4.1)-(4.11).

Denote the feasible set of Θ by S. Obviously, S is a subset of $\mathbb{R}^{|E|*|T|+|E|*|M|*|T|+|V|*|M|*|T|}$.

4.2 Theoretical Contributions

In this section we prove two theorems which are the theoretical contributions to this thesis. These theorems state that the set of constraints (4.5) is active at every local minimiser of the non-linear programming problem Θ. This implies that the link will be used at its full capacity at the same time that the transmission power is minimised.

We will introduce a new index notation for the SNR_e given in Subsection 2.3.2 to include time.

In the first theorem we analyse the signal to noise ratio defined in Subsection 2.3.2 and replace SR in (4.5) by $SNR_{e,t}$

$$SNR_{e,t} := \frac{p_{e,t}}{\eta^2}. \tag{4.13}$$

Therefore, the upper bound (4.5) for $c_{e,m,t}$ is given by:

$$\sum_{m \in M} c_{e,m,t} \leq B \log_2 \left(1 + \frac{p_{e,t}}{\eta^2} \right). \tag{4.14}$$

In the second theorem we analyse the signal to interference noise ratio defined in Subsection 2.3.2 and replace SR in (4.5) by $SINR_{e,t}$:

$$SINR_{e,t} := \frac{p_{e,t}}{\sum_{f \neq e} p_{f,t} + \eta^2}, \tag{4.15}$$

hence (4.5) is transformed into:

$$\sum_{m \in M} c_{e,m,t} \leq B \log_2 \left(1 + \frac{p_{e,t}}{\sum_{f \neq e} \rho_f p_{f,t} + \eta^2} \right). \tag{4.16}$$

Note that (4.14) and (4.16) imply a coupling condition between $c_{e,m,t}$ and $p_{e,t}$ and in particular, $p_{e,t} = 0$ implies $c_{e,m,t} = 0$. For the purpose of this thesis, the orthogonality between codes are all equal to one i.e. $\rho_f = 1, f \in E$.

4.2.1 First Theorem

Theorem 4.1: *Let* $\left(\bar{\mathbf{p}}, \bar{\mathbf{c}}, \bar{\mathbf{b}}\right)$ *be a local minimiser of the problem* Θ. *Then, the constraints (4.14) are active at* $\left(\bar{\mathbf{p}}, \bar{\mathbf{c}}, \bar{\mathbf{b}}\right)$ *i.e.*

$$\sum_{m \in M} \bar{c}_{e,m,t} = B \log_2 \left(1 + \frac{\bar{p}_{e,t}}{\eta^2} \right)$$

Proof: Suppose that there exist a local minimiser $\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right)$ of Θ and indices $\bar{e} \in E, \bar{t} \in T$ such that:

$$\sum_{m \in M} \check{c}_{\bar{e},m,\bar{t}} < B \log_2 \left(1 + \frac{\check{p}_{\bar{e},\bar{t}}}{\eta^2} \right). \tag{4.17}$$

As $\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right)$ is a local minimiser of Θ there exists an $\bar{\varepsilon} > 0$ such that

$$f\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right) \leq f\left(\mathbf{p}, \mathbf{c}, \mathbf{b}\right) \qquad \forall \left(\mathbf{p}, \mathbf{c}, \mathbf{b}\right) \in S \cap B_{\bar{\varepsilon}}\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right). \tag{4.18}$$

From (4.17) we obtain:

$$\frac{\sum_{m\in M}\check{c}_{\bar{e},m,\bar{t}}}{B} < \log_2\left(1+\frac{\check{p}_{\bar{e},\bar{t}}}{\eta^2}\right)$$

which implies

$$2^{\frac{\sum_{m\in M}\check{c}_{\bar{e},m,\bar{t}}}{B}} < 1+\frac{\check{p}_{\bar{e},\bar{t}}}{\eta^2},$$

and, furthermore,

$$\check{p}_{\bar{e},\bar{t}} > \left(2^{\frac{\sum_{m\in M}\check{c}_{\bar{e},m,\bar{t}}}{B}} - 1\right)\eta^2. \tag{4.19}$$

Let us define the right hand side of (4.19) as :

$$r := \left(2^{\frac{\sum_{m\in M}\check{c}_{\bar{e},m,\bar{t}}}{B}} - 1\right)\eta^2. \tag{4.20}$$

By (4.10) and (4.20), we have that $r \geq 0$.

Thus, (4.19) can be written as

$$\check{p}_{\bar{e},\bar{t}} - r > 0 \tag{4.21}$$

which implies

$$\check{p}_{\bar{e},\bar{t}} > 0.$$

Define

$$\varepsilon := \check{p}_{\bar{e},\bar{t}} - r. \tag{4.22}$$

Choose a $\lambda \in (0,1)$ with $\lambda\varepsilon < \bar{\varepsilon}$ and define the feasible point $\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right)$ as:

$$\begin{aligned} \widehat{\mathbf{c}} &= \check{\mathbf{c}} \\ \widehat{\mathbf{b}} &= \check{\mathbf{b}} \\ \widehat{p}_{e,t} &= \begin{cases} \check{p}_{e,t} & \text{if } (e,t) \neq \left(\bar{e}, \bar{t}\right) \\ \check{p}_{\bar{e},\bar{t}} - \lambda\varepsilon & \text{if } (e,t) = \left(\bar{e}, \bar{t}\right). \end{cases} \end{aligned} \tag{4.23}$$

Obviously, $\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right)$ is a small perturbation of $\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right)$ and, by construction, we have

$$\widehat{p}_{\bar{e},\bar{t}} > 0.$$

From (4.21), (4.22) and (4.23) we have that

$$\widehat{p}_{\bar{e},\bar{t}} > r. \tag{4.24}$$

Now, from (4.20) and (4.24) we have that

$$\widehat{p}_{\bar{e},\bar{t}} > \left(2^{\frac{\sum_{m\in M} \check{c}_{\bar{e},m,\bar{t}}}{B}} - 1\right)\eta^2.$$

Since $0 \leq r < \widehat{p}_{\bar{e},\bar{t}} < \check{p}_{\bar{e},\bar{t}}$, the point $\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right)$ fulfils all the constraints (4.1)- (4.11) i.e. it is a feasible point of the problem Θ.

The value of f at $\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right)$ is

$$\sum_{e\in E}\sum_{t\in T} \widehat{p}_{e,t} = \sum_{e\in E}\sum_{t\in T} \check{p}_{e,t} - \lambda\varepsilon$$

and since $\lambda\varepsilon > 0$ we have that

$$f\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right) = \sum_{e\in E}\sum_{t\in T} \widehat{p}_{e,t} < \sum_{e\in E}\sum_{t\in T} \check{p}_{e,t} = f\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right).$$

However, the latter inequality contradicts (4.18). Therefore, $\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right)$ is not a local minimiser of Θ, contradicting our initial assumption and, therefore, the constraints (4.14) have to be active. This completes the proof of Theorem 4.1. ■

4.2.2 Second Theorem

Based on the definition of $SINR_{e,t}$ in (4.15) we will prove that at every local minimiser, the set of constraints (4.16) for the non-linear programming problem Θ is active.

Theorem 4.2: *Let* $\left(\bar{\mathbf{p}}, \bar{\mathbf{c}}, \bar{\mathbf{b}}\right)$ *be a local minimiser of the problem* Θ*. Then, the constraints (4.16) are active at* $\left(\bar{\mathbf{p}}, \bar{\mathbf{c}}, \bar{\mathbf{b}}\right)$ *i.e.*

$$\sum_{m\in M} \bar{c}_{e,m,t} = B\log_2\left(1 + \frac{\bar{p}_{e,t}}{\sum_{f\neq e} \bar{p}_{f,t} + \eta^2}\right)$$

Proof: The proof of this theorem runs in an analogous manner as the proof of Theorem 4.1. Therefore, we will present only the main steps.

Suppose that there exist a local minimiser $\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right)$ of Θ and indices $\bar{e} \in E, \bar{t} \in T$ such that:

$$\sum_{m\in M} \check{c}_{\bar{e},m,\bar{t}} < B\log_2\left(1 + \frac{\check{p}_{\bar{e},\bar{t}}}{\sum_{f\neq \bar{e}} \check{p}_{f,\bar{t}} + \eta^2}\right). \tag{4.25}$$

As $\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right)$ is a local minimiser of Θ there exists an $\bar{\varepsilon} > 0$ such that

$$f\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right) \leq f\left(\mathbf{p}, \mathbf{c}, \mathbf{b}\right) \qquad \forall \left(\mathbf{p}, \mathbf{c}, \mathbf{b}\right) \in S \cap B_{\bar{\varepsilon}}\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right). \tag{4.26}$$

Analogously to (4.19), we obtain from (4.25) that

$$\check{p}_{\bar{e},\bar{t}} > \left(2^{\frac{\sum_{m\in M} \check{c}_{\bar{e},m,\bar{t}}}{B}} - 1\right)\left(\sum_{f\neq\bar{e}} \check{p}_{f,\bar{t}} + \eta^2\right). \tag{4.27}$$

Let us define the right hand side of (4.27) as :

$$r := \left(2^{\frac{\sum_{m\in M} \check{c}_{\bar{e},m,\bar{t}}}{B}} - 1\right)\left(\sum_{f\neq\bar{e}} \check{p}_{f,\bar{t}} + \eta^2\right). \tag{4.28}$$

By (4.9), (4.10) and (4.28), we have that $r \geq 0$.

Thus, (4.27) can be written as

$$\check{p}_{\bar{e},\bar{t}} - r > 0 \tag{4.29}$$

which implies

$$\check{p}_{\bar{e},\bar{t}} > 0.$$

Define

$$\varepsilon := \check{p}_{\bar{e},\bar{t}} - r. \tag{4.30}$$

Choose a $\lambda \in (0,1)$ with $\lambda\varepsilon < \bar{\varepsilon}$ and define the feasible point $\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right)$ as:

$$\begin{gathered}
\widehat{\mathbf{c}} = \check{\mathbf{c}} \\
\widehat{\mathbf{b}} = \check{\mathbf{b}} \\
\widehat{p}_{e,t} = \begin{cases} \check{p}_{e,t} & \text{if } (e,t) \neq \left(\bar{e}, \bar{t}\right) \\ \check{p}_{\bar{e},\bar{t}} - \lambda\varepsilon & \text{if } (e,t) = \left(\bar{e}, \bar{t}\right). \end{cases}
\end{gathered} \tag{4.31}$$

Analogously to proof of Theorem 4.1, it follows from (4.31) that

$$\widehat{p}_{\bar{e},\bar{t}} > 0,$$

$$\widehat{p}_{\bar{e},\bar{t}} > r \tag{4.32}$$

and the latter inequality and (4.28) imply that

$$\widehat{p}_{\bar{e},\bar{t}} > \left(2^{\frac{\sum_{m\in M} \check{c}_{\bar{e},m,\bar{t}}}{B}} - 1\right)\left(\sum_{f\neq\bar{e}} \check{p}_{f,\bar{t}} + \eta^2\right).$$

By construction $\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right)$ is a feasible point of the problem Θ and the value of f at $\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right)$ is

$$\sum_{e\in E}\sum_{t\in T} \widehat{p}_{e,t} = \sum_{e\in E}\sum_{t\in T} \check{p}_{e,t} - \lambda\varepsilon$$

and, by $\lambda\varepsilon > 0$, we have that

$$f\left(\widehat{\mathbf{p}}, \widehat{\mathbf{c}}, \widehat{\mathbf{b}}\right) = \sum_{e\in E}\sum_{t\in T} \widehat{p}_{e,t} < \sum_{e\in E}\sum_{t\in T} \check{p}_{e,t} = f\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right).$$

The latter inequality contradicts (4.26) and, hence, $\left(\check{\mathbf{p}}, \check{\mathbf{c}}, \check{\mathbf{b}}\right)$ is not a local minimiser of Θ, contradicting our initial assumption. Therefore, the constraints (4.16) have to be active i.e.

$$\sum_{m \in M} \bar{c}_{e,m,t} = B \log_2 \left(1 + \frac{\bar{p}_{e,t}}{\sum\limits_{f \neq e} \bar{p}_{f,t} + \eta^2} \right) \tag{4.33}$$

This completes the proof of Theorem 4.2. ■

We have shown that the set of constraints (4.16) is active at each local minimiser of Θ, which means that the total capacity of the channel is used with (local) minimum power.

4.3 Methodology for the Numerical Solutions

Our main goal is to solve the joint optimisation problem of routing, time-scheduling and power control, in order to minimise the power used by each node (device) in the network given by the optimisation problem Θ:

$$\text{minimise } f\left(\mathbf{p}, \mathbf{c}, \mathbf{b}\right) = \sum_{e \in E} \sum_{t \in T} p_{e,t}$$

subject to (4.1)-(4.11).

If we replace SR in (4.5) by $SINR_{e,t}$ (given in (4.15)), then Theorem 4.2 enables us to rewrite this problem as $\widehat{\Theta}$:

$$\text{minimise } f\left(\mathbf{p}, \mathbf{c}, \mathbf{b}\right) = \sum_{e \in E} \sum_{t \in T} p_{e,t}$$

subject to (4.1)-(4.4),(4.6)-(4.11) and (4.33).

In this section we will propose a methodology for the numerical solution of problem $\widehat{\Theta}$ where we embed time scheduling into the network topology.

Note that any example given in this section is related to the concepts and definitions given in previous sections.

Recall that:

- Digraph and coloured graph are defined in Section 3.2.
- Node, edge and time slot colouring maps are defined in Section 4.1.

The number of time slots $\tilde{t}_m$ needed to receive the whole message $m \in M$ depends on:

- The size of this message: z_m.
- The average rate of the buffer: R_m.
- The starting time slot for the message m: $t_{j(m)}$.

Therefore, $\tilde{t}_m$ for each message $m \in M$ is given by:

$$\tilde{t}_m = t_{j(m)+\lceil z_m/R_m \rceil} \qquad m \in M. \tag{4.34}$$

The total number of time slots needed to guarantee that all messages m have been received by their respective sink is given by:

$$t^{Max} = max\{t_{j(m)+\lceil z_m/R_m \rceil} \mid m \in M\}. \tag{4.35}$$

Furthermore, each time slot $t \in \left\{t_1, \ldots, t^{Max}\right\}$ has a pre-defined colour.

Let us define the following sets: $\widehat{Z} = \{z_m \mid m \in M\}$ is the set of different message sizes; $\widehat{R} = \{R_m \mid m \in M\}$ is the set of different average rates and $\widehat{T} = \left\{t_{j(m)} \mid m \in M\right\}$ is the set of different starting time slots. It may happen that a particular message (or part of it) cannot be sent completely through the node's outgoing edges at the same time slot t. Therefore, the remaining part of this message will be "sent-stored" which means that the part of this message that could not be sent will be stored at the buffer's node during this time slot.

From a graphical perspective, the buffer can be interpreted as a self-loop edge.

It is possible to embed the time slots into the network digraph as follows:

- First, let $|V| = r$ and we extend the set V by aggregating additional nodes $v_{i+r*\jmath}$, $i = 1, \ldots, r$, $\jmath = 1, \ldots, t^{Max}$:

$$\overline{V} = V \cup \left\{ \bigcup_{\jmath=1}^{t^{Max}} \{v_{i+r*\jmath} \mid i = 1, \ldots, r\} \right\}. \tag{4.36}$$

- Second, the set of buffer edges is defined by:

$$\daleth = \left\{ \left(v_{i+r*(j-1)}, v_{i+r*j}\right) \in \overline{V} \times \overline{V} \mid i = 1, \ldots, r,\ j = 1 \ldots t^{Max} \right\}. \tag{4.37}$$

- Third, the new set of edges $\overline{E}$ is given by:

$$\begin{aligned} \mho \quad = \quad & \bigcup_{\jmath=1}^{t^{Max}} \left\{ \left(v_{i+r*(\jmath-1)}, v_{j+r*\jmath}\right) \right. \\ & \left. \mid Co_v(v_i) = Co_t(t_\jmath), (v_i, v_j) \in E,\ i, j = 1, \ldots r, i \neq j \right\} \end{aligned} \tag{4.38}$$

and we obtain the new set of edges $\bar{E}$ as

$$\overline{E} = \mho \cup \Finv. \tag{4.39}$$

Note: In order to simplify the visualisation of the following figures, we enumerate the nodes instead of using the node notation introduced before. This is, the node v_i, $i = 1, \ldots r$ will be represented in the figures by the circle with corresponding number i.

Example 4.1: *Let us consider a network with* 6 *nodes and* 8 *edges. The digraph* $G_1(V, E)$ *associated with this network has the set of nodes* $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ *and the set of edges*

$$E = \{(v_1, v_2), (v_1, v_3), (v_2, v_4), (v_2, v_5), (v_3, v_4), (v_3, v_5), (v_4, v_6), (v_5, v_6)\}.$$

The graphical representation of $G_1(V, E)$ *is show in Figure 4.1:*

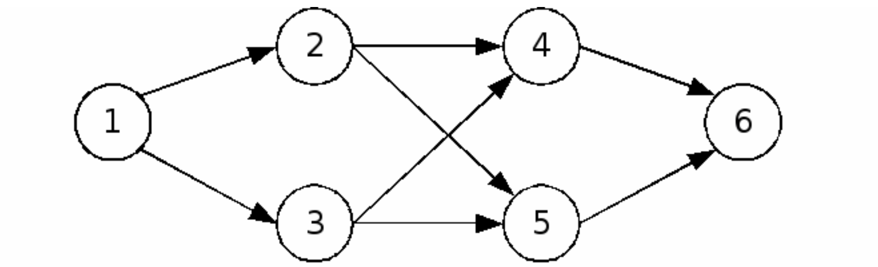

Figure 4.1: Six nodes network graph.

Let us propose the following colour assignation to the digraph $G_1 = (V, E)$ *shown in Figure 4.2.*

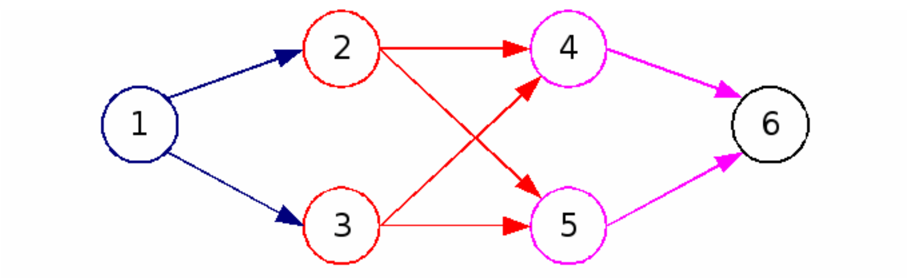

Figure 4.2: Coloured network graph.

We can see from Figure 4.2, that the digraph $G_1(V, E)$ has 4 colours associated and the following node colour assignation:

Node	Colour
v_1	blue
v_2, v_3	red
v_4, v_5	magenta
v_6	black

Table 4.1: Node colour assignation.

Therefore, we have the following set of colours:

$$C = \{"blue", "red", "magenta", "black"\}. \tag{4.40}$$

We will show how to get the new sets $\bar{V}$ and $\bar{E}$. First, consider only one time slot t_1 (the first one). According to (4.36) we obtain:

$$V \cup \{v_{i+r} \mid i = 1, \dots r\} = \{v_i, i = 1, \dots, 12\} \subset \bar{V}.$$

The corresponding buffer edges (4.37) are:

$$\{(v_i, v_{i+r}) \mid i = 1, \dots r\} = \{(v_1, v_7), (v_2, v_8), (v_3, v_9), (v_4, v_{10}), (v_5, v_{11}), (v_6, v_{12})\} \subset \daleth.$$

From (4.38) we have that

$$\{(v_i, v_{j+r}) \mid Co_v(v_i) = Co_t(t_1), (v_i, v_j) \in E, i, j = 1, \ldots r, i \neq j\} \subset \mho.$$

According to Table 4.1, let $Co_t(t_1) =$ *blue. Therefore, by Figure 4.2 the only node with the colour blue is* v_1*. Then, we choose corresponding edges* $(v_1, v_j) \in E$ *which are* (v_1, v_2) *and* (v_1, v_3)*. Since* $r = 6$ *we have :*

$$\{(v_1, v_8), (v_1, v_9)\} \subset \mho.$$

Therefore, according to (4.39) we have:

$$\begin{aligned} \bar{E} &= \mho \cup \Finv \\ &\supset \{(v_1, v_7), (v_2, v_8), (v_3, v_9), (v_4, v_{10}), (v_5, v_{11}), (v_6, v_{12}), (v_1, v_8), (v_1, v_9)\}. \end{aligned}$$

Figure 4.3 is the graphical representation of the corresponding digraph when the time slot under consideration is t_1 *(blue):*

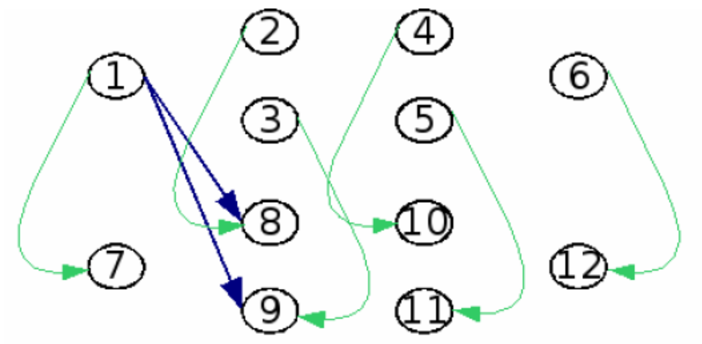

Figure 4.3: Time slot t_1 *blue* coloured.

Now, consider the time slot t_2 *is red* t_2*, i.e.* $Co_t(t_2) =$ *red. From Table 4.1 we select nodes with the colour red, which are* v_2 *and* v_3*. Then, the selected edges*

(v_i, v_j), such that $\{(v_i, v_j) \in E \mid i = 2, 3;\ j = 1, \ldots, r\}$, are $(v_2, v_4), (v_2, v_5), (v_3, v_4)$ and (v_3, v_5). Since $r = 6$ we have:

$$\{(v_2, v_{10}), (v_2, v_{11}), (v_3, v_{10}), (v_3, v_{11})\} \subset \mho.$$

Figure 4.4 is the graphical representation of the corresponding digraph.

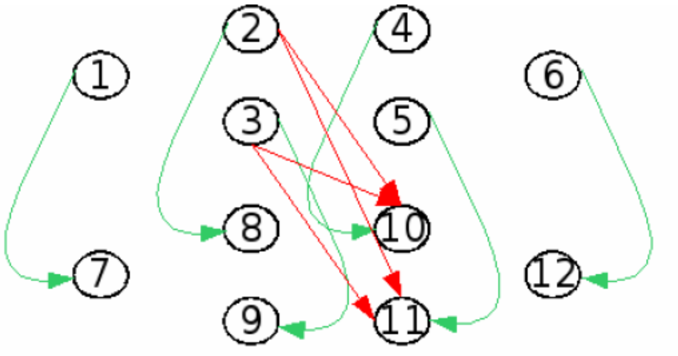

Figure 4.4: Time slot *red* coloured.

Now consider the first two time slots t_1 (blue) and t_2 (red). By (4.36) and (4.39), we have the following subsets of $\overline{V}$ and $\overline{E}$, respectively:

$$\{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}, v_{12}, v_{13}, v_{14}, v_{15}, v_{16}, v_{17}, v_{18}\} \subset \overline{V}$$

and

$$\{(v_1, v_7), (v_7, v_{13}), (v_2, v_8), (v_8, v_{14}), (v_3, v_9), (v_9, v_{15}), (v_4, v_{10}), (v_{10}, v_{16}), (v_5, v_{11}), \\ (v_{11}, v_{17}), (v_6, v_{12}), (v_6, v_{18}), (v_1, v_8), (v_1, v_9), (v_8, v_{16}), (v_8, v_{17}), (v_9, v_{16}), (v_9, v_{17})\} \subset \overline{E}.$$

The graphical representation of the corresponding digraph is shown in Figure 4.5.

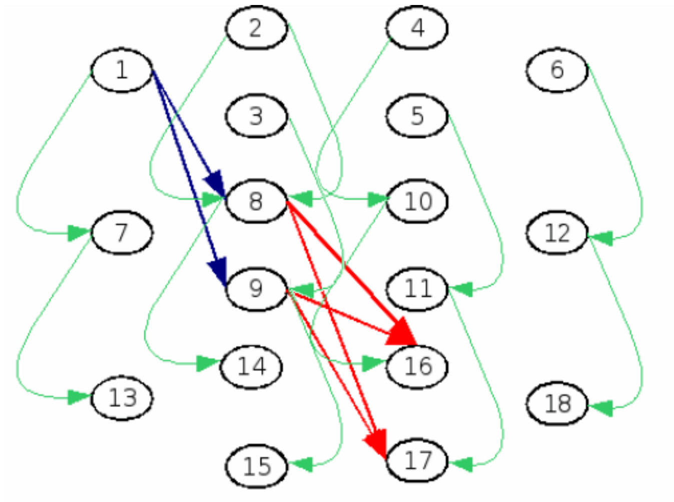

Figure 4.5: Two time slots with different colour associated.

Embedding the time slots into the network helped us to have a different vision of the problem. This embedding does not only help to "simplify" the modelling of the constraints into AMPL's notation, but also gives us the graphical representation of how the message(s) may flow or flow(s) through the network.

This completes Example 4.1.

In order to solve the non-linear programming problem $\widehat{\Theta}$ numerically, we used the package Knitro [30] and AMPL as the interface. (refer to Section 3.4 or [42]). The set of constraints (4.1)-(4.4),(4.6)-(4.11) and (4.33) will be written with AMPL's syntax and saved in a model file. For a specific example, the value of the parameter, upper and lower bounds, constants, etc. will be saved in a data file. In Appendix A, we have a *model file* which is the "transcription" of the general set of constraints and the objective function for the problem $\widehat{\Theta}$ into the AMPL's syntax.

AMPL offers a free student version which only allows us to solve problems of size 300 variables and 300 constraints. Let us consider a problem where sending a single message along a graph consisting of 4 nodes, 4 edges and 4 time slots is desired; in this case we have 240 variables and 298 constraints. But when we move on to graphs consisting of 6 nodes, 8 edges and 10 time slots for a single message, it generates a system of 1438 variables and 1172 constraints. In this case we exceed the restriction of the student's version on the number of constraints and variables. Then, it is necessary to send our model and data files plus a command file, which consist of those commands needed to solve and retrieve results, to an Internet server (NEOS see [10]). The NEOS server has a large variety of solvers and the number of variable and constraints is unrestricted. Then, the NEOS server will send back the obtained results by email.

4.4 Numerical Results and Sensitivity Analysis

In this section we present a set of experiments, their results and their analysis. The concepts of data, information or combination of messages are going to be considered as synonyms. By combination of messages we mean the adding of the sizes of two or more (pieces of) messages. We say, that the programming problem $\widehat{\Theta}$ becomes unfeasible, if it has no solution, i.e. the feasible set becomes empty. For the $SINR$ in this set of experiments, the spreading codes (fading) are all equal to one and the path-loss due to the distance between nodes is not considered.

The series of experiments to be presented have the following goals:

- To show graphically the flow of one or more messages when a local minimiser for the non-linear programming problem $\widehat{\Theta}$ is found.
- To solve the non-linear programming problem $\widehat{\Theta}$ for one message (single commodity problem), two messages and five messages through a (multi-commodity problem) network.

- To find the maximum size of a message that can be sent through a given network.
- To identify those constraints that cannot be fulfilled if the maximum size is exceeded. We will say that those constraints *make the problem* $\widehat{\Theta}$ *unfeasible*.
- To present a sensitivity analysis for the following parameters: Bandwidth, starting time slot, upper bounds (P^{Max}, B_v^{Max}).

We present four sets of experiments for different types of networks (digraphs) and parameters. The first set of experiments consists of sending a message through a given digraph. A sensitivity analysis of the results is presented. The second set of experiments consists of transmissions of two messages through a given digraph and its sensitivity analysis. The third set of experiments is for five messages with sensitivity analysis. The fourth set of experiments consists in sending a message through a larger network. For the following set of experiments we will consider the $SINR$

1. First set of experiments: Consider a message m_1 to be sent through the digraph $G_1(V, E)$ as given in Figure 4.2. The message's source is the node v_1 and the message's sink is the node v_6. The size of the message is $z_{m_1} = 3000\,Kbits$, the reduction rate of the buffer starting from $t_{j(m_1)} = t_3$ is $R_{m_1} = 600\,Kbits$. Therefore, we have that $\tilde{t}_{m_1} = t_{3+\lfloor 3000/600 \rfloor} = t_8$. Table 4.2 represents the colouring assignation for each time slot:

Time slot	Colour
t_1, t_4, t_7	blue
t_2, t_5, t_8	red
t_3, t_6	magenta

Table 4.2: Time slots colour assignation.

By (4.36) and (4.39), we obtain the graphical representation of the embedded time slots into the network in Figure 4.6:

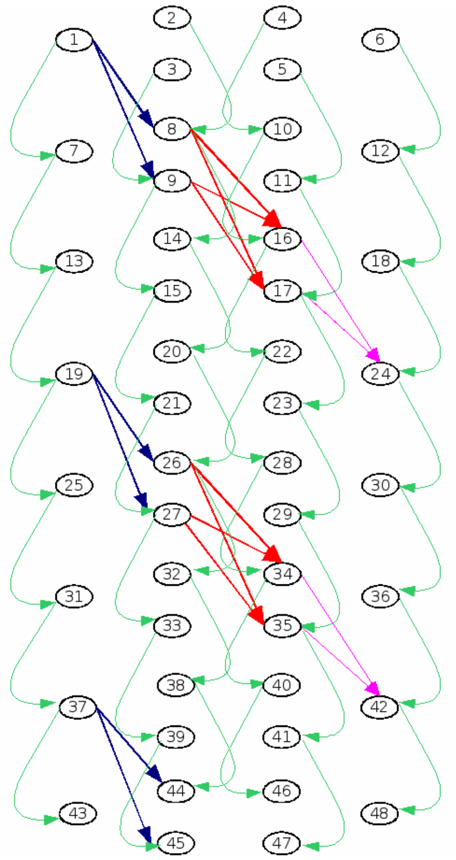

Figure 4.6: Time Coloured Network Topology.

Table 4.3 shows the set of fixed parameters and units used in order to solve the optimisation problem $\widehat{\Theta}$.

Parameter	Value	Units
P_v^{Max}	2	W
B_v^{Max}	4000	$kbit$
$P_{e,m}^{Max}$	2	W
η	0.008	mW
B	5	MHz

Table 4.3: List of parameters.

We solved this problem with the solver Knitro and obtained the following results: After 31 iterations which took 0.09 seconds, a local minimiser was found. The value of the objective function is 0.0012520468 mW. Figure 4.7 shows the edges used to send the message from the source to the sink. The sink is represented by those nodes which are marked by the letter "R" next to them. As we can see from Figure 4.7, the message leaves node v_1 and is divided into three parts. Two of them are sent along the edges (v_1, v_8) and (v_1, v_9), respectively. Here, an interesting pattern appears: both parts have the same size, i.e. $c_{(v_1,v_8),m_1} \approx c_{(v_1,v_9),m_1}$, and the remaining part is sent/stored at the node's buffer i.e. along the edge (v_1, v_7). At every node, we can verify how each part of the message flows through the network until the complete message is received. The reception of the message will start after the third time slot; then, the sink node must have at least R_m Kbits of the message and it can be "used" by the node (user). We can see that node v_{24} has two incoming edges: (v_{16}, v_{24}) and (v_{17}, v_{24}) as well as, one sent/stored edge (v_{24}, v_{30}). The reduction rate R is denoted by the arrow leaving the node. We can see almost the same pattern in nodes v_{30}, v_{36}, v_{42} and v_{48}.

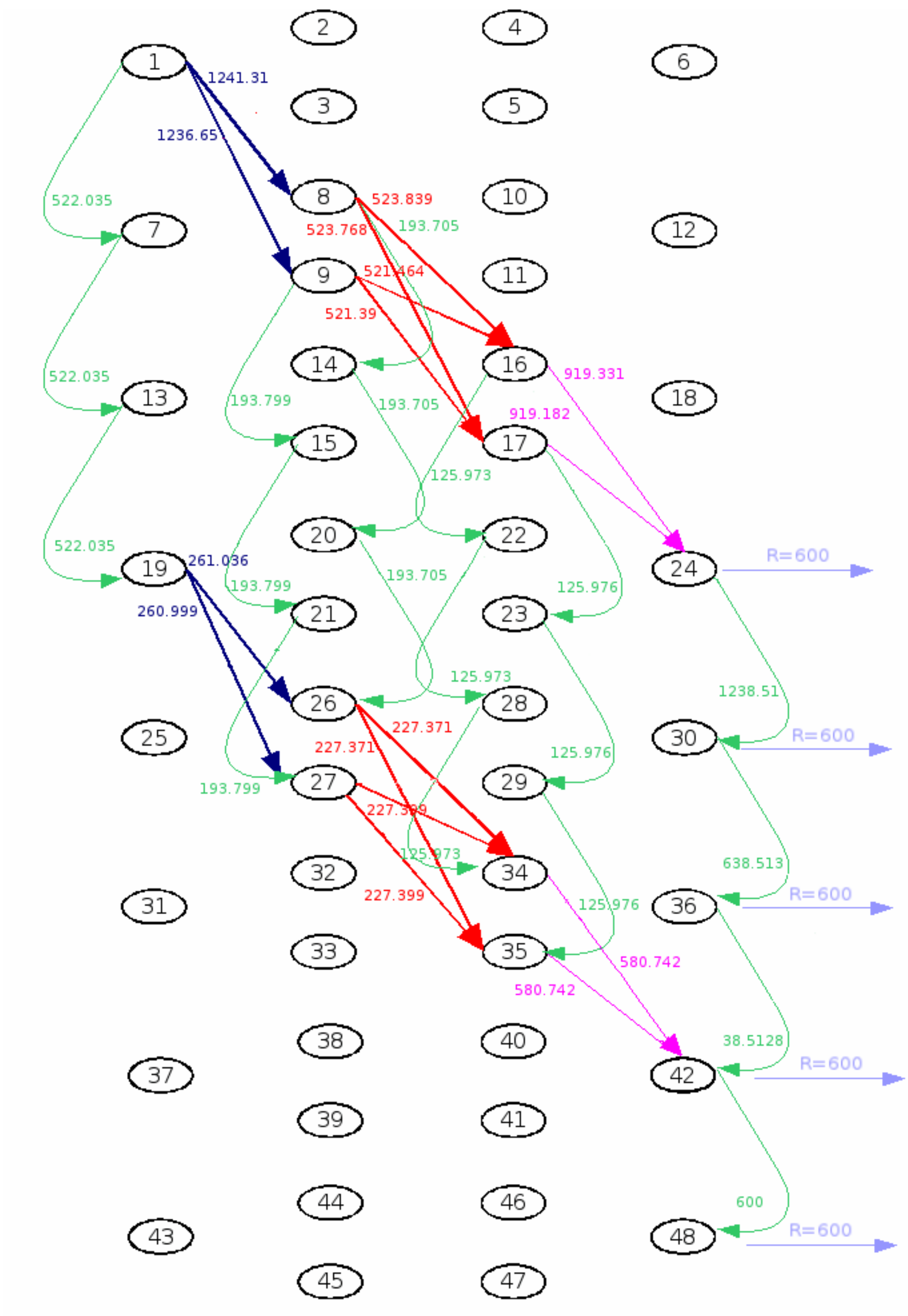

Figure 4.7: Message routing.

Tables 4.5 ,4.6 and 4.7 show the values of the transmission power p_e used at edge e to send the piece of message m_1 with size c_{e,m_1}, the size c_{e,m_1} and the size of that piece of message m_1 stored in the buffer edges e, respectively.

Edge	mW	$\sum_{i \neq e} p_{i,t}$	$SINR_e$(dBm)		Edge	mW	$\sum_{i \neq e} p_{i,t}$	$SINR_e$(dBm)
(v_1, v_8)	0.000184803	0.0009842	0.1877754		(v_{19}, v_{27})	0.000030604	0.0008306	0.0368448
(v_1, v_9)	0.000184167	0.0009848	0.1870083		(v_{26}, v_{34})	0.000028341	0.0008850	0.0320222
(v_8, v_{16})	0.00007778	0.0010326	0.0753215		(v_{26}, v_{35})	0.000028341	0.0008850	0.0320222
(v_8, v_{17})	0.00007777	0.0010327	0.0753110		(v_{27}, v_{34})	0.000028344	0.0008850	0.0320265
(v_9, v_{16})	0.00007744	0.0010330	0.0749667		(v_{27}, v_{35})	0.000028344	0.0008850	0.0320265
(v_9, v_{17})	0.000077429	0.0010330	0.0749563		(v_{34}, v_{42})	0.000073208	0.0008732	0.0838371
(v_{16}, v_{24})	0.000125841	0.0009258	0.1359238		(v_{35}, v_{42})	0.000073208	0.0008732	0.0838386
(v_{17}, v_{24})	0.000125822	0.0009258	0.1359001		(v_{37}, v_{44})	-	-	-
(v_{19}, v_{26})	0.000030608	0.0008307	0.0368448		(v_{37}, v_{45})	-	-	-

Table 4.5: Transmission power p_e used at the edge e.

Edge	Kbits		Edge	Kbits
(v_1, v_8)	1241.31		(v_{19}, v_{27})	261.00
(v_1, v_9)	1236.65		(v_{26}, v_{34})	227.37
(v_8, v_{16})	523.84		(v_{26}, v_{35})	227.37
(v_8, v_{17})	523.77		(v_{27}, v_{34})	227.40
(v_9, v_{16})	521.46		(v_{27}, v_{35})	227.40
(v_9, v_{17})	521.39		(v_{34}, v_{42})	580.74
(v_{16}, v_{24})	919.33		(v_{35}, v_{42})	580.75
(v_{17}, v_{24})	919.18		(v_{37}, v_{44})	0.0
(v_{19}, v_{26})	261.00		(v_{37}, v_{45})	0.0

Table 4.6: Size of the piece of message m_1 sent through edge e.

After finding a local minimiser to the non-linear programming problem $\widehat{\Theta}$, the following questions arise.

1. What is the maximum size of data such that the problem $\widehat{\Theta}$ has a local minimiser?

2. What happens if the starting time slot for the reception of the message is different?

3. Which constraints cannot be fulfilled for certain parameter values?

Edge	Kbits		Edge	Kbits
(v_1, v_7)	522.04		(v_{21}, v_{27})	193.8
(v_7, v_{13})	522.04		(v_{22}, v_{28})	125.97
(v_8, v_{14})	193.71		(v_{23}, v_{29})	125.98
(v_9, v_{15})	193.8		(v_{24}, v_{30})	1238.51
(v_{13}, v_{19})	522.04		(v_{28}, v_{34})	125.97
(v_{14}, v_{20})	193.71		(v_{29}, v_{35})	125.98
(v_{15}, v_{21})	193.8		(v_{30}, v_{36})	638.51
(v_{16}, v_{22})	125.97		(v_{36}, v_{42})	38.51
(v_{17}, v_{23})	125.98		(v_{42}, v_{48})	600
(v_{20}, v_{26})	193.71			

Table 4.7: Size of the piece of message m_1 sent through edge e (buffer edges).

We will answer these questions by applying sensitivity analysis. For the predefined set of parameters given in Table 4.3, the corresponding digraph, the starting time slot $t_{j(m_1)} = t_3$ and Table 4.2 we will try to find the maximum size of data m_1 for uniform reception rate R_{m_1} that can be sent through the network such that the problem $\widehat{\Theta}$ has still a local minimiser. The aim is to increase the size of the message z_{m_1} and R_{m_1} gradually until the problem $\widehat{\Theta}$ becomes unfeasible. For this purpose we define the following constants:

- The increment $\daleth$ on the size of the message (in Kbits).

- The increment $\beth$ on reception rate (in Kbits).

Then, we apply the following algorithm:

Algorithm 1 Increments on the size of the message

---

procedure Increments (Set of messages M, Set $\widehat{Z}$, Set $\widehat{R}$, constant$\daleth$, constant $\beth$, Set $\widehat{T}$, Coloured digraph $G\,(V,E)$, Coloured time slot Table)

for each $m \in M$

do

$Z_m = Z_m + \daleth$, $R_m = R_m + \beth$;

calculate $\overline{G}\left(\overline{V},\overline{E}\right)$for given $t_{j(m)}$, $G\,(V,E)$ and Coloured time slot table;

solve the non-linear programming problem $\widehat{\Theta}$ for given parameters $\left(P_v^{Max}, P_{e,m}^{Max}, B_v^{Max}, \eta, B, \Omega, t_{j(m)}, \overline{G}\left(\overline{V},\overline{E}\right), Z_m, R_m\right)$;

until (problem $\widehat{\Theta}$ becomes unfeasible);

end

end procedure

We ran Algorithm 1 for the following set of parameters $M = \{m_1\}\,, z_{m_1} = 3000, R_{m_1} = 600, \daleth = 500, \beth = 100, \widehat{T} = \{t_3\}$, digraph $G_1\,(V,E)$ and colour time slot Table 4.2. We obtained the following results: Algorithm 1 solves sixteen different experiments before the problem $\widehat{\Theta}$ becomes unfeasible. Figure 4.8 displays the sixteen different experiments for different sizes of messages and the total power used in the network for each one of them.

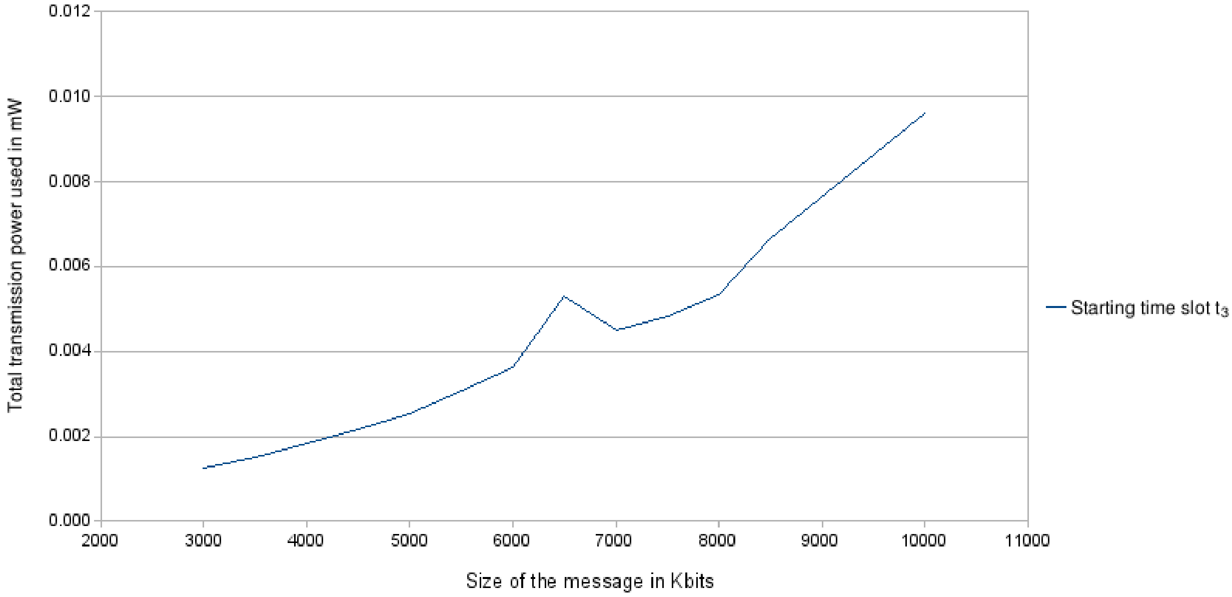

Figure 4.8: Minimum transmission power VS. size of message for starting time slot t_3.

Algorithm 1 stops either after 16 iterations finding that the message which can be sent through the network has the maximum size of $z_{m_1} = 10000\ Kbits$ at a constant reception rate of $R = 2000\ Kbits$. In particular, there is a bottleneck caused by the constraint (4.4). That is, the storage capacities of some nodes reach their limits, therefore it is not possible to store a bigger message. The time needed to solve these sixteen experiments is about 22 seconds and the average number of iterations is 260.

Now, let us repeat Algorithm 1 n times for different starting time slots, using the predefined set of parameters given in Table 4.3, the digraph $G_1(V, E)$, and Table 4.2. We present this idea in Algorithm 2:

Algorithm 2 Different starting time slot.

procedure Time_slot (Set of messages M, Set $\widehat{Z}$, Set $\widehat{R}$, constant$\daleth$, constant $\beth$, Set $\widehat{T}$, Coloured digraph $G\left(V,E\right)$, Coloured time slot Table, constant n)
  for $i \leftarrow 1$ *to* $i = n$
    Increments $(M$, Z_m, R_m, $\daleth$, $\beth$,$\widehat{T}$,$G\left(V,E\right)$, Coloured time slot Table);
    for each $m \in M$
      $t_{j(m)} = t_{j(m)+1}$;
    end
  end
end procedure

We ran the Algorithm 2 for the following set of parameters $M = \{m_1\}, z_{m_1} = \{3000\}, R_{m_1} = \{600\}, \daleth = 500, \beth = 100, \varkappa = 4, \widehat{T} = \{t_3\}$, digraph $G_1\left(V,E\right)$, and Table 4.2. We obtained the following results: Figure 4.9 shows different starting time slots $t^{Max} = \{t_3, t_4, t_5, t_6, t_7\}$.

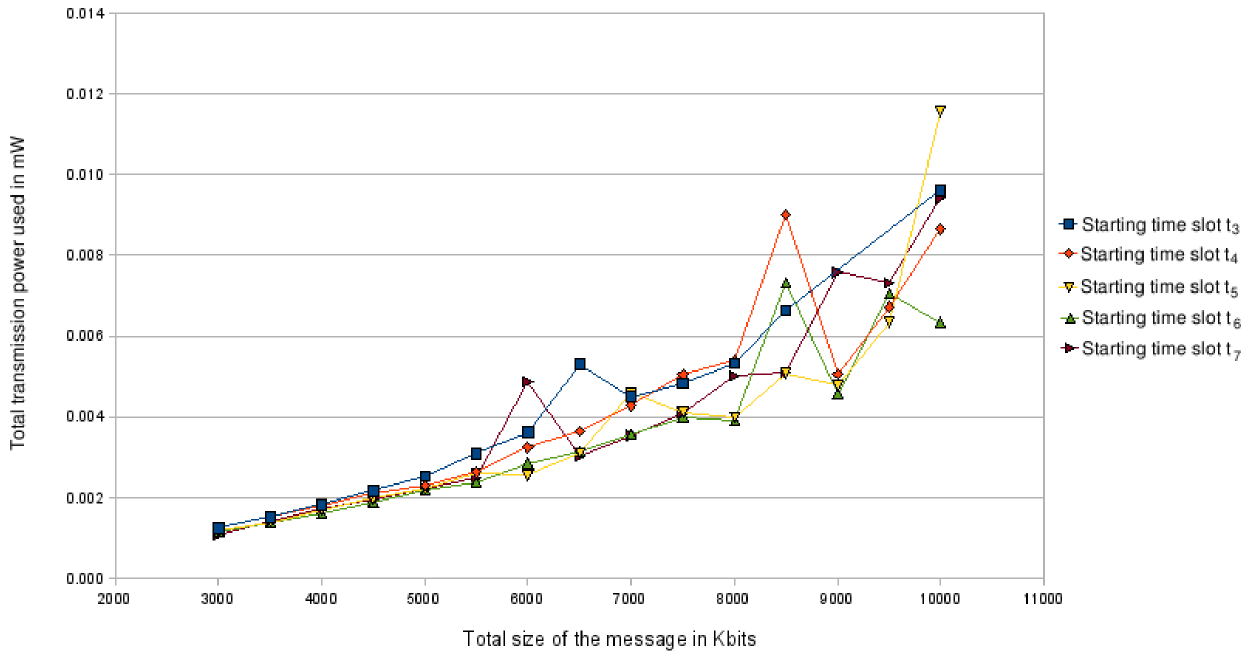

Figure 4.9: Transmission power VS. size of message at different starting time slots.

We obtained the following results, which are presented in Figure 4.9:

- Independently of the starting time slot $t_i, i = 3, \ldots, 7$ the total transmission power required increases as the size of the message z_{m_1} also increases. The relation remains similar up to messages of size $z_{m_1} = 5500$ Kbits.

- For messages greater than 5500 Kbits and for each starting time slot there is a specific size of the message where the power increases substantially, but immediately drops, i.e., there is a peak in the curve. For example: For starting time slot t_3 (blue line) for $z_{m_1} = 6000$ Kbits the transmission power is 0.0038 mW, but for $z_{m_1} = 6500$ Kbits the transmission power increases to 0.0055 mW and at $z_{m_1} = 8500$ Kbits the transmission power falls to 0.0044 mW. For starting time slot t_4 (red line) for $z_{m_1} = 8000$ Kbits the transmission power is 0.0055 mW, but for $z_{m_1} = 8500$ Kbits the transmission power increases to 0.009 mW and at $z_{m_1} = 9000$ Kbits the transmission power falls to 0.005 mW.

- The biggest message that can be sent does not depend on the starting time slot. This is not surprising since the capacity of the node's buffer is a bottleneck and it has not changed.

The time taken in order to solve this set of problems is around 1 minute and the average of number of iterations is 320.

2. For the second set of experiments consider the digraph $\widehat{G}\left(\widehat{V}, \widehat{E}\right)$, where

$$\widehat{V} = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\},$$

$$\begin{aligned}\widehat{E} = \{&(v_1, v_3), (v_1, v_4), (v_2, v_3), (v_2, v_4), (v_3, v_5), (v_3, v_6), \\ &(v_4, v_5), (v_4, v_6), (v_5, v_7), (v_5, v_8), (v_6, v_7), (v_6, v_8)\}\end{aligned}$$

and we have the following colour assignation to the nodes:

Node	Colour
v_1, v_2	blue
v_3, v_4	red
v_5, v_5	magenta
v_7, v_8	black

Table 4.8: Colour assignation for $v \in \widehat{V}$.

The graphical representation of $\widehat{G}\left(\widehat{V}, \widehat{E}\right)$ is given in Figure 4.10:

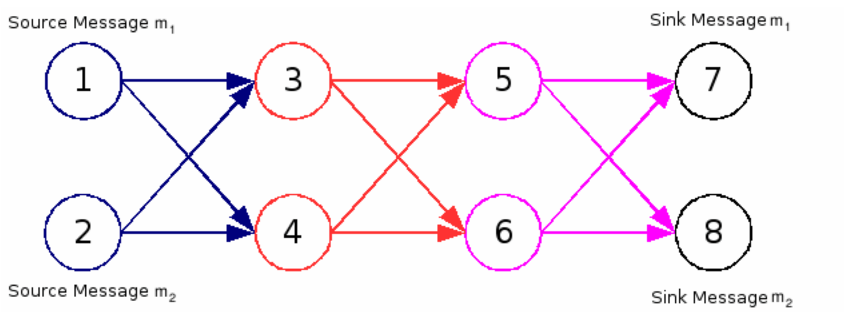

Figure 4.10: Eight node graph with two messages.

We consider two messages being sent along $\widehat{G}\left(\widehat{V}, \widehat{E}\right)$. The first message m_1 leaves from node v_1 and its sink node is v_7 while the second message m_2 leaves from node v_2 and its sink node is v_8. Since we have two messages to be sent, we are facing a multi-commodity network problem [67, 70]. As a matter of simplicity for the following set of experiments, both messages will have the same size and the same reduction rate: $z_{m_1} = z_{m_2} = 1500\ Kbits$ and $R_{m_1} = R_{m_2} = 300\ Kbits$. Furthermore, start the reception of both messages at $t_{j(m_1)} = t_{j(m_2)} = t_3$ and assume the following time slot assignation:

Time slot	Colour
t_1, t_4, t_7, t_{10}	blue
t_2, t_5, t_8, t_{11}	red
t_3, t_6, t_9, t_{12}	magenta

Table 4.9: Time slot assignation table for two messages.

From (4.35) we have that $t^{Max} = t_8$. We solved the non-linear programing problem $\widehat{\Theta}$ for these two messages along $\widehat{G}\left(\widehat{V}, \widehat{E}\right)$ with the set of parameters given in Table 4.3, node colouring and time slot colouring given in Tables 4.8 and 4.9, respectively. The following results were obtained: After 53 iterations made in 0.5 seconds, the solver Knitro finds a local minimiser with objective function value 0.001181189 mW. In Figures 4.11 and 4.12 we visualise the flow for message m_1 and m_2, respectively.

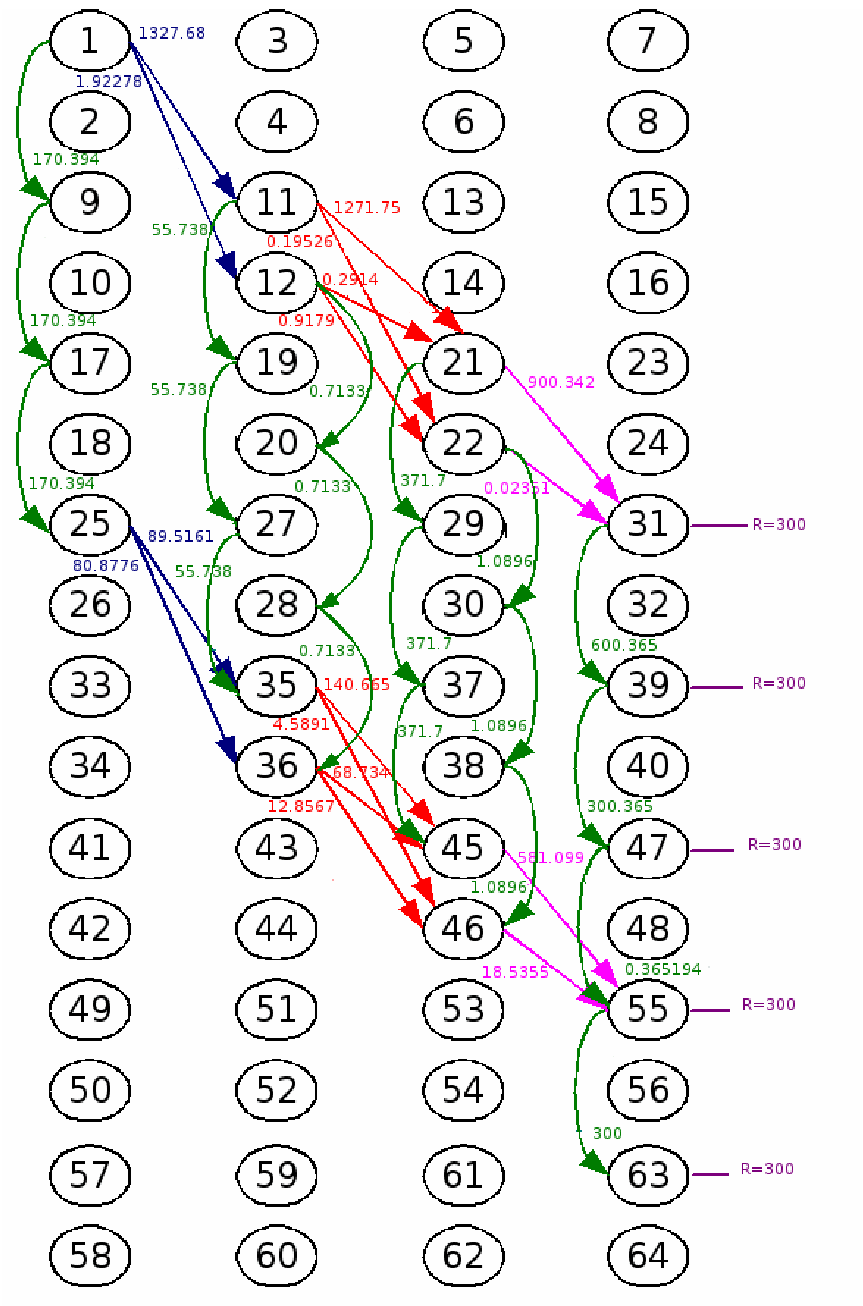

Figure 4.11: Message m_1 flowing through the network.

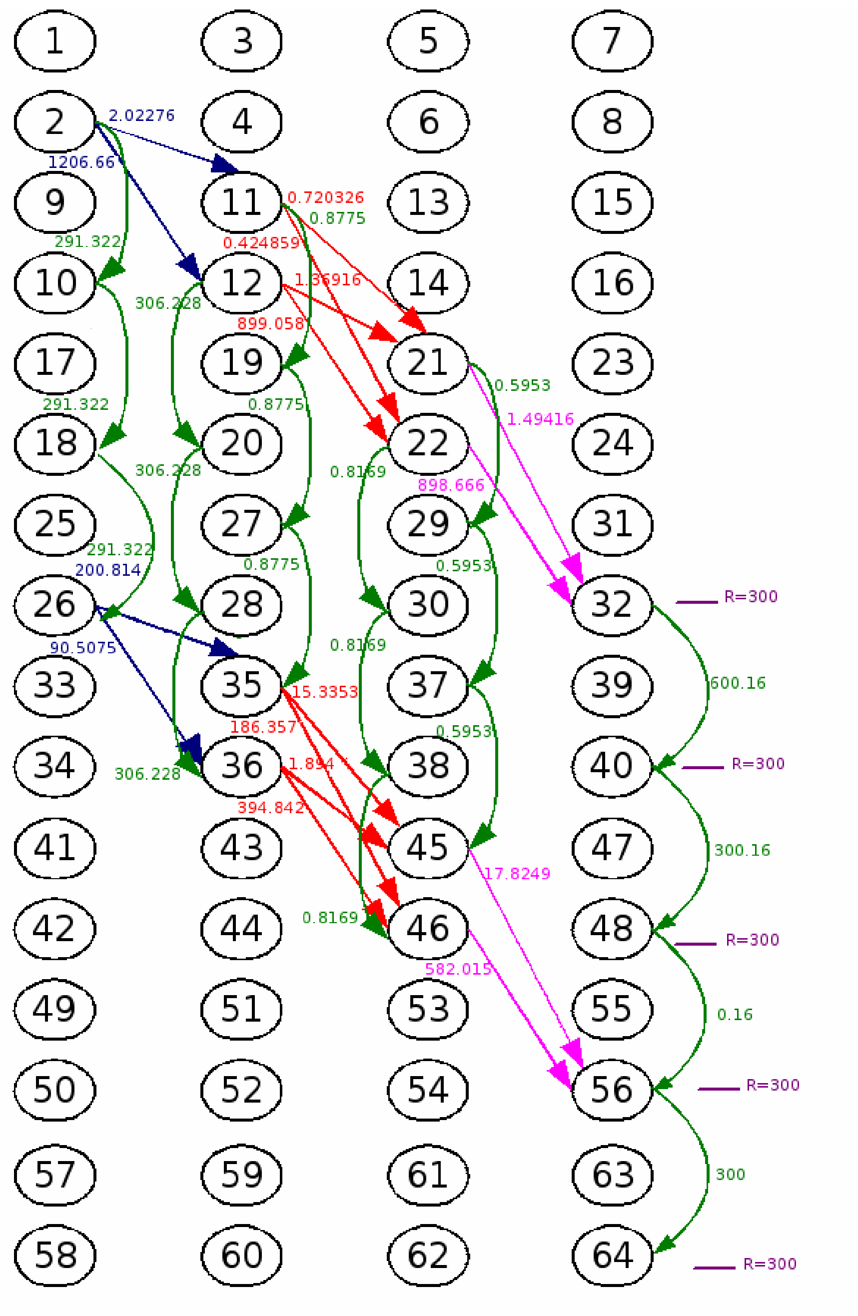

Figure 4.12: Message m_2 flowing through the network.

Figures 4.11 and 4.12 show how the messages are split into small parts and sent or stored by nodes until they are complete in their sink nodes. The sink nodes are those nodes with an "R" next to them. The values of the transmission power p_e

used at edge e to send a piece of a message with size $c_{e,m}$, the size $c_{e,m}$ of the piece of the message sent through edge e and the size of the piece of message m stored in the buffers are shown in the following Table 4.11.

	Power $p_{(v_i,v_j)}$		$c_{(v_i,v_j),m_1}$		$c_{(v_i,v_j),m_2}$
Edge (v_i, v_j)	W		Kbits		Kbits
(v_1, v_{11})	0.198567		1327.68		0
(v_1, v_{12})	0.000313		1.92		0
(v_2, v_{11})	0.000331		0		2.02
(v_2, v_{12})	0.181940		0		1206.66
(v_{11}, v_{21})	0.179521		1271.75		0.72
(v_{11}, v_{22})	0.000095		0.2		0.42
(v_{12}, v_{21})	0.000255		0.29		1.37
(v_{12}, v_{22})	0.130205		0.92		899.06
(v_{21}, v_{31})	0.122656		900.34		0
(v_{22}, v_{31})	0.000003		0.02		0
(v_{21}, v_{32})	0.000217		0		1.49
(v_{22}, v_{32})	0.122442		0		898.67
(v_{25}, v_{35})	0.010535		89.52		0
(v_{25}, v_{36})	0.009524		80.88		0
(v_{26}, v_{35})	0.023451		0		200.81
(v_{26}, v_{36})	0.010652		0		90.51
(v_{35}, v_{45})	0.019279		140.67		15.34
(v_{35}, v_{46})	0.023540		4.59		186.36
(v_{36}, v_{45})	0.008780		68.73		1.89
(v_{36}, v_{46})	0.049517		12.86		394.84
(v_{45}, v_{55})	0.073708		581.1		0
(v_{45}, v_{56})	0.002350		0		17.82
(v_{46}, v_{55})	0.002444		18.54		582.02
(v_{46}, v_{56})	0.073819		0		0

Table 4.11: Transmission powers and sizes of pieces of messages m_1 and m_2.

We ran Algorithm 1 for the following set of parameters: $M = \{m_1, m_2\}$, $z_{m_1} = z_{m_2} = 3000$, $R_{m_1} = R_{m_2} = 600$, $\daleth = 500, \beth = 100, \widehat{T} = \{3\}$, digraph $\widehat{G}\left(\widehat{V}, \widehat{E}\right)$ and Table 4.9. We obtained the following results:

Algorithm 1 solves eight different optimisation problems $\widehat{\Theta}$ for different sizes of messages before the problem $\widehat{\Theta}$ becomes unfeasible. We can see from Figure 4.13 that the combination of sizes for both messages cannot be greater than 7500 $Kbits$; otherwise, problem $\widehat{\Theta}$ would become unfeasible. Let us compare these results with the analogous results for the one message presented in Figure 4.8. We observe from Figure 4.8 and Figure 4.13 that the combined size of information for both messages is smaller than that of one message. That is, for one message we can send up to 10000 $Kbits$ while for the combination of two messages only 7500 $Kbits$. Furthermore, we can see in Figure 4.8 that the total power used to send one message of size 7500 $Kbits$ is one sixth of the power needed to send two messages of combined size of 7500 $Kbits$ in Figure 4.13. The difference is that it is the critical constraints that make problem $\widehat{\Theta}$ unfeasible for sending one message is (4.4), while for two messages it is the constraints (4.16). Here, we have an interesting pattern: If the constraints (4.4) make the problem $\widehat{\Theta}$ unfeasible, then the relationship between the size of the message and the total transmission power used resembles a quadratic function, as illustrated in Figure 4.8. But if the constraints (4.16) make the problem unfeasible, then this relationship resembles an exponential function, (as illustrated in Figure 4.13). In order to obtain a problem $\widehat{\Theta}$ that can also be solved when the combined size of messages is greater than 7500 $Kbits$, we can increase the size of the bandwidth, or as another option give more time before starting the reception of the messages. Increasing the size of the bandwidth is not a good idea, since as we already mentioned before, the bandwidth is a limited resource in telecommunication systems. The second option proposed is considered in the next experiment.

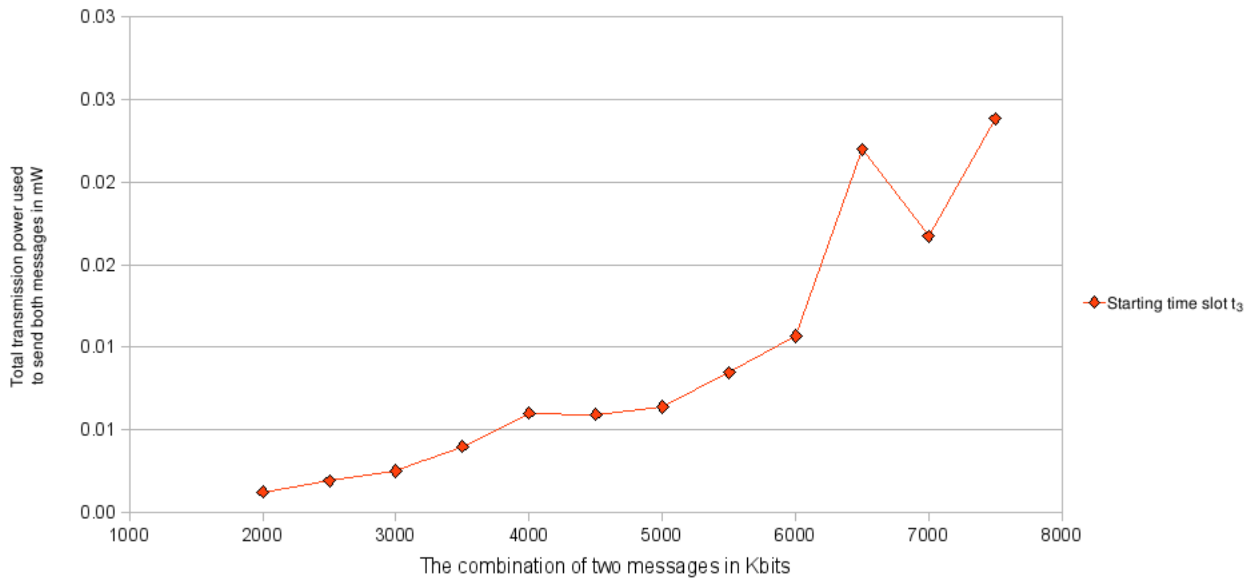

Figure 4.13: Minimum transmission power vs. size of message for starting time slot t_3.

Now applying Algorithm 2, we increase the size of each message for different starting time slots $t_{j(m)}$. Consider these parameters: $M = \{m_1, m_2\}$, $z_{m_1} = z_{m_2} = 3000$, $R_{m_1} = R_{m_2} = 600$, $\daleth = 500, \beth = 100, \varkappa = 4, t_{j(m_1)} = t_{j(m_2)} = t_3$, digraph $\widehat{G}\left(\widehat{V}, \widehat{E}\right)$ and Table 4.9. We obtained the following results presented in Figure 4.14.

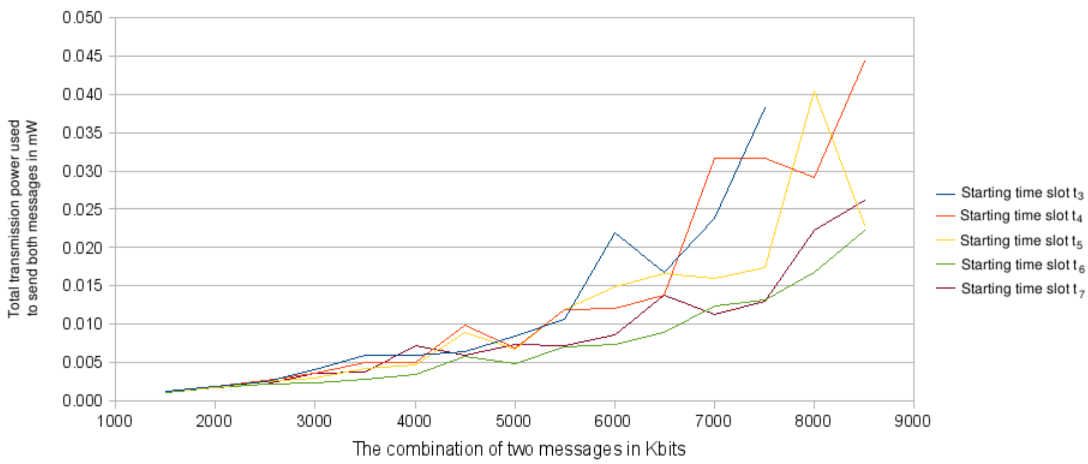

Figure 4.14: Transmission power vs. the combination of two messages sizes at different starting time slots.

- Independently of the starting time slot $t_i, i = 3, \ldots, 7$ the total transmission power required increases as the size of the combined size of messages z_{m_1} and z_{m_2} also increases. The relation remains the same up to messages of size $z_{m_1} + z_{m_2} = 5000$Kbits.

- For starting time slots t_3, t_4 and t_5 we have distinct transmission power peaks at 6000 Kbits, 7000 Kbits and 8000 Kbits respectively.

- For starting time slots t_6 and t_7 there are not peaks and the transmission power steadily increases to maximum transmission power 0.022 mW and 0.026 mW respectively.

- Only for the starting time slot t_5 the transmission power drops drastically after a combined size of 8000 Kbits while for starting time slots t_3 and t_4 the power increases until the combined size of messages reaches its maximum at 7500 Kbits and 8500 Kbits respectively. Since the starting time slot is set later, then the message has more time to reach the sink and the power needed is less, resulting in the drop observed.

- For starting time slot t_3 the maximum combined size of messages that can be sent is 7500 Kbits. For combined size of messages greater than 7500 Kbits, problem $\widehat{\Theta}$ becomes unfeasible due to the channel capacity i.e. constraint (4.16).

3. Third set of experiments: Consider five messages m_1, m_2, m_3, m_4, m_5 to be sent

along the digraph $\widetilde{G}\left(\widetilde{V},\widetilde{E}\right)$ where $\widetilde{V}=\{v_1,\ldots,v_{20}\}$,

$$\begin{aligned}\widetilde{E} \;=\; & \{(v_1,v_6),(v_1,v_7),(v_2,v_6),(v_2,v_7),(v_2,v_8),(v_3,v_7),(v_3,v_8),(v_3,v_9),(v_4,v_8),\\ & (v_4,v_9),(v_4,v_{10}),(v_5,v_9),(v_5,v_{10}),(v_6,v_{11}),(v_6,v_{12})\,(v_7,v_{11}),(v_7,v_{12}),\\ & (v_7,v_{13}),(v_8,v_{12}),(v_8,v_{13}),(v_8,v_{14}),(v_9,v_{13}),(v_9,v_{14}),(v_9,v_{15}),(v_{10},v_{14}),\\ & (v_{10},v_{15}),(v_{11},v_{16}),(v_{11},v_{17}),(v_{12},v_{16}),(v_{12},v_{17}),(v_{12},v_{18}),(v_{13},v_{17}),\\ & (v_{13},v_{18}),(v_{13},v_{19}),(v_{14},v_{18}),(v_{14},v_{19}),(v_{14},v_{20}),(v_{15},v_{19}),(v_{15},v_{20})\}\end{aligned}$$

and we have the following colour assignation to the nodes:

Node	Colour
v_1,v_2,v_3,v_4,v_5	blue
v_6,v_7,v_8,v_9,v_{10}	red
$v_{11},v_{12},v_{13},v_{14},v_{15}$	magenta
$v_{16},v_{17},v_{18},v_{19},v_{20}$	black

Table 4.12: Colour assignation for $v\in\widetilde{V}$.

The graphical representation $\widetilde{G}\left(\widetilde{V},\widetilde{E}\right)$ is given in Figure 4.15.

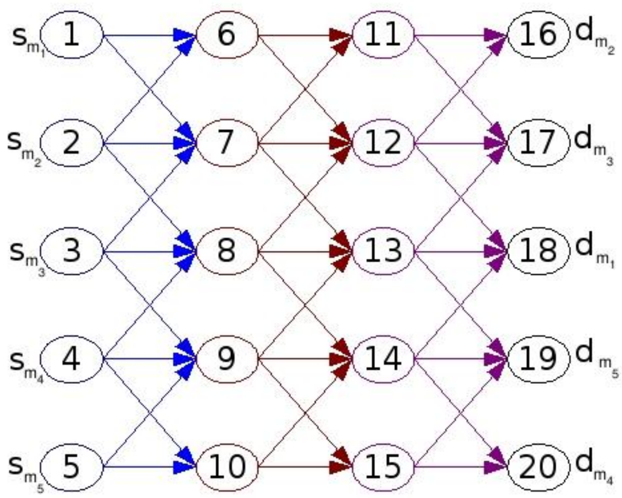

Figure 4.15: Five messages network topology.

Figure 4.15 shows the source nodes s_m and the sink nodes d_m for each message m. For example $s_{m_1} = v_1$ is the source node of message m_1 and the sink node for message m_1 is $d_{m_1} = v_{18}$. The size of each message is 500 $Kbits$ and the reduction rate at each sink node is 100 $Kbits$, we have $z_{m_i} = 500$, $R_{m_i} = 100$, $i \in 1, \ldots, 5$. Furthermore, assume the following time slot assignation:

Time slot	Colour
t_1, t_4, t_7, t_{10}	blue
v_2, v_5, v_8, v_{11}	red
v_3, v_6, v_9, v_{12}	magenta

Table 4.13: Time slot assignation table for five messages.

We solved the non-linear programing problem $\widehat{\Theta}$ for five messages being sent through $\widetilde{G}\left(\widetilde{V}, \widetilde{E}\right)$ with the set of parameters given in Table 4.3 as well as node colouring and time slot colouring given in Tables 4.12 and 4.13, respectively. After 33 iterations executed in 0.4 seconds, the solver Knitro found a local minimiser with objective function value of 0.01022744 mW. The results for p_e and $c_{e,m}$ at this local minimiser are shown in Table 4.15.

Edge (v_i, v_j)	Power $p_{(v_i,v_j)}$	$c_{(v_i,v_j),m_1}$	$c_{(v_i,v_j),m_2}$	$c_{(v_i,v_j),m_3}$	$c_{(v_i,v_j),m_4}$	$c_{(v_i,v_j),m_5}$
(v_1, v_6)	0.00090369	406.39				
(v_1, v_7)	0.00262205	1093.61				
(v_2, v_6)	0.00137324		574.05			
(v_2, v_7)	0.00144031		595.05			
(v_2, v_8)	0.00077044		330.91			
(v_3, v_7)	0.00188902			777.55		
(v_3, v_8)	0.00103666			441.74		
(v_3, v_9)	0.00064796			280.71		
(v_4, v_8)	0.00086544				367.77	
(v_4, v_9)	0.00137037				570.25	
(v_4, v_{10})	0.00135036				561.98	
(v_5, v_9)	0.00234516					980.26
(v_5, v_{10})	0.00119760					519.74
(v_6, v_{11})	0.0007858		327.83			
(v_6, v_{12})	0.0014920	406.39	246.21			
(v_7, v_{11})	0.0013658		273.81	299.76		
(v_7, v_{12})	0.0027324	529.9	321.24	270.54		
(v_7, v_{13})	0.0018488	563.72		207.24		
(v_8, v_{12})	0.0012440		330.91	207.05		
(v_8, v_{13})	0.0005612			234.69		
(v_8, v_{14})	0.0008661				367.77	
(v_9, v_{13})	0.0015993			280.71		397.34
(v_9, v_{14})	0.0013717				272.46	291.84
(v_9, v_{15})	0.0014213				297.78	291.08
(v_{10}, v_{14})	0.0012529				268.75	260.83
(v_{10}, v_{15})	0.0012972				293.23	258.91
(v_{11}, v_{16})	0.00142285		601.65			
(v_{11}, v_{17})	0.00072822			299.76		
(v_{12}, v_{16})	0.00213245		898.36			
(v_{12}, v_{17})	0.00115653			477.6		
(v_{12}, v_{18})	0.00220283	936.29				
(v_{13}, v_{17})	0.00170067			722.64		
(v_{13}, v_{18})	0.00135113	563.71				
(v_{13}, v_{19})	0.00096986					397.34
(v_{14}, v_{19})	0.00131290					552.67
(v_{14}, v_{20})	0.00214463				908.98	
(v_{15}, v_{19})	0.00130886					549.99
(v_{15}, v_{20})	0.00141074				591.02	

Table 4.15: Transmission power and sizes of pieces for messages $m_i, i = 1, \ldots 5$.

Furthermore, we applied Algorithm 1 with the following set of parameters: $M = \{m_i, i = 1, \ldots 5\}$, $\widehat{Z} = z_{m_i} = 600,\ i = 1, \ldots, 5,\ \widehat{R} = R_{m_i} = 100,\ i = 1, \ldots, 5,\ \gimel = 50,$ $\beth = 10,\ \widehat{T} = t_{j(m_i)} = t_3,\ i = 1, \ldots, 5,$ digraph $\widetilde{G}\left(\widetilde{V}, \widetilde{E}\right)$ and time slot assignation given in Table 4.13. Algorithm 1 solved ten different experiments before the problem $\widehat{\Theta}$ became unfeasible.

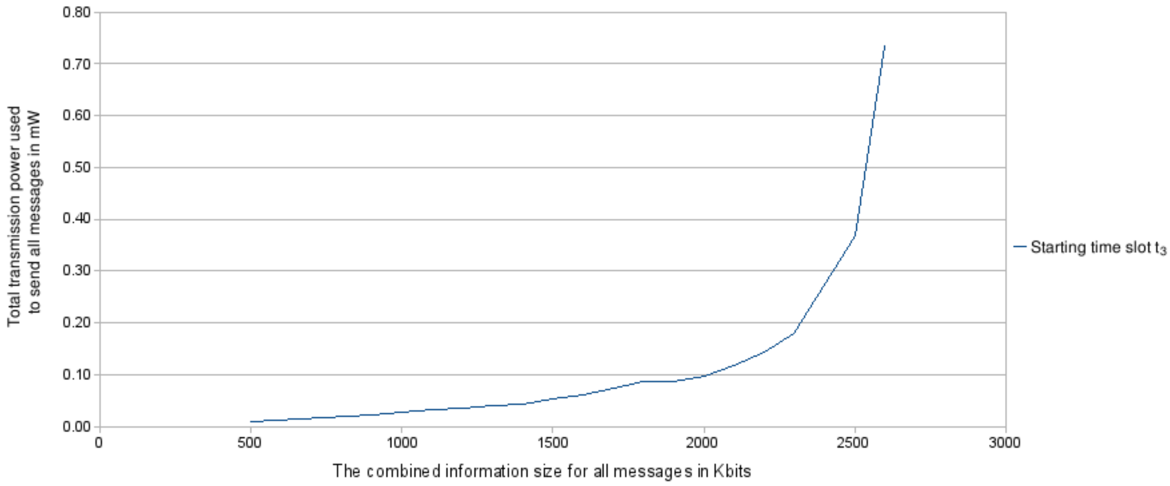

Figure 4.16: Total power used to send five messages.

In Figure 4.16, we can see a similar pattern as in Figures 4.8 and 4.13. That is, for one and two messages the maximum amount of data that can go through the network when the starting time slot is t_3, is 10000 $Kbits$ and 7500 $Kbits$, respectively, while for five messages it is 2500 $Kbits$.

Figure 4.17 illustrates the flows for each of the five messages.

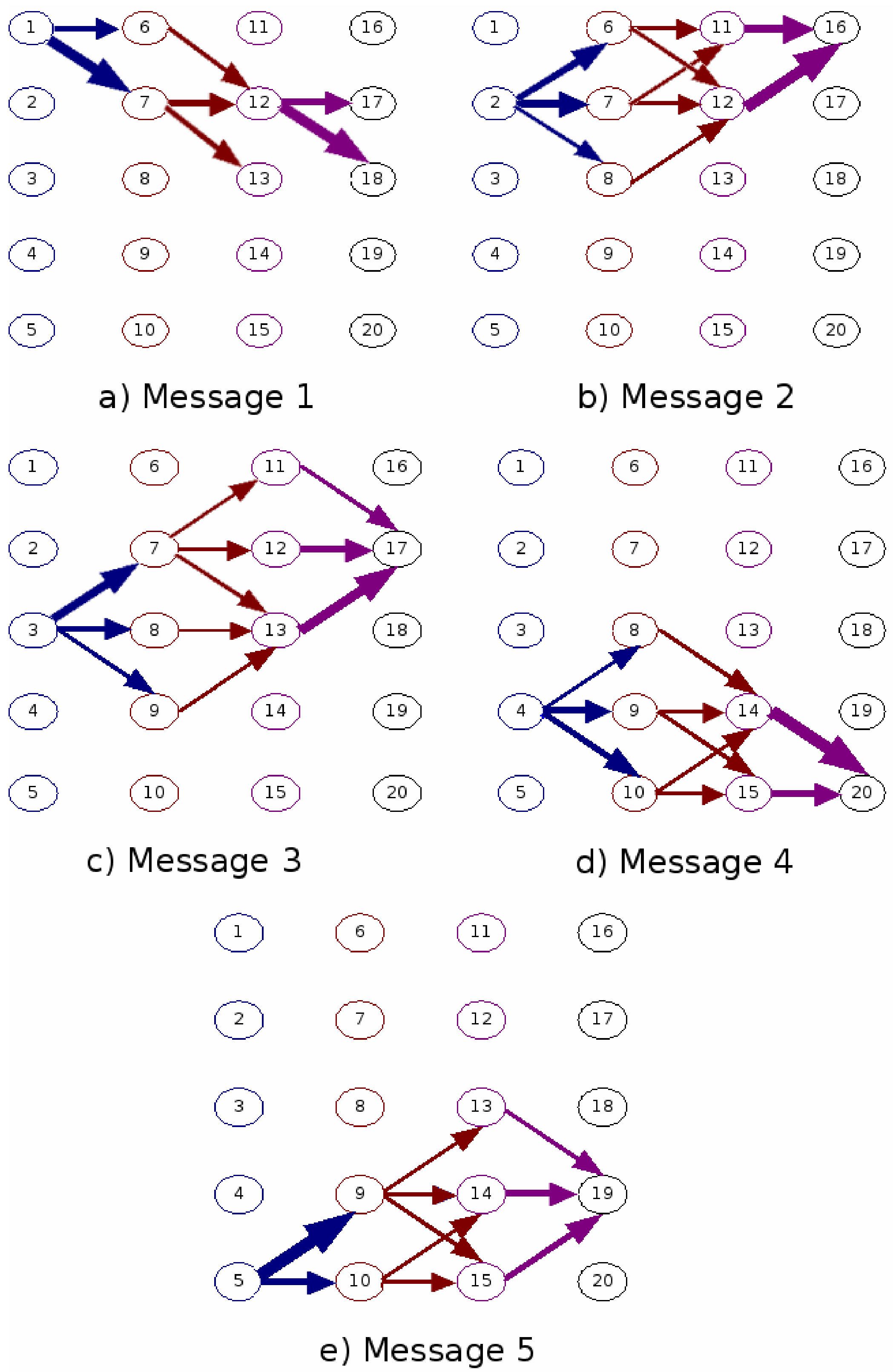

Figure 4.17: The flows of the five messages.

In Figure 4.17, the thickness of each arrow represents the size of information flowing through the edge.

4. Fourth set of experiments. This set of experiments refers to sending one message through a "larger" wireless network. These kinds of networks are related to so-called wireless mesh back-haul networks (for more details, see [87]). Figure 4.19 shows two representation of the same network where two messages with different sizes are sent.

We consider the digraph $G(V, E)$ where $V = \{v_1, \ldots, v_{15}\}$,

$$\begin{aligned} E \;=\; & \{(v_1, v_2), (v_1, v_3), (v_1, v_4), (v_1, v_5), (v_2, v_6), (v_2, v_7), (v_2, v_8), (v_2, v_9), (v_2, v_{10}), \\ & (v_3, v_6), (v_3, v_7), (v_3, v_8), (v_3, v_9), (v_3, v_{10}), (v_4, v_6)\,(v_4, v_7), (v_4, v_8), (v_4, v_9), \\ & (v_4, v_{10}), (v_5, v_6), (v_5, v_7), (v_5, v_8), (v_5, v_9), (v_5, v_{10}), (v_6, v_{11}), (v_6, v_{12}), (v_6, v_{13}), \\ & (v_6, v_{14}), (v_7, v_{11}), (v_7, v_{12}), (v_7, v_{13}), (v_7, v_{14}), (v_8, v_{11}), (v_8, v_{12}), (v_8, v_{13}), \\ & (v_8, v_{14}), (v_9, v_{11}), (v_9, v_{12}), (v_9, v_{13}), (v_9, v_{14}), (v_{11}, v_{15}), (v_{12}, v_{15}), (v_{13}, v_{15}), (v_{14}, v_{15})\} \end{aligned}$$

and we have the following colour assignation to the nodes:

Node	Colour
$v_1, v_6, v_7, v_8, v_9, v_{10}$	blue
$v_2, v_3, v_4, v_5, v_{11}, v_{12}, v_{13}, v_{14}$	red

Table 4.16: Colour assignation for $v \in \widetilde{V}$.

The graphical representation $G(V, E)$ is given in Figure 4.18.

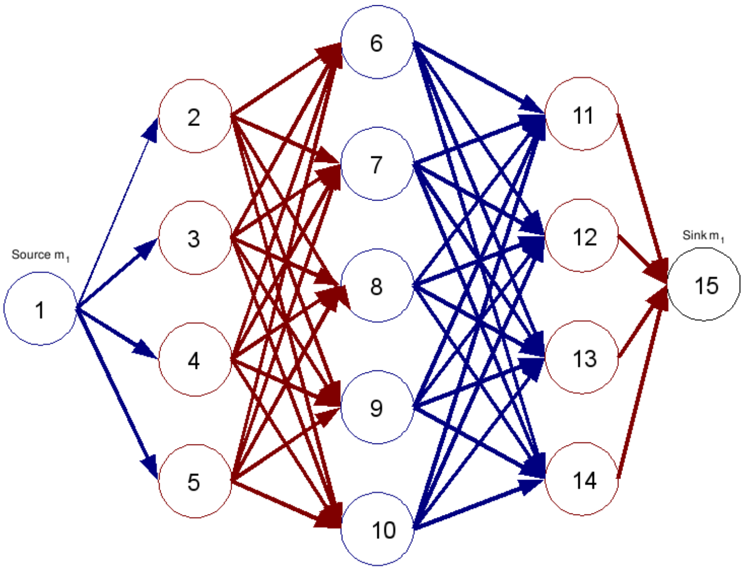

Figure 4.18: Five messages network topology.

Figure 4.18 shows that the source node of m_1 is v_1 and the sink node of m_1 is v_{15}. We present two experiments for two messages with different sizes. The first message m_1 has a size of 3500 $Kbits$ and the reduction rate at the sink node is 500 $Kbits$. The second message has a size of 22000 $Kbits$ and the reduction rate at the sink node is 500 $Kbits$. The starting time slot in both cases is t_4 which implies $t^{Max} = 48$ and assume the following time slot assignation:

Time slot	Colour
$t_i, i = \{1, 3, 5, \ldots 45, 47\}$	blue
$t_i, i = \{2, 4, 6, \ldots 46, 48\}$	red

Table 4.17: Time slot assignation table.

Figure 4.19a) shows the flow of a message with size $z_{m_1} = 3500$ $Kbits$ while Figure 4.19b) shows the flow of a message with size $z_{m_1} = 22000$ $Kbits$. Here, as above, the thickness of the arrow reflects the size of information flowing through the corresponding edge and the dotted lines reflect that no information is sent through the corresponding edge.

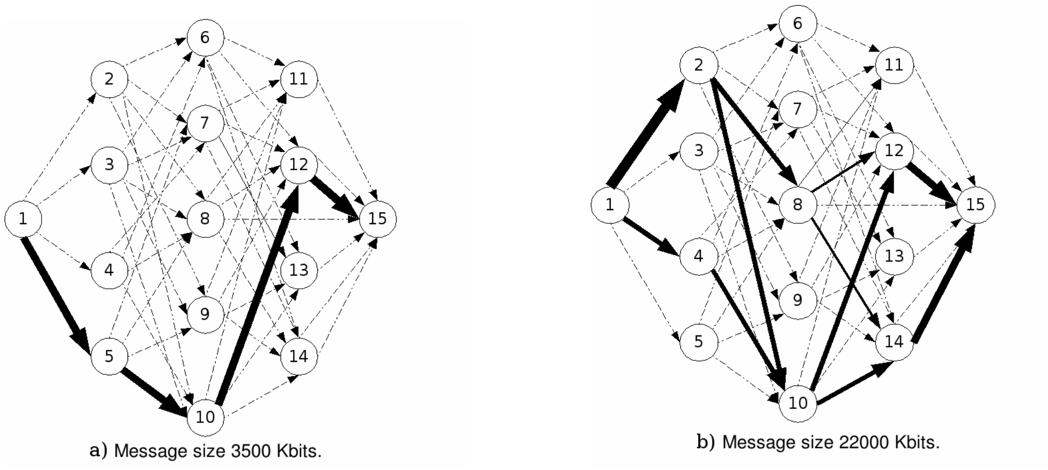

Figure 4.19: Back-haul network with 15 nodes graph.

We can see from Figure 4.19 that when the message is “small” the complete message flows through the same path, but when the size is “larger” it is spread into different paths. This is due to the node’s buffer constraint.

4.5 Summary

In this chapter we presented a mathematical model which describes an ad hoc wireless network. We proved two theorems showing that at a local minimiser of the corresponding non-linear programming problem, the maximum capacity of the edge $e \in E$ is exploited. From an engineering point of view, this means that the maximum capacity of the channel over a link is used, while assuring a good QoS for the link. The mathematical model includes the properties of TDMA and CDMA systems which allow multiple messages to flowing through a single link in a specific time slot.

We made a series of experiments for different initial conditions and topologies, with the goal of finding local minimisers. With the purpose of conducting a sensitivity analysis we repeatedly tested the programming problem by varying the initial conditions until it was no longer feasible.

We analysed an interesting constraint pattern: when the constraints (4.4) make the problem $\widehat{\Theta}$ unfeasible then the relationship between the combined size of messages and the total transmission power used is similar to a quadratic function while when the constraints (4.16) make the problem $\widehat{\Theta}$ unfeasible, then the relationship between the combined size of messages and the total transmission power used is similar to an exponential function.

Chapter 5

Second Mathematical Modelling Approach

In this chapter, we present a second mathematical model in this thesis that represents an ad hoc wireless network system. First, we define the set of decision variables. Second, we describe an ad hoc wireless network by a directed graph and we consider two filter types SUMF and MMSE. Third, we introduce the objective function and the constraints which represent properties of the ad hoc wireless network. The so obtained non-linear programming problem will be denoted by Φ. Fourth, we present the methodology used to solve the mathematical model. Fifth, the solution for different ad hoc network topologies using the SUMF and MMSE filters and different parameters are simulated. We recall Chapters 2 and 3 where the corresponding technical properties and mathematical concepts are discussed.

The requirements for this mathematical model are:

- Optimal use of battery: The nodes are powered by a battery whose electrical charge has a finite life. Therefore, it is important to make it last as long as possible.

- Guarantee a minimum level of SNR ratio target for each link in the network.
- It is also necessary to find at least one route for each message in the network.

In order to tackle these challenges, we will consider joint routing and power control for CDMA.

5.1 Mathematical Model

In this section we present the set of variables, the set of constraints and the objective function for the mathematical model of an ad hoc wireless network for CDMA systems. It is important to mention that for this model, we assume simultaneous transmission and reception at intermediate nodes. That is, we are not considering any half-duplex constraints.

Since each device in the network is allowed to send messages, there can be several messages flowing through the network concurrently (CDMA property).

The behaviour of the message going through the network will be mathematically described by a set of constraints, decision variables and lower and upper bounds. For the mathematical model, each device will be assigned a node and each link an edge.

Let us examine an ad hoc wireless network consisting of r devices. These devices can communicate with each other via a wireless link. Without loss of generality, consider a message originated at a particular source device and destined to arrive at a particular sink device. The rest of the devices are considered as intermediate devices. The intermediate devices will be source devices or sink devices according to their function (sending or receiving mode). The link connectivity between devices is given in Definition 2.1. The message is sent from the origin source device through the network until it reaches its destination via an unknown path. The path is set by the connectivity and link transmission power and capacity. In this scheme there are two possibilities:

- The whole message is sent as a single piece.
- The message is split into smaller pieces, depending on the capacity of the links. Then, each piece could be considered as a smaller message, which can be sent as a single piece.

Independently of how it is sent, the complete message will arrive at its destination integrated as a whole.

The behaviour of the message through the network will be mathematically described now by a set of constraints, decision variables and lower and upper bounds.

Since the network consists of r devices, the set of nodes is denoted by $V = \{v_1, \ldots, v_r\}$. The set of links is denoted by E according to Definition 2.1. Then, ad hoc wireless network, can be represented as a digraph which is an ordered pair $G = (V, E)$. For the set of edges E, we define E^+ and E^- as in section 4.1.

One can either send a message from the source node to the sink node by a single hop, (in case that there is a direct edge from source node to sink node), or by consecutive multi-hops. Also, since an ad hoc wireless network consists of several devices, it is possible for more than one message to go through the network (CDMA) at the same time. Let us define the set of messages $M = \{m_1, \ldots, m_L\}$ and let $z_{m_i} \in \mathbb{R}_+$, $i = 1, \ldots, L$ be the size of the message m_i. For each message $m_i \in M$, $i = 1, \ldots, L$ a source node $s_{m_i} \in V$ and a sink node $d_{m_i} \in V$ with $s_{m_i} \neq d_{m_i}$ exist. Since there are different senders and receivers in the network, we face a multi-commodity flow problem (refer to Section 3.3).

5.1.1 Design Variables

To model the characteristics of an ad hoc wireless network outlined in the previous section, it is necessary to define the decision variables involved and their units of measurement, the set of constraints and upper bounds unless otherwise specified, the indices e and m will always vary in the sets E and M respectively.

As we mentioned in the previous Section 5.1, a message m going through an ad hoc wireless network, may be sent as a complete message or it may be split into several parts. The size of the part of the message m going through the edge e is denoted by the decision variable $c_{e,m} \in \mathbb{R}_*$, measured in Kbits.

The transmission power allocated at the edge e is denoted by $p_e \in \mathbb{R}_*$, (measured in watts or dBm).

As a matter of simplicity, we will frequently make use of the following vector notations:

$$\begin{aligned} \mathbf{p} &= (p_e)_{e \in E} \\ \mathbf{c} &= (c_{e,m})_{e \in E, m \in M} \end{aligned} .$$

5.1.2 Interference Function

In Section 2.3.3 we introduced the notation of SUMF filter as well as MMSE filter respectively. Now, let us recall them.

An edge $e \in E$ is called *active* if a part of a message is sent through it. Each message $m \in M$ going through an active edge $e \in E$ in the system, is affected by interference, caused by the other neighbourhood devices in the system and by the background noise $\eta \in \mathbb{R}_+$. Let us define the function

$$\mathbf{I} : \mathbb{R}^{|E|} \to \mathbb{R}^{|E|}$$

by

$$I_e(\mathbf{p}) := \gamma_e \sum_{l \neq e} p_l + \eta^2$$

where $I_e(\mathbf{p})$, $e \in E$ denote the components of the function $\mathbf{I}(\mathbf{p})$. Then, the function $I_e(\mathbf{p})$ represents the effective interference caused by transmission via $l \in E,\ l \neq e$ that must be exceeded by the transmission power of the fixed edge $e \in E$.

We recall definitions of SUMF filter and MMSE filter given in Subsection 2.3.3:

SUMF filter:

$$I_e^{SUMF}(\mathbf{p}) := \gamma_e \left(\sum_{l \neq e} |r_{e,l}|^2 p_l + \eta^2 \right) \tag{5.1}$$

and *MMSE* filter:

$$I_e^{MMSE}(\mathbf{p}) := \frac{\gamma_e}{\left|(\mathbf{WR})_{(e,e)}\right|^2} \left(\sum_{l \neq e} \left|(\mathbf{WR})_{(e,l)}\right|^2 p_l + \beta_e \eta^2 \right). \tag{5.2}$$

They will be part of the set of constraints in this mathematical model. The values of the matrix $\mathbf{R}$ are calculated with the Matlab code presented in Appendix C. This code was developed by Dr. Armin Dekorsy (Qualcomm).

5.1.3 Parameters

In this subsection we present the parameters and their units of measurement as used in the set of constraints.

The maximum size of information that can go through an edge $e \in E$ is given by $N_e^{Max} \in \mathbb{R}_+$ (measured in Kbits).

The maximum bound for all transmission powers p_e is given as $P^{Max} \in \mathbb{R}_+$, (measured in watts or dBm).

We will further use the bandwidth $B \in \mathbb{R}_+$ and the background noise $\eta \in \mathbb{R}_+$(both defined in Subsection 4.1.2).

5.1.4 Set of Constraints

In this subsection we present the upper bounds for the decision variables and the further constraints for the mathematical model.

5.1.4.1 Upper bounds for our variables

For any edge $e \in E$, the sum of all sizes $c_{e,m}$ cannot exceed the maximum capacity N_e^{Max}:

$$\sum_{m \in M} c_{e,m} \leq N_e^{Max}. \tag{5.3}$$

For any edge $e \in E$, the transmission powers p_e cannot exceed the maximum power P^{Max}:

$$p_e \leq P^{Max}. \tag{5.4}$$

5.1.4.2 Constraints of the system

We need to assure for each message $m \in M$:

- The complete message m leaves the source node $s_m \in V$:

$$\sum_{e \in E^+(s_m)} c_{e,m} = z_m. \tag{5.5}$$

- The complete message m is received at the sink node $d_m \in V$:

$$\sum_{e \in E^-(d_m)} c_{e,m} = z_m. \tag{5.6}$$

- At each intermediate node (i.e. any node that is neither s_m nor d_m) the size of the ingoing messages is equal to the outgoing messages:

$$\sum_{e \in E^+(i)} c_{e,m} = \sum_{e \in E^-(i)} c_{e,m} \qquad i \in \{V \setminus \{s_m, d_m\}\}. \tag{5.7}$$

The capacity of a channel is given by Shannon's well known formula given in (2.9) for the MMSE filter as

$$\sum_{m \in M} c_{e,m} \leq B \log_2 \left(1 + \frac{p_e}{\sum_{l \neq e} \left| (\mathbf{WR})_{(e,l)} \right|^2 p_l + \eta^2} \right) \tag{5.8}$$

which implies that the sum of all sizes of the parts of the messages $c_{e,m}$ cannot exceed the maximal capacity of the channel.

For the SUMF filter the same constraint is written as

$$\sum_{m \in M} c_{e,m} \leq B \log_2 \left(1 + \frac{p_e}{\sum_{l \neq e} |r_{e,l}|^2 p_l + \eta^2} \right).$$

Note, that in particular if $p_{e,m} = 0 \Longrightarrow c_{e,m} = 0$.

It is necessary that the transmission power allocated at edge e is greater than the corresponding interference, therefore

$$(I_e(\mathbf{p}) - p_e) \leq 0. \tag{5.9}$$

That is, for those edges $e \in E$, it is necessary that the power p_e allocated along the edge e must be greater that the interference function $I_e(\mathbf{p})$.

The set of inequality constraints (5.9) can be modified as:

$$\left(I_e^{SUMF}(\mathbf{p}) - p_e\right) \leq 0 \tag{5.10}$$

or

$$\left(I_e^{MMSE}(\mathbf{p}) - p_e\right) \leq 0 \tag{5.11}$$

depending on the filter used (the SUMF filter (2.5) and the MMSE filter (2.8)).

5.1.4.3 Non Negativity Constraints

The non-negativity of p_e and $c_{e,m}$ is represented by:

$$p_e \geq 0 \tag{5.12}$$

$$c_{e,m} \geq 0. \tag{5.13}$$

5.1.5 The Non-Linear Programming Problem

The objective function is to minimise the total transmission power consumption in the system, with respect to all variables $p_e, c_{e,m}$; it is defined as follows:

$$f(\mathbf{p}, \mathbf{c}) = \sum_{e \in E} p_e. \tag{5.14}$$

Note that this function is monotonically increasing with respect to $(p_e)_{e \in E}$.

Now, we can define the non-linear programming problem (Φ):

$$\text{minimise } f(\mathbf{p}, \mathbf{c}) = \sum_{e \in E} p_e$$

$$\text{subject to (5.3)-(5.9) and (5.12)-(5.13).}$$

Denote the feasible set of Φ by S'. Obviously, S' is a subset of the set of $\mathbb{R}^{|E|*|M|+|E|}$.

In the non-linear problem Φ the constraint (5.9) can be replaced either by (5.10) or (5.11) depending on which filter (SUMF or MMSE) is used.

5.2 Methodology for the Numerical Solution

Our goal is to solve the optimisation problem Φ of routing and power control, in order to minimise the power used by each node in the network:

$$\text{minimise } f(\mathbf{p}, \mathbf{c}) = \sum_{e \in E} p_e$$

subject to (5.3)-(5.9), (5.12)-(5.13).

In the non-linear problem Φ the constraint (5.9) can be replaced either by (5.10) or (5.11) depending on which filter (SUMF or MMSE) is used.

In this section we will propose a methodology for the numerical solution of problem Φ. We used the packages Ipopt [88] and Knitro [30] as well as AMPL as the interface for these two packages (refer to Section 3.4 or [42]). The set of constraints is written with AMPL's syntax and is saved into a model file. For a specific example, the value of the parameters such as upper and lower bounds, constants, etc. will be saved in a data file. In the Appendix (B) there is a *model file* which is the "transcription" of the constraints and the objective function into AMPL's syntax. As we mentioned before, we have two different filters and, therefore, we have two different model files depending on the filter used.

Note: In order to simplify the visualisation of the following figures we enumerate the nodes instead of using the node notation used in the mathematical model.

5.3 Numerical Results

In this section we present three sets of experiments. We replace the interference function (5.9) by either the SUMF filter (5.1) or by the MMSE filter (5.2). Since all experiments presented in this section involve more than one message being sent through the network, we are facing a multi-commodity network flow problem (refer to Section 3.3). The first set of experiments has a small network graph where three messages are sent from three different sources to one sink only. The second set of experiments consists of two messages being sent through a network with six nodes and six edges. The third set of experiments consists of two messages being sent through a network with eight nodes and twelve edges. In each set of experiments we use the solvers Knitro and Ipopt. We present a comparative analysis between the solutions obtained from these solvers.

- First set of experiments: Consider a network with 4 nodes and 3 edges. The digraph $G(V,E)$ associated with this network has the set of nodes $V = \{v_1, v_2, v_3, v_4\}$ and the set of edges $E = \{(v_1, v_4), (v_2, v_4), (v_3, v_4)\}$. The graphical representation of $G(V,E)$ is given in Figure 5.1 and we will refer to it as a spider digraph:

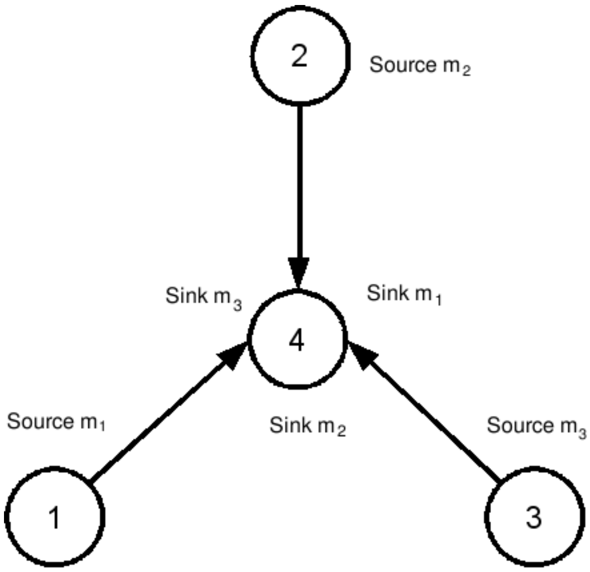

Figure 5.1: Spider network.

Consider three messages m_1, m_2 and m_3 to be sent through this network. The source node of message m_1, m_2 and m_3 is v_1, v_2 and v_3 respectively and the sink node of all messages v_4. The size of each message is $z_{m_1} = z_{m_2} = z_{m_3} = 1000\, Kbits$.

Table 5.1 shows the parameter values used to solve the optimisation problem Φ in this first experiment.

Parameter	Value	Units
N_e^{Max}	3000	$kbits$
P_e^{Max}	2	W
η	0.008	mW
B	5	MHz
γ	-13.5	dB
$\mathbf{R}$	$\begin{pmatrix} 1 & 0.06 & 0.25 \\ 0.06 & 1 & -0.19 \\ 0.25 & -0.19 & 1 \end{pmatrix}$	-

Table 5.1: Parameter values for the first experiment.

First, we solved this problem using the SUMF filter. The solvers Knitro and Ipopt calculated the following results:

Ipopt			Knitro		
Message	Edge	p_e	Message	Edge	p_e
m_1	(v_1, v_4)	0.0169311	m_1	(v_1, v_4)	0.0169311
m_2	(v_2, v_4)	0.0169311	m_2	(v_2, v_4)	0.0169311
m_3	(v_3, v_4)	0.0169311	m_3	(v_3, v_4)	0.0169311

Table 5.2: Transmission powers p_e for SUMF filter.

As we can see from Table 5.2 both solvers found the same local minimiser. The main difference between them is the number of iterations and the time taken to solve this problem. Knitro found the local minimiser after 11 iterations executed in 0.01 seconds while Ipopt took about 0.03 seconds and 4 iterations. The objective function value is 0.0507933.

Now, we solve this problem using the MMSE filter. The solvers Knitro and Ipopt calculated the following results:

Ipopt			Knitro		
Message	Edge	Power used (mW)	Message	Edge	Power used (mW)
m_1	(v_1, v_4)	0.0169311	m_1	(v_1, v_4)	0.0169311
m_2	(v_2, v_4)	0.0169311	m_2	(v_2, v_4)	0.0169311
m_3	(v_3, v_4)	0.0169311	m_3	(v_3, v_4)	0.0169311

Table 5.3: Transmission powers p_e for MMSE filter.

As we can see from Table 5.3 both solvers found the same local minimiser. The main difference between them is the number of iterations and the time consumed to solve this problem. Knitro found the local minimiser in about 10 minutes and after 50312 iterations while Ipopt took 7 iterations executed in 0.05 seconds. The objective function value is 0.0507933 mW.

Independently of the filter used, the minimum value of the objective function is 0.0507933 mW, but a significant difference in the consumption of computational time exists.

- Second set of experiments: Consider a network consisting of six nodes and six edges. The digraph $G(V, E)$ associated with this network has the set of nodes:

$$V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$$

and the set of edges:

$$E = \{(v_1, v_2), (v_1, v_3), (v_2, v_4), (v_3, v_5), (v_4, v_6), (v_5, v_6)\}.$$

The graphical representation of $G(V, E)$ is given in Figure 5.2.

Table 5.4 shows the set parameter values used to solve the optimisation problem Φ in this second experiment.

Parameter	Value	Units
N_e^{Max}	3000	$kbits$
P_e^{Max}	2	W
η	0.008	mW
B	5	MHz
γ	-13.5	dBm
$\mathbf{R}$	$\begin{pmatrix} 1 & 0.06 & \varepsilon & 0.25 & \varepsilon & 0.06 \\ 0.06 & 1 & -0.19 & -0.06 & -0.06 & -0.13 \\ \varepsilon & -0.19 & 1 & -0.25 & 0.5 & -0.06 \\ 0.25 & -0.06 & -0.25 & 1 & -\varepsilon & 0.06 \\ \varepsilon & -0.06 & 0.5 & -\varepsilon & 1 & -0.19 \\ 0.06 & -0.13 & -0.06 & 0.06 & -0.19 & 1 \end{pmatrix}$ where $\varepsilon = 6.94E - 018$	-

Table 5.4: Set of parameters for the second experiment.

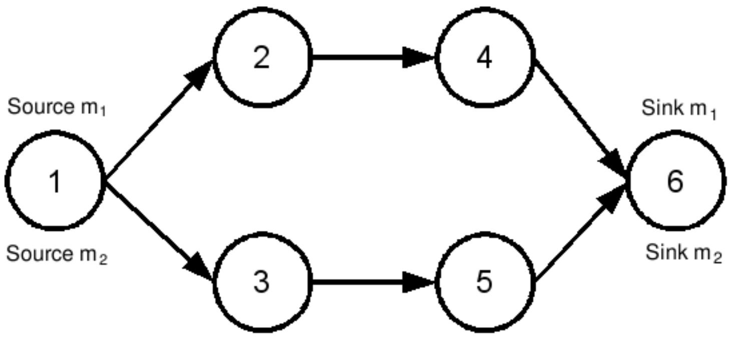

Figure 5.2: Six nodes graph with two messages.

There are two messages m_1 and m_2 to be sent through this network. Both messages have the same source node v_1 and the same sink node v_6. The size of m_1 and m_2 are $z_{m_1} = z_{m_2} = 1000\, Kbits$.

First, we solved this problem using the SUMF filter. The solvers Knitro and Ipopt calculated the following results:

Ipopt				Knitro			
Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}	Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}
(v_1, v_2)	500.014	500.014	0.0463774	(v_1, v_2)	21.3603	981.928	0.0465182
(v_1, v_3)	499.986	499.986	0.046375	(v_1, v_3)	978.64	18.072	0.0462339
(v_2, v_4)	500.014	500.014	0.0463774	(v_2, v_4)	21.3603	981.928	0.0465182
(v_3, v_5)	499.986	499.986	0.046375	(v_3, v_5)	978.64	18.072	0.0462339
(v_4, v_6)	500.014	500.014	0.0463774	(v_4, v_6)	21.3603	981.928	0.0465182
(v_5, v_6)	499.986	499.986	0.046375	(v_5, v_6)	978.64	18.072	0.0462339

Table 5.5: Transmission powers p_e for SUMF filter.

Table 5.5 displays the transmission powers used and the sizes of the pieces sent for each message. After 50008 iterations executed in 8 minutes, Knitro found a local minimiser with objective function value 0.2782563 mW, while Ipopt took 0.02 seconds and 7 iterations to find the local minimiser with objective function value 0.2782572 mW.

Then we solved this problem using the MMSE filter. The solvers Knitro and Ipopt calculated the following results:

Ipopt				Knitro			
Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}	Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}
(v_1, v_2)	500.044	500.044	0.04638	(v_1, v_2)	0	0	0
(v_1, v_3)	499.956	499.956	0.0463723	(v_1, v_3)	1000	1000	0.0708082
(v_2, v_4)	500.044	500.044	0.04638	(v_2, v_4)	0	0	0
(v_3, v_5)	499.956	499.956	0.0463723	(v_3, v_5)	1000	1000	0.0708082
(v_4, v_6)	500.044	500.044	0.04638	(v_4, v_6)	0	0	0
(v_5, v_6)	499.956	499.956	0.0463723	(v_5, v_6)	1000	1000	0.0708082

Table 5.6: Transmission powers p_e for MMSE filter.

Table 5.6 displays the transmission powers used and the sizes of the pieces sent for each message. After 52312 iterations executed in 10 minutes, Knitro found a local minimiser with objective function value mW, 0.21242465 while Ipopt took 0.05 seconds and 18 iterations to find the local minimiser with objective function value 0.2782569 mW.

Comparing these results found for each filter and solver, we get:

For the SUMF filter, Ipopt splits the message into two equal parts, while Knitro splits it in an approximate proportion of 98% to 2%, per edge. Both find almost the same value for the objective function.

For the MMSE filter, Ipopt again splits the message into two equal parts and routes the parts using all edges while Knitro routes the complete message using edges $(v_1, v_3), (v_3, v_5)$ and (v_5, v_6). Again, Knitro found a better objective function value, but the computational time is extremely long.

- Third set of experiments: Consider a network with 8 nodes and 12 edges. The digraph $G(V, E)$ associated with this network has the set of nodes

$$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$$

and the set of edges

$$\begin{aligned} E &= \{(v_1, v_3), (v_1, v_4), (v_2, v_3)\,(v_2, v_4), (v_3, v_5), (v_3, v_6) \\ &\quad (v_4, v_5), (v_4, v_6), (v_5, v_7)\,(v_5, v_8), (v_6, v_7), (v_6, v_8)\}. \end{aligned}$$

The graphical representation of $G(V, E)$ is given in Figure 5.3.

Table 5.7 shows the set of parameter values used to solve the optimisation problem Φ in this third experiment.

Parameter	Value	Units
N_e^{Max}	3000	$kbits$
P_e^{Max}	2	W
η	0.008	mW
B	5	MHz
γ	-13.5	dBm
R	$\begin{pmatrix} 1 & 0.06 & \varepsilon & 0.25 & \varepsilon & 0.06 & -0.13 & -0.44 & -0.25 & -0.25 & 0.19 & -0.13 \\ 0.06 & 1 & -0.19 & -0.06 & -0.06 & -0.13 & 0.19 & 0.13 & 0.06 & -0.06 & -0.13 & -0.19 \\ \varepsilon & -0.19 & 1 & -0.25 & 0.5 & -0.06 & -0.25 & 0.19 & -\varepsilon & \varepsilon & 0.06 & 0.13 \\ 0.25 & -0.06 & -0.25 & 1 & -\varepsilon & 0.06 & -0.13 & -0.31 & -0.13 & -\varepsilon & 0.06 & -0.38 \\ \varepsilon & -0.06 & 0.5 & -\varepsilon & 1 & -0.19 & -0.25 & -0.06 & -\varepsilon & -\varepsilon & -0.06 & -\varepsilon \\ 0.06 & -0.13 & -0.06 & 0.06 & -0.19 & 1 & 0.06 & -0.13 & -0.06 & -0.06 & -\varepsilon & -0.06 \\ -0.13 & 0.19 & -0.25 & -0.13 & -0.25 & 0.06 & 1 & 0.19 & \varepsilon & \varepsilon & 0.19 & 0.13 \\ -0.44 & 0.13 & 0.19 & -0.31 & -0.06 & -0.13 & 0.19 & 1 & 0.19 & 0.06 & 0.13 & -0.06 \\ -0.25 & 0.06 & -\varepsilon & -0.13 & -\varepsilon & -0.06 & \varepsilon & 0.19 & 1 & 0.13 & -0.06 & -\varepsilon \\ -0.25 & -0.06 & \varepsilon & -\varepsilon & -\varepsilon & 0.06 & \varepsilon & 0.06 & 0.13 & 1 & 0.19 & -\varepsilon \\ 0.19 & -0.13 & 0.06 & 0.06 & -0.06 & -\varepsilon & 0.19 & 0.13 & -0.06 & 0.19 & 1 & 0.19 \\ -0.13 & -0.19 & 0.13 & -0.38 & -\varepsilon & -0.06 & 0.13 & -0.06 & -\varepsilon & -\varepsilon & 0.19 & 1 \end{pmatrix}$ where $\varepsilon = 6.94E - 018$	-

Table 5.7: Set of parameters for the third experiment.

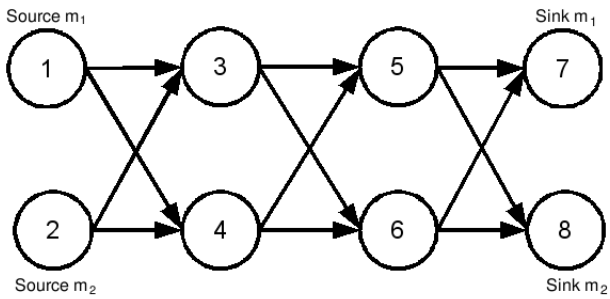

Figure 5.3: Eighth node graph with two messages.

We consider two messages being sent through this network as shown in Figure 5.3. The first message m_1 leaves from node v_1 and its sink node is v_7 while the second message m_2 leaves from node v_2 and its sink node is v_8. The sizes of m_1 and m_2 are $z_{m_1} = z_{m_2} = 1000\ Kbits$. First, we solved this problem using the SUMF filter. The solvers Knitro and Ipopt calculated the following results:

Ipopt				Knitro			
Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}	Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}
(v_1, v_3)	499.998	0	0.0272783	(v_1, v_3)	1000	0	0.0431389
(v_1, v_4)	500.002	0	0.0272785	(v_1, v_4)	0	0	0
(v_2, v_3)	0	500.011	0.027279	(v_2, v_3)	0	1000	0.0431389
(v_2, v_4)	0	499.989	0.0272778	(v_2, v_4)	0	0	0
(v_3, v_5)	249.996	250.01	0.0272787	(v_3, v_5)	0	0	0
(v_3, v_6)	250.002	250.001	0.0272785	(v_3, v_6)	1000	1000	0.0806935
(v_4, v_5)	249.998	249.999	0.0272782	(v_4, v_5)	0	0	0
(v_4, v_6)	250.004	249.99	0.0272781	(v_4, v_6)	0	0	0
(v_5, v_7)	499.994	0	0.027278	(v_5, v_7)	0	0	0
(v_5, v_8)	0	500.009	0.0272789	(v_5, v_8)	0	0	0
(v_6, v_7)	500.006	0	0.0272787	(v_6, v_7)	1000	0	0.0431389
(v_6, v_8)	0	499.991	0.0272779	(v_6, v_8)	0	1000	0.0431389

Table 5.8: Transmission powers p_e for SUMF filter.

For the SUMF filter, the results are presented in Table 5.8. In the solution of the Ipopt solver, the first message was split and their pieces followed the routes as indicated in Figure 5.4.

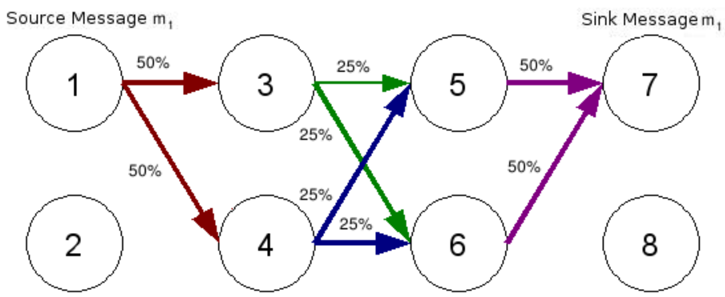

Figure 5.4: Message m_1 flowing through the network.

For message m_2, we have the routes followed by the pieces as indicated in Figure 5.5.

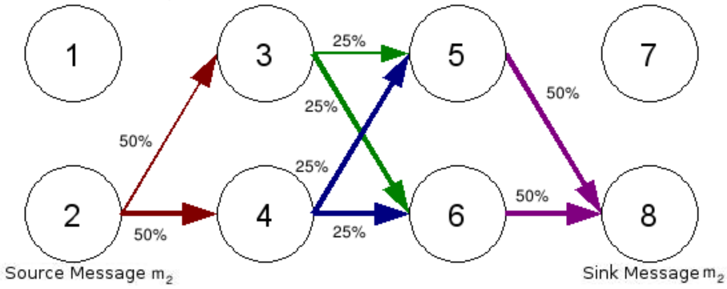

Figure 5.5: Message m_2 flowing through the network.

In the solution calculated by Knitro solver none of the messages was split and the edges used were (v_1, v_3) , (v_3, v_6) and (v_6, v_7) for message m_1 and (v_2, v_3) , (v_3, v_6) and (v_6, v_8) for message m_2. After 7 iterations executed in 0.03 seconds Ipopt found a local minimiser with objective function value 0.32734 mW, while Knitro found a local minimiser after 120 iterations executed in 1 minute with objective function value 0.25324 mW.

For the MMSE filter the results are presented in Table 5.9.

Ipopt				Knitro			
Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}	Edge	c_{m_1,e_i}	c_{m_2,e_i}	p_{e_i}
(v_1, v_3)	0	0	0	(v_1, v_3)	0.00739132	0	3.41458e-07
(v_1, v_4)	1000	0	0.0431389	(v_1, v_4)	999.992	0	0.0431393
(v_2, v_3)	0	0	0	(v_2, v_3)	0	0.0337449	1.55896e-06
(v_2, v_4)	0	1000	0.0431389	(v_2, v_4)	0	999.965	0.0431383
(v_3, v_5)	0	0	0	(v_3, v_5)	0.00359241	0.0179889	9.97025e-07
(v_3, v_6)	0	0	0	(v_3, v_6)	0.00379892	0.015756	9.03405e-07
(v_4, v_5)	1000	1000	0.0806936	(v_4, v_5)	0.0135845	0.0136702	1.25912e-06
(v_4, v_6)	0	0	0	(v_4, v_6)	999.979	999.953	0.0806925
(v_5, v_7)	1000	0	0.0431389	(v_5, v_7)	0.017177	0	7.93544e-07
(v_5, v_8)	0	1000	0.0431389	(v_5, v_8)	0	0.0316591	1.4626e-06
(v_6, v_7)	0	0	0	(v_6, v_7)	999.983	0	0.0431389
(v_6, v_8)	0	0	0	(v_6, v_8)	0	999.968	0.0431384

Table 5.9: Transmission powers p_e for MMSE.

For the MMSE filter in the solution of the Ipopt solver none of the messages was split and the message m_1 followed the route along the nodes $(v_1, v_4), (v_4, v_5)$ and (v_5, v_7), while the message m_2 followed the route along the nodes (v_2, v_4), (v_4, v_5) and (v_5, v_8). In the solution of the Knitro solver both messages were split. Nevertheless, as seen in Table 5.9, the size of the bigger piece of each message was very close to z_{m_1}.

After 113 iterations executed in 45.3 seconds Ipopt found a local minimiser with objective function value 0.253249mW while Knitro found a local minimiser with (almost the same) objective function value 0.253255mW in 6960 iterations executed in 4 minutes.

From these three set of experiments we can conclude that Ipopt is a much faster solver, but Knitro sometimes finds a better local minimiser. We could have expected to find better solutions for the MMSE filter since it is a more robust filter than the SUMF filter. We may find this over a larger ad hoc network or under different initial parameters.

5.4 Summary

In this chapter we had presented the second mathematical model which describes an ad hoc wireless network. This mathematical model includes a constraint which represents signal filtering. We found that Knitro takes longer to find a local minimiser than Ipopt, but it sometimes finds a better local minimiser. We compared the results of the non-linear programming problem for two different filters (SUMF filter and MMSE filter). We found that the results of the model which uses the MMSE filter are not substantially different from those of the model which uses the SUMF filter.

Chapter 6

Conclusions and Future Work

6.1 Summary and Conclusions

In this thesis we presented two slightly different models in order to describe some characteristic features of an ad hoc wireless network. The first model is presented in Chapter 4 and the second in Chapter 5. These two models are non-convex optimisation problems.

The first mathematical model proposed refers to optimisation for power control, scheduling and routing for TDMA/CDMA systems. In the second mathematical model we considered optimisation of power control and routing for CDMA systems.

As mentioned above, nodes in an ad hoc wireless network have to manage several performance aspects, such as: power management, system management, medium access, routing, etc. while providing a good QoS. For instance, both medium access and routing decisions have a significant impact on power consumption and joint consideration of both can yield more efficient power consumption. The goal of this thesis was to find efficient ways for the transmission of information through the network minimising the total transmission power in the network.

The problem integrates in a complex manner the following components:

- The power control problem of determining the optimal transmission powers to be allocated along the edges.
- The routing problem of determining the path of a message or messages from source to sink along the network.
- The problem of guaranteeing a good QoS for the links connectivity.
- The scheduling problem of determining the time slot when the device can send information (considered only in the first model).
- Filtering the signal (considered only in the second model).

These five elements need to be addressed in order to minimise the total transmission power consumed in the network.

In this thesis we have proved two theorems which in general terms state that, if there exist a local minimiser of the corresponding non-linear optimisation problem exists then the maximum capacity of each channel used is fully exploited.

In the course of our research, we analysed the power control algorithm proposed by Yates [93] and after battling with it for some months, we concluded that the conditions required by this power control algorithm, such as having a standard interference function, were not met by our model.

The model $\widehat{\Theta}$ in Chapter 4 presents an integrated solution of the joint routing, scheduling and power control problem for a multi-hop wireless network. A contribution to this thesis is the embedding of the scheduling problem into the network topology. This procedure helped us to visualise the behaviour of the messages flowing through the network at a certain time slot.

The model Φ in Chapter 5 presents an integrated solution of the joint routing and power control problem for a multi-hop wireless network. We included two different filters: the SUMF filter and the MMSE filter. In theory the MMSE filter should generate better solutions than the SUMF filter since it is a more robust filter. Nevertheless, our results for both filters are very similar. Probably it is possible to experiment with a larger ad hoc network or different initial parameters in order to observe this characteristic.

In the first instance, we developed a Matlab code to create a graphic interface to generate the data file used in AMPL. This data file contained the information corresponding to network connectivity defined by colouring and scheduling. Although it was only a graphic interface, the input of the initial parameters of the function in Matlab was too complex. Finally, instead of using Matlab matrix operators, we used sets and the different operations between them as given in AMPL. With this procedure it was easier to generate the data file for each experiment and it was very simple to modify the parameters and initial conditions. We ran all experiments by using Knitro and Ipopt and it turned out that the Ipopt package takes considerable less time than Knitro and that in some cases Knitro calculated a better local minimiser.

We solved numerically the non-convex optimisation problems for both models as well as for various ad hoc wireless network topologies and different initial conditions. For the model $\widehat{\Theta}$ presented in Chapter 4, we analysed and compared the results for different topologies by applying sensitivity analysis. Depending on those constraints which make the corresponding optimisation problem unfeasible, there is an interesting pattern between the calculated total transmission power and the size of the messages. If the constraint (4.4) makes the problem unfeasible, the relationship between the calculated total transmission power and the size of the messages resembles a quadratic function, while for the constraint (4.16), this relationship is similar to an exponential function. We have seen that our model can support higher data rate transmission with minimum power consumption.

The mathematical models presented in Chapters 4 and 5 describe the main characteristics of an ad hoc wireless network under different assumptions and it is obvious that the numerical examples can be extended to more complex network topologies.

6.2 Future Outlook

During the development of this thesis, a number of further promising ideas and research paths were discussed. Some of them are presented in this section.

Now, we propose ideas for the development of several mathematical models which include the following additional characteristics of an ad hoc wireless network:

- Mobility of the nodes, where the distances between them can vary.

  One of the main characteristic features of wireless network systems is the mobility of the devices in the system. In this thesis this property is not taken into account. We only considered wireless networks where the devices always kept the same distances and positions. The mobility of the devices has a strong impact on the transmission range and, consequently, on the routing of a message. For instance, if a device is within the transmission range of other devices and then it moves away and out of the transmission range of these devices, it would not be further considered as a possible intermediate device in the multi-hop scheme. This mobility may also cause changes in the interference level and link schedules. The mobility of the devices could be included in our model by:

  - A different colouring scheme where links are coloured depending on the distances between nodes. This would modify the topology of the wireless network at each time slot.
  - A random distance matrix for each time slot in which the distance between nodes can be characterised.

- Devices entering or leaving the network modify not only connectivity between devices i.e., routing, but also the quantity of interference. This situation can probably by analysed with similar approaches as those described above for mobility, since the effect on the wireless network system is the same.
- An analysis of the "relevant nodes", where relevant nodes is understood as those nodes which are "all the time" in the routing path of the messages.
- Analysis of the removal of those relevant nodes.
- Further analysis of the patterns found referring to the relationships between the total transmission power and the size of the message.
- Extension of the model $\widehat{\Theta}$ to Ultra Wide Band ad hoc wireless network [56].
- A TDMA/CDMA model where filtering of the signal is included. This can be done by adding the SUMF filter and MMSE filter constraints to the non-convex problem $\widehat{\Theta}$.

- Modification of the $SNIR_{e,t}$ defined in (4.15) to include a gain matrix $\mathbf{H}$ of size $|E| * |T| \times |E| * |T|$ as follows:

$$SINR_{e,t} := \frac{h_{e,t}p_{e,t}}{\sum_{f \neq e} h_{f,t}p_{f,t} + \eta^2}$$

  where each of its elements $h_{e,t}$ denotes the shadow-fading path loss co-efficient known as gain at the link $e \in E$ with values between $(0, 1]$.

Finally, we present the following additional numerical experiment with such a matrix $\mathbf{H}$.

Table 6.1 shows the set of fixed parameters and units used in order to solve the optimisation problem $\widehat{\Theta}$. The values of $h_{e,t}$ were randomly and arbitrarily chosen.

Parameter	Value	Units
P_v^{Max}	2	W
B_v^{Max}	4000	*kbit*
$P_{e,m}^{Max}$	2	W
η	0.008	mW
B	5	MHz
Gain Matrix H	$\begin{pmatrix} 1 & 0.3 & 0.5 & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & 1 & 0.7 & 0.9 & \varepsilon & \varepsilon & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 1 & \varepsilon & 0.6 & 0.5 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & 1 & 0.4 & 0.3 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & 0.4 & 0.3 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & 0.4 & 0.6 \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 & \varepsilon \\ \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & \varepsilon & 1 \end{pmatrix}$ $\varepsilon = 0.0001$	

Table 6.1: List of parameters.

For this experiments, we will consider the same digraph $G(V, E)$ as in the second set of experiments in Chapter 4. We have

$$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\},$$

$$\begin{aligned} E = & \{(v_1, v_3), (v_1, v_4), (v_2, v_3), (v_2, v_4), (v_3, v_5), (v_3, v_6), \\ & (v_4, v_5), (v_4, v_6), (v_5, v_7), (v_5, v_8), (v_6, v_7), (v_6, v_8)\} \end{aligned}$$

and the colour assignation to the nodes is given in Table 4.8. The graphical representation of $G(V, E)$ is given in Figure 4.10.

We consider two messages being sent along $G(V, E)$. The first message m_1 leaves from node v_1 and its sink node is v_7 while the second message m_2 leaves from node v_2 and its sink node is v_8. Both messages have the same size and the same reduction rate: $z_{m_1} = z_{m_2} = 1500\ Kbits$ and $R_{m_1} = R_{m_2} = 300\ Kbit$. The reception of both messages starts at $t_{j(m_1)} = t_{j(m_2)} = t_3$ and the time slot assignation is given in Table 4.9

By (4.35), it is $t^{Max} = t_8$. We solved the non-linear programing problem $\widehat{\Theta}$ for these two messages along $G(V, E)$ and obtained the following results:

The solver Knitro calculated after 0.4 seconds a local minimiser with objective function value 0.0027637108 mW after 13 iterations. We see that the objective value for this experiment is nearly the double as the objective function value found for the corresponding experiment in Chapter 4.

Figures 6.1 and 6.2 show the flow for message m_1 and m_2 respectively.

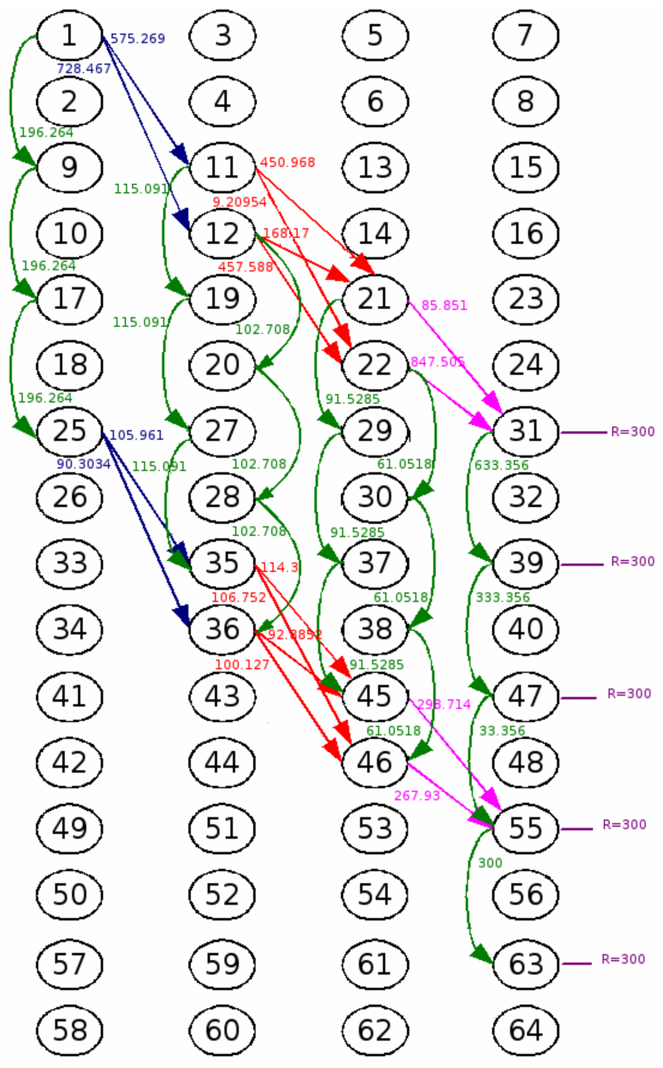

Figure 6.1: Message m_1 flowing through the network.

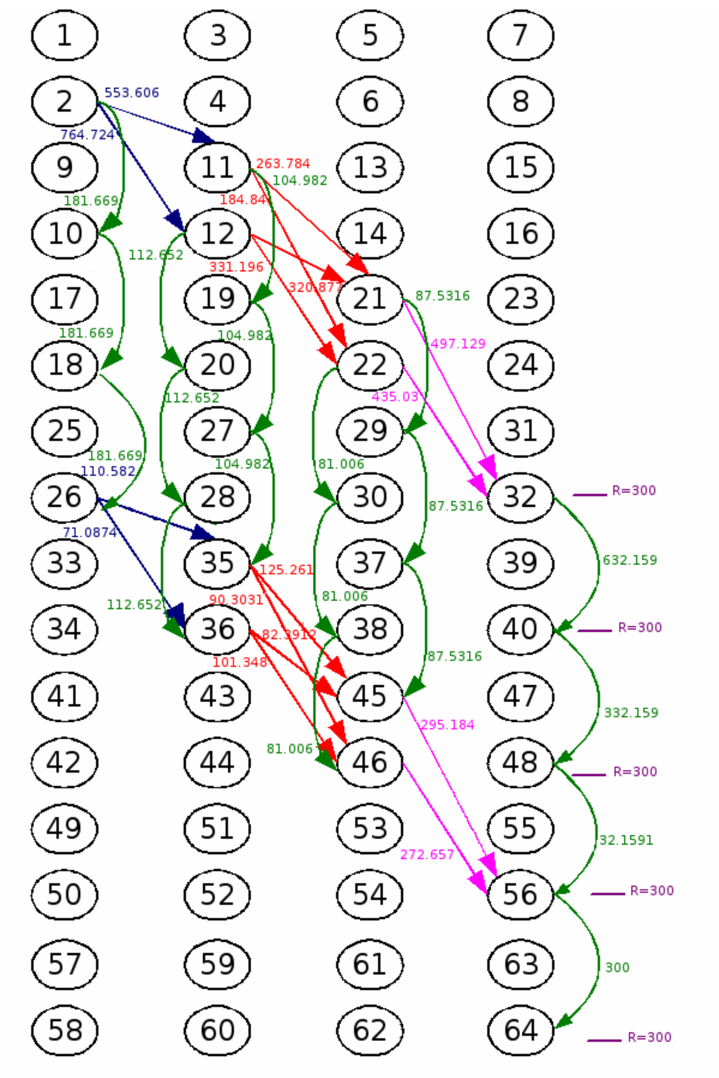

Figure 6.2: Message m_2 flowing through the network.

The values of the transmission powers p_e used to send a piece of message with size $c_{e,m}$, the size $c_{e,m}$ of the piece of the message sent through edge e and the size of the piece of message stored in the buffers are shown in Table 6.3.

	Power $p_{(v_i,v_j)}$		$c_{(v_i,v_j),m_1}$		$c_{(v_i,v_j),m_2}$
Edge (v_i, v_j)	W		Kbits		Kbits
(v_1, v_{11})	0.313107		575.27		0
(v_1, v_{12})	0.235418		728.47		0
(v_2, v_{11})	0.129327		0		553.61
(v_2, v_{12})	0.136959		0		764.72
(v_{11}, v_{21})	0.069805		450.97		263.78
(v_{11}, v_{22})	0.190298		9.21		184.84
(v_{12}, v_{21})	0.184812		168.17		320.88
(v_{12}, v_{22})	0.389391		457.59		331.2
(v_{21}, v_{31})	0.031457		85.85		0
(v_{22}, v_{31})	0.168420		847.51		0
(v_{21}, v_{32})	0.118046		0		497.13
(v_{22}, v_{32})	0.069161		0		435.03
(v_{25}, v_{35})	0.041020		105.96		0
(v_{25}, v_{36})	0.020998		90.3		0
(v_{26}, v_{35})	0.018341		0		110.58
(v_{26}, v_{36})	0.009195		0		71.09
(v_{35}, v_{45})	0.049000		114.3		125.26
(v_{35}, v_{46})	0.048509		106.75		90.3
(v_{36}, v_{45})	0.054016		92.89		82.39
(v_{36}, v_{46})	0.082636		100.13		101.35
(v_{45}, v_{55})	0.095922		298.71		0
(v_{45}, v_{56})	0.063208		0		295.18
(v_{46}, v_{55})	0.049268		267.93		272.66
(v_{46}, v_{56})	0.038983		0		0

Table 6.3: Transmission powers and sizes of pieces message m_1 and m_2.

From Table 6.1, the corresponding gain value over the edges (v_1, v_2) is 0.3 and the gain value over the edge (v_1, v_3) is 0.4 respectively. After embedding scheduling into the network topology the corresponding edges of (v_1, v_2) and (v_1, v_3) are (v_1, v_{11}) and (v_1, v_{12}) respectively. The sizes of the piece of message flowing through these edges are $c_{v_1,v_{11},m_1} = 575.269$ and $c_{v_1,v_{12},m_1} = 728.467$. That is, for those edges where the gain value is greater, the piece of the size of message sent is also greater. This is not surprising since it is consistent with the theory found in the literature [44].

As a resume: in this thesis we have presented two mathematical models which incorporate some of the main characteristics of an ad hoc wireless network. However, in this final chapter we have proposed further ideas for future research that, hopefully, lead also to further improved mathematical models.

Bibliography

[1] *Ampl benefits.* http://www.ilog.com/products/ampl/ last visited 05-11-2007.

[2] *Conopt.* http://www.conopt.com/ last visited 12-03-2009.

[3] *Flodding protocol.* http://perso.crans.org/~raffo/papers/phdthesis/thesisch1.html last visited 05-11-2007.

[4] *General algebraic modeling system gams.* http://www.gams.com/ last visited 12-03-2009.

[5] *Lindo.* http://www.lindo.com/ last visited 12-03-2009.

[6] *Minos.* http://www.ampl.com/ last visited 12-03-2009.

[7] *A modeling language for mathematical programming.* http://www.ampl.com/ last visited 05-11-2007.

[8] *Mosek.* http://www.mosek.com/ last visited 12-03-2009.

[9] *Neos guide.* http://www-fp.mcs.anl.gov/otc/Guide/index.html last visited 23-03-2009.

[10] *Neos server for optimization.* http://www-neos.mcs.anl.gov/ last visited 05-11-2007.

[11] *Node radio range.* http://www.nets.rwth-aachen.de/content/research/projects/sub/bionics/ last visited 05-11-2007.

[12] *Nonlinear programming faq.* “http://www.faqs.org/faqs/nonliear-programming-faq last visited 12-06-2009”.

[13] *Optimisation technology centre*. http://www.ece.northwestern.edu/OTC/otcmission.htm last visited 23-03-2009.

[14] *Optimisation technology centre research*. http://www.ece.northwestern.edu/OTC/otc_research.htm last visited 23-03-2009.

[15] *Optimisation technology centre software*. http://www.ece.northwestern.edu/OTC/otcsotware.htm last visited 23-03-2009.

[16] *The radio spectrum*. http://news.bbc.co.uk/nolpda/ukfs_news/hi/newsid_7048000/7048548.stm/ last visited 24-10-2008.

[17] G. Aggélou, *Wireless Mesh Networking.*, McGraw Hill International Editions, New York, 2009.

[18] A. Agrawal, J. Andrews, J. Cioffi, and T. Meng, *Iterative power control for imperfect successive interference cancellation.*, IEEE, JNL, 4 (2003), pp. 878 – 884.

[19] I. Akyildiz, X. Wang, and W. Wang, *Wireless mesh networks: A survey*, IEEE, JNL, 43 (2005), pp. S23 – S30.

[20] L. Alonso and R. Agusti, *Automatic rate adaptation and energy-saving mechanisms based on cross-layer information packet-switched data networks.*, IEEE,, 42 (2004), pp. S15–S20.

[21] A. Bahai and B. Saltzberg, *Multi-Carrier Digital Communications : Theory and Applications of OFDM.*, Kluwer Academic Publishers, Boston, 1999.

[22] M. Bazaraa, J. Jarvis, and H. Sherali, *Linear Programming and Netwok Flows*, Addison Wesley, Boston, 1990.

[23] M. Bazaraa, H. Sherali, and C. Shetty, *Nonlinear Programming: Theory and Algorithms.*, John Wiley & Sons, Inc. Canada, 2006.

[24] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, Massachusetts, 1999.

[25] R. Bhatia and M. Kodialam, *On power efficient communication over multi-hop wireless networks: Joint routing, scheduling and power control.*, IEEE, CNF, 2 (2004), pp. 1457–1466.

[26] E. Biglieri and S. Benedetto, *Principles of Digital Transmission With Wireless Applications.*, Kluwer Academic Publishers, Norwell USA, 1999.

[27] J. Broch, D. Maltz, D. Johnson, Y. Hu, and J. Jetcheva, *A performance comparison of multi-hop wireless ad hoc network routing protocols.*, Proceedings of the The IEEE Computer Society's on Mobile and Computing and Networking (Mobicom 98), (1998).

[28] S. Brueck and A. Dekorsy, *On system capacity and coverage improvements by linear miltiuser detection for utms*, IEEE, 1 (2003), pp. 2162–2166.

[29] R. Byrd, C. Gilbert, and J. Nocedal, *A trust region method based on interior point techniques for nonlinear programming*, Mathematical Programming, (2000), pp. 149–185.

[30] R. Byrd, R. Waltz, and J. Nocedal, *Knitro: An Integrated Package for Nonlinear Optimization.*, 2006.

[31] C. Comaniciu, N. Mandatam, and V. Poor, *Wireless Networks Multiuser Detection in Cross-Layer Design*, Springer, 2005.

[32] A. R. Coon, N. I. M. Gould, and P. L. Toint, *LANCELOT: a Fortran Package for Large Scale Nonlinear Optimization*, vol. 17, Springer-Verlag New York, Inc., 1992.

[33] R. Cruz and A. Santhanam, *Optimal routing, link scheduling and power control in multihop wireless networks.*, IEEE, CNF, 1 (2003), pp. 702–711.

[34] G. Dantzig, *Linear Programming and Extensions*, Princeton University Press, 1963.

[35] A. Dekorsy, *Umts course.* http://www.ant.uni-bremen.de/whomes/dekorsy/ last visited 05-11-2007.

[36] R. Diestel, *Graph Theory.*, Springer Verlang, 2000.

[37] O. Dousse, F. Baccelli, and P. Thiran, *Impact of interferences on connectivity in ad hoc networks*, IEEE, ACM, (2005).

[38] Y. Eldar and A. Chan, *An optimal whitening approach to linear multiuser detection.*, IEEE, JNL, 49 (2003), pp. 2156 – 2171.

[39] A. Ephremides and T. ElBatt, *Joint scheduling, power control, and routing algorithm for ad-hoc wireless networks.*, IEEE, JNL, 3 (2002), pp. 74–85.

[40] R. Fletcher and S. Leyffer, *User manual for filterSQP*, University of Dundee, Scotland, U.K., 1998.

[41] G. Foschini and Z. Miljanic, *A simple distributed autonomous power control algorithm and its convergence.*, IEEE, JNL, 42 (1993), pp. 641–646.

[42] R. Fourer, D. Gay, and B. Kerninghan, *AMPL A Modeling Language for Mathematical Programming.*, Thomson Brooks/Cole, California, 2003.

[43] R. Gallager, *A minimum delay routing algorithm using distributed computation.*, IEEE, JNL, 25 (1997), pp. 73 – 85.

[44] A. Goldsmith, *Wireless Communication.*, Cambridge University Press, Cambridge, 2005.

[45] A. Goldsmith and S. Wicker, *Desing callenges for ennergy-constrained ad hoc wireless network*, IEEE, 9 (2002), pp. 8–27.

[46] J. Gomez, A. Campbell, M. Naghshineh, and C. Bisdikian, *Paro: Supporting dynamic power controlled routing in wireless ad hoc networks.*, Wireless Network, Springer, 9 (2003), pp. 443–460.

[47] S. Grandhi, J. Zander, and R. Yates, *Constrained power control.*, Wireless Personal Communications Springer, Netherlands, 1 (1994), pp. 257–270.

[48] W. Grover, *Mesh-based Survivable Transport Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking.*, Prentice Hall, New Jersey, 2003.

[49] A. Gupta, I. Wormsbecker, and C. Williamson, *Experimental evaluation of tcp performance in multi-hop wireless ad hoc networks.*, IEEE, CNF, 1 (2004), pp. 3–11.

[50] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, New Jersey, 2002.

[51] S. Haykin and M. Moher, *Modern Wireless Communications.*, Prentice Hall, New Jersey, 2004.

[52] IEEE Int. Conf. Commun. ICC'93, *Increasing Spectral Efficiency Through Power Control.*, 1993.

[53] B. Jabbari, C. Jean, C. A. S., and A. Zadeh, *Combined routing, channel scheduling and power control in packet radio ad hoc networks with cellular overlay.*, IEEE, 4 (2002), pp. 1960–1964.

[54] J. H. Jeong and M. Kavehrad, *On the equivalence of turbo multiusers detector using mmse with a-priori information and soft interference cancellation followed by mmse filtering*, International Journal of Wireless Information Networks, 14 (2007).

[55] M. Johansson, L. Xiao, and S. Boyd, *Simultaneous routing and power allocation in cdma wireless data networks.*, IEEE, CNF, 1 (2003), pp. 51–55.

[56] R. Jurdak, *Wireless Ad Hoc and Sensor Networks*, Springer Verlang, 2007.

[57] N. Karmankar, *A new polynomial time algorithm for linear programming*, Combinatorica, 4 (1994), pp. 373–395.

[58] B. Kobb, *Wireless Spectrum Finder*, McGraw Hill Professional, 2001.

[59] M. Kocvara and M. Stingl, *Pennon a generalized augmented lagrangian method for semidefinite programming*, tech. rep., Institute of Applied Mathematics, University of Erlangen, Germany, 2002.

[60] D. Lay, *Linear Algebra and its Applications.*, Addison Wesley, Boston, 2005.

[61] J. Lee and L. Miller, *CDMA System Engenering Handbook*, Artech House, Publisher., 1998.

[62] D. Luenberger, *Linear and Nonlinear Programming.*, Addison Wesley, Boston, 1984.

[63] R. Lupas and S. Verdu, *Linear multiuser detector for synchrounous code-division multiple access channels*, IEEE, TIT, 35 (1989), pp. 123–136.

[64] I. J. Lustig, R. Marsten, and D. Shanno, *Computational experience with a primal- dual interior point method for linear programming*, Linear Algorithm, (1991), pp. 191–222.

[65] B. Makarevitch, *Distributed scheduling for winmax mesh network.*, IEEE, CNF, 1 (2006), pp. 1–5.

[66] G. Marconi, *Improvements in transmitting electrical impulses and signals, and in apparatus therefor.* British Patent 12039, 1896.

[67] D. Medhi and K. Ramasamy, *Network Routing.*, Elsevier, New York, 2007.

[68] R. Merris, *Graph Theory.*, Addison Wesley, Boston, 2000.

[69] S. Nash and A. Sofer, *Linear and Nonlinear Programming.*, McGraw Hill International Editions, New York, 1996.

[70] M. Pióro and D. Medhi, *Routing, Flow, and Capacity Design in Communication and Computer Networks.*, Elsevier, New York, 2004.

[71] B. Radunovic and J.-Y. Boudec, *Joint scheduling, power control and routing in symmetric, one dimensional, multi-hop wireless networks.*, Technical report, EPFL, (2002).

[72] ——, *Optimal power control, scheduling and routing uwb networks.*, IEEE,, (2004).

[73] A. Rahman and P. Gburzynski, *On constructing minimum-energy path-preserving graphs for ad-hoc wireless networks.*, IEEE, CNF, 5 (2005), pp. 3083–3087.

[74] T. Saaty and R. Busacker, *Finite Graphs and Networks*, McGraw Hill Book Company, 1965.

[75] J. Schiller, *C01-introduction.* http://cst.mi.fu-berlin.de/ last visited 05-11-2007.

[76] C. Shuguang, R. Madan, A. Goldsmith, and S. Lall, *Joint routing, mac, and link layer optimization in sensor networks with energy constraints.*, IEEE, CNF, 2 (2005), pp. 725–729.

[77] S. Singh and C. Raghavendra, *Pamas power aware multi-access protocol with signalling for ad hoc networks.*, ACM SIGCOMM, 28 (1998), pp. 5–26.

[78] T. Stern, *A class of decentralized routing algorithms using relaxation.*, IEEE, JNL, 25 (1977), pp. 1092–1102.

[79] G. Strang, *Linear Algebra and its Applications.*, Academic Press., second ed., 1980.

[80] G. Stuber, *Principles of Mobile Communication.*, Kluwer Academic Publishers, Boston, 2001.

[81] R. T., *Wireless Communications.*, Prentice Hall, New Jersey, 2002.

[82] J. Tang and X. Zhang, *Cross-layer-model based adaptive resource allocation for statistical qos guarantees in mobile wireless networks.*, IEEE, JNL, 7 (2008), pp. 2318 – 2328.

[83] A. Tenenbaum, *Computer Networks.*, Prentice Hall, New Jersey, 2002.

[84] R. Vanderbei, *Loqo user's manual.*, tech. rep., Princeton University, Princeton, 2006.

[85] S. Verdu, *Multiuser Detection.*, Cambridge University Press, Cambridge, 1998.

[86] S. Verdu and S. Shamai, *Spectral efficiency of direct-sequence spread-spectrum multiaccess with random spreading.*, IEEE, CNF, 1 (1998), pp. 64–65.

[87] H. Viswanathan and S. Mukherjee, *Throughput range tradeoff of wireless mesh backhaul networks.*, IEEE, JNL, 24 (2006).

[88] A. Wächter and L. Biegler, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming.*, Mathematical Programming, 106 (2006), pp. 25–57.

[89] B. Walke, S. Mangold, and L. Berlemann, *IEEE 802 Wireless Systems Protocols, Multi-hop Mesh/Relaying Performance and Spectrum Coexistence.*, John Wiley & Sons, Inc. Canada, 2006.

[90] X. Wang and K. Kar, *Cross-layer rate optimization in multi-hop aloha networks.*, IEEE, CNF, 5 (2005), pp. 2942–2946.

[91] D. Wetteroth, *OSI Reference Model for Telecommunications*, McGraw Hill International Editions, New York, 2001.

[92] L. Xiao, M. Johansson, and S. Boyd, *Simultaneous routing and resource allocation via dual decomposition.*, IEEE, JNL, 52 (2002), pp. 1136 – 1144.

[93] R. Yates, *A framework for uplink power control in cellular radio systems.*, IEEE, JNL, 13 (1995), pp. 1341–1348.

[94] R. Yates and C. Huang, *Integrated power control and base station assignment.*, IEEE, JNL, 44 (1995), pp. 638–644.

[95] ——, *Rate of convergence for minimum power assignment algortihms in cellular radio systems*, Springer, Netherlands, 4 (1998).

[96] R. Yates and S. Ulukus, *Adaptive power control and mmse interference suppression.*, Springer, Netherlands, 4 (1998), pp. 489–496.

[97] R. Yates, S. Ulukus, and A. Yener, *Interference management for cdma systems through power control, multiuser detection, and beamforming.*, IEEE, JNL, 49 (2001), pp. 1227–1239.

[98] F. Yu and V. Krishnamurthy, *Cross-layer radio resource allocation in packet cdma wireless mobile networks with lmmse receivers.*, Springer Berlin / Heidelberg, 3462 (2005), pp. 1243–1254.

[99] W. Yu and J. Yuan, *Joint source coding, routing and resource allocation for wireless sensor networks.*, IEEE, ICC, 2 (2005), pp. 737–741.

[100] L. Yun and A. Ephremides, *Joint scheduling, power control, and routing algorithm for ad-hoc wireless networks.*, IEEE, CNF, 9 (2005), pp. 1–10.

[101] Y. Zhang, J. Luo, and H. Hu, *Wireless Mesh Networking.*, Auerbach, New York, 2007.

Appendix A

AMPL Model File and Data file

In this appendix we present two files: the first one is the transcription of the mathematical model $\hat{\Theta}$ into AMPL notation and the second one is the data file. These files are for the set of experiments in Chapter 4.

Model file:

```
param TMax>0, integer;

param PvMax>0;

param PEMax>0;

param Num_Nodes >0, integer;

param BvMax>0;

param Sigma>0;

param BandW>0;

set OriNodes;
```

set Time:= {1..TMax} ordered;

set NewNodes := {1..(TMax+1)*Num_Nodes} ordered;

set OriGain;

set Colour;

set Message;

set Source_Nodes within OriNodes;

set Destiny_Nodes within OriNodes;

Two dimentional sets

set Buffers := setof {i in OriNodes} (i,i); # Buffers set Edges within (OriNodes cross OriNodes);

set TimeColour within (Time cross Colour);

set Nodes_Message within (OriNodes cross Message);

set DesNodMes within (Destiny_Nodes cross Message);

set FinalNodes_Message := setof { (i,j) in DesNodMes, t in 1.. TMax+1 } (i+(Num_Nodes*(t-1)),j);

param R {NewNodes cross Message} >=0, default 0;

param SM {NewNodes cross Message} >=0, default 0;

param GAIN {OriGain cross OriGain} >=0, <=1 default 0;

Four dimentional sets

set Edges_Messages_Colour := setof{(i,j,k) in Edges_Messages, (l,m,n) in Edges-Colour: i=l and j=m} (i,j,k,n);

set Edges_MessageFlow := setof { (i,j,k,l) in Edges_Messages_Colour, (m,n) in TimeColour : l=n } (i+(m-1)*Num_Nodes,j+(m*Num_Nodes),k);

set Differ := setof {(i,j,k) in Edges_MessageFlow, (l,m) in FinalNodes_Message: j=l and k!=m } (i,j,k);

set FinalEdgesMessage := Edges_MessageFlow diff Differ;

set EndNoFlow := setof { (i,j,k) in EdgesColour, (m,n) in TimeColour : k=n } (i+(m-1)*Num_Nodes,j+(m*Num_Nodes));

set EndNoFlowOne := setof { (i,j,k) in EdgesColour, (m,n) in TimeColour : k=n and k<>’P’} (i+(m-1)*Num_Nodes,j+(m*Num_Nodes));

set EndNoFlowOneTime := setof { (i,j,k) in EdgesColour, (m,n) in TimeColour : k=n and k<>’P’} (i+(m-1)*Num_Nodes,j+(m*Num_Nodes),m);

set EndNoFlowDif :=(EndNoFlow diff EndNoFlowOne); set PowerEdges := setof {(i,j,k) in FinalEdgesMessage, (l,m) in EndNoFlowOne: i=l and j=m } (i,j);

var PE{(i,v) in EndNoFlowOne}>=0, <= PEMax; var CPA{(i,j) in EndNoFlow, m in Message}>=0;

Objective Function

minimize Total_Power: sum{ (i,j) in PowerEdges} PE[i,j];

subject to Flow {k in NewNodes, m in Message}:

sum{(i,k) in EndNoFlow :(i,k,m) in FinalEdgesMessage } CPA[i,k,m] - sum{(k,j) in EndNoFlow :(k,j,m) in FinalEdgesMessage} CPA[k,j,m]=R[k,m]-SM[k,m];

subject to UpperBoundBuf {(i,j) in EndNoFlowDif }:

sum{m in Message} CPA[i,j,m]<= BvMax;

subject to UpperBoundPvMax {v in NewNodes}:

sum{ (i,j) in EndNoFlowOne : v=i } PE[i,j]<= PvMax;

subject to Coupling { (i,j,k) in EndNoFlowOneTime, t in Time: (i,j,t) in EndNoFlowOneTime }: sum{m in Message, (l,n,o) in Edges_MessageFlow :i=l and j=n and m=o } CPA[i,j,m]=BandW*(log(1+((1/1)*(PE[i,j]*GAIN[i mod Num_Nodes,j mod Num_Nodes]/ (sum{ (v,l,t1) in EndNoFlowOneTime: (i<>v || j<>l) and t=t1 } PE[v,l]*GAIN[v mod Num_Nodes,l mod Num_Nodes] + Sigma))))/log(2));

Data file for 2 Messages

param TMax=9;

param PvMax=2;

param BvMax=4000;

param PEMax=2;

param Num_Nodes=8;

param Sigma =0.0008;

param BandW=5000;

param R:=

[*,1] 31 70 39 70 47 70 55 70 63 70

[*,2] 32 70 40 70 48 70 56 70 64 70;

param SM:=

[1,1] 350

[2,2] 350;

set OriNodes := 1 2 3 4 5 6 7 8;

set Nodes_Message:= (*,1) 1 3 4 5 6 7 (*,2) 2 3 4 5 6 8;

set OriGain := 1 2 3 4 5 6 7 0;

param GAIN:=

[1,*] 3 0.3 4 0.5

[2,*] 3 0.7 4 0.9

[3,*] 5 0.6 6 0.5

[4,*] 5 0.4 6 0.3

[5,*] 7 0.4 0 0.6

[6,*] 7 0.7 0 0.9;

set Colour := B R G P;

set Message := 1 2;

set Edges := (1,3) (1,4) (2,3) (2,4) (3,5) (3,6) (4,5) (4,6) (5,7) (5,8) (6,7) (6,8);

set Destiny_Nodes:= 7 8;

set DesNodMes:= (7,1) (8,2);

set EdgesColour := (1,3,B) (1,4,B) (2,3,B) (2,4,B) (3,5,R) (3,6,R) (4,5,R) (4,6,R) (5,7,G) (5,8,G) (6,7,G) (6,8,G) (1,1,P) (2,2,P) (3,3,P) (4,4,P) (5,5,P) (6,6,P) (7,7,P) (8,8,P);

set TimeColour := (1,B) (1,P) (2,R) (2,P) (3,G) (3,P) (4,B) (4,P) (5,R) (5,P) (6,G) (6,P) (7,B) (7,P) (8,R) (8,P) (9,G) (9,P);

Appendix B

Model file

In this appendix we present three files: the first is the transcription of the mathematical model (Φ) into AMPL notation for SUMF, the second is the transcription of the mathematical model (Φ) into AMPL notation for MMSE filter and the third is the data file for 2 third numerical experiment presented in Chapter 5.

Model file for SUMF

```
param PvMax>0;

param BvMax>0;

param PEMax>0;

param Num_Nodes >0, integer;

param Sigma>0;

param BandW>0;

param Gamma>0;

set OriNodes;

set Message;
```

```
set Destiny_Nodes within OriNodes;

param E;

# Two dimentional sets

set Edges within (OriNodes cross OriNodes);

set Nodes_Message within (OriNodes cross Message);

set DesNodMes within (Destiny_Nodes cross Message);

set FinalEdgesMessage within (Edges cross Message);

param R {OriNodes cross Message} >=0, default 0;

param SM {OriNodes cross Message} >=0, default 0;

param matrixR {Edges cross Edges};

# Three dimentional sets

set Edges_Messages:= setof {(i,j) in Edges, (k,l) in Nodes_Message: i=k } (i,j,l);

var PE{(i,j) in Edges}>=0, <= PEMax;

var CPA{(i,j) in Edges, m in Message}>=0;

var QVEC{1..E}>=0;

var IEK{(i,j) in Edges};

minimize Total_Power: sum{ (i,j) in Edges} PE[i,j];

subject to Flow {k in OriNodes, m in Message}:

sum{(i,k) in Edges } CPA[i,k,m] - sum{(k,j) in Edges } CPA[k,j,m]=R[k,m]-SM[k,m];

subject to UpperBoundBuf {(i,j) in Edges }:

sum{m in Message} CPA[i,j,m]<= BvMax;
```

subject to Coupling { (i,j) in Edges }:

sum{m in Message, (l,n,o) in Edges_Messages :i=l and j=n and m=o } CPA[i,j,m]=BandW*(log(1

(sum{ (v,l) in Edges: (i<>v || j<>l) } PE[v,l] + Sigma))))/log(2));

subject to Iek {(i,j) in Edges}:

IEK[i,j] = Gamma*(sum {(k,l) in Edges: i<>k and j<>l } matrixR[i,j,k,l]*PE[k,l]

+ Sigma);

subject to Coupling1 {(i,j) in Edges}:

PE[i,j]*(IEK[i,j]-PE[i,j]) <= 0;

Model file for MMSE filter

param PvMax>0;

param BvMax>0;

param PEMax>0;

param Num_Nodes >0, integer;

param Sigma>0;

param BandW>0;

param Gamma>0;

set OriNodes;

set Message;

set Destiny_Nodes within OriNodes;

param E;

Two dimentional sets

```
set Edges within (OriNodes cross OriNodes);

set Nodes_Message within (OriNodes cross Message);

set DesNodMes within (Destiny_Nodes cross Message);

set FinalEdgesMessage within (Edges cross Message);

param R {OriNodes cross Message} >=0, default 0;

param SM {OriNodes cross Message} >=0, default 0;

param matrixR {Edges cross Edges};

# Three dimentional sets

set Edges_Messages:= setof {(i,j) in Edges, (k,l) in Nodes_Message: i=k } (i,j,l);

var PE{(i,j) in Edges}>=0, <= PEMax;

var CPA{(i,j) in Edges, m in Message}>=0;

var QVEC{1..E}>=0;

var MAT_A {Edges cross Edges};

var MAT_W {Edges cross Edges};

var MAT_WR {Edges cross Edges};

var MAT_WRW {Edges cross Edges};

var IEK{(i,j) in Edges};

minimize Total_Power: sum{ (i,j) in Edges} PE[i,j];

subject to Flow {k in OriNodes, m in Message}:

sum{(i,k) in Edges } CPA[i,k,m] - sum{(k,j) in Edges } CPA[k,j,m]=R[k,m]-SM[k,m];

subject to UpperBoundBuf {(i,j) in Edges }:
```

sum{m in Message} CPA[i,j,m]<= BvMax;

subject to Coupling { (i,j) in Edges }:

sum{m in Message, (l,n,o) in Edges_Messages :i=l and j=n and m=o } CPA[i,j,m]=BandW*(log(1
(sum{ (v,l) in Edges: (i<>v || j<>l) } PE[v,l] + Sigma))))/log(2));

subject to MatA { (i,j) in Edges, (k,l) in Edges } :

MAT_A[i,j,k,l] = if (i=k and j=l) then (matrixR[i,j,k,l]*PE[i,j]+Sigma) else (matrixR[i,j,k,l]* (PE[i,j]));

subject to MatW {(i,j) in Edges, (k,l) in Edges }:

sum {(m,n) in Edges} MAT_A[i,j,m,n] * MAT_W[m,n,k,l] = if(i=k and j=l) then PE[i,j] else 0;

subject to MatWMatR {(i,j) in Edges, (k,l) in Edges }: MAT_WR[i,j,k,l]= sum {(m,n) in Edges} MAT_W[i,j,m,n] * matrixR[m,n,k,l];

subject to MatWRMatWt {(i,j) in Edges, (k,l) in Edges}:

MAT_WRW[i,j,k,l]= sum {(m,n) in Edges} MAT_WR[i,j,m,n] * MAT_W[k,l,m,n];

subject to Iek {(i,j) in Edges}:

IEK[i,j]*(MAT_WR[i,j,i,j]^2) = Gamma*(sum {(k,l) in Edges: i<>k and j<>l } ((MAT_WR[i,j,k,l]^2)*PE[k,l]) + ((MAT_WRW[i,j,i,j])*Sigma));

subject to Coupling1 {(i,j) in Edges}:

PE[i,j]*(IEK[i,j]-PE[i,j]) <= 0;

Data file

param PvMax=2;

param PEMax=2;

param Num_Nodes=8;

param Sigma =0.08;

param BandW=5000;

param BvMax=7000;

param E =12;

param Gamma=.01;

param R:= [*,1] 7 1000 [*,2] 8 1000;

param SM:= [1,1] 1000 [2,2] 1000;

set OriNodes := 1 2 3 4 5 6 7 8;

set Nodes_Message:= (*,1) 1 3 4 5 6 7 (*,2) 2 3 4 5 6 8;

set Message := 1 2;

set Edges := (1,3) (1,4) (2,3) (2,4) (3,5) (3,6) (4,5) (4,6) (5,7) (5,8) (6,7) (6,8);

set Destiny_Nodes:= 7 8;

set DesNodMes:= (7,1) (8,2);

set FinalEdgesMessage (5,7,1) (5,8,2) (6,7,1) (6,8,2);

param: matrixR:= [1, 3,*,*] 1 3 1 1 4 0.06 2 3 6.94E-018 2 4 0.25 3 5 6.94E-018 3 6 0.06 4 5 -0.13 4 6 -0.44 5 7 -0.25 5 8 -0.25 6 7 0.19 6 8 -0.13 [1, 4,*,*] 1 3 0.06 1 4 1 2 3 -0.19 2 4 -0.06 3 5 -0.06 3 6 -0.13 4 5 0.19 4 6 0.13 5 7 0.06 5 8 -0.06 6 7 -0.13 6 8 -0.19 [2, 3,*,*] 1 3 6.94E-018 1 4 -0.19 2 3 1 2 4 -0.25 3 5 0.5 3 6 -0.06 4 5 -0.25 4 6 0.19 5 7 -6.94E-018 5 8 6.94E-018 6 7 0.06 6 8 0.13 [2, 4,*,*] 1 3 0.25 1 4 -0.06 2 3 -0.25 2 4 1 3 5 -6.94E-018 3 6 0.06 4 5 -0.13 4 6 -0.31 5 7 -0.13 5 8 -6.94E-018 6 7 0.06 6 8 -0.38 [3, 5,*,*] 1 3 6.94E-018 1 4 -0.06 2 3 0.5 2 4 -6.94E-018 3 5 1 3 6 -0.19

4 5 -0.25 4 6 -0.06 5 7 -6.94E-018 5 8 -6.94E-018 6 7 -0.06 6 8 -6.94E-018 [3, 6,*,*] 1 3 0.06 1 4 -0.13 2 3 -0.06 2 4 0.06 3 5 -0.19 3 6 1 4 5 0.06 4 6 -0.13 5 7 -0.06 5 8 0.06 6 7 -6.94E-018 6 8 -0.06 [4, 5,*,*] 1 3 -0.13 1 4 0.19 2 3 -0.25 2 4 -0.13 3 5 -0.25 3 6 0.06 4 5 1 4 6 0.19 5 7 6.94E-018 5 8 6.94E-018 6 7 0.19 6 8 0.13 [4, 6,*,*] 1 3 -0.44 1 4 0.13 2 3 0.19 2 4 -0.31 3 5 -0.06 3 6 -0.13 4 5 0.19 4 6 1 5 7 0.19 5 8 0.06 6 7 0.13 6 8 -0.06 [5, 7,*,*] 1 3 -0.25 1 4 0.06 2 3 -6.94E-018 2 4 -0.13 3 5 -6.94E-018 3 6 -0.06 4 5 6.94E-018 4 6 0.19 5 7 1 5 8 0.13 6 7 -0.06 6 8 -6.94E-018 [5, 8,*,*] 1 3 -0.25 1 4 -0.06 2 3 6.94E-018 2 4 -6.94E-018 3 5 -6.94E-018 3 6 0.06 4 5 6.94E-018 4 6 0.06 5 7 0.13 5 8 1 6 7 0.19 6 8 -6.94E-018 [6, 7,*,*] 1 3 0.19 1 4 -0.13 2 3 0.06 2 4 0.06 3 5 -0.06 3 6 -6.94E-018 4 5 0.19 4 6 0.13 5 7 -0.06 5 8 0.19 6 7 1 6 8 0.19 [6, 8,*,*] 1 3 -0.13 1 4 -0.19 2 3 0.13 2 4 -0.38 3 5 -6.94E-018 3 6 -0.06 4 5 0.13 4 6 -0.06 5 7 -6.94E-018 5 8 -6.94E-018 6 7 0.19 6 8 1;

Appendix C

Matlab file

Matlab code used to generate the Matrix **R** used in equations (5.10) and (5.11).

% PROJECT: Matrix R for the Interference Function

%DESCRIPTION: Calculation of the matrix R that we will use within AMPL % N = number of nodes (Devices) plus number of messages

% n_Pg Processing-Gain of CDMA (fix and constant parameter depending on the data rate the user transmits).

% In UMTS-Rel 99 we have n_Pg=16 for 64 kbps

% Author: Armin Dekorsy 24.01.03

function [R]= Matrix_R(N)

n_Pg=1; while n_Pg <= N, n_Pg=n_Pg*2; end if n_Pg < 16 n_Pg=16 end

C=1./sqrt(n_Pg)*(2*round(rand(n_Pg,N))-1); H=eye(N); C_eff=C*H; R=C_eff'*C_eff;