



# Radio Mesh Networks and the Round Weighting Problem

Cristiana Gomes

## ► To cite this version:

Cristiana Gomes. Radio Mesh Networks and the Round Weighting Problem. Networking and Internet Architecture [cs.NI]. Université Nice Sophia Antipolis, 2009. English. NNT: . tel-00449856

**HAL Id: tel-00449856**

**<https://theses.hal.science/tel-00449856>**

Submitted on 22 Jan 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE NICE-SOPHIA ANTIPOLIS  
ÉCOLE DOCTORALE STIC  
Sciences et Technologies de l'information et de la Communication

# THÈSE

pour obtenir le titre de

## Docteur en SCIENCES

de l'Université de Nice-Sophia Antipolis

Mention : INFORMATIQUE

présentée et soutenue par

**Cristiana Maria Nascimento GOMES**

financée par la fondation CAPES

*Ministère de l'Education du Brésil, Caixa Postal 250, Brasília – DF  
70040-020, Brésil*

## Radio Mesh Networks and the Round Weighting Problem

Thèse dirigée par **Jean-Claude BERMOND** et **Jérôme GALTIER**  
et préparée au sein du projet MASCOTTE ( I3S(CNRS/UNSA) - INRIA )

soutenue le **1 decembre 2009**

### Jury :

M.	Bermond	Jean-Claude	Directeur de recherche CNRS	Directeur
M.	Caminada	Alexandre	Professeur U. Belfort Montbéliard	Rapporteur
M.	Corrêa	Ricardo	Professeur U. Federal do Ceará	
M.	Galtier	Jérôme	Ingénieur Orange Labs	Co-Directeur
M.	Gourdin	Eric	Ingénieur Orange Labs	
M.	Klasing	Ralf	Chargé de recherche CNRS	
M.	Mahey	Philippe	Professeur U. Blaise Pascal	Rapporteur
M.	Mateus	Geraldo R.	Professeur U. Federal de Minas G.	Rapporteur
M.	Pérennes	Stéphane	Chargé de Recherche CNRS	Invité



*To my father (in memoriam).*



# Remerciement

Merci à la famille Mascotte qui m'a offert autant de bonheur que de connaissance.

Merci à celui qui m'a présenté le monde de la non-linéarité; c'est magnifique!

Merci à mon oracle, pas toujours présent mais il fallait aussi du temps pour que je réfléchisse par moi-même... aux prochaines questions.

Merci à celui qui me supportait le soir et les week-ends à m'écouter parler de mes super problèmes, pas super pour tout le monde!

Merci à celui qui m'a fait arriver là où je voulais aller. Il ne m'a pas seulement pris par la main, il a eu aussi la patience de m'apprendre à marcher avec mes propres jambes. Un jour je veux courir comme lui... dans les grands marathons.

Merci à Dieu pour tout faire si parfait (sauf mes graphes, heureusement!).



# Abstract

In this thesis, we address the joint routing and slot assignment problem between the routers and the gateways in a radio mesh access networks. We model the problem as a Round Weighting Problem (RWP) in which the objective is to minimize the overall period of slot activations providing enough capacity to satisfy the bandwidth requirements of the routers.

Solving the full problem means generating an exponential set of simultaneous transmission rounds which is intractable even for small networks. To cope with this issue, we implement a mathematical multi-objective model to solve the problem using a column generation method. We observe that the bottleneck is usually located in a limited region around a gateway.

We propose a method to obtain lower bounds (considering only a limited probable bottleneck region) and upper bounds for general graphs. Our methods are applied to grid graphs providing closed formulae for the case of uniform demands, and also optimal routing strategies considering non-uniform demands.

Motivated by the results of the existence of a limited (bottleneck) region capable of representing the whole network, we consider a variant of the RWP dealing also with bandwidth allocation, but considering SINR conditions in a CDMA network. We give sufficient conditions for the whole network to be reduced to a single-hop around the gateway. It is due to the fact that the problem is convex under some conditions that are often met. We are interested in optimal solutions in which each flow going through a bottleneck receives a fair share of the available bandwidth.





# Résumé

Dans cette thèse, nous étudions le problème joint du routage et de l'attribution des "slots" entre les routeurs et les points d'accès dans les réseaux radio maillés. Nous le modélisons comme un problème de "Round weighting" dont l'objectif est de minimiser la période d'activation des "slots" en assurant une capacité suffisante pour répondre aux demandes de bande passante des routeurs.

Résoudre le problème dans son intégralité nécessite la génération d'un ensemble exponentiel de "rounds", ce qui est hors de portée même pour des petits réseaux. Par conséquent, nous développons un modèle mathématique multicritère qui résout le problème en utilisant une méthode de génération de colonnes. Nous observons que le goulot d'étranglement est en général situé autour d'un point d'accès.

Nous proposons une méthode pour obtenir des bornes inférieures et des bornes supérieures pour les graphes généraux. Nous appliquons ces méthodes aux grilles obtenant des formules closes pour des demandes uniformes et des stratégies optimales de routage pour des demandes non-uniformes.

Motivé par les résultats sur l'existence d'une région limitée capable de représenter le réseau dans sa totalité, on considère une variante du RWP qui traite aussi de l'allocation de bande mais en considérant le SINR dans un réseau CDMA. Nous donnons des conditions suffisantes pour qu'un réseau puisse être réduit à un réseau mono-saut autour du point d'accès. Cela est dû au fait que le problème est convexe. Nous nous intéressons aux solutions optimales pour lesquelles chaque flot dans le goulot reçoit une partie juste de la bande passante disponible.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Wireless media access . . . . .	6
1.2	Outline of the thesis . . . . .	9
<b>2</b>	<b>RWP definition and a mathematical formulation</b>	<b>11</b>
2.1	Distance- $d$ model of interference . . . . .	12
2.2	A simple example . . . . .	14
2.3	RWP with fixed flow: a coloring problem . . . . .	16
2.4	RWP with fixed rounds: a flow problem . . . . .	18
2.4.1	Maximum-flow (min-cut) problem . . . . .	19
2.4.2	Minimum-cost flow problem . . . . .	19
2.5	Column generation method . . . . .	21
2.5.1	Master problem formulation . . . . .	22
2.5.2	Sub-problem formulation . . . . .	23
2.5.3	Cases solved in polynomial time . . . . .	24
<b>3</b>	<b>Load versus time: a multi-objective analysis</b>	<b>25</b>
3.1	Multi-objective formulation . . . . .	25
3.1.1	$\epsilon$ -restricted technique . . . . .	26
3.1.2	Min time versus min-max load . . . . .	27
3.2	Conclusion . . . . .	31
<b>4</b>	<b>Extracting information of bottlenecks</b>	<b>33</b>
4.1	Minimum maximal weighted clique . . . . .	35
4.2	Experimental results . . . . .	38
4.3	Conclusion . . . . .	43
<b>5</b>	<b>General lower bound methods and application to grid graphs</b>	<b>45</b>
5.1	Definitions . . . . .	45
5.2	Lower bounds: general results . . . . .	47
5.2.1	Lower bounds using one call-clique . . . . .	48
5.2.2	Lower bounds using many call-cliques . . . . .	49
5.2.3	Lower bounds using Critical Edges . . . . .	53
5.2.4	Relationship with duality . . . . .	55

5.3	Lower bound for grids . . . . .	56
5.3.1	Gateway in the middle: a lower bound . . . . .	56
5.3.2	Gateway in the corner: a lower bound . . . . .	57
5.4	Conclusion . . . . .	61
<b>6</b>	<b>General upper bound methods and application to grid graphs</b>	<b>63</b>
6.1	Upper bounds: general results . . . . .	63
6.1.1	Interference free $\gamma$ -labeled <i>paths</i> ( <i>cycles</i> ) . . . . .	63
6.1.2	Distance- $d$ model of interference and the <i>Width</i> . . . . .	64
6.2	Upper bounds for grids . . . . .	67
6.2.1	Gateway in the middle: routing the demand of a single node	69
6.2.2	Gateway in the middle: routing the demand of a combination of nodes . . . . .	77
6.2.3	Gateway in the corner: routing the demand of a single node	82
6.2.4	Gateway in the corner: routing the demand of a combination of nodes . . . . .	85
6.3	Conclusion . . . . .	87
<b>7</b>	<b>Bottleneck region and physical interference model</b>	<b>93</b>
7.1	Model definition . . . . .	95
7.2	Theoretical approach with fairness and optimality . . . . .	98
7.3	Method usage example . . . . .	100
7.4	Conclusion . . . . .	105
<b>8</b>	<b>Conclusion and perspectives</b>	<b>107</b>

# Chapter 1

## Introduction

In radio networks, the communication channels are shared among the terminals. Thus, one of the major problems faced is the reduction of capacity due to interferences caused by simultaneous transmissions [1]. Due to low power or poor propagation conditions the radio range of the nodes is limited and therefore it may be necessary in many cases to relay messages (packets) over multiple nodes before the final destination is reached. In such multi-hop packet radio networks, a message may travel long distances by means of the store-and-forward mechanism: a node transmits a packet on the radio link to another node, which in turn forwards the packet.

A Radio Mesh Network (RMNs) is composed of radio gateways, routers and clients as shown in figure 1.1. The gateways interconnect the RMN with the backbone network (e.g. Internet). The routers constitute a multi-hop radio network that serves as backhaul providing network access for the mobile clients. This thesis considers only the radio mesh network backhaul: routers and gateways.

We consider the problem of exchanging data between stationary routers and gateways in the radio network backhaul. A router can access to the backbone network by using *any* gateway. This problem was also proposed by FRANCE TELECOM (now ORANGE LABS) under the name of “how to bring Internet in the villages” where there is no high speed access everywhere. The houses of the village are equipped with radio devices and have access to a central node having high speed access to Internet (e.g. using a satellite).

We address a special case of the Round Weighting Problem (RWP) in Radio Mesh Networks. The original RWP was introduced in [2] and deals with a multicommodity flow as each source has a specific destination. In this thesis, a special case of RWP is considered as the source nodes (routers) are not associated with a specific destination (gateway). In both approaches, the RWP is composed of two sub-problems: the *routing* and the *slot assignment* problems.

We explain these two subproblems in the next paragraphs. Before that, let us describe our network modeling. The network topology is represented by a *communication graph*  $G = (V, E)$ , where the nodes  $V$  represent the sources and

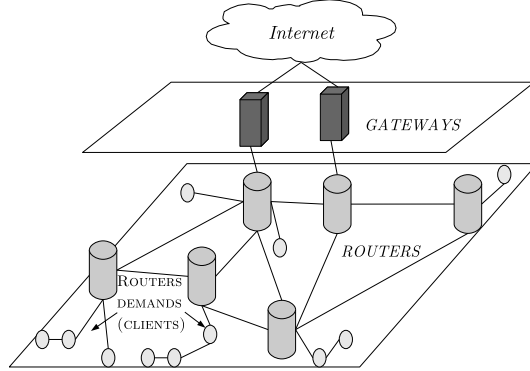


Figure 1.1: Radio Mesh Network.

destinations, and the edges  $E$  represent the links (possible calls). Two nodes can communicate directly if they are connected by an edge of the communication graph.

In general this graph is directed and a call implies a sender and a receiver. However, motivated by *reliable* protocols in which the nodes send an acknowledgment (confirmation of reception), we assume the communication graph is undirected (if a node is able to send to another node the converse is true). Doing so, we consider that an edge activation can allow communication in both directions. For instance, if the edge  $\{u, v\}$  is activated the node  $u$  can send data to the node  $v$ , and  $v$  can send an acknowledgment to  $u$  with the same activation. On the contrary, it can be that node  $v$  sends data to  $u$ , and  $u$  sends an acknowledgment to  $v$ . Figure 1.2 shows an example of a communication graph where node 1 can send data to nodes 2 and 4 but cannot receive data from any other node.

We have to take into account some incompatibility constraints among the links, as it is not possible that all links communicate at the same time. Let  $\mathcal{R} \subseteq 2^{|E|}$  be the sets of links that can communicate at the same time. For instance the links  $\{e_1, e_2\}$  can be possibly activated at the same time if and only if  $\{e_1, e_2\} \in \mathcal{R}$ . Each set in  $\mathcal{R}$  will be called a *round*. To define the set of valid rounds  $\mathcal{R}$ , we use a given set of *validation rules*. Let us explain that on a simple example, consider the communication graph in figure 1.2.

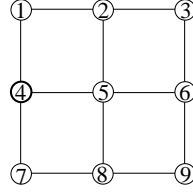


Figure 1.2: A communication graph showing the nodes that can communicate directly.

On this graph the set of possible simultaneous communications (a round) are described in figure 1.3 by a simple rule, that is a *round* is simply defined by any set of edges pairwise at distance at least 2 (number of hops). For example, the edge  $\{1, 2\}$  can be in the same round with either  $\{7, 8\}$ ,  $\{8, 9\}$  or  $\{6, 9\}$ , but these 3 edges cannot be in the same round. So the rounds containing  $\{1, 2\}$  are  $\{\{1, 2\}\}$ ,  $\{\{1, 2\}, \{7, 8\}\}$ ,  $\{\{1, 2\}, \{8, 9\}\}$  and  $\{\{1, 2\}, \{6, 9\}\}$ . Note that the edge  $\{2, 5\}$  cannot be activated with any other edge, and so do the edges  $\{4, 5\}$ ,  $\{5, 6\}$ ,  $\{5, 8\}$ . Note also that it is not possible to define a round with 3 edges in this example.

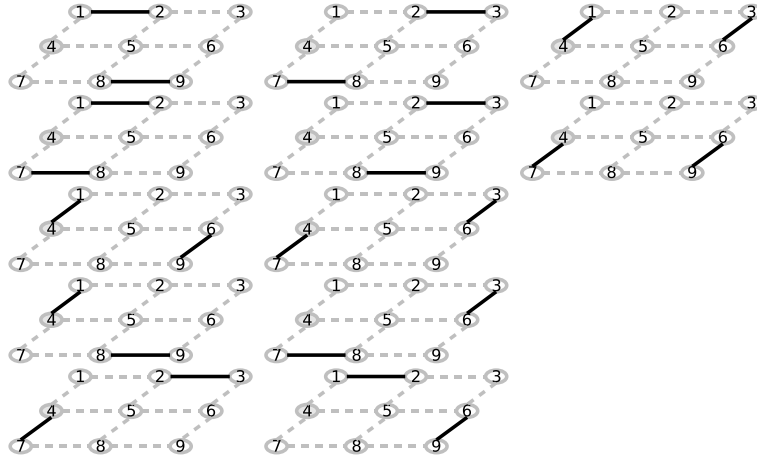


Figure 1.3: All the rounds with two edges are depicted here. There is more 12 possible rounds with one edge. The adopted validation rule is the following: activated edges have to be pairwise at distance at least 2.

Now we describe the two subproblems mentioned before, we start with the *routing subproblem*. It was modeled as a network flow problem and deals with flow relaying over multiple source nodes before the final destinations are reached. As we consider a continuous flow transmission, it defines in fact a *bandwidth reservation* with the flow representing the bandwidth requirements of the source nodes.



In this thesis, a special case of RWP is considered as the source nodes (routers) are not associated with a specific destination (gateway). It can then be turned into a single-commodity flow problem as follows. The set of gateways can be connected to a super-sink node draining all the flow by cost-free links (cost-free = without causing interferences). A super-source can also be created to feed all the routers. The router requirements of bandwidth are assigned to the capacity of each cost-free edge that leaves the super-source to the corresponding source node.

The other subproblem is the *slot assignment subproblem*. It is modeled as a labeling problem: each link in a round receives the same slot represented by a label. In this work, we consider a slot as an interval of time (see section 1.1 for other possible interpretations). For this specific case the nodes must have *synchronized* clocks. Each pair of nodes (an edge of the communication graph) will only communicate during their time-slots. In order to allow continuous flow transmission, the overall interval of slot activations is repeated periodically, each period satisfying the total demand requirements. In fact, several pair of nodes (set of links) can actually receive the same time-slot (label) under the condition they define a round. Note that each round is identified by a time-slot, which has an activation time implying a link capacity (bandwidth).

The RWP objective is to minimize the overall period of slot activations providing enough capacity to satisfy the routers requirements of bandwidth. The *input* of the RWP problem are the communication graph, the set of *validation rules* describing the possible rounds, and the network bandwidth requirements for each router. The *output* is a positive flow for each edge representing the edge bandwidth from the routing problem solution. At the same time, we have to find a set of *rounds* with their activation times (weights) satisfying the routers bandwidth.

In figure 1.4, we give a simple example of RWP solving. Consider the grid in the left side of the figure. The nodes 1, 3 and 8 require a bandwidth of 1 to the gateway node 4 and all the other nodes do not require bandwidth. The RWP solution gives the paths followed by the flow and the selected rounds to cover it. In this example, the *validation rules* define a round as a set of edges which are pairwise at distance at most 2. The right side of the figure shows the used rounds (layers): round  $R_a = \{\{1, 4\}\}$  with activation time equals to 2, round  $R_b = \{\{1, 2\}, \{7, 8\}\}$  with activation time equals to 1 and  $R_c = \{\{2, 3\}, \{4, 7\}\}$  with activation time equals to 1.

So, we have a bandwidth of 1 reserved to the node 1 using the path  $P_1 = \{\{1, 4\}\}$ , a bandwidth of 1 to the node 3 using the path  $P_2 = \{\{2, 3\}, \{1, 2\}, \{1, 4\}\}$ , and a bandwidth of 1 to the node 8 using the path  $P_3 = \{\{7, 8\}, \{7, 4\}\}$ .

Figure 1.5 shows the rounds activations, for example the slot for the round  $R_a$  starts at the time  $t_1$  and finishes at the time  $t_3$ . Note that we use a period of 4 for the activations of this sequence of rounds:  $R_a R_a R_b R_c$ . This period is repeated continuously guaranteeing the bandwidth requirements for the nodes 1, 3 and 8 by period. Note that the order of the round activations does not matter.

Considering a given integer demand we want to obtain for some applications

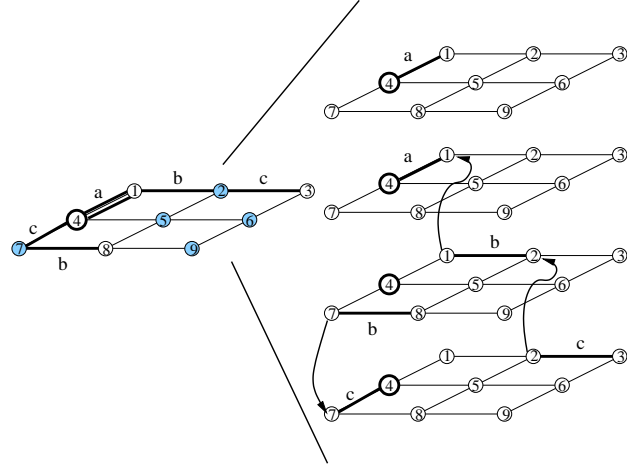


Figure 1.4: Path deduction from a set of rounds.

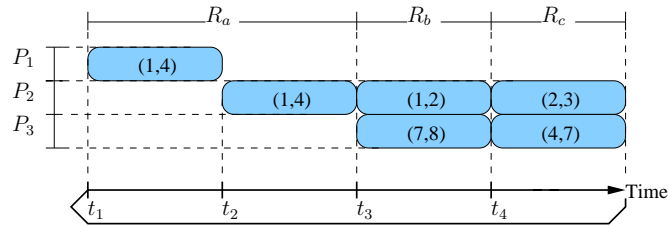


Figure 1.5: Events on the time line.

a solution where the activation times (weights) are integers. This problem will be called *Integer Round Weighting Problem (IRWP)*. Thus it deals with *integer flow* (as consequence of the integer round weights) and each flow of  $x$  (not divisible) will be considered a *packet*. Integer round activation times (weights) allow to consider a fixed time-slot size (e.g. multiple of  $x$ ) as defined in several real protocols. We consider the simplest case when a flow of  $x = 1$  represents a packet, and the time-slot (round) has a fixed size (weight) of 1 that we consider enough to send 1 packet. Note that a node can use several paths (splittable flow) to route its packets to the gateway.

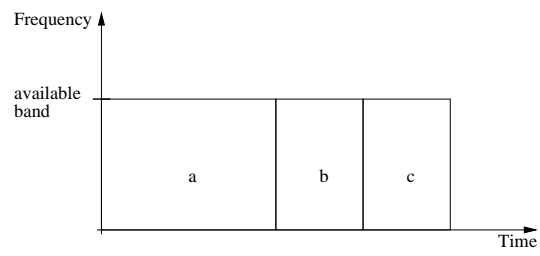
It might be useful, instead of finding simply an integer flow, to find a unique path for each node. It is known as *unsplittable flow* or *mono-routing*. It avoids dealing with the packet-reordering problem, as the packets arrive at the destination in the same order they were sent. The demands being integer, it can be considered a special case of the IRWP.

## 1.1 Wireless media access

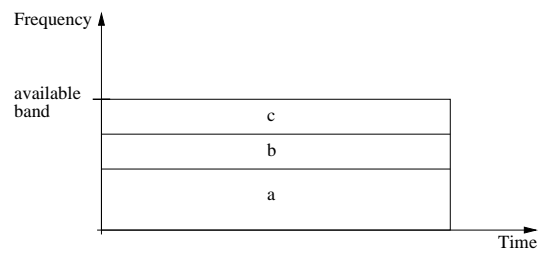
A multiple access method allows several devices to share the capacity of the same communication medium. The multiple access control mechanisms are provided by the MAC (Media Access Control) layer. The MAC-layer is a sub-layer of the Link Layer of the TCP/IP model. The access coordination can be accomplished via several ways: by multiplexing the various signals, by allowing the signals to contend for the access (without pre-coordination), or by combining these two approaches.

In the *contention-based multiple access* techniques, the medium access policy is based on competition and the main concern is to avoid collisions. Whenever a node needs to transmit a packet, it tries to get access to the medium. The advantage of this method is the simplicity in the sense that it does not deal with pre-allocations. On the other hand, it is difficult to provide QoS since access to the network cannot be guaranteed. Due to a conflict free access, the *contention-free multiple access* protocols have the advantage of using the medium efficiently considering high-load traffic. Instead, system with bursty traffic typically use contention-based multiple access techniques due to its dynamicity.

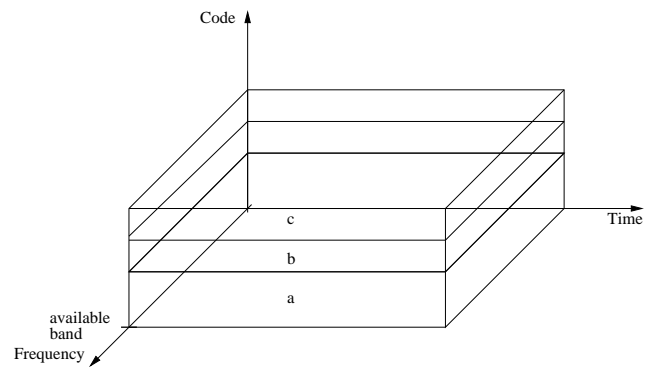
In figure 1.6 we can see the three main contention-free multiple access technologies which are used by the wireless networks: frequency division multiple access (FDMA), time division multiple access (TDMA), and code division multiple access (CDMA). The resource available is split into non-overlapping slots (frequency slot, time slot, and code slot) and each signal receives a slot. In the case of the CDMA, this notion can be applied only if orthogonal codes (non-interfering codes) are considered. Usually, the users receive a different instance of the noise carrier that are not strictly orthogonal, that is different codes may interfere with each other. All links are allowed to communicate at the same time consequently, each rate of communication is limited by the others. The power transmitted by each user is defined to maintain the SINR (Signal to Interference plus Noise Ratio) above a given threshold (we deal with this case in chapter 7).



(a) Time Division Multiple Access.



(b) Frequency Division Multiple Access.



(c) Code Division Multiple Access.

Figure 1.6: Contention-free multiple access technologies.

The slot allocation problem in wireless networks is done classically by a vertex coloring of the graph induced by the interference relations. This keeps the length of a schedule small and thus helps to improve the throughput and reduce overhead. Schedules are based on pairwise link compatibility. More realistic models taking SINR conditions have been proposed. We have considered both models.

## Interference models

There are two widely used models of interference for wireless network defining the rounds, the physical model and the protocol model (also called binary model). These models give the conditions (validation rules) for a successful transmission, that will occur if there is sufficiently small interference. The physical model treats interference as noise, thus a transmission is successful if the SINR at the receiver exceeds a threshold. This model takes into account interference due to *simultaneous* transmissions by other nodes. As SINR calculation is a non-convex function with respect to the transmission powers, the computational complexity is very high for large-sized networks.

Under the protocol model, a successful transmission occurs when a node falls inside the transmission range of its intended transmitter and falls outside the interference ranges of other non-intended transmitters. Under the protocol model, the impact of interference from a transmitting node is *binary* and is solely determined by whether or not a receiver falls within the interference range of this transmitting node. That is, if a node falls in the interference range of a non-intended transmitter, then this node is considered to be interfered and thus cannot receive correctly from its intended transmitter; otherwise, the interference is assumed to be negligible. Due to the simplification, the protocol model has been widely used in developing algorithms and protocols in wireless networks and can be easily applied to analyze large-sized wireless networks.

Note that the definition of round is flexible and permits the utilization of either binary models or physical models, it only depends of the adopted validations rules. In this thesis, in order to obtain precise results, we will use a model based on distance in the graph of communication called the distance- $d$  model. In this model a round consists of edges pairwise at distance  $\geq d$  (see section 2.1 for precise definition).

## Protocol model versus physical model

The arguments against the protocol model is that a binary decision of whether or not interference exists (based on interference range) does not accurately capture physical layer characteristics. For the case when a node falls in the interference range of a non-intended transmitter, the protocol model assumes that this node cannot receive correctly from its intended transmitter (due to interference). But this is overly conservative, based on capacity formula, as there could still be some capacity even with interference. On the other hand, for the case when a node falls outside the interference range of each non-intended

transmitter, the protocol model assumes that there is no interference. But this is somewhat optimistic as small interferences from different transmitters can aggregate and may not be negligible in capacity calculation [3].

The *reality check* method [3] derives an achievable result for a given protocol model solution. In reality check, it is only needed to re-compute capacities and adjust flow rates. They use the same capacity formula as that in physical model, this formula is only used for simple calculations instead of being part of a complex optimization problem as under the physical model. Therefore, the complexity of the reality check mechanism is very small.

According to [3], the efficacy of the protocol model depends on the performance gap between its reality check result and the result obtained under the physical model. If this performance gap is small, then the protocol model is a good approximation and can be used as an effective tool for analyzing wireless networks. Reality check result cannot exceed the optimal result under the physical model since that both solutions employ the same accurate link capacity computation and the final results (i.e. objective function values) are both feasible.

## 1.2 Outline of the thesis

We address the round weighting problem (RWP) that was introduced in [2]. We now give a brief description of the main results of this thesis.

In *Chapter 2*, we present a RWP definition as a joint flow (routing) and coloring problem (slot assignment). We discuss each sub-problem separately. For a given flow satisfying the flow constraints, the RWP becomes a coloring problem. Otherwise, for a given set of rounds, a solution for RWP can be obtained solving such a minimum cost flow problem with a dynamic cost function  $c(u, v)$  that depends of the weight assigned to the rounds covering each edge  $\{u, v\}$  and the classical simple flow constraints. Of course, the optimal solution for the RWP can only be obtained by considering the flow and the coloring problems together.

We present also a cross-layer formulation of the problem that jointly computes the routing and slot allocation. We use a column generation algorithm with a simple flow model to solve the problem.

In *Chapter 3*, we propose a multi-objective study for the *RWP*. The first one is to balance the load in the routers (*MinMaxLoad*). It increases the security in case of failure. The second objective is to minimize the communication time (*MinTime*), that is the original objective of the RWP. This multi-objective formulation uses the model of Chapter 2; both the model and the multi-objective study are published in [4].

*Chapter 4* gives a superficial introduction to methods to find lower bounds derived from a probable bottleneck region for a general graph. Our objective is to define good limited regions in  $G$  where we can look for lower bounds for RWP. Our analyses are based on the study of the flow cutting across these regions. We use the model of Chapter 2 to run experiments on networks from the literature,

with different numbers of gateways. We compare the results for IRWP, RWP and our lower bound.

*Chapter 5* presents formally methods to obtain lower bounds (inspired by *Chapter 4*) for general graphs. We present lower bounds independent of the adopted binary interference model. Then we give more precise lower bounds for the distance- $d$  model (with any value of  $d$ ). Our methods are applied to grid graphs (using distance- $d$  model) providing closed formulae (as proved in *Chapter 6*) for the case considering uniform demand. A preliminary version of this case was published in [5] in which we give a lower bound considering  $d = 2$  by using duality.

*Chapter 6* presents upper bounds (given by feasible routings) for the RWP and IRWP. We show several routing strategies for grids reaching optimal solution equal to the lower bound (*Chapter 5*). We consider different cases according to the position of the gateway and demands.

Motivated by the results of the existence of a limited (bottleneck) region capable to represent the whole network, in *Chapter 7*, we consider a variant of the RWP that also deals with bandwidth allocation, but using the interference model with SINR (Signal to Interference plus Noise Ratio) conditions. In this case, we do not attempt to allocate a separate slot to each link. Instead the links are allowed to communicate at the same time; consequently the rate of the communication is limited by the others.

The power transmitted by each user is defined to maintain the SINR above a given threshold. The model presented here is valid for UMTS and other systems that tolerate interferences. In this context, we give sufficient conditions for the multi-hop problem to be reduced to a single-hop problem by only changing the utility functions. These conditions are represented by our description of utility functions. This work was published in [6].

## Chapter 2

# RWP definition and a mathematical formulation

Recall that the RWP was introduced in [2], where they show that the problem is NP-hard for gathering. Furthermore, they give a 4-approximation algorithm for general topologies and show that RWP is polynomial for paths. They also asked the question of finding simple efficient algorithms and the complexity of the problem for grids.

The RWP can be seen as a relaxation of the *Round Scheduling Problem (RSP)* (or Minimum Time Gathering problem) where each node has some data to send to the gateway and we want to minimize the completion time of the gathering. The RSP is also important for sensor networks (see example in [7]) where we want to collect data (alerts) in a Base Station. This problem admits a 4-approximation algorithm [8, 9]; polynomial or 1-approximation algorithms are also given for specific topologies like paths, trees (see the survey in [10] for more details). It has been solved for grids in case of unitary traffic (each node has one unit of traffic to send to the gateway), for the asymmetrical model for  $d$  odd [11] and for even and also hexagonal grids [12].

The RWP and the RSP have similar behavior when the network links are completely filled from the source to the destination (steady state). They differ by the additional time to “fill” and “drain” the network (transient states) that is only taken into account by *RSP*.

In this chapter we introduce some notation and the RWP solving method that is based on column generation [4].

Recall that the network is represented by a *communication graph*  $G = (V, E)$ . The bandwidth should be allocated between the routers in  $V_r$  and the gateways in  $V_g$ , with  $V = V_r \cup V_g$  and  $V_r \cap V_g = \emptyset$ .

Each router has a demand of bandwidth  $b(v)$  that will be treated as a flow, then the flow in an edge represents the bandwidth associated with this edge. The flow can be split on several paths reaching one or more gateways. Let  $B$  be the sum of all the router demands,  $B = \sum_v b(v)$ . The set of edges  $E \subseteq V \times V$



corresponds to the communication links of the network. If a router  $u$  is located within the transmission range of a router  $v$ , considering distance, obstacles, directional antennas, and so on, then there is an edge  $\{u, v\} \in E$ .

The set of validation rules for the rounds is here modeled as follows. The set  $I_{u,v}$  is composed by all the links interfering with the link  $\{u, v\}$ . Each edge  $\{u, v\}$  in  $G$  corresponds to a node  $v_{u,v}$  in a conflict graph  $C(G) = (V_c, E_c)$ . An edge  $\{v_{u,v}, v_{x,y}\} \in E_c, \forall \{u, v\}, \{x, y\} \in E$  means that  $\{x, y\} \in I_{u,v}$ .

Notice that this set permits the use of several interference models. We can consider a symmetric model or an asymmetric model. The distance of interference may be based on the number of hops, on a propagation model, or on the geometric distance between the nodes, etc. As our analysis is based on flow contention (bottleneck) similar results to these ones presented in this work should be obtained independently of the adopted model. To put it in a nutshell, any binary interference model can be used as it can be modeled by a conflict graph. Considering the conflict graph as an input of the problem, we are completely independent of the way it is generated.

Note that the definition of round is flexible and permits the utilization of either binary models or physical models, it depends only the validations rules adopted. It is easy to adapt our model to generate rounds taking into account the SINR constraints (see [13]).

For the sake of simplicity, our tests and examples consider a symmetric interference model defined in the next subsection.

## 2.1 Distance- $d$ model of interference

To give precise results, we will specify the binary model of interference by using distances in graphs. The model can be viewed as a symmetric variant of the interference model used for example in [2, 14] where a node causes interference in all the nodes at distance at most  $d_I$  from it (nodes in its zone of interference); in their model, two directed calls  $(s, r)$  and  $(s', r')$  interfere if the distance between the nodes  $s$  and  $r'$  is  $d(s, r') \leq d_I$  or  $d(s', r) \leq d_I$  (asymmetrical interference model).

Here we suppose that both calls  $(u, v)$  and  $(v, u)$  can be done. In particular when  $u$  sends something to  $v$ ,  $v$  sends an acknowledgement (confirmation of reception) to  $u$ ; that is used in *reliable* protocols. In our model, two symmetrical calls interfere if one node of a call is in the interference zone of a node of the other call.

Let us define the distance between two edges (calls)  $e = \{u, v\}$ ,  $e' = \{u', v'\}$  as the minimum distance  $d(e, e') = \min_{x \in \{u, v\}, y \in \{u', v'\}} d(x, y)$  between their end vertices. So, two calls (or edges) interfere if  $d(e, e') \leq d_I$ . For several reasons, we will use the parameters  $d = d_I + 1$  ( $d$  can be seen as the minimum distance of reuse of the same frequency for two calls during a time-slot).

**Definition 1** *In the distance- $d$  model, two calls (edges)  $e$  and  $e'$  interfere if their distance  $d(e, e') < d$ .*

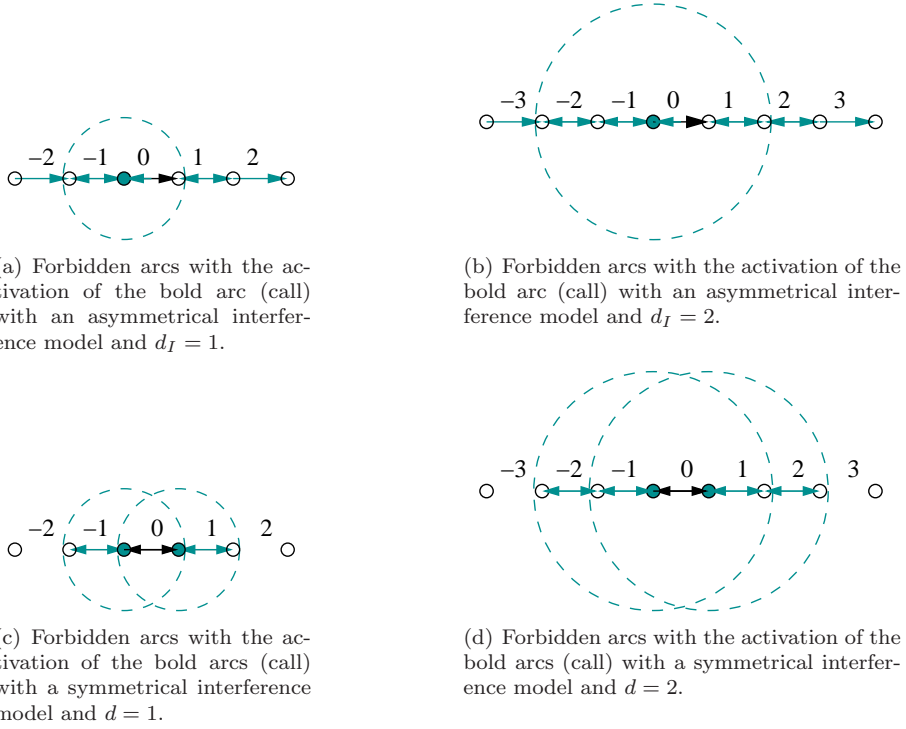


Figure 2.1: Symmetrical versus asymmetrical models.

Consequently, a *round consists of edges which are pairwise at distance at most  $d$* . The particular case  $d = 1$  is nothing else than the primary node interference model or node-exclusive interference model [15]. In that case, a round is a matching. In the case  $d = 2$  we get the so called distance-2 interference model [16, 17, 18, 19]. In this case, a round is an induced matching.

One of the reasons to use  $d$  (and not  $d_I$ ) is to be coherent with these two particular models. Furthermore let the conflict graph be the graph whose vertices represent the edges (calls) of  $G$ , two vertices being joined if the corresponding calls interfere.

Then, in the case  $d = 1$ , the conflict graph is nothing else than the line graph  $L(G)$  of  $G$ . (The vertices of  $L(G)$  represent the edges of  $G$  and two vertices are joined in  $L(G)$  if their corresponding edges intersect). More generally, for any  $d$ , the conflict graph is the  $d$ -th power of  $L(G)$  (The  $k$ -th power of a graph being the graph with two vertices joined if their distance is less than or equal to  $k$ ).

The differences between the symmetrical and the asymmetrical models are shown in Figure 2.1 on a path, where are indicated the arcs can not be activated if we want the call (the bold arc(s)) to be successful. For a given  $d$  the set of arcs in the symmetrical model is smaller than that for the asymmetrical model with  $d_I = d$  but bigger than  $d_I = d - 1$ . Consequently, any RWP solution using

the distance- $d$  symmetrical interference model represents a lower bound for the RWP using the asymmetrical model with  $d_I = d$  for general graphs. It is as if the RWP with symmetrical model “relaxes” some interfering arcs.

Finally, notice that the solution for grid graphs we obtain in the distance- $d$  model with symmetrical model is very close to the solution of the asymmetrical model with  $d_I = d$ . As our problem deal with a gathering, the solution is a flow toward a unique node so the additional interference of the asymmetrical model does not make an important difference. The formulae that we obtain are then similar to that of [11, 12] that uses the asymmetrical model.

When a round  $R_i$  is active, a set of edges with activation time of  $w(R_i)$  is available. The RWP minimizes the overall period  $W = \sum_i w(R_i)$  to allow a routing from  $V_r$  to  $V_g$  of a flow of  $B$ . Consider  $W_{min}$  the minimum period, this time interval is repeated periodically providing a throughput of  $\frac{b(v)}{W_{min}}, \forall v \in V_r$ . Recall that the flows on the edges of the problem solution represent the allocated bandwidth.

## 2.2 A simple example

In figure 2.2, we recall the example of RWP solving in the introduction chapter (now, using the formal notation). We consider a distance of interference  $d = 2$ . The nodes 1, 3 and 8 require a bandwidth connection of  $b(v) = 1$  to the gateway node 4. As  $b(v) = 0$  for the other nodes of  $V_r$ , we obtain  $B = 3$ . The example shows two different routings. The left routing needs a total time of  $W = 5$ . Note that the right routing is better because it also admits a routing of  $B$  using only  $W = 4$ . This is the optimal solution to RWP. It means that at each period of  $W_{min}$  the nodes 1, 3 and 8 can send a flow of 1, thus we obtain a throughput for each node of  $\frac{b(v)}{W} = \frac{1}{4}$  (flow/time).

Figure 2.3 shows the paths followed by the nodes and the selected rounds to cover them, round  $R_a = \{\{1, 4\}\}$  with capacity  $w(R_a) = 2$ , round  $R_b = \{\{2, 1\}, \{8, 7\}\}$  with capacity  $w(R_b) = 1$  and  $R_c = \{\{3, 2\}, \{7, 4\}\}$  with capacity  $w(R_c) = 1$ .

It is interesting from a practical point of view to know the paths used by the source nodes to reach the gateways. It allows to configure routing tables, to permit the use of the same path for response from the gateway, to provide QoS, and so on. These paths can be deduced using a MSSD (Multiple Source-Single Destination) flow model getting as input the capacities computed by the SSSD (Single Source-Single Destination) model.

We can extract the paths followed by the nodes because the flow variable (in MSSD) contains the information about the source node using each link. The SSSD models does not have such information. For example, in a SSSD model we can not know if the packet transmitted by the node 1 in the slot  $a_1$  (of the round  $R_a$ ) is from itself or from the node 3.

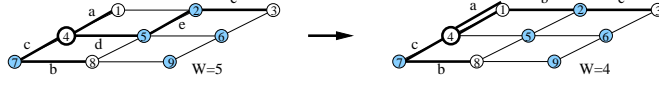


Figure 2.2: Deciding the best routing.

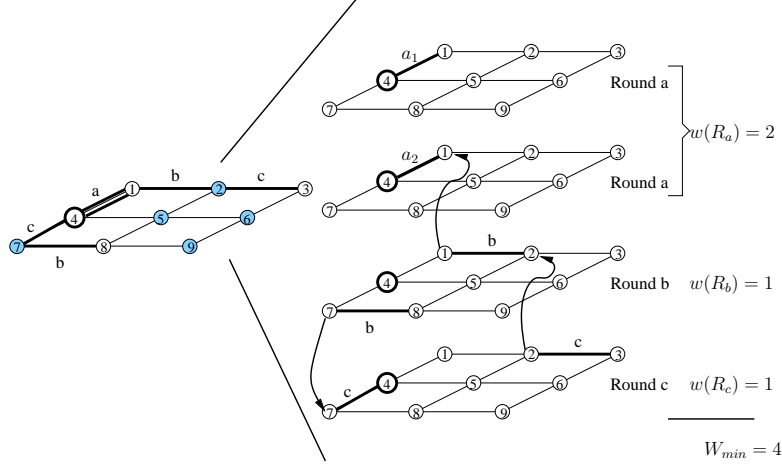


Figure 2.3: Path deduction from a set of rounds.

The problem can be defined as follows:

PROBLEM:	Round weighting problem (RWP)
INPUT:	The communication graph $G(V_r \cup V_g, E)$ , the conflict graph $C(G)$ representing the edge interferences, and the network bandwidth $b(v)$ for each router $v \in V_r$ .
SOLUTION:	Each edge $\{i, j\} \in E$ receives a positive flow $\phi(i, j)$ that represents the edge bandwidth from the routing problem solution. At the same time, we have to find a set of rounds $R$ with their weights $w(R_i)$ achieving the routers bandwidth ( $\sum_{R_i \ni (i, j)} w(R_i) \geq \phi(i, j)$ ).
OBJECTIVE:	Minimize the overall period $W$ of round activations providing edge capacities $c(u, v)$ enough to achieve the routers requirements of bandwidth. We call $W_{min}$ the minimum period.

Considering a given integer demand we want to obtain for some applications a solution where the activation times (weights) are integers. This problem will be called *Integer Round Weighting Problem (IRWP)*.

The RWP is a joint flow and coloring problem. To contextualize the problem in the next subsections, we describe each sub-problem separately. Suppose that, by some means, we could know the solution of each sub-problem that gives the

optimal solution for the RWP. We call simply optimal sub-problem solution. Thus, if we consider a given optimal sub-problem flow solution, the RWP is reduced to a coloring problem. Otherwise, if we consider a given optimal solution for the coloring sub-problem, the RWP is reduced to a flow problem. Of course, the optimal solution for the sub-problems can only be obtained from the RWP solution, that is considering the flow and the coloring problems together.

## 2.3 RWP with fixed flow: a coloring problem

If we consider a given optimal sub-problem flow solution, the RWP becomes a weighted coloring problem. Let the graph  $G'$  represent  $G$  with edge weights given by the optimal flow solution. Let the graph  $C(G)'$  represent  $C(G)$  with node weights derived from  $G'$ . Thus a node  $v$  in  $C(G)'$  has a weight determined by the quantity of flow crossing the corresponding link in  $G'$ .

We need the following definitions to understand the RWP as a weighted coloring problem. A  $k$ -coloring of a graph  $H = (V, E)$  is a partition  $I = (I_1, \dots, I_k)$  of the vertex set  $V$  of  $H$  into independent sets  $I_i$ . The classical coloring problem looks for the vertex coloring minimizing  $k$ , that is the chromatic number  $\chi(H)$ . A generalization of this problem is obtained by the *weighted coloring*, that considers a strictly positive weight  $w(v)$  for any vertex  $v \in V$ , and defines the weight of the independent set  $I$  of  $H$  as  $w(I) = \max w(v) : v \in I$ . The objective is to define a vertex coloring  $I = (I_1, \dots, I_n)$  of  $H$  minimizing  $\sum_{i=1}^n w(I_i)$ .

For any graph  $H$ , let  $\Delta(H)$  be the maximum degree,  $\omega(H)$  be the clique number (i.e. the size of the largest clique), and  $\chi_f(H)$  be the fractional chromatic number of  $H$  (defined later). The most known bounds are  $\omega(H) \leq \chi_f(H) \leq \chi(H) \leq \Delta(H) + 1$  for any graph  $H$ .

A known upper-bound defined as a function of the graph weights is  $\chi(H, w) \leq 1 + |\mathcal{W}|(\chi(H) - 1)$  [20], where  $|\mathcal{W}|$  represents the number of different weights  $w$  used in the nodes of  $H$ ,  $\mathcal{W} = w(v) : v \in V$ .

Considering each class of color a round, the RWP can be seen as a *weighted coloring* and  $W_{min} = \chi(C(G)')$ . For example, we describe how RWP solves the slot allocation problem with a coloring problem using a symmetric interference model. Given a network graph  $G$ , its line graph  $L(G)$  is a graph such that each vertex of  $L(G)$  represents an edge of  $G$ ; and two vertices of  $L(G)$  are adjacent if and only if their corresponding edges share a common endpoint in  $G$ . The conflict graph  $C(G)$  is the line graph power  $d$ ,  $(L(G))^d$ . The  $d$ th power of the graph  $L(G)$  is the graph with the same set of vertices as  $L(G)$  and an edge between two vertices if and only if there is a path of length at most  $d$  between them.

In a symmetric model, the RWP is then a *distance- $d$*  coloring in which any two edges within distance  $d$  of each other must get different colors. The most basic case is  $d = 1$  where the set of rounds  $\mathcal{R}$  is simply the set of the matchings of  $G$ , that is a set of edges without common vertices. If  $d = 2$ , a *round* represents an *induced matching* in  $G$ . An induced matching is a matching in which no two edges are linked by an edge of  $G$ .

Considering that the rounds can be active for a fractional time interval (fractional rounds weight), RWP can be seen as a *weighted fractional coloring* of  $C(G)'$ . Grotschel and al. [21] proved that the weighted fractional coloring problem is NP-hard for general graphs. They showed that if the problem of computing the weight of the largest weighted independent set is NP-hard, so the weighted fractional coloring problem is also NP-hard. See [22] for further information about *fractional coloring*.

To be generic and independent of an interference model, a *round* is a set of edges in  $G$  that correspond to an *independent set* of  $C(G)$ . At first, we describe the RWP as a *weighted fractional coloring* of the  $C(G)'$ .

We present a linear program for the *weighted fractional coloring problem*. Let  $\mathcal{I}$  denote the family of all independent sets of a graph  $H$ . The graph  $H$  has a weight  $f(v)$  for each vertex  $v \in V(H)$ . The *weighted fractional coloring problem* consists of assigning a non-negative real capacity  $w(I)$  to each independent set in  $\mathcal{I}$ . The value  $w(I)$  has to be greater or equal to the weight  $f(v)$  for each vertex  $v \in I$ . The objective is to minimize  $\sum_{I \in \mathcal{I}} w(I)$ . The *minimum weighted fractional coloring* can be formulated as follows:

$$\chi_f = \min \sum_{I \in \mathcal{I}} w(I) \quad (2.1)$$

$$\sum_{I: v \in I} w(I) \geq f(v), \forall v \in V(H) \quad (2.2)$$

$$w(I) \geq 0, \forall I \in \mathcal{I} \quad (2.3)$$

The dual problem is the *maximum weighted fractional clique* that asks for a vector  $x \in \mathcal{R}^{V(H)+}$  subject to the constraints 2.5, such that  $\sum_{v \in V(H)} x(v)f(v)$  is as large as possible, given the weight function  $f \in \mathcal{R}^{V(H)+}$ . The dual problem can be expressed as:

$$\omega_f = \max \sum_{v \in V(H)} x(v)f(v) \quad (2.4)$$

$$\sum_{v \in I} x(v) \leq 1, \forall I \in \mathcal{I} \quad (2.5)$$

$$x(v) \geq 0, \forall v \in V(H) \quad (2.6)$$

A *fractional clique* can be seen as a non-negative real-valued function on  $V$  such that the sum of the values of the function on the vertices of any independent set is at most one. Notice that a fractional clique is a relaxation of the integer clique. It is simple to understand, consider a function that assigns 1 to the vertices of a clique in a graph  $H$  and zero otherwise. It respects the condition that each independent set weight in  $G$  is 1 (each independent set intersects the clique in at most one vertex). Thus this function is a fractional clique, whose

weight is the number of vertices in the clique. As a clique with  $n$  vertices can be seen as a fractional clique of weight  $n$ , for any graph  $H$ ,  $\omega(H) \leq \omega_f(H)$ .

By the *strong duality theorem* of Linear Programming [22], the optimum solution for the *weighted fractional coloring problem* equals to the optimum solution for the *weighted fractional clique problem*, that is  $\chi_f = \omega_f$ .

Now, we show the same model in the context of RWP to be more clear. So  $\mathcal{I}$  (independent nodes) is replaced by the set of rounds  $\mathcal{R}$  (independent set in  $C(G)$ ) in RWP.

The *weighted fractional coloring problem* consists of assigning a non-negative real capacity  $w(R)$  to each round in  $\mathcal{R}$ . The capacity  $w(R)$  has to be enough to support the flow  $\phi(i, j)$  for each edge  $(i, j) \in E$ . The objective is to minimize  $\sum_{R \in \mathcal{R}} w(R)$ . The primal problem for RWP given an optimal routing  $(\phi(i, j), \forall (i, j) \in E)$  can be expressed as:

$$\min \sum_{R \in \mathcal{R}} w(R) \quad (2.7)$$

$$\sum_{R: (i, j) \in R} w(R) \geq \phi(i, j), \forall (i, j) \in E(G) \quad (2.8)$$

$$w(R) \geq 0, \forall R \in \mathcal{R} \quad (2.9)$$

The dual version of the problem is:

$$\max \sum_{(i, j) \in E(G)} d(i, j) \phi(i, j) \quad (2.10)$$

$$\sum_{(i, j) \in R} d(i, j) \leq 1, \forall R \in \mathcal{R} \quad (2.11)$$

$$d(i, j) \geq 0, \forall (i, j) \in E(G) \quad (2.12)$$

As we consider fractional round weights  $w(R)$ , the RWP becomes a weighted fractional coloring problem as we discuss above. If we consider integer round weights  $w(R)$ , RWP becomes a (integer) weighted coloring problem. It means that when the nodes are activated they must remain active continuously for an integer duration of time. The fractional coloring may provide a shorter  $W$  due to the utilization of the fractional activation times, thus there is an integrality gap. We compare  $W_{min}^i$  ( $W_{min}$  with integer  $w(R)$ ) with  $W_{min}^f$  ( $W_{min}$  with fractional  $w(R)$ ) in chapter 2.

## 2.4 RWP with fixed rounds: a flow problem

The RWP becomes a flow problem if the optimal solution for the coloring sub-problem (rounds) is known. In this subsection we discuss about flow problems in the literature that can be related to our problem. At first, we recall the basic notions of flow according to [23].

**Definition 2 (Flow)** Suppose  $G(V, E)$  is a finite directed graph in which every edge  $(u, v) \in E$  has a non-negative, real-valued capacity  $c(u, v)$ . If  $(u, v) \notin E$ , we assume that  $c(u, v) = 0$ . We distinguish two vertices: a source  $s$  and a sink  $t$ . A flow network is a real function  $\phi : V \times V \rightarrow \mathbb{R}$  with the following three properties for all nodes  $u$  and  $v$ :

- *Capacity constraints:*  $\phi(u, v) \leq c(u, v)$ . The flow from one vertex to another must not exceed the capacity.
- *Skew symmetry:*  $\phi(u, v) = -\phi(v, u)$ . It is a notational convenience that says that the flow from a vertex  $u$  to a vertex  $v$  is the negative of the flow in the reverse direction.
- *Flow conservation:*  $\sum_{w \in V} \phi(u, w) = 0$ , unless  $u = s$  or  $u = t$ . It says that the total flow out of a vertex other than a source or sink is 0.

We recall the classical definitions in the context of the related problems.

#### 2.4.1 Maximum-flow (min-cut) problem

Given a directed graph  $G(V, E)$ , where each edge  $u, v$  has a capacity  $c(u, v)$ , the maximum flow problem is to find a feasible flow (see Definition 2) from some given single-source to a given single-sink that is maximum. The maximum source-to-sink flow in a network is equal to the minimum source-to-sink cut in the network, as stated in the *Max-flow min-cut theorem*.

The RWP considers the capacity  $c(u, v)$  as a variable representing the edge activation time. While the maximum flow problem considers a given fixed capacity  $c(u, v)$ . Let  $\mathcal{R}_{u,v}$  be the subset of rounds in  $\mathcal{R}$  containing the edge  $(u, v)$ . In RWP, the capacity of an edge is defined as  $c(u, v) = \sum_{R \in \mathcal{R}_{u,v}} w(R)$  and it is optimal if  $W = \sum_{R \in \mathcal{R}} w(R)$  is minimum. Thus, our objective is not to find the maximal flow. Our objective is to find an optimal network capacity  $c(u, v)$  in a way to admit a *maximum flow* of  $B$ , so  $B$  is fixed and it is an input of our problem.

Now, considering a given optimal capacity  $c(u, v)$  (rounds and also their weights), the RWP becomes a flow problem (without objective). We need only to check that the flow  $B$  can go through the network links respecting the given  $c(u, v)$ . Of course, if we can put more flow than  $B$  means that the given capacity is not optimal implying a  $W > W_{min}$ . In other words, if we solve the maximum flow problem in a network with a given optimal capacity we have to find a maximum flow equals to  $B$ .

#### 2.4.2 Minimum-cost flow problem

Given a flow network  $G(V, E)$  with source  $s \in V$  and sink  $t \in V$ , where edge  $(u, v) \in E$  has capacity  $c(u, v)$ , flow  $\phi(u, v)$  and cost  $q(u, v)$ . The cost of sending this flow is  $\phi(u, v)q(u, v)$ . The objective is to minimize the total cost



of the flow  $\sum_{u,v \in V} q(u,v)\phi(u,v)$ , subject to a required flow  $\sum_{w \in V} \phi(s,w) = B$  and the flow constraints in Definition 2.

Both problems are flow problems and they have to respect the flow constraints in Definition 2. They have also a given fixed amount of flow  $B$  as an input of the problem.

Both objective functions want to minimize the cost of the flow. But our cost function is more complicated. The minimum cost flow problem considers a fixed cost  $q(u,v)$  as input of the problem. In our problem, we consider such a *dynamic cost* that is dependent of the flow.

Given optimal rounds (without their weights), our objective is to find a network capacity  $c(u,v)$  of *minimum cost*  $W_{min} = \min \sum_i w(R_i)$  that admits a flow of  $B$ . So, our cost depends of the capacity definition  $c(u,v) = \sum_{R \in \mathcal{R}_{u,v}} w(R)$  that consequently depends of the flow due to the capacity constraints  $c(u,v) \geq \phi(u,v)$  in Definition 2.

Let  $f(u,v,R)$  be the flow in the edge  $(u,v)$  that is transmitted during the time interval  $w(R)$ . If we want to reuse the same objective function that is  $\min \sum_{u,v \in V} q(u,v)\phi(u,v)$ , the edge cost can be defined as the following  $q(u,v) = \sum_{R \in \mathcal{R}_{u,v}} f(u,v,R)M(u,v,R)/\phi(u,v)$ . The binary variable  $M(u,v,R)$  is 1 if  $f(u,v,R) = \max_{(i,j) \in E} f(i,j,R)$ , and 0 otherwise; with  $\sum_{(u,v) \in R} M(u,v,R) = 1, \forall R \in \mathcal{R}$ .

According to [23], many of the asymptotically fastest maximum-flow algorithms are push-relabel algorithms, and the fastest actual implementations of maximum-flow algorithms are based on the push-relabel method. Other flow problems, such as the minimum-cost flow problem, can be solved efficiently by push-relabel methods. The Goldberg's "generic" maximum-flow algorithm has a simple implementation that runs in  $O(V^2E)$  time, thereby improving upon the  $O(VE^2)$  bound of the Edmonds-Karp algorithm. This generic algorithm can be refined to obtain another push-relabel algorithm that runs in  $O(V^3)$  time.

Notice that, if all the round weights are integer we have integer edge capacities. Then, by the *Integrality theorem*, there is an optimum flow  $\phi(e)$  whose values are all integers ( $\phi : E \rightarrow \mathbb{Z}$ ). It was proved by [24] and follows from the observation that the constraints matrix respects the property of *total unimodularity*, i.e., every subdeterminant of this matrix is 0, 1, or -1.

An analysis of the combinatorial meaning of the *Integrality theorem* is given by [25]: Let  $\phi$  be any  $(s,t)$ -flow with value  $v(\phi) > 0$ . It is easy to find a directed  $(s,t)$ -path  $P$  such that  $\phi(e) > 0$  for all  $e \in P$ . Let  $X^P$  denote the incidence vector of  $P$ , then  $X^P$  is a special kind of  $(s,t)$ -flow with value 1. Set  $\lambda_P = \min\{\phi(e) : e \in P\}$ , then  $\phi' = \phi - \lambda_P X^P$  is a  $(s,t)$ -flow with value  $v(\phi') = v(\phi) - \lambda_P$ . Going on similarly, we can decompose  $f$  into directed  $(s,t)$ -paths and possibly into a flow with value 0:

$$\phi = \lambda_{P_1} X^{P_1} + \dots + \lambda_{P_m} X^{P_m} + \phi_0,$$

where  $v(\phi_0) = 0$ . So  $v(\phi) = \lambda_{P_1} + \dots + \lambda_{P_m}$ . Now if  $\phi(e)$  is integral then so are the  $\lambda_P$ , and we may view this decomposition as a collection of  $(s,t)$ -paths, in which  $P_i$  occurs with multiplicity  $\lambda_{P_i}$ . The condition that  $\phi(e) \leq c(e)$  can

then be translated into the condition that this family of paths uses each edge  $e$  at most  $c(e)$  times. The objective is then just the number of paths in the collection.

## 2.5 Column generation method

Since the number of rounds is exponential, a column generation (CG) algorithm is used to avoid dealing with the complete set of rounds. Our CG algorithm has an exponential complexity (in the number of links) as well. Nevertheless, in most of the practical cases CG algorithm is efficient. It has only worst-case exponential complexity, whereas the brute force enumerating algorithms have average-case exponential complexity.

An algorithm enumerating a tractably large subset of simultaneous transmission rounds has been developed in order to compute an approximated solution for maximum throughput using linear programming (LP) in [26]. Solving the full LP problem means generating an exponential set of scenarios which is intractable even for small networks as seen in [27]. To cope with this issue, column generation methods have been considered, e.g. [28], [4], [13], [29] and [30]. The work in [28] and [4] present mathematical models for RWP, based on cut constraints and flow constraints respectively. Similar problems can be found in works dealing with CG algorithms for graph coloring (see section 2.3 to know the relation to our problem), see [31] for more details.

The mathematical model is decomposed into a master problem and a sub-problem models. We solve the master problem with a small subset of columns (rounds)  $\mathcal{R}' \subseteq \mathcal{R}$ , which serves as an initial basis. The sub-problem is then solved to check the optimality of the solution under the current master basis and to generate new columns for the master problem. This procedure repeats until the master problem contains all columns necessary to find the optimal solution of the original problem. Considering the RWP, each column corresponds to one round.

In each iteration, if the sub-problem can find a new column that may improve the master solution, this column is inserted in the master basis and a new master solution is computed. If the sub-problem can not find, linear programming and duality theory ensure that the solution of the problem is optimal. This algorithm is represented in figure 2.4.

The master problem (that has a set of rounds as input) is a linear programming problem that is known to be solved in polynomial time using, for example, Ellipsoid method [Yudin, Nemirovski and Shor, 1972], randomized polynomial-time Simplex algorithm [32] (polynomial time with high probability), etc. The complexity is solving the sub-problem which is combinatorial, and finding the optimal solution is NP-hard.

We will see in section 4.2 that in our tests this complexity is delimited to only a small part of the graph, the bottleneck region.

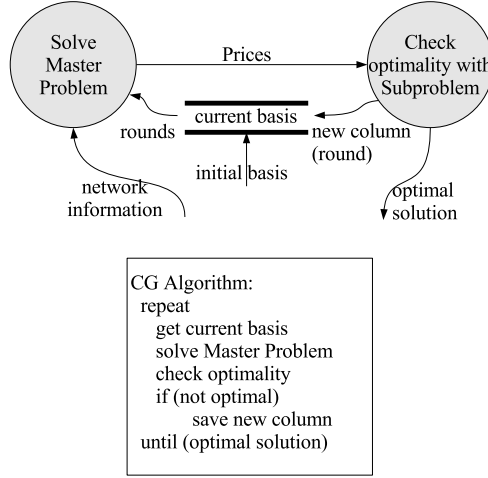


Figure 2.4: Column generation algorithm and data flow diagram.

### 2.5.1 Master problem formulation

In this subsection we define some variables of the master model. Let the variable  $\phi(i, j)$  be the flow over link  $(i, j)$ . The demand from each router  $v$  is represented by the integer parameter  $b(v)$ . Recall that  $B = \sum_v b(v)$ . Let the binary parameter  $a_{i,j}^R$  be 1 if the link  $(i, j)$  is active in the round  $R$ , and 0 otherwise.

Recall that set  $I_{u,v}$  is composed by all links interfering with  $(u, v)$ . We define  $F_{(u,v)}^{(i,j)} = 0$  if  $(i, j) \in I_{u,v}$  and 1, otherwise. We define  $w(R)$  as the fraction of time that round  $R \in \mathcal{R}'$  is active. Consequently, there is an induced edges capacity  $c(i, j) = \sum_{R \in \mathcal{R}} a_{i,j}^R w(R), \forall (i, j) \in E$ .

The master problem can be defined as follow: Given a graph  $G(V_r \cup V_g, E)$ , a set of routers demand  $b(v)$  and a set of rounds  $\mathcal{R}' \subseteq \mathcal{R}$ , the problem is to assign a weight  $w(R)$  to each round  $R \in \mathcal{R}'$ . The weights represent the amount of time a round will be active. The total amount of time needed to satisfy all demand will be  $W = \sum_{R \in \mathcal{R}'} w(R)$ . The master problem expressed as a linear programming model is the following:

$$W_{min} = \min \sum_{R \in \mathcal{R}'} w(R) \quad (2.13)$$

$$\sum_{i \in V_f(v,i) \in E} \phi(v,i) = b(v), \forall v \in V_r \quad (2.14)$$

$$\sum_{j \in V_g} \sum_{i \in V_r / (i,j) \in E} \phi(i,j) = B \quad (2.15)$$

$$\phi(i,j) = -\phi(j,i), \forall (i,j) \in E \quad (2.16)$$

$$\sum_{R \in \mathcal{R}'} a_{i,j}^R \cdot w(R) \geq \phi(i,j), \forall (i,j) \in E \quad (2.17)$$

Constraints (2.14-2.16) correspond to the flow constraints in Definition 2 based in [23] (see section 2.4). Constraints (2.14) define the flow leaving its source router. Constraints (2.15) define the flow arriving in the gateway set. Constraints (2.16) represent the skew symmetry (responsible for the opposite flow cancellation). Constraints (2.14), (2.15) and (2.16) correspond to the flow conservation. See example of the use of the flow constraints in figure 2.5. Constraints (2.17) assign weights to the rounds to satisfy the flow in the edges.

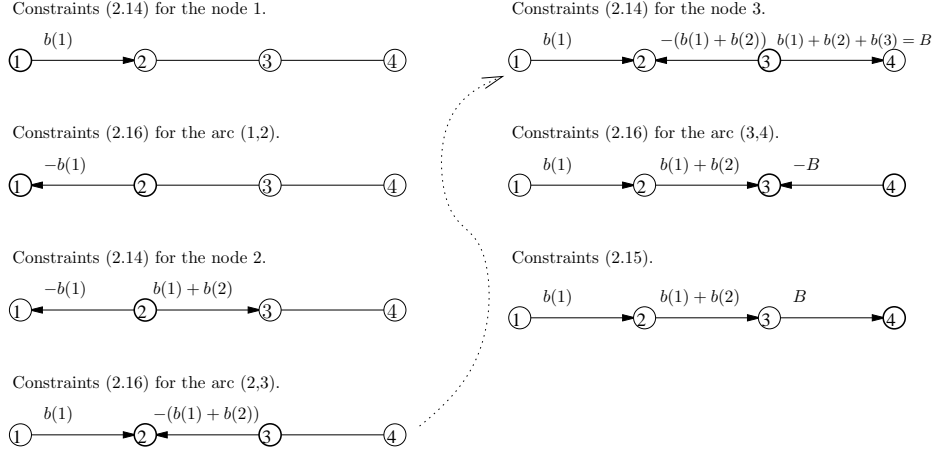


Figure 2.5: Example of flow problem using the formulation presented.

## 2.5.2 Sub-problem formulation

To express the sub-problem as a integer linear programming model, we have to define some additional notations. Let the parameter  $p_{(i,j)}$  be given by the dual variable associated with the constraints (2.17) of the master problem. Consider the binary variable  $u_{(i,j)} = 1$  indicating if the edge  $(i,j)$  enters the round to be added to  $\mathcal{R}'$ ,  $u_{(i,j)} = 0$  otherwise.

The sub-problem model generates a round  $R$  with the minimal *reduced cost*  $\left(1 - \sum_{(i,j) \in E} p_{(i,j)} \cdot a_{i,j}^R\right)$  to enter the master basis. It can be seen as the *maximum weighted independent set problem* which is NP-hard [33]. The parameter  $p_{(i,j)}$  corresponds to the weight of the edges. The objective function is to max-

imize the sum of the weights of all active edges respecting interferences. The formulation of the sub-problem is the following:

$$\max \sum_{(i,j) \in E} p_{(i,j)} u_{(i,j)} \quad (2.18)$$

$$u_{(i,j)} + u_{(k,l)} \leq 1 + F_{(i,j)}^{(k,l)}, \forall (i,j) \in E, \forall (k,l) \in E \quad (2.19)$$

The objective function (2.18) searches the maximum weight, which is equivalent to the minimum reduced cost. The parameter  $p_{(i,j)}$  guides the column generation to select the best round. Constraints (2.19) avoid interferences according to the interference model in  $F$ .

If the value of the objective function in the sub-problem is strictly greater than 1 (e.g. the reduced cost is negative), a new column  $u_{(i,j)}$  is found and the master basis is expanded. Otherwise, the master problem already gives the optimal solution to the original problem.

### 2.5.3 Cases solved in polynomial time

There are some cases of RWP that can be solved in polynomial time with the proposed *CG* formulation, as proved in theorem 1.

**Theorem 1** *Consider a linear model with exponential number of variables and a fixed number of constraints. If the pricing problem for this linear model can be solved in polynomial time then the linear model can be solved in polynomial time.*

**Proof:** The pricing problem of our CG model is the separation problem, as the CG algorithm corresponds to the dual of the cutting plane method. According to Grötschel, Lovász, and Schrijver on the ellipsoid method [34], a cutting plane algorithm runs in polynomial time if and only if the separation problem can be solved in polynomial time. ■

**Corollary 1** *The RWP can be solved in polynomial time if the maximum weighted independent set can be solved in polynomial time for  $C(G)$ .*

An example for the corollary 1 is the RWP considering the distance- $d$  model and  $d = 1$ . In this case, the pricing problem is the maximum weight matching problem, that is known to be solved in polynomial time.

## Chapter 3

# Load versus time: a multi-objective analysis

In this chapter we present the results of the work in [4]. We propose a multi-objective approach for the *RWP*, considering two objectives. The first one is to balance the load in the routers (*MinMaxLoad*). It increases the security in case of failure. The second objective is to minimize the communication time (*MinTime*). It corresponds to the time required to route all router demands. We observe that the worst bottleneck is located around the gateway in the instances of test.

*Multi-objective optimization* does not compute a unique solution, but a set of “best” solutions, called the *Pareto optimal set*, capturing the trade-offs between the different metrics. Solving a multi-objective problem consists in finding the Pareto optimal set, from which the decision maker choose the solution that fits the best his needs. In this work, each point of the Pareto set is obtained by solving an optimization problem.

The main contribution of this work is to give a multicriteria vision of the Round Weighting Problem. As far as we know there is no multi-objective analysis in this subject.

This chapter is organized as follows. The section 3.1 presents our multi-objective approach with  $\epsilon$ -restricted technique that uses the model presented in chapter 2. In section 3.1.2 are presented some of the experimental results obtained. We conclude the chapter and give the future directions in section 3.2.

### 3.1 Multi-objective formulation

To evaluate the overall quality of our solutions, we use the following metrics:

- *MinMaxLoad* ( $f^1$ ): Balancing the quantity of flow in the routers. The rounds are chosen in a way to minimize the maximum load  $l_v$  in the routers  $V_r$ .

- *MinTime* ( $f^2$ ): Minimizing the time of the communication. It chooses the rounds in a way that the round activations time will be minimum, that is, it minimizes the total weight  $w(R)$  of the schedule.

The objective function of the master problem with objective *MinMaxLoad* and *MinTime* are the following, respectively

$$\min(f^1 = \max_{v \in V_r}(l_v)) \quad (3.1)$$

$$\min(f^2 = w(R)) \quad (3.2)$$

To study the trade-offs between these two metrics, we consider the problem as a multi-objective one. The main idea of multi-objective optimization is to find out all the possible *non-dominated* solutions of an optimization problem. A solution is *dominated* if there is another solution improving simultaneously all the metrics. A solution is non-dominated if there is no other solution dominating it. Informally speaking, it means that if a solution is non-dominated within the whole solution space, it is not possible to improve one of the metrics without worsening at least one of the other metrics. The set of all non-dominated solutions is the *Pareto set* [35].

In multi-objective optimization, the solution space is a part of  $\mathcal{R}^m$  where  $m$  is the number of metrics. In our case  $m = 2$ . The optimization is performed on the plane and as there is no total order relation in  $\mathcal{R}^2$ , there is not a single but many “best solutions”.

A multi-objective optimization problem can be defined in the following way:

$$\bar{F} = \left\{ \begin{array}{l} \min_x F(x) \\ x \in \mathcal{P} \end{array} \right\} \quad \text{where } F : \left\{ \begin{array}{ll} \mathcal{P} & \rightarrow \mathcal{R}^2 \\ x & \mapsto \begin{pmatrix} f^1(x) \\ f^2(x) \end{pmatrix} \end{array} \right\} \quad (3.3)$$

$\mathcal{P}$  is the set of feasible solutions, defined by constraints (2.14) to (2.17).

### 3.1.1 $\epsilon$ -restricted technique

The idea of the  $\epsilon$ -restricted technique is to add additional constraints preventing the solver to return one of the optimum solution of one of the induced mono-objective problems, as described in [36] and [37]. More precisely, the  $\epsilon$ -restricted technique corresponds to generating and solving mono-objective problems under the form:

$$\left\{ \begin{array}{l} \min_x f^i \\ x \in \mathcal{P} \\ f^j \leq \epsilon^j; j \neq i \end{array} \right. \quad (3.4)$$

The  $\epsilon^i$  are chosen such that  $\bar{f}^i \leq \epsilon^i$ , where  $\bar{f}^i$  corresponds to the optimum value of the mono-objective problem minimizing objective  $f^i$ . Figure 3.1 illustrates the key idea of the  $\epsilon$ -method: Solving the classical mono-objective problem minimizing  $f^1$  gives  $\bar{f}^1$ . If the restriction  $f^2(x) \leq \epsilon^2$  is added to the

problem, minimizing  $f^1$  will not return  $\bar{f}^1$  anymore but another points of the Pareto optimal set. The same can be applied when minimizing  $f^2$ .

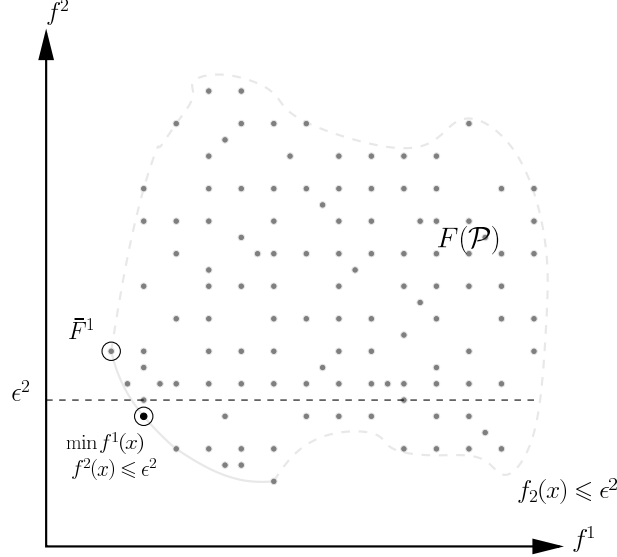


Figure 3.1:  $\epsilon$  based method minimizing  $f^1$

The Pareto set provides to the decision maker the trade-offs, allowing him to choose the solution that he considers as the best one.

### 3.1.2 Min time versus min-max load

The model was coded using the AMPL modeling language and it was solved using the commercial software Cplex version 10, on a desktop PC with one gigabyte of RAM. We used the mesh networks instances from [38]. We defined a simple interference model where each edge interferes with another one if the distance between them in graph  $G$  is lower than 2.

We represent some of the obtained results on figures 3.2 to 3.4. The results are represented in the solution space: The  $x$  axis represents the communication time, and the  $y$  axis represents the maximum load. Each point corresponds to a solution.

This approach using column generation and multi-objective optimization appears to be quite efficient, as the computation time to solve any instance is low, of the order of tenths of seconds. The overall time  $f^1$  as well as the maximum load  $f^2$  decrease as the number of gateways increases.

As it was expected, the routing generates bottlenecks located around the gateway(s), because all the flow goes toward them. We observe that when the routing use distinct paths to route the flow, it allows to activate different edges in the same round, reducing the overall transmission time. Informally speaking,



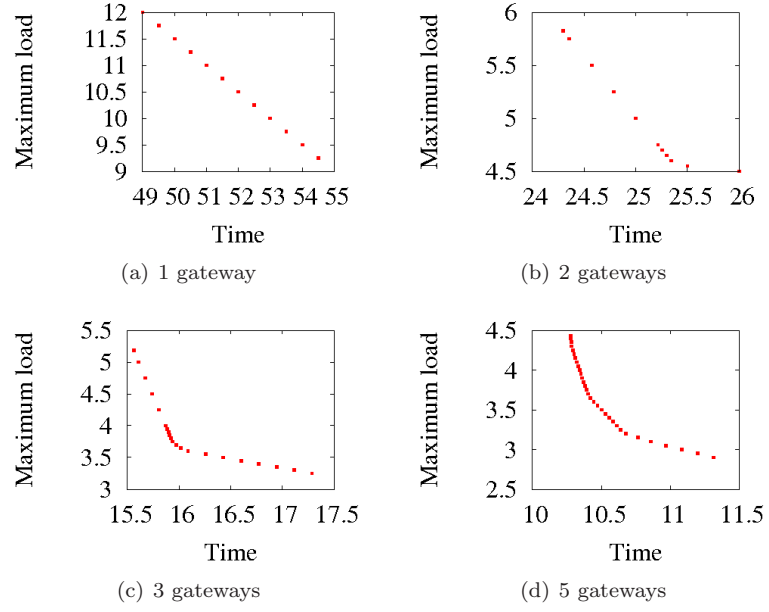


Figure 3.2: 39 nodes mesh network (giul69 instance)

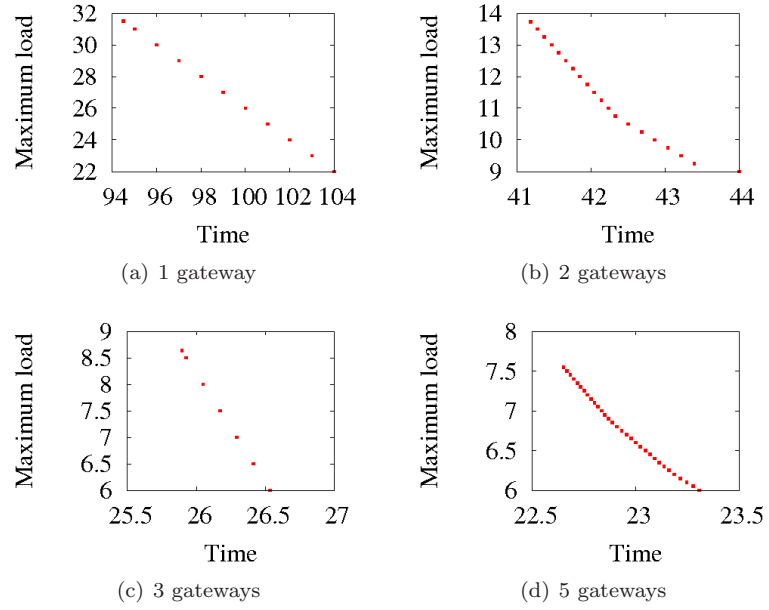


Figure 3.3: 65 nodes mesh network (ta2\_65 instance)

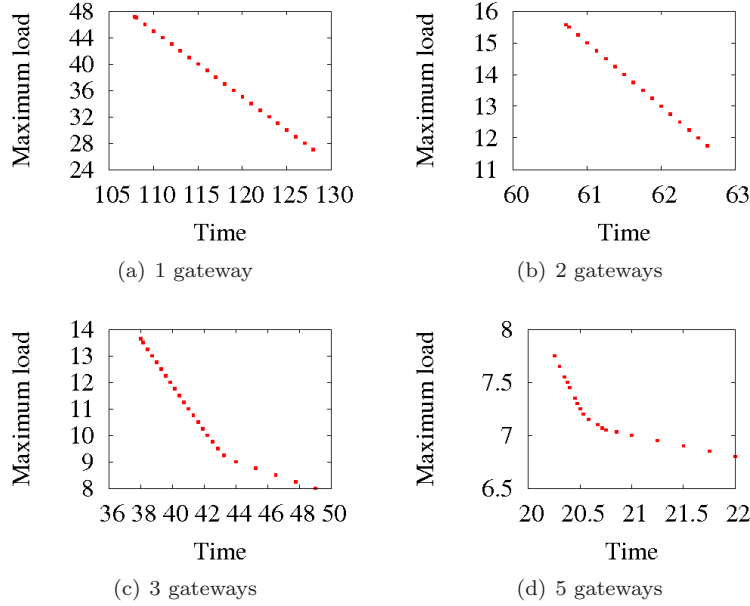


Figure 3.4: 54 nodes mesh network (zib54 instance)

it may be more efficient to follow different routes that do not interfere one with another than following shorter routes resulting in more interferences.

Minimizing the time increases the maximum load of the routers. We observe that the relation between the maximum load and the transmission time seems convex and piecewise linear. The linear parts corresponds to the following situations: As we make tighter the value of the maximum load, an amount of flow is deported on another path. Using this other path results in an increase of the overall transmission time. Hence, for each unit of flow following the second path, the overall transmission time increases by a given value (the difference of time between the first path and the second one).

Each disruption in the graphs is due to the happening of a new saturated node (bottleneck), forcing a flow transfer on a path that is not the best possibility. This can happen, for example, due to a transfer of flow from a gateway to another one having the neighbor nodes less saturated. It may activate some path that was not in use. As a consequence, the rate of time per flow increases.

A disruption situation is illustrated on figure 3.5, where routers 2,4 and 5 send data to gateway 1. We assume that each router has only one unit of traffic to send,  $b(v) = 1$ . For clarity reason, we consider that the time unit is the second (s) and the flow unit is the Gigabit (Gb), even if the formulation is independent from the unit chosen. Let us consider only the flow from router 4, because routers 2 and 5 send directly their flow to the gateway. The flow from router 4 follows three different paths  $p_1 = 4 - 5 - 1$ ,  $p_2 = 4 - 3 - 2 - 1$

and  $p_3 = 4 - 3 - 7 - 8 - 6 - 1$  to reach the gateway. When the maximum authorized load is 1.25Gb, the flow from router 4 is divided the following way: 0.25Gb follow  $p_1$ , 0.17Gb follow  $p_2$  and 0.58Gb follow  $p_3$ . Router 5 is the only bottleneck of the network. With a tighter maximum load of 1.2Gb, 0.05Gb of flow are deported from path  $p_1$  to paths  $p_2$  and  $p_3$ , resulting in an increase of required time of 0.02s, that is, with a rate of  $-0.05/0.02 = -2.5\text{Gb/s}$ . But now there are two bottlenecks: routers 2 and 5. With a tighter maximum load of 1.15Gb, flow from paths  $p_1$  and  $p_2$  are deported to the path  $p_3$ , resulting in an increase of required time of 0.1s, that is, with a rate of  $-0.05/0.1 = -0.5\text{Gb/s}$ . The overall results obtained with this example are represented on figure 3.6.

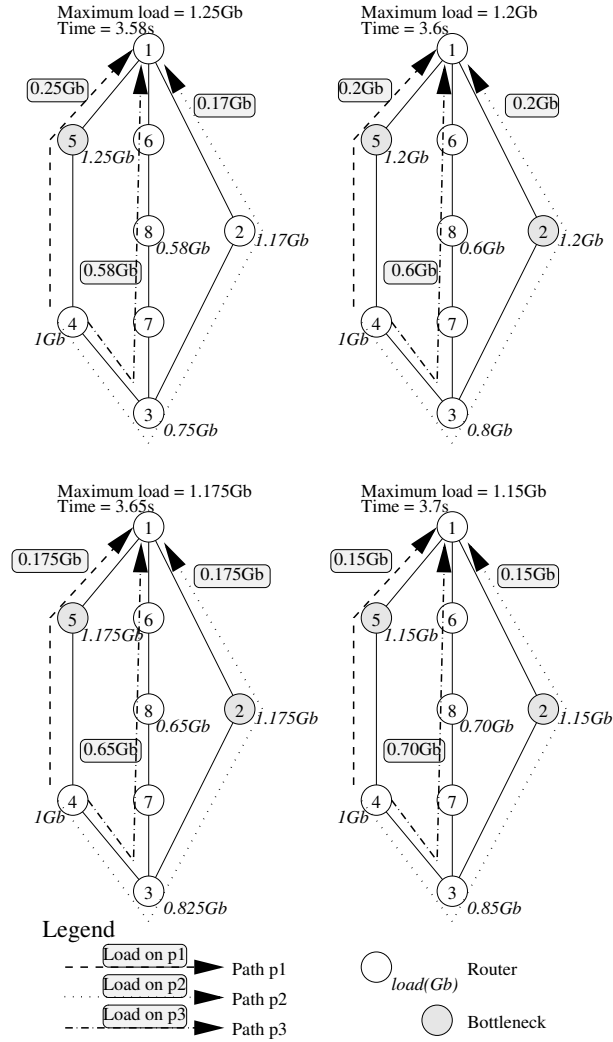


Figure 3.5: Small example for piecewise linear functions.

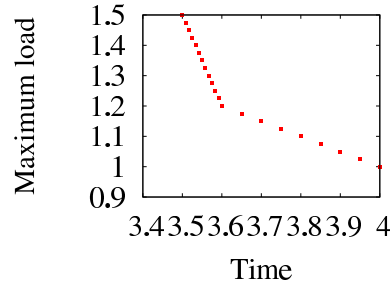


Figure 3.6: Result obtained with the example from figure 3.5.

## 3.2 Conclusion

In this chapter, we propose a multi-objective approach relating the overall transmission time, expressed in number of rounds, and the maximum load for the RWP. The problem is solved using a column generation approach.

We make experiments with some networks with different number of gateways. The multi-objective approach allows us to obtain results about the relationship between the maximum load and the overall transmission time. This relationship corresponds to a convex piecewise linear function. Each linear parts corresponds to the increase of time resulting by transferring part of the load from a path to another. Each disruption is due to the happening of a new bottleneck, forcing a flow transfer from a path to another.



## Chapter 4

# Extracting information of bottlenecks

In this chapter, we present a method which computes *lower bounds* for RWP. First we explain how to identify bottleneck regions (also called critical regions) in  $G$ . Then the method checks if these regions can give good lower bounds for RWP by studying the flow cutting across these regions. Of course, the maximum lower bound found among the selected regions is the best one.

A probable bottleneck is defined as a function of the topology of the graph, demand distribution and routing conditions. For example, a probable bottleneck can be a subgraph containing edges in narrow areas (e.g. min-cut of  $G$ ) or edges concentrating a high demand (close to  $B$ ) of flow going through it. Thus, depending on the topology of the graph and the demand distribution, a bottleneck can be located anywhere in the network.

Solving RWP in a subgraph of  $C(G)$  gives a *lower bound* for the problem independently of how this subgraph is defined. In general, solving RWP even in such a delimited region (see definition 3) remains NP-hard.

**Definition 3 (Localized RWP)** *Solving RWP in a bottleneck region means to solve RWP for the graph  $G$  computing the rounds only to cover the flow cutting across this given bottleneck region. Note that part of the flow may avoid (if it is possible) this region using then zero rounds. For the unavoidable flow, there are several ways of cutting across the given region. The objective is the same: Finding the rounds and their activation times giving a minimum period length  $W_{min}$ . The rounds have to provide edge capacities  $c(u,v)$  only for the bottleneck region admitting an unavoidable flow  $B' \leq B$  crossing this region.*

Figure 4.1(a) shows a first simple case, where we study the bottleneck region considering the incoming edges of the gateway. In this case  $B'$  will be the total flow  $B$ . It is also true for the case considering multiple gateways in figure 4.1(b). Figure 4.1(c) shows the bottleneck region of a relay node  $v$ , considering its outgoing and incoming edges. Thus there is a flow of  $B'$  crossing the region

and, a flow of  $B'$  plus the node demand  $b(v)$  leaving. The same thing happens for the bottleneck region considering the outgoing and incoming edges of several relay nodes together as illustrated in figure 4.1(d). It can represent a min-cut in  $G$  or, any other cut that will be very charged due to the demand distribution and routing conditions.

Figure 4.1(e) shows a special case considering a unique source (see example in figure 4.6), when we study the bottleneck region considering the outgoing edges of a node  $v$ . In this case  $B'$  will be  $b(v)$ . Similarly, we can also consider several sources together. These cases are reserved for big demand concentrations (close to  $B$ ) in these nodes, and we do not need consider the other nodes of the network.

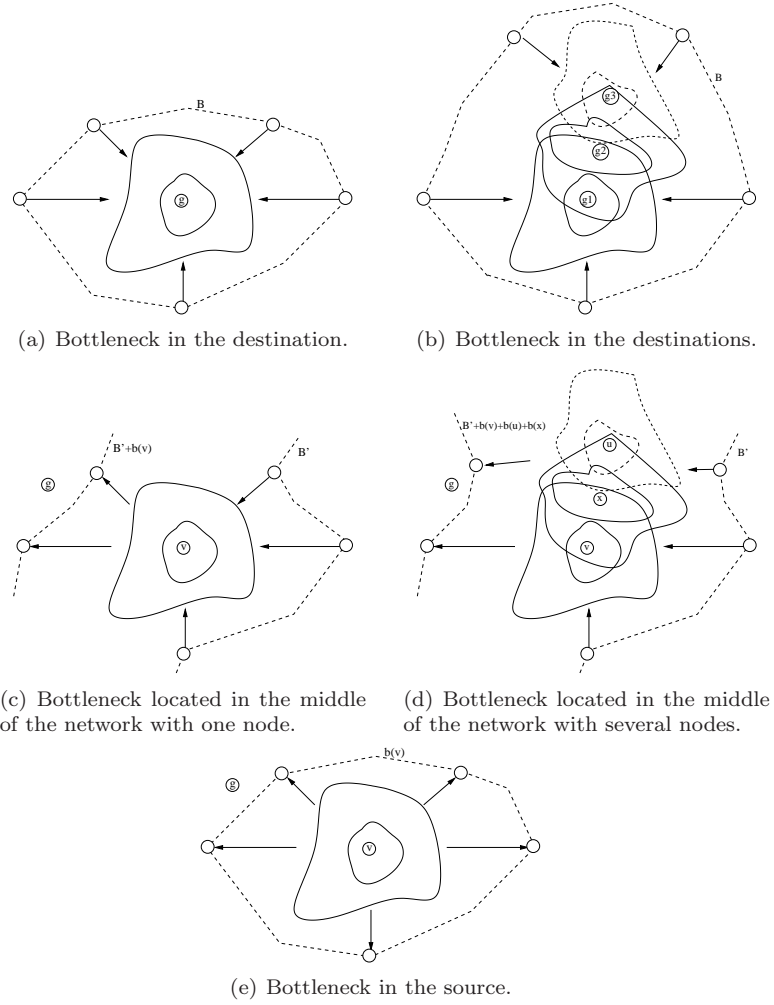


Figure 4.1: Types of bottlenecks and positions.

In the next section, we discuss a specific region defined by some cliques in  $C(G)$ .

## 4.1 Minimum maximal weighted clique

We saw that we can derive a lower bound by solving RWP in any bottleneck region. In this section, we introduce definition 4 that describes how to find lower bounds taking into account a region covered by a set of cliques. Let a *clique* represent a set of nodes in  $C(G)$  representing edges in  $G$  pairwise interfering.

**Definition 4 (Min-Max weighted clique)** *The weight of a clique is represented by the sum of the weight (or flow) of its edges. The min-max weighted clique is obtained by computing the routing problem of RWP (without the rounds computation) that minimizes the weight of the maximal weighted clique among the given cliques into a bottleneck region.*

The definition of clique is important because all flow (weight) on the edges of a clique will be considered in the rounds. In fact, they are added together as all edges in a clique interfere one with each other. That is why the objective is to minimize the weight of the maximal weighted clique. We will see that in our results and examples, some cliques are often enough to obtain a tight lower bound. This bottleneck region is composed by the closest maximal cliques to a probable bottleneck in  $G$ . We call  $LB_c$  the lower bound obtained with the maximal weighted clique given by definition 4.

Minimizing the maximal weighted clique means paralleling more activations, thus it minimizes the total number of colors needed. So we deduce a lower bound that is  $LB_c \leq \omega(C(G)')$ . Using the graph transformation of Mycielski [39], it was already proved in [40] that the difference between the fractional chromatic number  $\chi_f$  and  $\omega$  can be large. In our case, the weights given by a flow seem to make  $\chi_f(C(G)')$  getting close to  $\omega(C(G)')$  hence to  $LB_c$  (see next subsection).

Consider the example of a grid graph with the gateway (black node) in the middle and  $d = 2$ . Figures 4.2(a), 4.2(b), 4.2(c) and 4.2(d) depict the selected cliques around the gateway.

Figure 4.2(e) shows an example of a bad routing (the clique has not the minimum weight) giving a worse result. Figure 4.2(f) gives the best routing. The best configuration shows that the 4 gray nodes close to the gateway have to go backwards and then use the central axes to reach the gateway. In fact, this strange routing minimizes the maximal weighted clique of weight  $LB_c = B^{\frac{5}{4}} - 1$  instead of  $LB_c = B^{\frac{5}{4}} - \frac{1}{2}$  (with the non-MinMax weighted cliques). We improve the time of  $\frac{1}{2}$ , and it corresponds to the optimal solution as proved by the work in [5]. Figures 4.3(a) and 4.3(b) depict the selected cliques around the gateway. Figure 4.3(c) shows a bad routing for a grid graph with the gateway in the corner. Figure 4.3(d) shows a flow routing giving the minimum maximal weighted clique of weight  $LB_c = B^{\frac{3}{2}} - \frac{1}{2}$ . This lower bound was also proved optimal by the work in [5].



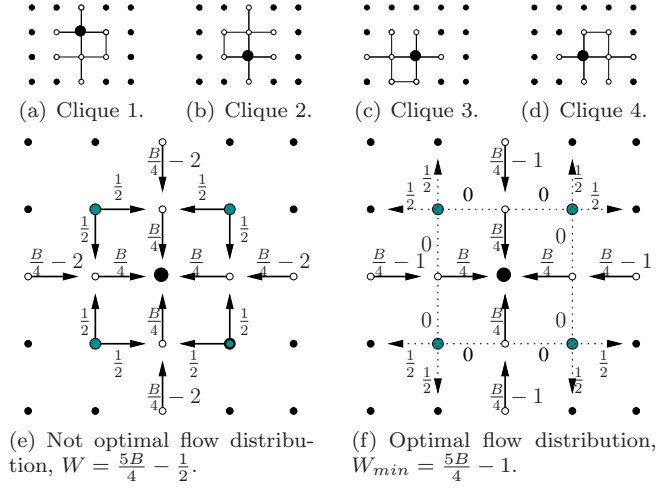


Figure 4.2: Looking for the best routing in a grid graph with the gateway in the middle with uniform demand (d=2).

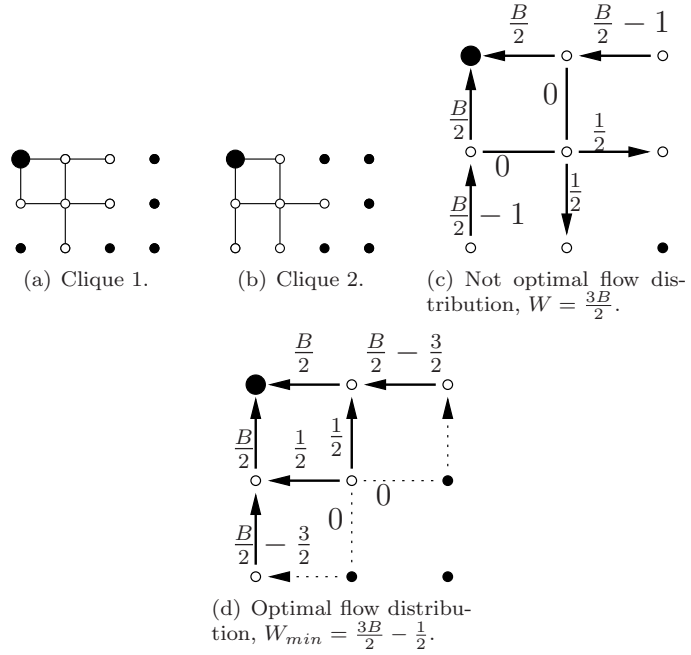


Figure 4.3: Looking for the best routing in a grid graph with the gateway in the corner with uniform demand (d=2).

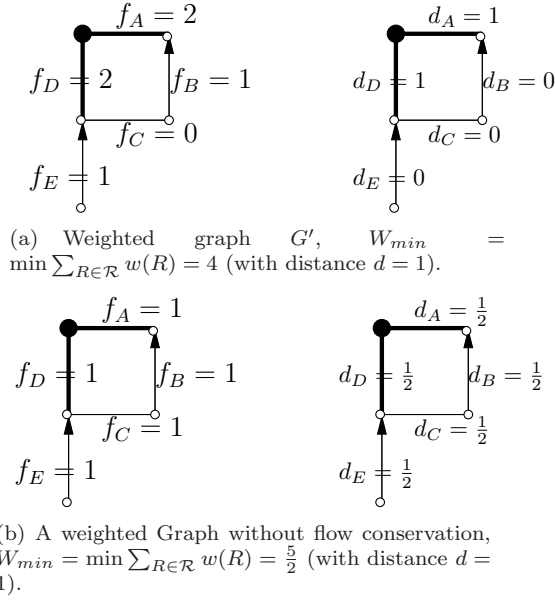


Figure 4.4:  $LB_c$  (cliques) formation from the dual weights.

## The duality principle

In this subsection, we explain why the bottleneck regions are so important for the quality of the lower bound. Indeed the non-zero variables of the dual problem for RWP (defined in section 2.3) are concentrated in a delimited region, the bottleneck region.

This is illustrated by the following example.  $G$  is a small graph with 5 nodes. Figure 4.4.a depicts the case in which the weights on the edges are obtained by a routing (i.e. with flow conservation), while the weights in figure 4.4.b are uniform (no flow conservation). Recall that the graph  $G'$  represents the graph  $G$  with the flow  $\phi(i, j)$  on its edges  $(i, j)$  given by an optimal routing.

Note that when the variables  $\phi(i, j)$  represent real flows the dual weights  $d_{i,j}$  concentrate in the bottleneck around the gateway as shown in figure 4.4.a. When the variables  $\phi(i, j)$  represent uniform weights, the dual weights  $d_{i,j}$  spread to other areas of the network.

We now explain why the dual values are concentrated in the bottleneck region. We notice that the closer the node is to the gateway, the bigger is the load due to the relaying. The links outside this region have slacks of activation. An edge has a *slack* when it has several possible options to get activated forming a *round* with edges on the bottleneck region. It is easy to assign time slots (colors) to the edges outside this region because once this critical region covered they will have several possibilities to get activated without modifying the total routing time  $W_{min}$ .

Suppose  $(i, j)$  is an edge in  $G'$  with a weight  $\phi(i, j)$ . Consider  $M$  the sum of the weights of the heaviest edges  $(u, v) \neq (i, j)$  from each round containing  $(i, j)$ . If  $\phi(i, j) \geq M$ , then it is more interesting to assign  $d(i, j) = 1$ . We can write  $M = \sum_{(u,v) \in R | (i,j) \in R, (i,j) \neq (u,v)} \phi(u, v) h_{u,v}, \forall R \in \mathcal{R}$ , where  $h_{u,v} = 1$  if  $\phi(u, v) = \max_{(k,l) \in R \setminus \{(i,j)\}} \phi(k, l)$ , and  $h_{u,v} = 0$  otherwise. To be clearer, for the cases where  $\chi_f = \omega$ , all these edges  $(i, j)$  compose the clique and no more than the colors already used by them are needed. Moreover, each round contains at least one edge of the clique.

In figure 4.4(a), there is a graph with a given routing. The edge  $A$  gets a weight of 2 therefore  $\max \sum_{(i,j) \in V(G')} d_{i,j} \phi(i, j) = 2d_A + 1d_B + 0d_C + 2d_D + 1d_E = 4$ . It implies  $d_A = 1$  and  $d_D = 1$ . It means that we only need to analyze the min-max weighted clique formed by the edges  $A$  and  $D$  (in bold) to obtain the optimum solution, as explained above. We observe that the bottleneck makes  $\chi_f$  get close to  $\omega$  and consequently, to the  $LB_c$ .

## 4.2 Experimental results

In this section, we confirm the validity of our bounds experimentally. We use the model presented in section 2.5<sup>1</sup> that is implemented using the AMPL modeling language. The instances are solved using the mathematical programming engine CPLEX version 10, on a desktop PC with two gigabyte of RAM. We use the public network graph representation from [38], so that our experiments can be reproduced. We make also some experiments with grid graphs.

In our experiments, we consider many gateways randomly selected, a symmetric interference model with  $d = 2$ , and equal bandwidth requirements  $b(v) = 1, \forall v \in V_r$  (uniform bandwidth).

Table 4.1 [38] gives the network topology characteristics. The solutions  $W_{min}^i$  represent  $W_{min}$  with integer round weights, that is the optimal solution for the IRWP. Table 4.1 also shows the solutions  $W_{min}^f$  that are  $W_{min}$  considering fractional round weights, that is the optimal solution for the RWP. Both solutions are computed using only the CG algorithm; for  $W_{min}^i$  we set as integer the variables  $w(R_i)$ . Our integer results are guaranteed to be optimal only because we have a feasible solution that respects  $W_{min}^i = \lceil W_{min}^f \rceil$  as shown in Table 4.1. Of course, to deal with general graphs we would need a Branch and Price algorithm [41]. The computation time to solve any of these instances was low, of the order of seconds.

If there are 2 or more gateways,  $W_{min}$  is not exactly divided by the number of gateways because they have different absorption rates or because they are close and their critical regions interfere one with another, as seen in table 4.1.

Figure 4.5 shows the giul network with  $W_{min}^f = 49$  that is equal to  $\omega(C(G'))$  and to  $LB_c$  (the weight of a min-max clique around the gateway). It means that the colors from this clique are enough to cover the paths from all 38 nodes to

<sup>1</sup>The source code can be found at <http://www-sop.inria.fr/members/Cristiana.Gomes/implementations.html>

Table 4.1: Networks topologies and results

Network	Gateways	Nodes	Edges	$W_{min}^f$	$W_{min}^i$
pdh	1	11	34	16	16
pdh	2	11	34	9.5	10
polska	1	12	18	15	15
atlanta	1	15	22	17.67	18
atlanta	3	15	22	7.71	8
newyork	1	16	49	18.5	19
newyork	3	16	49	6.67	7
france	1	25	45	54	54
france	3	25	45	14.5	15
nobel	1	28	41	38	38
giul	1	39	172	49	49

Table 4.2: Results to Grids with gateway in a corner

Network	Nodes	Edges	$W_{min}^f$	$W_{min}^i$
Grid3x3	9	12	11.5	12
Grid4x4	16	24	22	22
Grid5x5	25	40	35.5	36
Grid7x7	49	84	71.5	72
Grid8x8	64	112	94	94
Grid10x10	100	180	148	148

the gateway. The cliques are composed by the edges containing the numerical values representing the flow arriving in the gateway.

Usually the  $\omega(C(G)')$  is derived from the bottleneck in the gateway as in all our tests. But sometimes it may exist a clique larger than the one at the gateway that can not be avoided by the routing. When such clique exists,  $\omega(C(G)')$  can be derived from bottlenecks in other parts of the network. Figure 4.6 shows a unique source node at the corner that sends its traffic to the gateway node at the center of the grid, then  $B = b(v)$ . The  $\omega(C(G)')$  can be derived by the min-max weighted clique at the corner, so  $LB_c = \frac{3b(v)}{2}$ . Analyzing only the cliques around the gateway node at the center, we get a lower value of  $LB_c = \frac{5b(v)}{4}$ .

Table 4.2 (respectively 4.3) provides the results for grid graphs with the gateway in a corner (respectively with the gateway in the middle). The same phenomenon is observed: The lower bound given by the  $LB_c$  around the gateway is tight (see section 4.1). It is known that the coloring of perfect graphs<sup>2</sup> present the clique number  $\omega$  tight, but our conflict graphs are not perfect.

<sup>2</sup>A perfect graph has the chromatic number of every induced subgraph equals the clique number of that subgraph.

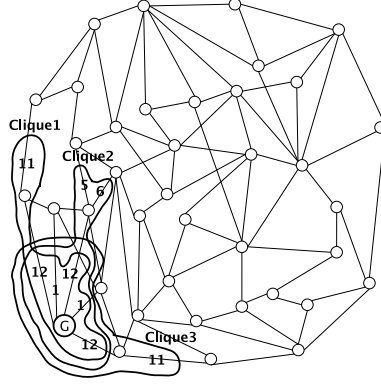


Figure 4.5: Giul network ( $B = 38$ ), solution  $W_{min}^f = 49$  ( $d=2$ ).

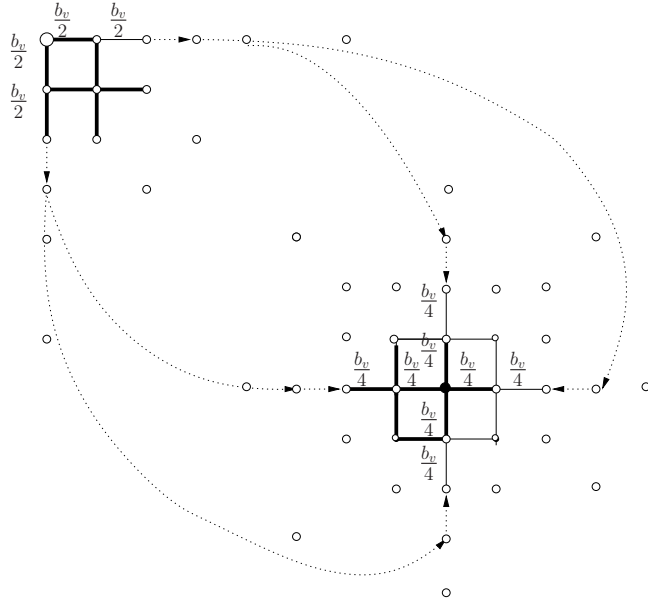


Figure 4.6: Bottleneck at the source node  $W_{min}^f = \frac{3}{2}$  ( $d=2$ ).

Table 4.3: Results to Grids with gateway in the middle

Network	Nodes	Edges	$W_{min}^f$	$W_{min}^i$
Grid3x3	9	12	10	10
Grid4x4	16	24	20	20
Grid5x5	25	40	29	29
Grid7x7	49	84	59	59
Grid8x8	64	112	77.75	78
Grid10x10	100	180	122.75	123

## Integer RWP

In this section we discuss the results for the Integer RWP. Table 4.1, 4.2 and 4.3 show that  $W_{min}^i = \lceil W_{min}^f \rceil$ . It usually happens due to the fact that there is a huge concentration of traffic in the critical region.

The traffic around the gateway is well distributed between the cliques (minimizing the maximum weighted clique) which results in cliques of the same size (or almost the same size for the integer result when  $B$  is not exactly divisible by the number of cliques). Figure 4.7 shows the cliques (represented by the bold edges) for fractional and integer cases on the Newyork network. Their sizes are respectively 18.5 and 19. The letters represent the rounds/colors.

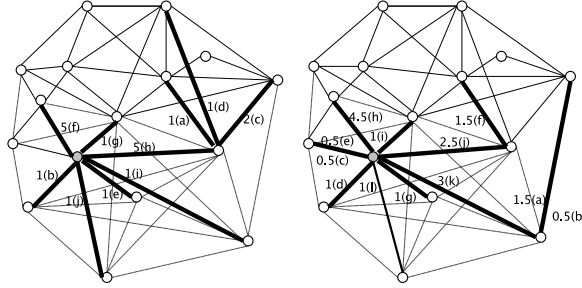


Figure 4.7: Newyork network ( $B = 15$ ) with solution  $W_{min}^f = 18.5$  and  $W_{min}^i = 19$  ( $d=2$ ).

Figure 4.8(a) shows one of the min-max cliques with size of only 12 (the other cliques are given by rotation of this one). Figure 4.8(b) shows a fractional solution with  $W_{min}^f = 12.5$  and Figure 4.8(c) shows an integer solution with  $W_{min}^i = 13$ . This example shows that  $W_{min}$  can be greater than the min-max clique. This result is not surprising if we consider that the conflict graph has

a  $C_5 = \{v_{1,9}, v_{2,8}, v_{3,7}, v_{4,6}, v_{5,10}\}$  (cycle of length 5). The  $C_5$  has the largest clique of size 2, but needs 3 colors to be covered.

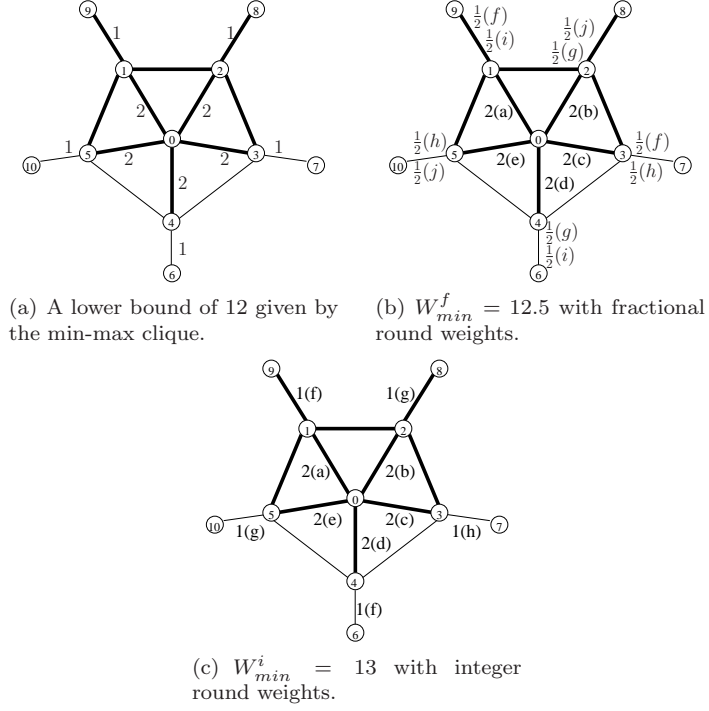


Figure 4.8: Example of lower bound with  $W_{min} \neq \omega(C(G)')$  ( $d = 2$ ).

Although according to our tests  $W_{min}^i = \lceil W_{min}^f \rceil$  seems to be true for our problem, a counter-example on Figure 4.9 shows that we can achieve  $W_{min}^i > \lceil W_{min}^f \rceil$  considering uniform demand. Notice that considering integer round weights we obtain a clique of weigh 2. The  $W_{min}^i = 3$  because considering  $d = 2$  we need 3 colors to cover a simple path.  $W_{min}^f = \frac{3b(v)}{2}$  as shown in Figure 4.6, thus we have  $W_{min}^i > \lceil W_{min}^f \rceil$ .

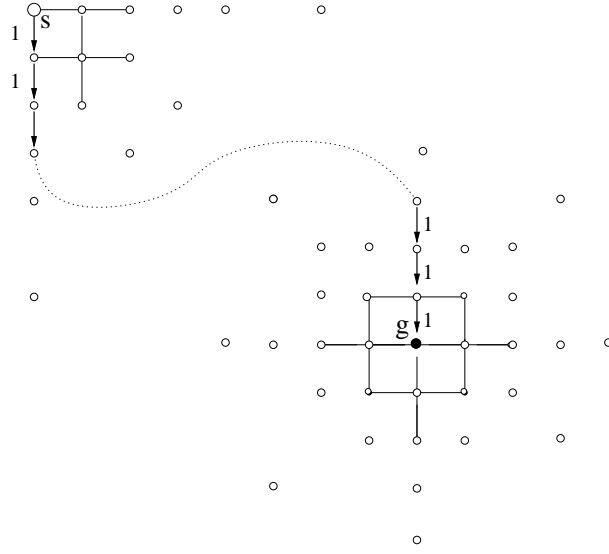


Figure 4.9: Example of  $W_{min}^i \neq \lceil W_{min}^f \rceil$  for RWP (d=2).  $W_{min}^i = 3$  and  $\lceil W_{min}^f \rceil = \frac{3}{2}$  (see figure 4.6).

### 4.3 Conclusion

In this chapter, we presented a method to compute *lower bounds* derived from a probable bottleneck region. We made experiments with networks with different numbers of gateways. In our experimental results only some cliques in the bottleneck region was enough to find the optimal solution for RWP.

Although experimentally  $W_{min}^i = \lceil W_{min}^f \rceil$  seems to be true for our problem, we showed a counter-example.





## Chapter 5

# General lower bound methods and application to grid graphs

This chapter presents formally methods to obtain lower (inspired by *chapter 4*) for general graphs. We present lower bounds independent of the adopted binary interference model. Then we give more precise lower bound (LB) for the distance- $d$  model (with any value of  $d$ ). Our methods are applied to grid graphs.

### 5.1 Definitions

In this section we will present some definitions that will be useful in future sections. We recall some definitions from Chapter 2 in a way to concentrate all definitions in the same place.

#### Definitions related to the edges of $G$

- $G(V, E)$ : Transmission graph with  $V$  as set of nodes (vertices)  $E$  the set of edges (calls).
- $L(G)$ : A graph whose vertices represent the edges of  $G$  and two vertices are joined in  $L(G)$  if their corresponding edges intersect.
- $d(u, v)$ : distance between  $u$  and  $v$ , that is the length of the shortest path between  $u$  and  $v$  (e.g. the neighbors of  $g$  are at distance 1 of  $g$ ).
- $d(e, e')$ : distance between edges  $e = (u, v)$  and  $e' = (u', v')$  which corresponds to  $\min_{x \in \{u, v\}, y \in \{u', v'\}} d(x, y)$ .
- $E_l$ : set of edges at level  $l$ , i.e. edges joining a node at distance  $l$  from the gateway to a node  $l - 1$ . More precisely,  $E_l = \{e = (u, v) \in E \mid d(g, u) =$

$l$  and  $d(g, v) = l - 1$ . Thus for example,  $E_1$  are all the edges which are adjacent to the gateway  $g$ .

- $K_0$ : set of edges in  $G$  at level at most  $\lceil \frac{d}{2} \rceil$  of the gateway  $g$ ,  $K_0 = \bigcup_{1 \leq l \leq \lceil \frac{d}{2} \rceil} E_l$ .
- $V_{K_0}$ : set of nodes in  $G$  at distance at most  $\lceil \frac{d}{2} \rceil$  of the gateway  $g$ .

#### Definitions related to the cliques

- $C(G)$ : conflict graph of  $G$ , that is the graph whose vertices represent the edges of  $G$ , two vertices are joined if the corresponding edges (which represent calls) interfere.
- Distance- $d$  model. Two edges (calls)  $e$  and  $e'$  interfere in the distance- $d$  model if  $d(e, e') < d$ . (See definition in section 2.1). The conflict graph is the  $d$ -th power of the line graph  $L(G)$ .
- call-clique: set of pairwise interfering edges. The corresponding vertices form a clique in  $C(G)$ . For example in the distance- $d$  model,  $K_0$  is a call-clique.

#### Definitions related to the flow and admissible rounds

- $b(v)$ : demand due to node  $v$ .
- $\phi$ : In what follows,  $\phi$  will always denote a feasible flow satisfying the demand  $b(v)$  defined by

$$\begin{aligned} \sum_{i \in V_l | (v, i) \in E} \phi_v(v, i) &= b(v), \forall v \in V_r \\ \sum_{j \in V_g} \sum_{i \in V_r | (i, j) \in E} \phi_v(i, j) &= b(v), \forall v \in V_r \\ \sum_{i \in V_r | (i, j) \in E} \phi_v(i, j) &= \sum_{k \in V_l | (j, k) \in E} \phi_v(j, k), \forall j, v \in V_r. \end{aligned}$$

- $\phi_v(e)$ : flow sourced at node  $v$  traversing the edge  $e$ .
- $\phi(e)$ : flow traversing the edge  $e$ .  $\phi(e) = \sum_{v \in V} \phi_v(e)$ .
- $R$  (Round): set of non-interfering edges. It corresponds to an independent set in  $C(G)$  (See definition in section 2.1).
- $\mathcal{R}$ : set of all rounds  $R$ .
- $\mathcal{R}_e \subset \mathcal{R}$ : set of all the rounds containing the edge  $e$ .
- $w(R)$ : weight of the round  $R$ .

- $c_w(e)$ : the capacity of the edge  $e$  in function of the rounds weight in  $\mathcal{R}_e$ ,  
 $c_w(e) = \sum_{R \in \mathcal{R}_e} w(R) = \sum_{R \in \mathcal{R}} w(R) |R \cap \{e\}|$ .

We will say that the weights  $w(R)$  assigned to the rounds  $R \in \mathcal{R}_e$  are admissible for the flow  $\phi$  if

$$c_w(e) \geq \phi(e) \quad \forall e \quad (5.1)$$

A weight function  $w$  is admissible if there exists a flow for which it is admissible.

- $\phi(E')$ :  $\sum_{e \in E'} \phi(e)$ . Sum of the flow on a set of edges  $E'$ .
- $c_w(E')$ :  $\sum_{e \in E'} c_w(e) = \sum_{e \in E'} \sum_{R \in \mathcal{R}_e} w(R) = \sum_{R \in \mathcal{R}} w(R) |R \cap E'|$ , the capacity of the edges  $E' \subseteq E$  is a measure derived of the rounds weight covering these edges.

Our *objective* is to minimize  $W = \sum_{R \in \mathcal{R}} w(R)$  on all the admissible weight functions. The minimum will be denoted  $W_{\min}$ . Now, we will show how to use call-cliques (in particular those centered at the gateway) to obtain lower bounds.

### Definitions related to grids

We will study in more details the grid topology. We consider the rectangular  $p \times q$  grid with  $N = pq$  vertices. We will use a coordinate system  $(x, y)$  where  $-p_1 \leq x \leq p_2$  with  $p_1 + p_2 + 1 = p$  and  $-q_1 \leq y \leq q_2$  with  $q_1 + q_2 + 1 = q$  ( $p_1, p_2, q_1, q_2$  being integers). The gateway will always have coordinate  $(0, 0)$ . The result will strongly depend of the position of the gateway in the grid. We will mainly consider two extremal cases to illustrate the results:

- Gateway in the *corner*, that is  $p_1 = 0, q_1 = 0$ .
- Gateway in the *middle*, which means that the gateway is far enough of the borders. In the distance- $d$  model, we will express that by supposing that  $\min(p_1, p_2, q_1, q_2) \geq f(g) \geq \frac{d+1}{2}$ , where  $f(g)$  is a function which associates to  $g$  its minimum distance to the borders.

-Rotation: we define the function rotation  $\rho$  as the one to one mapping  $\rho((x, y)) = (-y, x)$  which corresponds to a rotation in the plane of  $\frac{\pi}{2}$  around the central node  $(0, 0)$ . This definition works perfectly when  $p_1 = p_2 = q_1 = q_2$ . We can extend it to any grid by doing the rotation in a super grid with size  $(2p' + 1, 2q' + 1)$  with  $p' = \max(p_1, p_2, q_1, q_2)$  and ignoring the vertices not in the original grid.

## 5.2 Lower bounds: general results

In this section, we present lower bounds for the problem of RWP considering the distance- $d$  model. We give more precise results for grid graphs.

### 5.2.1 Lower bounds using one call-clique

Recall that a call-clique is a set of edges pairwise interfering. It means that, if two transmissions occurs in a call-clique, then they cannot be performed simultaneously. Thus, the sum of the capacities of the edges in a call-clique sets up a lower bound for the RWP as we will see in the following lemma.

**Lemma 1** *Let  $K \subseteq E$  a call-clique. Then  $c_w(K) \leq W$ .*

**Proof:** We know that  $c_w(K) = \sum_{R \in \mathcal{R}} w(R) |R \cap K|$ . As each round  $R$  is a set of independent edges,  $R$  contains at most one edge of  $K$ . Then  $|R \cap K| \leq 1$  and consequently  $c_w(K) \leq \sum_{R \in \mathcal{R}} w(R) = W$ . ■

For  $F$  a set of edges, and a path  $P_{v,g}$  between  $v$  and  $g$ , let  $\text{LB}(P_{v,g}, F)$  denote the number of edges that  $P_{v,g}$  and  $F$  have in common. Therefore,  $\text{LB}(P_{v,g}, F) = |P_{v,g} \cap F|$ . We define  $\text{LB}(v, F)$  as the minimum  $\text{LB}(P_{v,g}, F)$  over all the paths  $P_{v,g}$  between  $v$  and  $g$ .

**Lemma 2**  $c_w(F) \geq \sum_{v \in V} b(v) \text{LB}(v, F)$ .

**Proof:** For any flow  $\phi$  and any node  $v$ ,  $\phi_v(F) \geq b(v) \text{LB}(v, F)$ . ■

The first idea consists in choosing particular sets  $F$ . A natural candidate is the set  $E_l$  (of edges at level  $l$ ). The nodes *outside*  $E_l$ , i.e. the nodes at distance to the gateway at least  $l$ , must cross the edges  $E_l$  to reach the gateway. So, if  $d(v, g) \geq l$ , then  $\text{LB}(v, E_l) \geq 1$  and we have the following corollary.

**Corollary 2**  $c_w(E_l) \geq \sum_{v; d(v,g) \geq l} b(v)$ .

We will use corollary 2 to give a lower bound for  $c_w(K_0)$  where we recall that  $K_0$  is the set of edges around the gateway at level at most  $\lceil \frac{d}{2} \rceil$ .

First, we introduce the following definition that will be useful later.

**Definition 5**  $S_0 = \sum_{v \in V_{K_0}} d(v, g) b(v) + \lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v)$ .

It enables us to get a lower bound on  $c_w(K_0)$  which will be useful in the distance- $d$  model.

**Lemma 3** *In the distance- $d$  model  $c_w(K_0) \geq S_0$ .*

**Proof:** As  $K_0 = \bigcup_{l \leq \lceil \frac{d}{2} \rceil} E_l$  and the levels  $E_l$  for  $1 \leq l \leq \lceil \frac{d}{2} \rceil$  are pairwise disjoint, then  $c_w(K_0) = \sum_{l \leq \lceil \frac{d}{2} \rceil} c_w(E_l) \geq \sum_{l \leq \lceil \frac{d}{2} \rceil} \sum_{v; d(v,g) \geq l} b(v) = S_0$ . ■  
Note that the value  $S_0$  is independent of the function  $w$ . Therefore,

**Proposition 1** *In the distance- $d$  model  $W_{\min} \geq S_0$ .*

In some cases, the lower bound  $S_0$  is attained. It happens for the grid with the gateway in the middle and  $d$  odd (see theorem 8). The figure 5.1 shows an example of a clique in this case.

In some other cases we use lemma 2 with a maximum call-clique  $K$  containing  $K_0$ . For example, for the grid with  $d$  odd and the gateway in the corner, the

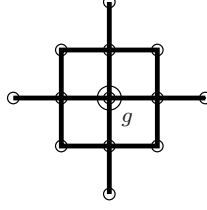


Figure 5.1: Clique for  $d$  odd with  $g$  in the middle. In this scheme,  $d = 3$ .

maximum call-clique is larger than  $K_0$  (see figure 5.7) and gives a better bound than  $S_0$  (Theorem 2). We will show that the bound is attained for uniform demand. However, using only one call-clique does not necessary give a tight bound.

### 5.2.2 Lower bounds using many call-cliques

We present a result similar to lemma 2 but improved for multiple sets of edges. We denote  $\mathcal{P}_{v,g}$  the set of all the paths between  $v$  and  $g$ .

**Lemma 4** *Given  $F_1, \dots, F_q$  sets of edges, then*

$$\sum_{i=1}^q c_w(F_i) \geq \sum_v b(v) \min_{P_{v,g} \in \mathcal{P}_{v,g}} \left( \sum_{i=1}^q \text{LB}(P_{v,g}, F_i) \right)$$

**Proof:** For any flow  $\phi$  and any node  $v$ ,  $\sum_{i=1}^q \phi_v(F_i) \geq b(v) \min_{P_{v,g} \in \mathcal{P}_{v,g}} \sum_{i=1}^q \text{LB}(P_{v,g}, F_i)$ . ■

Consider the example of a grid with the gateway at the corner and the distance-2 model ( $d = 2$ ) depicted in figure 5.2. We have two maximum call-cliques containing  $K_0$ :  $K_1$  and  $K_2$  which also contain the four edges leaving vertex  $(1, 1)$ . Furthermore  $K_1$  contains the edge  $e_1 = ((1, 0), (2, 0))$  and  $K_2$  contains the edge  $e_2 = ((0, 1), (0, 2))$ . For vertex  $v^* = (1, 1)$  both  $\text{LB}(v^*, K_1) = \text{LB}(v^*, K_2) = 2$ . For any vertex  $v$  different from  $(0, 1)$ ,  $(1, 0)$  and  $(1, 1)$  any path  $P_{v,g}$  from  $v$  to  $g$  must use one edge at level 2 either  $e_1$  or  $e_2$ , then  $\text{LB}(v, E_2) \geq 1$ . That implies that  $\text{LB}(P_{v,g}, K_1) + \text{LB}(P_{v,g}, K_2) \geq 2 \text{LB}(P_{v,g}, E_1) + \text{LB}(P_{v,g}, E_2) \geq 3$ . In this way, one of the call-clique will carry at least  $3/2$  of the flow of the vertices different from  $(0, 1)$ ,  $(1, 0)$  and  $(1, 1)$ . Using lemma 4, we get that

$$\begin{aligned} c_w(K_1) + c_w(K_2) &\geq \sum_v b(v) \min_{P_{v,g} \in \mathcal{P}_{v,g}} (\text{LB}(P_{v,g}, K_1) + \text{LB}(P_{v,g}, K_2)) \\ &\geq 2b((0, 1)) + 2b((1, 0)) + 4b((1, 1)) + 3 \sum_{v \notin \{(0, 1), (1, 0), (1, 1)\}} b(v) \end{aligned}$$

and so, one of this two call-cliques is greater than  $\frac{1}{2}$  of this value. Therefore, we have the following bound and we will see after that this bound is attained.

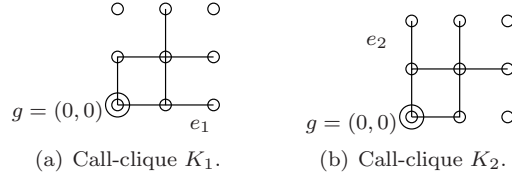


Figure 5.2: Two maximum call-cliques  $K_1$  and  $K_2$  for the case  $d = 2$ .

**Proposition 2** *For the grid with the gateway in the corner in the distance-2 model ( $d = 2$ )*

$$W_{min} \geq b(0, 1) + b(1, 0) + 2b(1, 1) + \frac{3}{2} \sum_{v \notin \{(0,1), (1,0), (1,1)\}} b(v)$$

In general we have the following lemma.

**Lemma 5** *Let  $K_1, \dots, K_q$  be a family of call-cliques. Then one of the call-cliques  $K^*$  satisfy  $c_w(K^*) \geq \frac{1}{q} \sum_{v \in V} b(v) \min_{\mathcal{P}_{v,g}} \sum_{i=1}^q \text{LB}(P_{v,g}, K_i)$*

**Proof:** By lemma 4,  $\sum_i c_w(K_i) \geq \sum_{v \in V} b(v) \min_{\mathcal{P}_{v,g}} \sum_{i=1}^q \text{LB}(P_{v,g}, K_i)$  and so one of the call-cliques, denoted  $K^*$ , has value  $c_w(K^*)$  greater than or equal to the mean. ■

**Corollary 3** *Let  $K_1, \dots, K_q$  be a family of call-cliques such that each edge of  $E_l$  appears at least  $\lambda_l$  times in the call-cliques, then  $W_{min} \geq \sum_l \sum_{v; d(v,g) \geq l} \frac{\lambda_l}{q} b(v)$ .*

**Proposition 3** *Let  $G$  be the grid with the gateway in the middle and  $d = 2k$  be even. Then*

$$W_{min} \geq S_0 + \frac{1}{4} \sum_{v; d(v,g) > k} b(v)$$

**Proof:** Consider the 4 following call-cliques (see figure 5.3 for  $d = 2$ ): They all contain the edges of  $K_0$ . Furthermore,  $K_1$  contains the edge  $((k+1, 0), (k, 0))$  and the edges at level  $k+1$  with positive coordinates:  $((k+1-i, i), (k-i, i))$  and  $((k+1-i, i), (k+1-i, i-1))$  for  $1 \leq i \leq k$ . The call-cliques  $K_2, K_3$  and  $K_4$  are obtained by successive rotation of  $\frac{\pi}{2}$  the previous call-clique. In this way the edges in  $E_l$ ,  $1 \leq l \leq k$  are covered 4 times and the edges in  $E_{k+1}$  are covered once. ■

We will see after that this lower bound is attained.

In some cases, we have to use the lemma with call-cliques which are not easy to find and do not necessary contain the gateway. An example of that is the case of the grid for  $d = 4$  with the gateway at the corner and the demand concentrated only in one node: the node  $(3, 2)$ . A lower bound consists in considering two call-cliques containing  $K_{max}$ .

A better lower bound for the same example consists in using call-cliques which do not all cover the gateway. The new lower bound uses the call-cliques

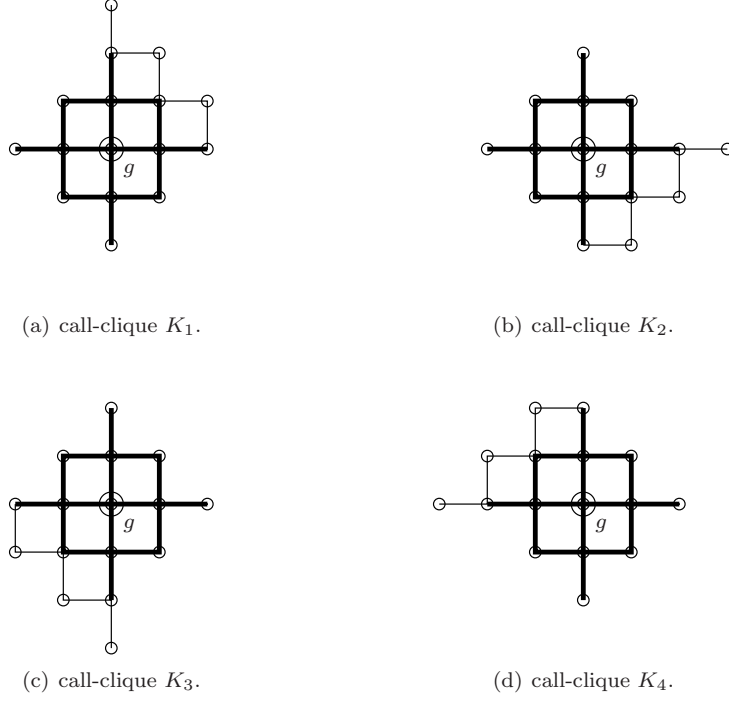


Figure 5.3: Case  $d$  even and  $g$  in the middle. The 4 call-cliques combined covers  $E_i$ ,  $1 \leq i \leq k+1$  for  $d = 2k$ . In this scheme,  $d = 4$ .

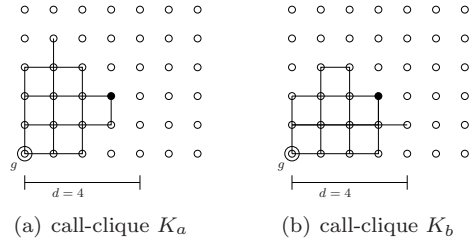


Figure 5.4: Example of a specific lower bound when the demand is concentrated in one node. In this example,  $d = 4$  and the demand is concentrated in node  $(3, 2)$ . A lower bound of  $\frac{5}{2}b((3, 2))$  is attained using the two call-cliques  $K_a$  and  $K_b$ .



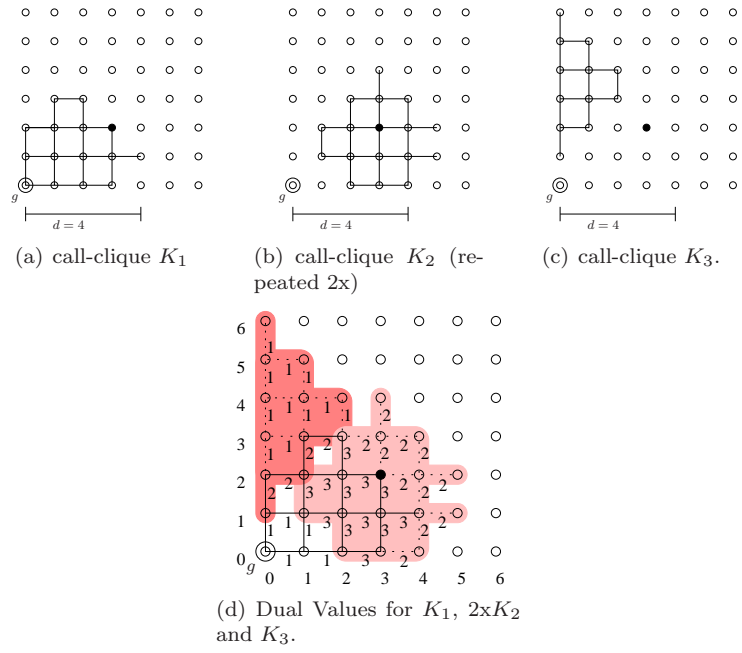


Figure 5.5: Example with  $d = 4$  and the demand is concentrated in node  $(3, 2)$ . Four call-cliques are needed to obtain a tight lower bound of  $\frac{11}{4}b((3, 2))$  which is higher than  $\frac{5}{2}b((3, 2))$ .

Table 5.1: Possible Paths from (3, 2) to the gateway (0, 0) cost less than 11.

$v_5$	$v_4$	$v_3$	$v_2$	$K_1$	$K_2(\times 2)$	$K_3$	Total
-	(4, 0)	(3, 0)	-	3	4	-	11
-	(3, 1)	(3, 0)	-	5	3	-	11
-	-	(2, 1)	-	5	3	-	11
-	(2, 2)	(1, 2)	(1, 1)	5	3	-	11
-	(2, 2)	(1, 2)	(0, 2)	5	2	2	11
-	(1, 3)	(1, 2)	(1, 1)	4	3	2	12
-	(1, 3)	(1, 2)	(0, 2)	4	2	4	12
-	(0, 4)	(0, 3)	-	2	2	5	11
(1, 4)	(1, 3)	(0, 3)	-	2	2	5	11
(2, 3)	(1, 3)	(0, 3)	-	3	2	4	11

depicted in figure 5.5. The call-clique  $K_2$  (see figure 5.5(b)) is used twice and  $K_1$  (see figure 5.5(a)) and  $K_3$  (see figure 5.5(c)) once. Consider a path from (3, 2) to the gateway (0, 0).

We consider different cases according the way the path arrives in  $g$ . More precisely, we consider the last vertex  $v_i$  at distance  $i$  from  $g$  used by the path with  $i \in \{2 \dots 5\}$ . We indicate in the following table the number of edges of  $K_1$ ,  $K_2$  (repeated twice) and  $K_3$  the path uses.

It is simple to check in figure 5.5(d) that we do not have path from (3, 2) to the gateway (0, 0) that costs less than 11 (see the table 5.1). Then  $\min_{P \in \mathcal{P}_{(3,2),g}} (\text{LB}(P, K_1) + 2\text{LB}(P, K_2) + \text{LB}(P, K_3)) \geq 11$ . Then, one of the call-clique  $K^*$  satisfy  $c_w(K^*) \geq \frac{11}{4}b((3, 2))$ .

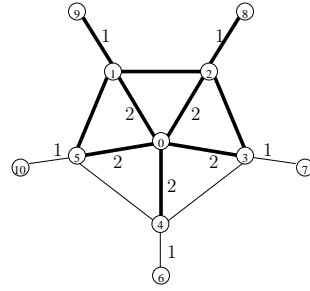
### 5.2.3 Lower bounds using Critical Edges

Lemma 5 does not attain the best lower bounds in all cases. Consider the example of figure 5.6 with  $d = 2$ . We have 5 maximal call-cliques all containing the edges at level 1 plus two consecutive edges at level 2. Then, applying corollary 3 and noting that each edge at level 2 appears exactly in two call-cliques we get  $W \geq \sum_{v;d(v,g) \geq 1} b(v) + \frac{2}{5} \sum_{v;d(v,g) \geq 2} b(v)$ .

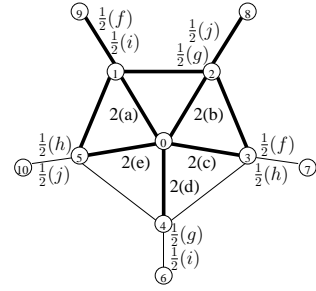
In the particular case where  $b(v) = 1$  for the 10 vertices of the figure we get a lower bound  $W \geq 10 + \frac{2}{5} \cdot 5 = 12$ .

Figure 5.6(c) shows an integer solution for IRWP with  $W_{\min}^i = 13$  and figure 5.6(b) a fractional solution with  $W_{\min} = 12.5$ . In fact 12.5 is the exact value. Indeed each round  $R$  can contain at most 2 edges at level 2 and so the best we can do is to transmit at level 2 a flow of value  $2w(R)$ . The flow contribution to  $W$  from vertices at level 2 is at least  $\frac{5}{2}$  and so  $W \geq 10 + 2.5 = 12.5$ .

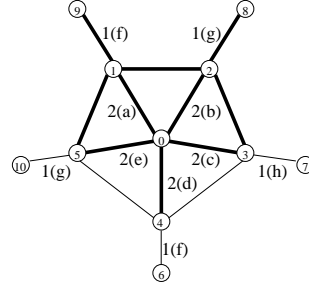
This result is not surprising if we consider the conflict graph. Indeed the subgraph of the conflict graph induced by the edges at level 2 form a cycle of length 5 and a maximal independent set is of size 2. But, we need 3 labels implying in the integer case a lower bound of 3 and so  $W_{\min}^i \geq 13$ . In the fractional case, it is known that we can use a fractional coloring with  $\frac{5}{2}$  labels.



(a) A lower bound of 12 given by a call-clique.



(b)  $W_{min} = 12.5$  with fractional round weights.



(c)  $W_{min}^i = 13$  with integer round weights.

Figure 5.6: Example of lower bound calculation. In this case  $d = 2$ .

For a set of edges  $F$ , let us denote by  $\alpha(F)$  the maximum number of independent edges. It corresponds to the independent (stability) number of the subgraphs of the conflict graph generated by  $F$ .

**Definition 6** *Let  $K$  be a call-clique. An edge  $e \notin K$  is said to be critical for  $K$  if  $K \cup \{e\}$  is a call-clique.*

**Lemma 6** *Let  $K$  be a call-clique and  $F$  a set of edges all critical for  $K$ , then  $W \geq c_w(K) + \frac{c_w(F)}{\alpha(F)}$ .*

**Proof:** As  $K \cup \{e\}$  is a call-clique for any  $e$  in  $F$  a round can contain at most one edge of  $K \cup \{e\}$ . Then

$$W = \sum_{R \in \mathcal{R}} w(R) \geq \sum_{R; R \cap K \neq \emptyset} w(R) + \sum_{R; R \cap F \neq \emptyset} w(R) \quad (5.2)$$

and  $\sum_{R; R \cap K \neq \emptyset} w(R) \geq c_w(K)$ . But, by definition  $R$  contains independent edges, then  $|R \cap F| \leq \alpha(F)$  and  $c_w(F) = \sum_R w(R) |R \cap F| = \sum_{R; |R \cap F| \neq \emptyset} w(R) |R \cap F| \leq \alpha(F) \sum_{R; |R \cap F| \neq \emptyset} w(R)$ . Finally, by (5.2), we have that  $W \geq c_w(K) + \frac{c_w(F)}{\alpha(F)}$ . ■

By taking  $K = K_0$  and  $F$  the set of edges at level  $\lceil \frac{d+1}{2} \rceil$  and noting that any path from a vertex at distance at least  $\lceil \frac{d+1}{2} \rceil$  should use an edge of  $E_{\lceil \frac{d+1}{2} \rceil}$ , we get the following result.

**Corollary 4** *If all the edges of  $E_{\lceil \frac{d+1}{2} \rceil}$  are critical for  $K_0$ , then*

$$W \geq S_0 + \frac{1}{\alpha(E_{\lceil \frac{d+1}{2} \rceil})} \sum_{v; d(v, g) \geq \lceil \frac{d+1}{2} \rceil} b(v)$$

For example, if we apply corollary 4 for the grid with the gateway in the middle and  $d = 2k$ , as all the edges of  $E_{k+1}$  are critical for  $K_0$  and the 4 edges  $((k+1, 0), (k, 0))$ ,  $((0, k+1), (0, k))$ ,  $((0, -k-1), (0, -k))$ ,  $((-k-1, 0), (-k, 0))$  are independent, we have a new proof of Proposition 3.

#### 5.2.4 Relationship with duality

In the following, we show that a set of call-cliques may be associated with a dual solution.

The dual formulation of RWP has been studied in [2]. A dual solution for the RWP for gathering instances can be described with the following property.

**Property 1 ([2])** *The dual problem of round weighting consists of finding a metric  $m : E \rightarrow \mathbb{R}^+$  onto the edge set maximizing the total distance that the traffic needs to travel ( $W = \sum_{v \in V} d_m(g, v) b(v)$ ) and such that the maximum length of a round is 1 ( $(\forall R \in \mathcal{R}) w(R) = \sum_{e \in R} d_m(e) \leq 1$ ).*

Now, we will show that it is possible to construct a feasible dual solution for RWP starting from the call-cliques.

Let  $\mathcal{K}$  a set of call-cliques. First, for each edge  $e$  in a call-clique of  $\mathcal{K}$  we define  $\mathcal{K}_e = \{K \in \mathcal{K} \mid e \text{ is an edge of } K\}$ . Let us define a metric  $m : E \rightarrow \mathbb{R}^+$  is such that  $m(e) = \frac{|\mathcal{K}_e|}{|\mathcal{K}|}$ . Let us now check that  $m$  is a feasible dual solution. To check this, we need to know that for any non-interfering set of edges  $E' \subseteq E$ , the sum  $\sum_{e \in E'} m(e)$  must be less than (or equal to) 1. In fact, as  $E'$  is a set of non-interfering edges, the sets  $\{\mathcal{K}_e\}_{e \in E'}$  are pairwise disjoint. Thus,  $\sum_{e \in E'} m(e) = \sum_{e \in E'} \frac{|\mathcal{K}_e|}{|\mathcal{K}|} \leq 1$ .

## 5.3 Lower bound for grids

### 5.3.1 Gateway in the middle: a lower bound

In the following, we consider the case of uniform demand ( $b(v) = b, \forall v$ ) in a grid of size  $p \times q$  and  $N$  vertices. In this case, the total demand is  $N - 1$ . We derive formulas only in function of the  $d$  that compute a lower bound for grid graphs. In subsection 6.2.2, we prove that these formulas give the optimal solution.

Recall that by definition (see section 5.1) gateway in the “middle” means a gateway far from the borders. In this section, we suppose  $\min(p_1, p_2, q_1, q_2) \geq \lceil \frac{d+1}{2} \rceil$ .

By proposition 1 a lower bound is  $S_0 = \sum_{v \in V_{K_0}} d(v, g)b(v) + \lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v)$ . For  $d$  even, this lower bound can be improved to  $S_0 + \frac{1}{4} \sum_{v; d(v, g) > k} b(v)$  as shown in proposition 3.

In the particular case of uniform demand we can obtain closed formula. In the proofs we will denote by  $N_i$  the number of vertices at distance  $i$  to the gateway. For  $i \leq \min(p_1, p_2, q_1, q_2)$ , in particular for  $i \leq k + 1$  we have  $N_i = 4i$ .

**Proposition 4** *Given a grid  $p \times q$  with  $\min(p_1, p_2, q_1, q_2) \geq \lceil \frac{d+1}{2} \rceil$  and  $N$  vertices with the gateway in the middle. Considering uniform demand and  $d = 2k - 1$  odd, then  $W_{\min} \geq (k(N - 1) - \frac{4}{6}k(k + 1)(k - 1))b$ .*

**Proof:** By proposition 1,  $W_{\min} \geq S_0 = \sum_{v \in V_{K_0}} d(v, g)b(v) + \lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v)$ . Consider  $b(v) = 1$  for all  $v$ , then  $\sum_{v \in V_{K_0}} d(v, g)b(v) = \sum_{v \in V_{K_0}} d(v, g) = \sum_{i \leq k} iN_i$  and  $\lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v) = k((N - 1) - \sum_{i \leq k} N_i)$ . Then we have:

$$\begin{aligned} W_{\min} &\geq \sum_{i \leq k} iN_i + k((N - 1) - \sum_{i \leq k} N_i) \\ &= \sum_{i \leq k} 4i^2 + k(N - 1) - k \sum_{i \leq k} 4i \\ &= k(N - 1) - 4 \left[ k \frac{k(k + 1)}{2} - \frac{k(k + 1)(2k + 1)}{6} \right] \end{aligned}$$

$$= k(N-1) - \frac{4k(k+1)(k-1)}{6}.$$

Making  $b(v) = b$ , then  $W_{\min} \geq (k(N-1) - \frac{4}{6}k(k+1)(k-1))b$ .  $\blacksquare$

**Proposition 5** *Given a grid  $p \times q$  with  $\min(p_1, p_2, q_1, q_2) \geq \lceil \frac{d+1}{2} \rceil$  and  $N$  vertices with the gateway in the middle. Considering uniform demand and  $d = 2k$  even, then  $W_{\min} \geq ((k + \frac{1}{4})(N-1) - \frac{k(k+1)(4k-1)}{6})b$ .*

**Proof:** Consider  $b(v) = 1$  for all  $v$ , by proposition 3,  $W_{\min} \geq S_0 + \frac{1}{4} \sum_{v; d(v,g) > k} b(v) = S_0 + \frac{1}{4}((N-1) - \sum_{i \leq k} N_i)$ . From proposition 4,  $S_0 = k(N-1) - \frac{4}{6}k(k-1)(k+1)$  and

$$W_{\min} \geq (k + \frac{1}{4})(N-1) - \frac{k(k+1)(4k-1)}{6}.$$

Making  $b(v) = b$ , then  $W_{\min} \geq ((k + \frac{1}{4})(N-1) - \frac{k(k+1)(4k-1)}{6})b$ .  $\blacksquare$

In section 6.2.2, we will prove that these formulas give the optimal solution.

### 5.3.2 Gateway in the corner: a lower bound

#### Case $d$ odd

In this section, we study the case when  $d = 2k - 1$  is odd. Notice that, when the gateway is placed at the corner, we can construct call-cliques bigger than  $K_0$ . In fact, the maximum call-clique  $K_{\max}$  containing  $K_0$  is strictly bigger than  $K_0$  for  $d \geq 3$ . In this way, we will use call-cliques bigger than  $K_0$  and we obtain better lower bounds (which will be attained).

We define  $K_{\max}$  as the call-clique composed by the edges delimited by the vertices  $V_{K_0} \cup S_{\text{od}}$  where  $S_{\text{od}} = \{v \mid d(v, g) \leq 2k \text{ and } d(v, v^*) \leq k\}$  and  $v^*$  denotes the node  $(k, k)$ . An example of  $K_{\max}$  for  $d = 9$  ( $k = 5$ ) is depicted in figure 5.7. Another example for  $d = 15$  ( $k = 8$ ) is depicted in Figure 5.8 where the values of the lower bound, given in the next lemma, are also indicated.

**Lemma 7** *For the grid with the gateway at the corner and  $d = 2k - 1$ , then*

$$\text{LB}(v, K_{\max}) \geq \begin{cases} \frac{d+1}{2} & \text{if } v \notin K_0 \cup S_{\text{od}} \\ \min\{d(v, g); 3\frac{d+1}{2} - d(v, g); 2\frac{d+1}{2} - d(v, v^*)\} & \text{if } v \in K_0 \cup S_{\text{od}} \end{cases}$$

**Proof:** If  $v \in V_{K_0}$  any path from  $v$  to  $g$  uses  $d(v, g)$  edges in  $K_0$  (and so in  $K_{\max}$ ). Note that, in that case,  $2k - d(v, v^*) = d(v, g)$  as  $d(v^*, g) = 2k$ . Otherwise, any path has to use  $k$  edges in  $K_0$  giving the lower bound for  $v \notin S_{\text{od}}$ . If  $v \in S_{\text{od}}$  any path from  $v$  to  $g$  will use  $k$  edges in  $K_0$  plus certain edges in  $S_{\text{od}}$ . The number of edges used in  $S_{\text{od}}$  is either  $d(v, g) - k$  needed to attain a vertex of  $K_0$ ; or  $2k - d(v, g)$  to attain the diagonal bordering  $S_{\text{od}}$  composed by the vertices at distance  $2k$  from  $g$  ( $x + y = 2k$ ); or  $k - d(v, v^*)$  to attain the diagonals bordering  $S_{\text{od}}$  below ( $y = x + k$ ) or above ( $x = y + k$ ).  $\blacksquare$

**Theorem 2** *For the grid with the gateway at the corner and  $d = 2k - 1$ ,*

$$W_{\min} \geq \sum_v b(v) \text{LB}(v, K_{\max})$$

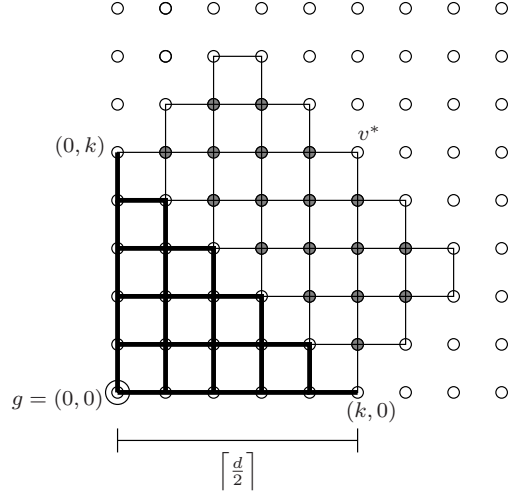


Figure 5.7: Call-clique  $K_{\max}$  for  $d$  odd with  $g$  at the corner. In this scheme,  $d = 9$ . The call-clique  $K_0$  consists in all the bold edges.

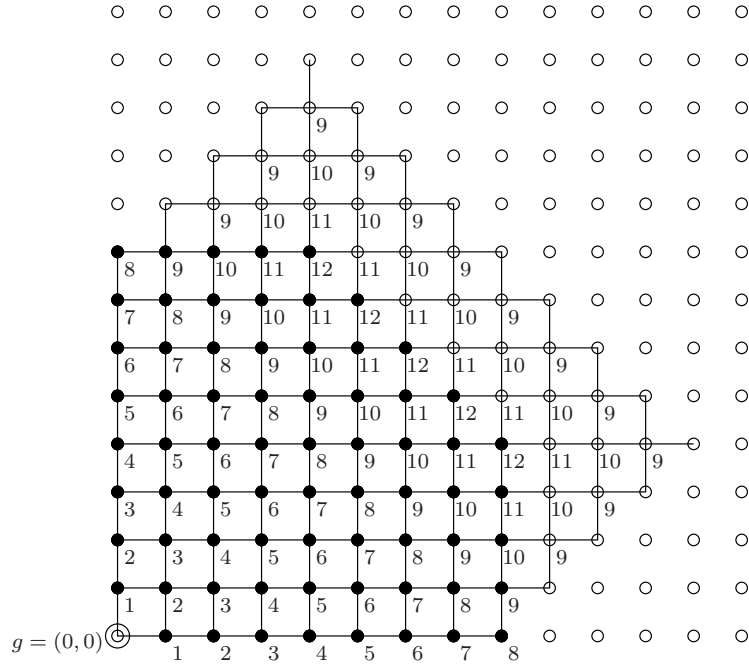


Figure 5.8: Lower bound per node in uniform demand case. The black nodes indicate the nodes whose lower bound correspond to their distance to the gateway. In this scheme,  $d = 15$ .

**Proof:**  $W \geq c_w(K_{\max}) \geq \phi(K_{\max}) \geq \sum_v \phi_v(K_{\max}) \geq \sum_v b(v) \text{LB}(v, K_{\max})$  ■

Using Theorem 2 we can derive an explicit formula for the lower bound when the demand is uniform and when the grid is far enough to contain the vertices of  $K_{\max}$ . That is  $\min(p, q) = \frac{3k}{2}$ .

**Proposition 6** *For the grid with  $N = pq$  nodes ( $\min(p, q) \geq k + \lceil \frac{k}{2} \rceil$ ) with the gateway at the corner and  $d = 2k - 1$ , if  $b(v) = 1$  for all  $v$ , then*

$$W \geq \frac{d}{2}(N - 1) + f(d)$$

where  $f(d) = \frac{\lambda}{12}(\lambda - 1)(\lambda - 5)$  if  $d = 4\lambda - 1$ ; and  $f(d) = \frac{\lambda - 1}{12}\lambda(\lambda - 5)$  if  $d = 4\lambda + 1$ .

**Proof:**

We have to count  $\sum_v \text{LB}(v, K_{\max})$ . For all the vertices not in  $V_{K_0} \cup S_{\text{od}}$ ,  $\text{LB}(v, K_{\max}) = k$  (Recall that  $S_{\text{od}}$  is defined as  $\{v \mid d(v, g) \leq 2k - 1 \text{ and } d(v, v^*) \leq k - 1\}$ ). For the vertices in  $V_{K_0}$ ,  $\text{LB}(v, K_{\max}) = d(v, g) \leq k$  and for  $v \in \text{Int}(S_{\text{od}})$ ,  $\text{LB}(v, K_{\max}) \geq k$ . In  $K_0$  we have  $i + 1$  vertices at distance  $i$  from  $g$  giving a difference compared to  $k$  of  $k - i$ ; so for the vertices of  $K_0$  we have a total loss of  $A_k = \sum_{i=1}^{k-1} (i + 1)(k - i) = \frac{(k-1)k(k+4)}{6}$ . The vertices  $(x, y)$  in  $S_{\text{od}}$  give an excess for those at distance  $i > 0$  from one of the 4 diagonals delimiting  $S_{\text{od}}$  namely  $x + y = k$ ;  $x + y = 2k$ ;  $x = y + k$ ;  $y = x + k$ . We distinguish two cases depending on the parity of  $k$ . For the case even  $k = 2\lambda$ , the number of vertices in  $S_{\text{od}}$  with an excess of  $i$  (that is a value  $k + i$ ) is  $3k - 4i$  for  $1 \leq i \leq \lambda - 1$ , and  $\lambda + 1$  for  $i = \lambda$ . For the case odd  $k = 2\lambda + 1$ , they are in number  $3k - 4i$  for  $1 \leq i \leq \lambda$ .

All together they give an excess  $B_k$ . For the case  $k = 2\lambda$ ,  $B_k = \sum_{i=1}^{\lambda-1} i(3k - 4i) + \lambda(\lambda + 1) = \frac{k}{6}(5\lambda^2 + 1)$ . For the case  $k = 2\lambda + 1$ ,  $B_k = \sum_{i=1}^{\lambda} i(3k - 4i) = \frac{k}{6}(5\lambda(\lambda + 1))$ .

Finally, we get  $B_k - A_k$  in order to obtain the total excess. For the case  $k = 2\lambda$ ,  $B_k - A_k = \frac{k}{6}(\lambda - 1)(\lambda - 5)$ . For the case  $k = 2\lambda + 1$ ,  $B_k - A_k = \frac{k}{6}\lambda(\lambda - 5)$ . ■

**Case  $d$  even.**

As we have seen in the example of figure 5.2, we have to consider in that case two cliques  $K_1$  and  $K_2$ . These two cliques contain a clique  $K_{\max}$  consisting of the vertices of  $V_{K_0} \cup S_{\text{ev}}$  where  $S_{\text{ev}} = \{v \in V \mid d(v, g) \leq 2k + 1 \text{ and } d(v, v^*) \leq k\}$  and  $v^* = (k, k)$ . Furthermore,  $K_1$  contains the  $\lfloor \frac{k}{2} \rfloor + 1$  vertices  $v = (x, y)$  such that  $x + y \leq 2k + 1$  and  $x = y + k + 1$ . In the same way,  $K_2$  contains the  $\lfloor \frac{k}{2} \rfloor + 1$  vertices  $v$  such that  $x + y \leq 2k + 1$  and  $y = x + k + 1$ .

**Theorem 3** *For the grid with the gateway at the corner and  $d = 2k$ ,*

$$W_{\min} \geq \sum_{v \in K_0 \cup S_{\text{ev}}} \min[d(v, g), \frac{3(d+1)}{2} - d(v, g), d+1 - d(v, v^*)]b(v) + \frac{d+1}{2} \sum_{v \notin V_{K_0} \cup S_{\text{ev}}} b(v)$$



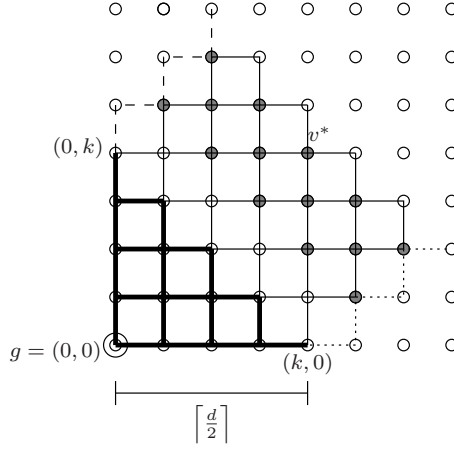


Figure 5.9: Two overlapped cliques for  $d$  even with  $g$  at the corner. In this scheme,  $d = 8$ . Edges in dash belongs to  $K_1$  and in dots belongs to  $K_2$ .

**Proof:** We use lemma 5 with the two cliques  $K_1$  and  $K_2$ . If  $v \in V_{K_0}$  any path from  $v$  to  $g$  uses  $d(v, g)$  edges in  $K_{\max}$ . If  $v \notin V_{K_0} \cup S_{\text{ev}}$ , any path from  $v$  to  $g$  use  $k$  edges in  $K_{\max}$  and at least one in  $K_1$  or one in  $K_2$  giving the lower bound  $k + 1/2 = \frac{d+1}{2}$ .

If  $v \in S_{\text{ev}}$ , we distinguish 3 cases depending on the number of edges needed to attain the border of  $S_{\text{ev}}$ :

- $d(v, g) - k$  to attain a vertex  $x + y = k$ , and so  $d(v, g)$  edges in  $K_{\max}$
- $2k + 1 - d(v, g)$  to attain the diagonal composed by the vertices at distance  $2k + 1$  (i.e,  $x + y = 2k + 1$ ) but then we need at least  $k$  edges in  $K_{\max}$  and one in  $K_1$  or  $K_2$  so altogether  $3k + 1 + 1/2 - d(v, g) = 3(\frac{d+1}{2}) - d(v, g)$ .
- $k - d(v, v^*)$  to attain the diagonal below (i.e,  $y = x + k$ ) or above ( $x = y + k$ ). Then we use  $k$  edges in  $K_{\max}$  and either 2 in  $K_1$  (vertices above) or 2 in  $K_2$  (vertices below) so altogether  $2k - d(v, v^*) + 1/2 \cdot 2 = d + 1 - d(v, v^*)$ .

■

Note that the formula is identical to that of the case  $d$  odd. When the demand is uniform and the grid large enough to contain the vertices in  $K_1$  and  $K_2$  that is  $\min(p, q) \geq \frac{3k}{2} + 1$ , computation analog to that of proposition 6 gives the following result.

**Proposition 7** For the grid with  $N = pq$  nodes ( $\min(p, q) \geq k + \lceil \frac{k}{2} \rceil$ ) with the gateway at the corner and  $d = 2k$ , if  $b(v) = 1$  for all  $v$ , then

$$W \geq \frac{d+1}{2}(N-1) + f(d),$$

where  $f(d) = \frac{\lambda}{12}(4\lambda^2 - 2(\lambda - 1))$  if  $d = 4\lambda$ ; and  $f(d) = -\frac{1}{2} + \frac{\lambda}{12}(\lambda + 1)(4\lambda - 19)$  if  $d = 4\lambda + 2$ .

## 5.4 Conclusion

In this chapter, we presented formally methods to obtain lower bounds for general graphs. Our methods are applied to grid graphs (using distance- $d$  model) providing closed formulae (as proved in chapter 6) for the case considering uniform demand. It was considered the gateway placed either in the middle or in the corner.



## Chapter 6

# General upper bound methods and application to grid graphs

This chapter presents several routing methods for grids reaching optimal solution equals to our lower bound. It was considered different cases (changing gateway position and demand characteristics) considering fractional and integer solutions.

### 6.1 Upper bounds: general results

To find upper bounds we will propose routing strategies giving a small total weight  $W$ . For that, to each vertex  $v$ , we will associate  $\pi(v)$  paths from  $v$  to  $g$  carrying the demand  $b(v)$ . More precisely, each path  $P_i(v, g)$  will carry some flow  $\phi_v^i$  and  $\sum_i \phi_v^i = b(v)$ .

Furthermore, we will assign to the paths labels (or colors)  $c_j$ . Each label  $c_j$  corresponds to a round  $R_j$  and so we have to insure that the edges with the same label do not interfere. Therefore we introduce the notion of interference free  $\gamma$ -labeled *paths* (cycles).

#### 6.1.1 Interference free $\gamma$ -labeled *paths* (cycles)

**Definition 7 (Interference free  $\gamma$ -labeled *paths*)** *A set of paths (or cycles) are said to be interference free  $\gamma$ -labeled if we can assign to the edges  $\gamma$  labels such that two edges with the same label do not interfere.*

In order to obey the inequalities in (5.1),  $c_w(e) \geq \phi(e)$ , we will give to each round  $R_j$  a weight  $w(R_j)$  equal to the maximum of the flow on each arc with label  $c_j$ . With this strategy we get the following proposition.

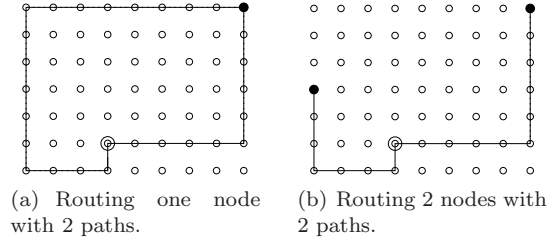


Figure 6.1: Main routing strategies.

**Proposition 8** *Let  $G = (V, E)$  and  $V' = \{v_1, \dots, v_\pi\}$  a family of nodes (not necessarily distinct). If there exist  $\pi$  pairwise interference free  $\gamma$ -labeled paths from  $V'$  to  $g$ , then we can satisfy a demand of  $\pi$  with a total weight  $\gamma$ .*

**Proof:** We send a flow of 1 in each path. After that, each edge labeled with one of the  $\gamma$  labels  $c_j$  is associated with a round  $R_j$  with weight 1. The set of edges used by  $R_j$  are non-interfering, as the paths are interference free  $\gamma$ -labeled. Furthermore, the inequalities in (5.1) are respected. ■

We will use Proposition 8 mainly in two cases: all  $v_i$  distinct and all  $v_i$  equal to the same vertex  $v$ . In the latter case proposition 8 gives the following.

**Corollary 5** *If there exists  $\pi$  pairwise interference free  $\gamma$ -labeled paths from  $v$  to  $g$ , then we can route the demand  $b(v)$  with rounds having a total weight  $W = \frac{\gamma}{\pi} b(v)$ .*

**Proof:** By proposition 8, with all  $v_i = v$ , we can route a flow of  $\pi$  in  $\gamma$  rounds of weight 1 and so a flow of  $b(v)$  in  $\gamma$  rounds of weight  $\frac{b(v)}{\pi}$  each. ■

We will use two main *routing strategies*. Either we route the total demand  $b(v)$  of a vertex  $v$  by finding interference free paths from  $v$  to  $g$  and applying corollary 5; or we combine paths issued from  $v$  with paths issued from other nodes. We might have to do different combinations to be able to route all the demands (see Figure 6.1).

### 6.1.2 Distance- $d$ model of interference and the *Width*

We will present results concerning the distance- $d$  model of interference. In some applications, we need to route the demand from  $v$  via a single path. If we use a shortest path and we give to each edge a different label, we obtain:

**Proposition 9** *We can route the demand  $b(v)$  of a node  $v$  using a single path with a weight  $W \leq b(v)d(v, g)$ .*

The particular case of a node  $v \in K_0$  has  $W \geq b(v)d(v, g)$  by corollary 2, then we obtain:

**Corollary 6** *In the distance- $d$  model of interference, the demand  $b(v)$  of a node  $v$  of  $K_0$  can be satisfied with  $W_{min} = b(v)d(v, g)$  with a single shortest path.*

If  $v \notin K_0$  the lower bound is  $W_{min} \geq \lceil \frac{d}{2} \rceil b(v)$  and so cannot be attained using a single path. In fact, in the case of a single path considering the distance- $d$  model, if the path is of length  $\geq d+1$ , we need at least  $d+1$  labels as  $d+1$  consecutive edges always interfere. If we want to have an interference free path with  $d+1$  labels, the only way is to repeat a sequence of  $d+1$  different labels in order that  $d+1$  consecutive edges have different labels. This construction does not always work (see later an example in figure 6.2 in which the path in a grid turns back at distance shorter than  $d$  making a “short  $U$ ”).

In that case, there are two edges far away (that are at distance  $\geq d$  on the path), but at distance  $< d$  in the graph. Thus, the path can not be interference-free  $d+1$ -labeled if these two edges receive the same label. Therefore, we introduce the following definition:

**Definition 8 (Width  $d$ )** *A path (or a cycle) has width  $d$ , if two edges at distance  $\geq d$  in the path (or cycle) are also at distance  $\geq d$  in the graph.*

**Proposition 10** *In the distance- $d$  model of interference, a path of width  $d$  can be interference free  $(d+1)$ -labeled.*

**Proof:** We can label the edges of the path by repeating a chain of  $d+1$  labels. If two edges have the same label, then they are necessarily at distance  $\geq d$  in the path and, by definition of the width, they are also at distance  $\geq d$  in the graph and so do not interfere. ■

**Proposition 11** *In the distance- $d$  model of interference, a shortest path can be interference free  $(d+1)$ -labeled.*

**Proof:** By proposition 10, it suffices to prove that a shortest path has width  $d$ . But two edges at distance  $\geq d$  in the path are also at distance  $\geq d$  in  $G$ , otherwise we will have a shortcut creating a shortest path, that is a contradiction. ■

**Corollary 7** *Considering the distance- $d$  model of interference and a general graph, we can route the demand  $b(v)$  using a shortest path with weight  $W \leq (d+1)b(v)$ .*

Consequently, if we route the demand of each node with a shortest path we obtain the following approximation.

**Theorem 4** *In the distance- $d$  model with  $d > 1$ , there exists a  $\frac{d+1}{\lceil \frac{d+1}{2} \rceil}$ -approximation for the RWP problem.*

**Proof:** We have, by proposition 1, a lower bound of  $\frac{d+1}{\lceil \frac{d+1}{2} \rceil} b(v)$  and by corollary 7 an upper bound of  $d+1$ . ■

Note that, it gives for  $d$  odd a 2-approximation and for  $d$  even an  $\frac{\alpha}{d}$ -approximation with  $\frac{\alpha}{d} = \frac{2}{d} + 2$  and so is the worst case ( $d = 2$ ) a 3-approximation.

We can also use proposition 11 to design 2 interference free  $d+1$ -labeled paths in the following case.

**Corollary 8** *If  $d(v_1, v_2) = d(v_1, g) + d(g, v_2)$  then we can send a flow of 1 from  $v_1$  and a flow of 1 from  $v_2$  with  $d + 1$  rounds.*

**Proof:** The path formed by the union of a shortest path from  $v_1$  to  $g$  and the shortest path from  $v_2$  to  $g$  is a shortest path between  $v_1$  and  $v_2$ , then it can be  $d + 1$ -labeled by proposition 11. ■

Cycles play an important rule and illustrate the *routing strategies* in figure 6.1. Indeed, a cycle containing  $g$  induces for any vertex  $v$  two paths from  $v$  to  $g$  and, for any pair of vertices  $v_1$  and  $v_2$  two paths, one from  $v_1$  to  $g$  and another from  $v_2$  to  $g$ .

**Proposition 12** *In the distance- $d$  model of interference, a cycle of width  $d$  can be interference free  $(d + 1)$ -labeled if and only if its length is a multiple of  $d + 1$ .*

**Proof:** Let us start from some edge  $e_1$  of the cycle and label the path with repetitions of the chain  $C = c_1 \dots c_{d+1}$  of labels. If the length is a multiple of  $d + 1$ , then the edges labeled  $c_i$  are at a distance multiple of  $d$  on the cycle, and so by definition of the width at distance  $\geq d$  in the graph. If the length is not a multiple of  $d + 1$  then the last edge of the path labeled  $c_1$  is at distance  $< d$  of the first edge  $e_1$  also labeled  $c_1$ , therefore these edges interfere. ■

From proposition 12, we obtain:

**Corollary 9** *In the distance- $d$  model of interference, if there exists a cycle containing  $v$  and  $g$  of width  $d$  and multiple of  $d + 1$  then the demand  $b(v)$  of a node  $v$  can be satisfied with a weight  $W \leq \frac{d+1}{2}b(v)$ .*

**Proof:** We have two interference free  $(d + 1)$ -labeled paths from  $v$  to  $g$ . We can route then half of the demand in each path obtaining, by corollary 5,  $W_{min} \leq \frac{\gamma}{\pi}b(v) = \frac{d+1}{2}b(v)$ . ■

Our objective is to find upper bounds that consists in interference free paths with a number of labels corresponding to the lower bounds. If  $d$  is odd, we have a lower bound equal to  $\frac{d+1}{2}$  and so by corollary 9 we obtain:

**Theorem 5** *In the distance- $d$  model of interference with  $d$  odd, if there exists a cycle containing  $v$  and  $g$  (two paths from  $v$  to  $g$ ) of width  $d$  and multiple of  $d + 1$  then the demand  $b(v)$  can be satisfied with a weight  $W_{min} = \frac{d+1}{2}b(v)$ .*

We can also use two paths interference free  $(d + 1)$ -labeled issued from two different vertices, as shown in the next theorem.

**Theorem 6** *Let  $G$  be a 2-connected graph and let  $d = 1$ . If  $\sum_{v \notin K_0} b(v)$  is even and,  $b(v) \leq \frac{1}{2} \sum_{v \notin K_0} b(v)$ ,  $\forall v \notin K_0$ ,  $W_{min} = \sum_{v \in V_{K_0}} d(v, g)b(v) + \frac{d+1}{2} \sum_{v \notin V_{K_0}} b(v)$  is solution for IRWP.*

**Proof:** For vertices in  $K_0$ , we use corollary 6. For the other vertices, we can always route together a remaining demand of two nodes as  $b(v) \leq \frac{1}{2} \sum_{v \notin K_0} b(v)$ . For that, we need a pair of disjoint paths (so interference free as  $d = 1$ ) from  $u$  to  $g$  and from  $v$  to  $g$ , that exist as  $G$  is 2-connected. ■

As we will see later, in some cases we will also need more complicated routing (like 4 paths or 2 cycles). In the section 6.2 we give solutions for the case of grids as example of application of the presented methodology. The following definition will be useful to find interference free paths.

**Definition 9 (Path distance  $d(P, Q)$ )** *The distance  $d(P, Q)$  between two paths  $P$  and  $Q$  is the minimum of the distance between any edge of  $P$  and any edge of  $Q$ ,  $d(P, Q) > \min_{e_1 \in P, e_2 \in Q} d(e_1, e_2)$ .*

**Proposition 13** *In the distance- $d$  model of interference, two paths  $P$  and  $Q$  at distance  $\geq d$  do not interfere.*

## 6.2 Upper bounds for grids

In a grid the paths or cycles have a specific structure. Indeed they are formed by a succession of horizontal and vertical subpaths. To describe such a path or cycle, we will only give the vertices where there is a change of direction. So between two vertices  $(x, y_0) — (x', y_0)$ , we have an *horizontal* path consisting of all the vertices  $(u, y_0)$  with  $x \leq u \leq x'$  if  $x < x'$ , or  $x' \leq u \leq x$  if  $x > x'$ . Similarly between two vertices  $(x_0, y)(x_0, y')$  we have a *vertical* path  $(x_0, y) — (x_0, y')$ . We introduce the following definitions that will be necessary to describe our methods of routing in the next sections.

**Definition 10 (Monotonic path)** *We will say that a path is monotonic (has a “stair” shape), if the first and second coordinates of the vertices where there is a change of direction are ordered in a monotonic way.*

We have 2 types of monotonic paths according to  $x_i$  and  $y_i$  vary in the same way or not. For example, a monotonic path  $P = (x_0, y_0) — (x_1, y_0) — (x_1, y_1) — (x_1, y_2) — (x_2, y_2) — (x_2, y_3) \dots — (x_m, y_n) — (x_m, y_{n+1})$  is a monotonic of *negative type*  $-+$  (or  $-+$ ), if the  $x_i$  are increasing  $x_0 \leq x_1 < x_2 \dots < x_m$  and the  $y_i$  are decreasing  $y_1 > y_2 \dots > y_n \geq y_{n+1}$  (as the path is undirected by considering the vertices in the opposite order we have decreasing  $x$  and increasing  $y$ ). See figure 6.3(b) for an example of this case. When the vertices have both increasing (resp. decreasing)  $x$  and  $y$ , the path is said to be monotonic of *positive type*  $++$  (resp.  $--$ ).

**Proposition 14** *Let  $G$  be a monotonic path in a 2-dimensional grid. It can be interference free  $(d + 1)$ -labeled.*

**Proof:** The  $x$  and  $y$  in this path are monotonic, then it has *width*  $d$  as the distance in the path is exactly that in the graph. Thus, proposition 10 says it can be interference free  $(d + 1)$ -labeled. ■

The figure 6.2 gives an example with a unique non monotonic path showing that the path can make interference with itself (“short U”).



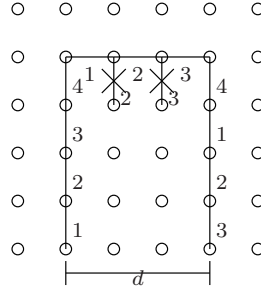


Figure 6.2: To turn back, a path needs to maintain a width  $\geq d$  to be interference free  $(d+1)$ -labeled. In this example,  $d = 3$ .

**Definition 11 (Diagonal of an edge)** *The positive (resp. negative) diagonal of an edge  $e$ , denoted  $S_e^+$  (resp.  $S_e^-$ ) consists of the edges of a monotonic positive (resp. negative) path where all the subpaths are of length 1 (stairs of step 1). Figure 6.3(a) shows the negative diagonal associated with the edges labeled 2 and Figure 6.3(b), positive diagonal associated with the edges labeled 4.*

Now we define relations between monotonic paths.

**Definition 12 ( $d$ -Parallel paths)** *Two monotonic paths  $P$  and  $Q$  are said  $d$ -parallel, if they are of the same type negative (respectively positive) and, if  $e' \in Q$  and  $e \in S_e^+$  (respectively  $S_e^-$ ) with  $e \in P$ , then  $d(e, e') \geq d$ .*

See figure 6.3(b) for an example with two parallel negative paths.

**Property 1** *Given two  $d$ -parallel paths  $P$  and  $Q$  in a 2-dimensional grid  $G$ , they can be interference free  $(d+1)$ -labeled.*

**Proof:** We start labeling the path  $P$  with  $d+1$  labels. Each edge of  $Q$ , that is in a *diagonal set*  $S_e$  of an edge  $e$  in  $P$ , receives the same label of  $e$ . If there exist edges in  $Q$  that are not in a *diagonal set* of  $P$ , they receive the continuation of the sequence of labels derived from the edges in *diagonal sets* of  $P$ . There is no interference between the edges with the same label as, by definition of  $d$ -parallel paths, two edges in the same diagonal are at distance  $\geq d$ . ■

In figure 6.3, we illustrate the property 1 with two pair of parallel paths. In particular two horizontal (or vertical) paths  $P$  and  $Q$  at distance  $d(P, Q) \geq \lceil \frac{d+1}{2} \rceil$  are  $d$ -parallel. Indeed in that case the distance between two edges of  $P$  and  $Q$  in the same diagonal is  $2d(P, Q) - 1 \geq d$  (see figure 6.3(a) for  $d = 3$ ). Similarly if two general monotonic paths have their horizontal and vertical subpaths at distance  $\geq \lceil \frac{d+1}{2} \rceil$  the distance between two edges at the same diagonal is  $\geq d$ . It is the case when one path is obtained from the other by translation of vector  $(\lceil \frac{d+1}{2} \rceil, \lceil \frac{d+1}{2} \rceil)$ , see figure 6.3(b).

Paths uniquely horizontal or vertical can be considered of any type, either  $++$  or  $+-$  (see example in figure 6.3(a)).

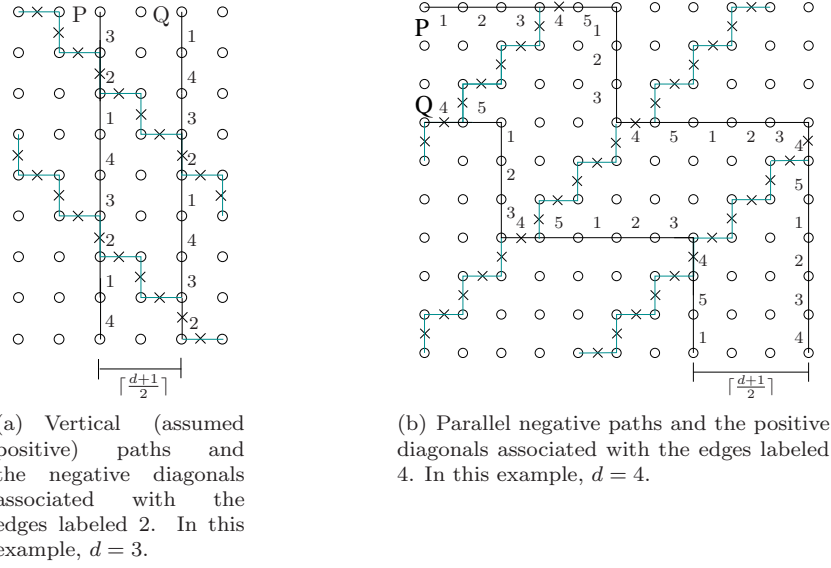


Figure 6.3: Interference free  $(d + 1)$ -labeled paths.

### 6.2.1 Gateway in the middle: routing the demand of a single node

We consider here the first strategy where a demand of  $b(v)$  in a single node  $v$  is routed, we call *single routing*.

**Definition 13 (Regions of the grid)** We split the grid in 4 regions:  $R_A, R_B, R_C$  and  $R_D$ , as shown in Figure 6.2.1.

Note that we could chose different splittings. The results will be valid as soon as the regions are obtained by rotation of  $\frac{\pi}{2}$  of the first one. For the presentation we choose as first region  $R_A$  containing the vertices  $(x, y)$  with  $x \geq 0$ ,  $y \geq 1$  and  $x + y \geq \lceil \frac{d+1}{2} \rceil$  to exclude the vertices of  $K_0$ . Indeed for the vertices of  $K_0$ , as we have seen in corollary 6, we can route their demand  $b(v)$  in  $b(v)d(v, g)$  rounds by using the shortest path with  $d(v, g)$  different labels.

#### Case $d$ odd

**Theorem 7** Let  $G$  be a 2-dimensional grid with  $\min(p_1, p_2, q_1, q_2) \geq d$  and gateway  $g$  in the middle, and let  $d$  be odd ( $d = 2k - 1$ ). Considering the demand  $b(v)$  of a single node  $v$ , we have  $W_{\min} = d(v, g)b(v)$  if  $v \in K_0$  using one shortest path from  $v$  to  $g$ . If  $v \notin K_0$ , there exist 2 paths from  $v$  to  $g$  that can be  $(d + 1)$ -labeled, therefore  $W_{\min} = kb(v)$  for the single routing of  $v$ .

**Proof:** If  $v \in K_0$ , by corollary 6,  $W_{\min} = b(v)d(v, g)$ . If  $v \notin K_0$ , the lower bound is  $S_0 = kb(v)$ . To prove the theorem we will construct for any  $x$  a

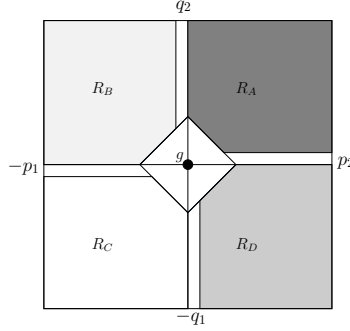


Figure 6.4: The grid regions and  $K_0$ .

generic cycle containing all the vertices of the column  $x, y \geq 0$  and satisfying the hypothesis of the theorem 5 (length multiple of  $d + 1$  and width  $\geq d$ ). So  $W_{min} \leq kb(v)$  for all nodes  $(x, y)$ . The cycle consists of the following subdipaths (we indicate the vertices where there is a change of direction).

$$(0, 0) \rightarrow (x, 0) \rightarrow (x, q_2) \rightarrow (-d, q_2) \rightarrow (-d, -\alpha) \rightarrow (0, -\alpha) \rightarrow (0, 0).$$

See Figure 6.5 for an example with  $d = 3$  ( $k = 2$ ),  $x = 5$  and  $q_2 = 5$ . The length of the cycle is  $2(x + d) + 2(q_2 + \alpha)$ . We chose  $\alpha$  as the smallest possible integer  $0 \leq \alpha < k$ , such that  $2(x + d) + 2(q_2 + \alpha) \equiv 0 \pmod{2k}$ . So the length of the cycle is a multiple of  $d + 1$ . In the example in Figure 6.5, the length of the cycle is  $26 + 2\alpha$  so we chose  $\alpha = 1$  (length  $28 \equiv 0 \pmod{4}$ ). As  $q_2 \geq d$  the horizontal paths are at distance  $\geq d$  and as we chose the vertical line at  $-d$  (choice possible, as we have  $p_1 \geq d$ ), the vertical paths are also at distance  $\geq d$ . ■

Using for each node  $v$  a routing as described above, we have by theorem 7 the following.

**Theorem 8** *Let  $G$  be a 2-dimensional grid with  $\min(p_1, p_2, q_1, q_2) \geq d$  and gateway in the middle, and let  $d$  be odd. We have  $W_{min} = S_0$ .*

#### Case $d$ even

When the distance  $d$  is even the situation is more complex if we want to route the demand of a single node. For vertices in  $K_0$  we have a lower bound of  $d(v, g)$  attained by using a shortest path. But for  $v \notin K_0$  we have by proposition 3 a lower bound of  $k + \frac{1}{4}$ . Based on the proof of the proposition 3 and more precisely on corollary 5, the only way to reach the bound is to find 4 paths from  $v$  to  $g$  crossing  $E_{k+1}$  using the 4 edges which do not interfere  $(0, k + 1)(0, k)$ ,  $(k + 1, 0)(k, 0)$ ,  $(0, -(k + 1))(0, -k)$  and  $(-(k + 1), 0)(-k, 0)$ ; then we need to use the shortest path from  $(0, k)$  and its rotated  $(-k, 0), (0, -k), (k, 0)$  to  $g$ .

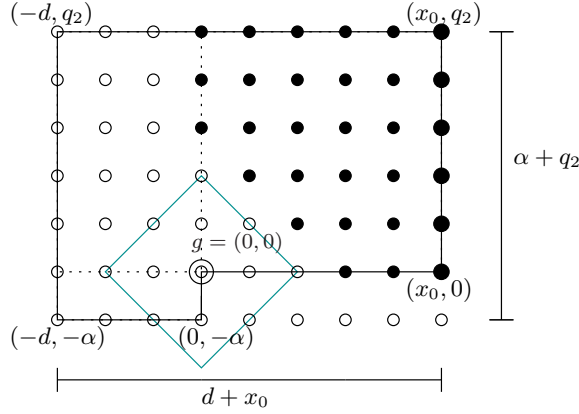


Figure 6.5: Routing method for a node  $v$  with a cycle and  $d$  odd ( $d=3$ ).

Let us call a *cross* this set of 4 edges plus the shortest paths. More generally, a cross centered in  $v = (x, y)$  will consist of the  $4k+4$  edges  $(x, y+i)(x, y+i+1)$ ,  $(x-i, y)(x-i-1, y)$ ,  $(x, y-i)(x, y-i-1)$ ,  $(x+i, y)(x+i+1, y)$  for  $0 \leq i \leq k$ .

The situation is the same around  $v$  where the 4 paths leaving  $v$  should use the edges of the cross centered in  $v$ , using  $4k+1$  labels, with the same label being given for the 4 edges  $(x+k, y)(x+(k+1), y)$ ,  $(x-k, y)(x-(k+1), y)$ ,  $(x, y+k)(x, y+(k+1))$  and  $(x, y-k)(x, y-(k+1))$ . If  $v$  is too close from  $g$ , it is not possible to find two compatible labeling for the two crosses (see Figure 6.7). That happens if  $d(v, g) \leq d+1$  and  $v = (x, y)$  with  $x \neq 0$ ,  $y \neq 0$ . When  $x = 0$  (or  $y = 0$ ) the clique regions may overlap, but, as we will see later, we can reuse the labels and all together use  $4k+1$  labels (see example of Figure 6.10). Figure 6.6 shows the zones considering an even  $d$ .

- $Z_B$ : Zone composed by all nodes at distance  $\leq d+1$  and  $\geq k+1$  (except these in the axes).
- $Z_C$ : Zone composed by all nodes that are at distance at most  $k$  of the vertical and horizontal borders.
- $Z_D$ : Zone composed by all nodes that are at distance at most  $k$  of the border not including the nodes of  $Z_C$ .
- $Z_A$ : All other positions that are not considered in  $Z_B$ ,  $Z_C$  and  $Z_D$ . These nodes have  $W_{min} \geq k + \frac{1}{4}$ . We will prove that the nodes in this region can reach the gateway with four  $(4k+1)$ -labeled paths with  $\frac{1}{4}$  of weight each one. There exist special zones in  $Z_A$ , they are  $Z'_A$  ( $y > 2k$  and  $1 \leq x \leq k+1$  in  $Z_A$ ) and  $Z''_A$  ( $x > 2k$  and  $1 \leq y \leq k+1$  in  $Z_A$ ) that have a more complicated routing.

For nodes in  $Z_B$ ,  $Z_C$ ,  $Z_D$  the lower bound is  $> k + \frac{1}{4}$  (see Figure 6.7) but it is not easy to be precise. The case of  $v$  in one corner can be done with a cycle

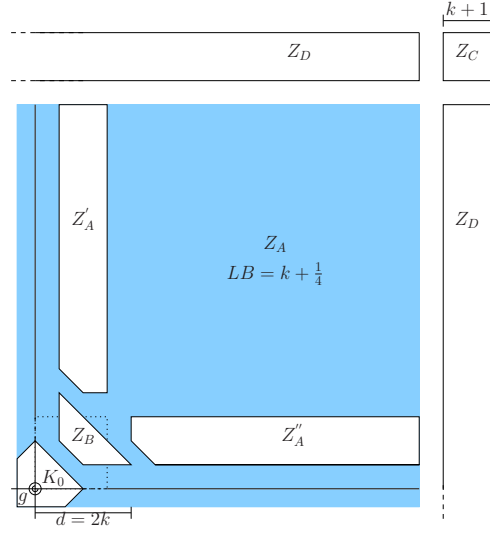


Figure 6.6: Lower bound obtained for each single routing ( $b(v) = 1$ ) from the depicted grid parts, considering  $d$  even ( $k = \frac{d}{2}$ ).

as shown in theorem 7 and has lower bound  $k + \frac{1}{2}$ . Note that, zone  $Z_A$  covers the majority of nodes for grids with large  $p_1, p_2, q_1$  and  $q_2$ . For the single routing of a node in  $Z_A$ , theorem 9 shows that  $W_{min} = k + \frac{1}{4}$ . By similar methods, we can give an upper bound of  $k + \frac{1}{3}$  for nodes in  $Z_D$  and  $k + \frac{1}{2}$  for the other nodes, so obtaining a  $1 + \frac{1}{4k+1}$ -approximation.

**Theorem 9** *Let  $G$  be a 2-dimensional grid with gateway  $g$  in the middle and  $v \in Z_A$ , and let  $d$  be even. There exist 4 paths from  $v$  to  $g$  that can be  $(4k+1)$ -labeled. Therefore,  $W_{min} = \frac{2d+1}{4}b(v)$  for the single routing of  $v$ .*

**Proof:** We distinguish three cases according to the node position:

- case 1: node  $v \in Z_A \setminus \{Z'_A \cup Z''_A\}$ . We consider the following paths:

$$P_1: (-k, 0) \rightarrow (0, 0) \rightarrow (0, -k);$$

$$P_2: (0, y) \rightarrow (x, y) \rightarrow (x, 0);$$

$$P_3: (-(k+1), y + (k+1)) \rightarrow (x + (k+1), y + (k+1)) \rightarrow (x + (k+1), -(k+1)).$$

To label these paths, we follow the steps below:

1. We first label  $P_2 = (0, y) \rightarrow (x, y) \rightarrow (x, 0)$  with the sequence  $Ce'$  (see scheme in Figure 6.8 and example in Figure 6.9).

Furthermore (to respect the labeling of the cross in  $(x, y)$ ), we assign to the edges  $(x, y+k)(x, y+k+1)$  and  $(x+k, y)(x+k+1, y)$  the same label as that of the edge  $(x-(k+1), y)(x-k, y)$  (label identical to that of  $(x, y+k)(x, y+k+1)$ ).

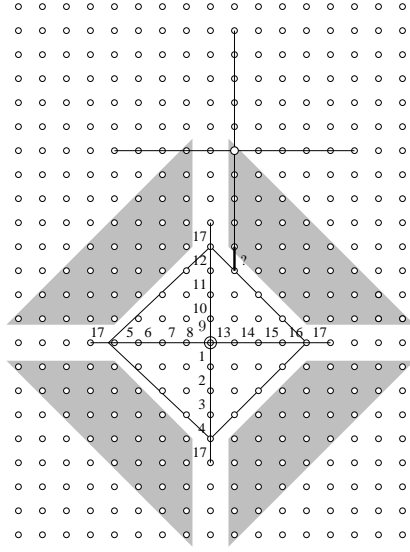


Figure 6.7: Routing the demand of the node  $(1,8)$  in  $Z_B$ , it is impossible to assign a label to the bold edge without use a new label. In this example,  $d = 8$ .

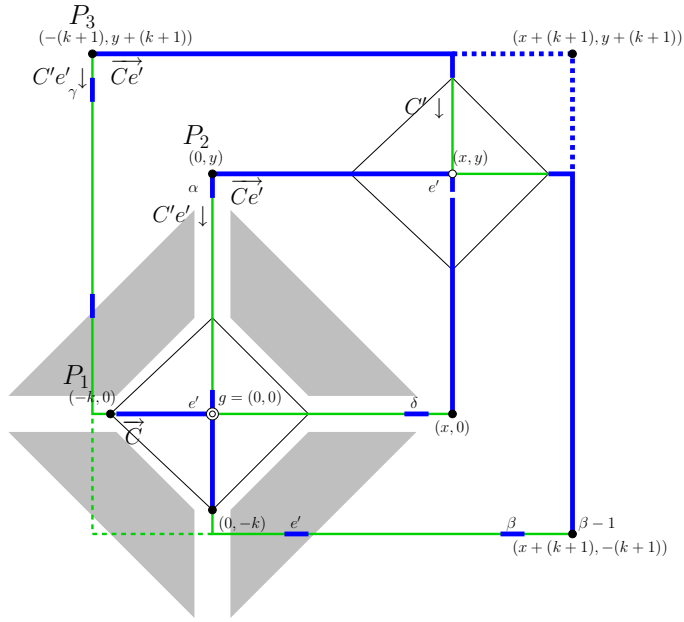


Figure 6.8: labeling paths for the nodes in  $Z_A$ .

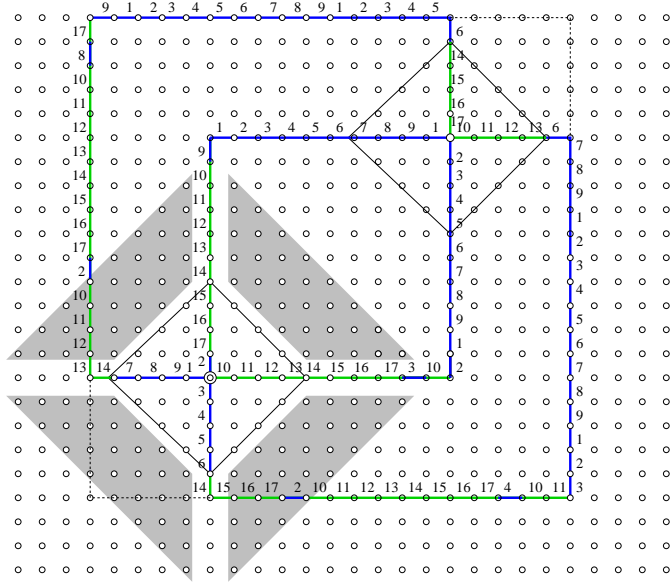


Figure 6.9: Routing the demand of the node  $(10,10)$  in  $Z_A$ . In this example,  $d = 8$ .

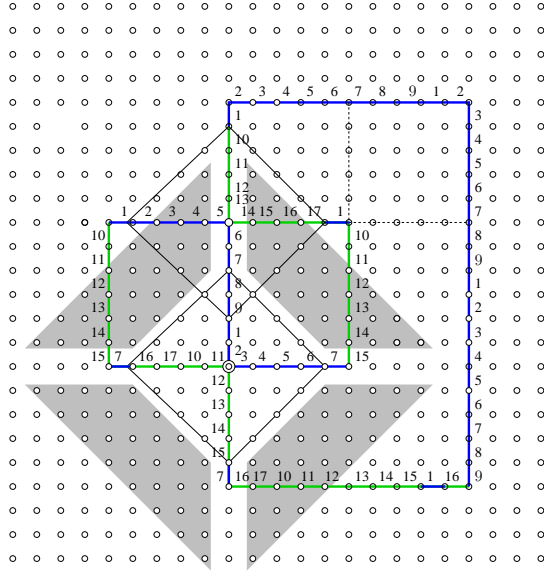


Figure 6.10: Routing the demand of the node  $(0,6)$  in  $Z_A$ . In this example,  $d = 8$ .

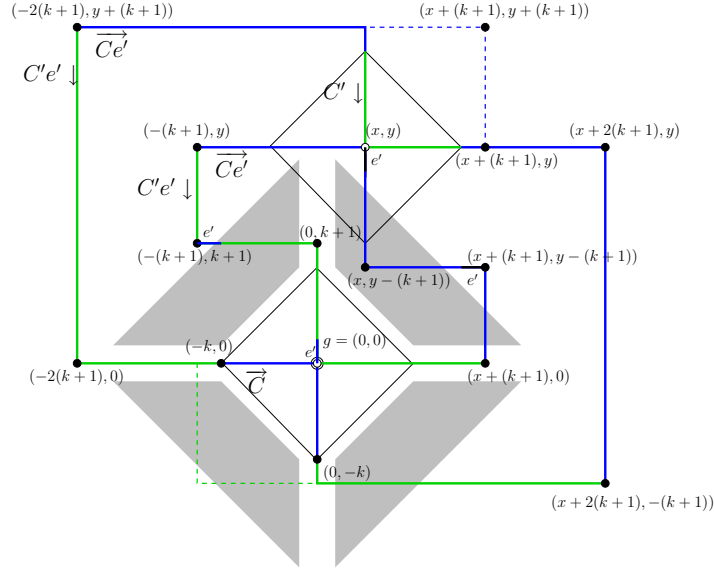


Figure 6.11: labeling paths for the nodes in  $Z'_A$ .

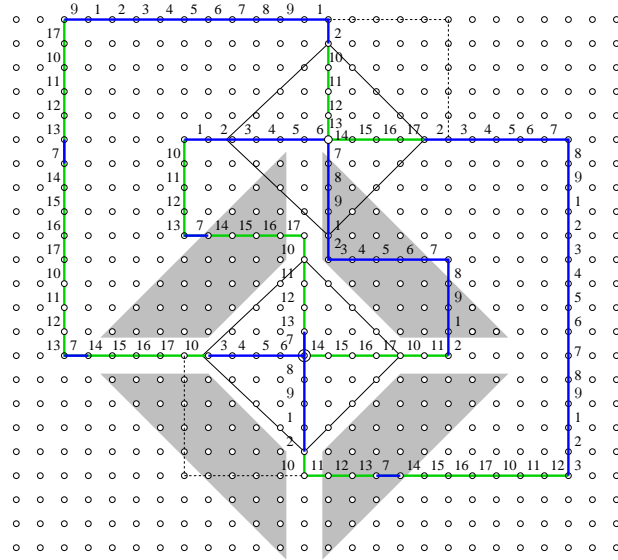


Figure 6.12: Routing the demand of the node  $(1,9)$  in  $Z'_A$ . In this example,  $d = 8$ .



We assign to the edge  $(1, 0)$  the same label  $e'$  given to the edge  $(x, y - 1)(x, y)$ .

2. Now, we label the paths  $P_1 = (-k, 0) \text{---} (0, 0) \text{---} (0, -k)$  with  $C$  by giving to the first edge  $(-k, 0)(-k + 1, 0)$  the same label as the edge of  $P_2$  on its positive diagonal. That is possible as we have  $d$  labels and  $P_1$  is of length  $d$ .
3. Then we label the paths  $P_3 = (-(k + 1), y + (k + 1)) \text{---} (x + k + 1, y + k + 1) \text{---} (x + k + 1, -(k + 1))$  with  $Ce'$ . We do the labeling in such a way the label given to  $(x - 1, y + k + 1)(x, y + k + 1)$  is the one preceding the label given to  $(x, y + k + 1)(x, y + k)$  in step 1. There is no interference as the paths  $P_1$ ,  $P_2$  and  $P_3$  are  $d$ -parallel and the labels are these given in the proof of property 1 (same being translated by one).

4. We also label the *reflected paths*:

$$P'_1 = (x, y + k) \text{---} (x, y) \text{---} (x + k, y);$$

$$P'_2 = (0, y) \text{---} (0, 0) \text{---} (x, 0);$$

$$P'_3 = (-(k + 1), y + k + 1) \text{---} (-(k + 1), -(k + 1)) \text{---} (x + k + 1, -(k + 1)).$$

We use the sequence  $C'e'$  that works perfectly except for the edges of the  $P'_j$  labeled  $e'$  at distance  $< d$  from the path  $P_1$ ,  $P_2$  and  $P_3$ .

For these edges in  $P'_j$ , we give as label the one just before or after the label of the last edge of the path  $P_i$ . For example, if the edge  $(x + k + 1, -k)(x + k + 1, -(k + 1))$  has label  $\beta - 1$  in the sequence  $Ce'$ , the edge initially labeled  $e'$  in  $P'_3$  near that one will get label  $\beta$  (see example in Figure 6.9).

- case 2: node  $v = (0, y)$  on the axis. The proof is similar to the case 1, but we consider the following paths (see example in Figure 6.8 for  $d = 8$ ):

$$P_1: (-(k + 1), y) \text{---} (0, y) \text{---} (0, 0) \text{---} (k + 1, 0);$$

$$P_2: (0, y + (k + 1)) \text{---} (2(k + 1), y + (k + 1)) \text{---} (2k + 1, -(k + 1)).$$

To label these paths, we use the sequence  $Ce'$ . We also label the reflected paths:

$$P'_1: (0, y) \text{---} (k + 1, y) \text{---} (k + 1, 0);$$

$$P'_2: (-(k + 1), y) \text{---} (-(k + 1), 0) \text{---} (0, 0) \text{---} (0, -(k + 1)) \text{---} (-2(k + 1), -(k + 1)).$$

To label these paths, we use  $C'e'$  (with the edges  $e'$  labeled with labels in  $C$  as a continuation from  $P_1$  or  $P_2$ ).

- case 3: node  $v$  in  $Z'_A$ . The proof is similar to the case 1, but we consider the following paths and reflected paths (see example in Figure 6.12 for  $d = 8$ ):

$$P_1: (-k, 0) \text{---} (0, 0) \text{---} (0, -k);$$

$$\begin{aligned}
P_2: & (-(k+1), y) \text{---} (x, y) \text{---} (x, y - (k+1)) \text{---} (x + (k+1), y - (k+1)); \\
P_3: & (-2(k+1), y + (k+1)) \text{---} (x + (k+1), y + (k+1)) \text{---} (x + (k+1), y) \text{---} (x + 2(k+1), y).
\end{aligned}$$

■

### 6.2.2 Gateway in the middle: routing the demand of a combination of nodes

In this section, we present the second routing strategy, which enables to route simultaneously the same flow (less than or equal to the smallest demand) from 2 (for  $d$  odd) or 4 vertices (for  $d$  even).

#### Case $d$ odd

As we saw, this case is solved for the demand of a single node with a cycle. That is, we can attain the lower bound for the problem, but not necessarily with *integer round weights*. Here, we present solutions that deal with this requirements. In our method, it suffices to find one path for each of the two selected vertices. They can be in the same region, in two adjacent regions, or in two opposite regions.

**Theorem 10** *Let  $G$  be a 2-dimensional grid with  $\min(p_1, p_2, q_1, q_2) \geq \frac{3d}{2}$ , gateway  $g$  in the middle and a pair of vertices  $v_1$  and  $v_2$  not in  $K_0$ , and let  $d$  be odd ( $d = 2k - 1$ ). There exist 2 paths that can be  $(d + 1)$ -labeled, one from each node  $v_i$  to the gateway.*

**Proof:** To prove that, we use the splitting in 4 regions introduced in Definition 13 and distinguish 3 cases:

- case 1: the two nodes are in opposite regions  $R_A$  and  $R_C$  (or  $R_B$  and  $R_D$ ). Let  $v_1 = (x_1, y_1)$  with  $x_1 \geq 0, y_1 > 0$ ; and  $v_2 = (x_2, y_2)$  with  $x_2 \leq 0, y_2 > 0$ . In that, we use corollary 8 with the shortest path  $(x_1, y_1) \text{---} (0, y_1) \text{---} (0, 0) \text{---} (0, y_2) \text{---} (x_2, y_2)$ .
- case 2: the two nodes are in adjacent regions  $R_A$  and  $R_B$  (or  $R_B$  and  $R_C$ , or  $R_C$  and  $R_D$ , or  $R_D$  and  $R_A$ ). Let  $v_1 = (x_1, y_1)$  with  $x_1 \geq 0, y_1 > 0$ ; and  $v_2 = (x_2, y_2)$  with  $x_2 < 0, y_2 \geq 0$ .

In that case, the node with the smallest distance to the gateway uses the shortest path and the other node makes a detour guaranteeing a  $d + 1$ -labeling of the two paths (see Figure 6.13 and 6.14). We suppose that  $d(v_2, g) \leq d(v_1, g)$  as in Figure 6.13. Let  $P_2 = (x_2, y_2) \text{---} (x_2, 0) \text{---} (0, 0)$  and label it with a chain  $c = \{1, 2, \dots, 2k\}$  (so  $|c| = d + 1$ ) starting at  $(x_2, y_2)$ .

For  $v_1$ , we use the path  $P_1$  from  $g$  to  $v_1$ :

$$P_1 = (0, 0) \text{---} (0, -\alpha) \text{---} (p_2, -\alpha) \text{---} (p_2, y_1) \text{---} (x_1, y_1)$$

We label  $P_1$  with the colors  $C$  starting at the edge  $(0,0)(0,-1)$  with the label following that of the edge  $(-1,0)(0,0)$ . We chose  $\alpha$  in such way the last edge of  $P_1$  receives the label  $2k$  or  $2k-1$  according to the parity.  $\alpha$  is chosen as the smallest possible integer such that  $2\alpha + 2p_2 - x_1 + y_1 - x_2 + y_2 \equiv \beta \pmod{2k}$  where  $\beta = 0$  if  $-x_1 + y_1 - x_2 + y_2 \equiv 0 \pmod{2}$  or  $\beta = 1$  otherwise. In the example  $x_1 = 0$ ,  $y_1 = 6 - x_2 = -1$ ,  $y_2 = 5$ ,  $p_2 = 7$ , so  $2\alpha + 26 \equiv 0 \pmod{8}$  gives  $\alpha = 3$ .

There exist such an  $\alpha$ , as an increase of  $\alpha$  by 1 increases the length of  $P_1$  by 2. The two paths are interference free.

- case 3: two nodes in the same region. We suppose the region is  $R_A$  and let  $v_1 = (x_1, y_1)$  with  $x_1 \geq 0$ ,  $y_1 > 0$ ; and  $v_2 = (x_2, y_2)$  with  $x_2 \geq 0$ ,  $y_2 > 0$ . This case is not complicated and we subdivide it into 3 subcases, but give the formal proof only for the first subcase.

- subcase 1:

Consider  $P_1 = (0,0) \text{---} (0,-\alpha) \text{---} (p_2,-\alpha) \text{---} (p_2,y_1) \text{---} (x_1,y_1)$  and  $P_2 = (0,0) \text{---} (x_2,0) \text{---} (x_2,y_2)$  (see Figure 6.13). We start labeling the inverse of  $P_2$  that is  $(x_2,y_2) \text{---} (x_2,0) \text{---} (0,0)$  with the chain  $C^+ = 1, 2, \dots, 2k$ . Note that the size of  $|P_2| = x_2 + y_2$  is fixed, so we label the path  $P_1$  with the same chain  $C^+ = 1, 2, \dots, 2k$  but starting with the label used in the edge  $(-1,0)$  of  $P_2$  plus one. We choose the value of  $\alpha$  in order to obtain the last edge of  $P_1$  receiving the label  $2k$  or  $2k-1$ . This configuration is possible as  $|P_1| + |P_2| = 2\alpha + 2p_2 - x_1 + y_1 + x_2 + y_2$ . The  $\alpha$  is chosen the smallest possible such that  $2\alpha + 2p_2 - x_1 + y_1 + x_2 + y_2 \equiv k - \beta \pmod{2k}$  where  $\beta = 0$  if  $-x_1 + y_1 + x_2 + y_2 \equiv k \pmod{2}$  and  $\beta = 1$  otherwise. That is possible as an increase of  $\alpha$  of 1 increases the length of the path  $P_1$  by 2. The paths are then interference free.

- subcase 2 ( $x_1 < x_2$  and  $y_1 < y_2$ ): with  $y_2 - x_2 \geq y_1 - x_1$  or  $y_2 - x_1 < y_1 - x_1$ .

■

Note that when  $y_1 = y_2$  (similarly  $x_1 = x_2$ ) we can apply the method presented in the subsection 6.2.1 (case  $d$  odd) that is able to route a complete column (or a complete line) of a region. Applying theorem 10 we obtain:

**Theorem 11** *Let  $G$  be a 2-dimensional grid with gateway  $g$  in the middle, and let  $d$  be odd ( $d = 2k - 1$ ). If  $\sum_{v \notin K_0} b(v)$  is even then  $W_{min} = S_0$  is solution for IRWP.*

**Proof:** For vertices in  $K_0$ , we use corollary 6. For  $v \notin K_0$ , if  $b(v)$  is an even integer, we send the demand in  $\frac{d+1}{2}b(v)$  rounds by theorem 7. If  $b(v)$  is an odd integer, we send a flow of  $b(v) - 1$  using  $\frac{d+1}{2}(b(v) - 1)$  rounds by theorem 7. An even number of vertices remains with a demand of 1, as the total demand  $\sum_{v \notin K_0} b(v)$  is even. Then, we use theorem 10 to send the demand of each two nodes in  $d + 1$  rounds. ■

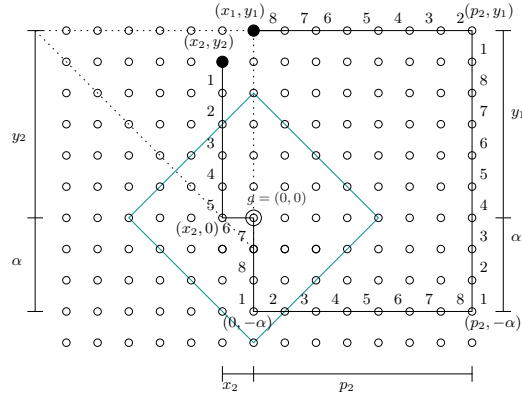


Figure 6.13: Routing strategy with  $d(v_2, g) \leq d(v_1, g)$ . In this example,  $d = 7$ .

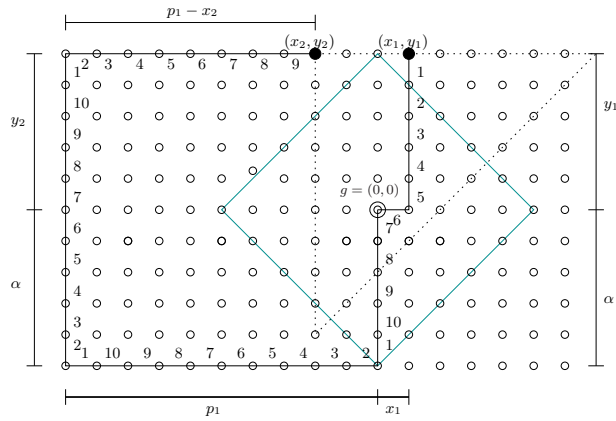


Figure 6.14: Routing strategy with  $d(v_2, g) > d(v_1, g)$ . In this example,  $d = 9$ .

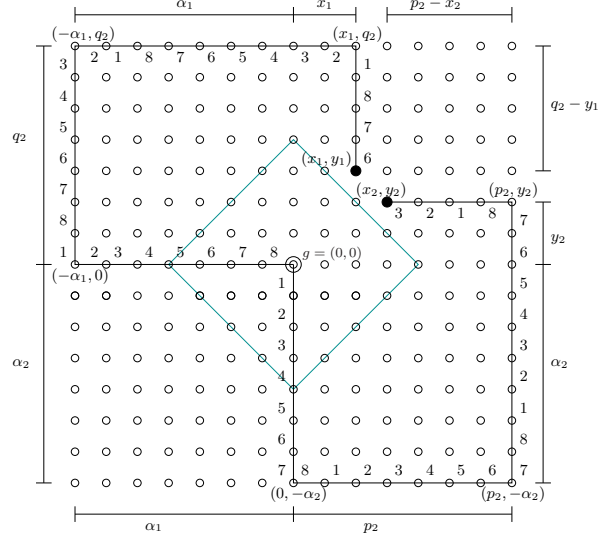


Figure 6.15: Case 3 ( $x_1 < x_2$  and  $y_1 > y_2$ ). In this example,  $d = 7$ .

### Case $d$ even

In what follows, we consider only nodes in the region  $R_A$  (see Definition 13). Without loss of generality, we give the labels only for the path that is completely contained in  $R_A$ , connecting a node in this region to the gateway. The constructions can be done for the 3 other regions by rotating the paths of  $\frac{\pi}{2}$  for  $R_B$ ,  $\pi$  for  $R_C$  and  $\frac{3\pi}{2}$  for  $R_D$  (see figures 6.16(a) and 6.16(b)).

We will use  $4k + 1$  labels  $a_1 \dots a_k$ ,  $b_1 \dots b_k$ ,  $c_1 \dots c_k$ ,  $d_1 \dots d_k$  and  $e$  to label the paths. In fact, we use chains of labels, for example the chain  $A^+$  represents the sequence of labels  $a_1, a_2 \dots a_k$  and  $A^-$  represents the inverted chain  $(a_k, a_{k-1} \dots a_1)$ , similarly for  $B^-$ ,  $B^+$ ,  $C^-$ ,  $C^+$ ,  $D^-$  and  $D^+$ . We use also concatenation of these chains, for example  $A^+eC^-$  means  $a_1, a_2 \dots a_k, e, c_k, c_{k-1} \dots c_1$ .

Let  $m(l_1) = l_2$  denotes the mapping of a given label  $l_1$  into another label  $l_2$ . We define  $m(a_i) = b_i$ ,  $m(b_i) = c_i$ ,  $m(c_i) = d_i$  and  $m(d_i) = a_i$ . So  $m^2(a_i) = c_i$ ,  $m^3(a_i) = d_i$  and  $m^4(a_i) = a_i$ . If an edge  $e$  in a path  $P$  is labeled  $l$ , the edge  $\rho(e)$  of  $\rho(P)$  is labeled  $m(l)$ . For example, if we use an horizontal path  $(0, a) - (p_2, a)$  in  $R_A$ , the rotated path in  $R_B$  will be a vertical path  $(-a, 0) - (-a, q_2)$ , it is the path  $(0, -a) - (-p_1, -a)$  in  $R_C$  and  $(a, 0) - (a, -q_1)$  in  $R_D$ .

**Theorem 12** *Let  $G$  be a 2-dimensional grid with  $\min(p_1, p_2, q_1, q_2) \geq k + 1$ , gateway  $g$  in the middle and  $v_A \in R_A$ ,  $v_B \in R_B$ ,  $v_C \in R_C$  and  $v_D \in R_D$ , and let  $d$  be even ( $d = 2k$ ). There exist 4 paths that can be  $(4k + 1)$ -labeled, one from each node  $v_i$  to the gateway.*

**Proof:** We start showing that two edges on a path in  $R_A$  with the same label do not interfere (as their distance is  $\geq d$ ). We distinguish three cases according

to the vertex  $(x, y) \in R_A$  position:

- Case 1: For a vertex  $(0, y)$  (in the axis), we use the vertical path  $(0, y) - (0, 0)$ ;  
We start from  $(0, 0)$  using the repetition of chain  $A^+eC^-A^+eC^- \dots$ . Doing so, edges  $(0, i-1+\lambda(d+1))(0, i+\lambda(d+1))$  are labeled  $a_i$  and edges  $(0, -i+(\lambda+1)(d+1))(0, -(i-1)+(\lambda+1)(d+1))$  are labeled  $c_i$ . Edges  $(0, k+\lambda(d+1))(0, k+1+\lambda(d+1))$  are labeled  $e$ , for  $1 \leq i \leq k$  and  $\lambda \geq 0$ .
- Case 2: For a vertex  $(x, y)$  with  $x \geq 0, y \geq k+1$ , we use a shortest path that goes first horizontally and then vertically:  $(x, y) - (0, y) - (0, 0)$ .  
We start from  $(0, y)$  labeling the part  $(0, y) - (x, y)$  of the path by repeating the chain  $B^-D^+e$ . So edges  $(i-1+\lambda(d+1), y)(i+\lambda(d+1), y)$  are labeled  $b_i$ , edges  $(-i+(\lambda+1)(d+1), y)(-(i-1)+(\lambda+1)(d+1), y)$  are labeled  $d_i$ , and edges  $(2k+\lambda(d+1), y_0)(2k+1+\lambda(d+1), y_0)$  are labeled  $e$ . The rest of the path, that is  $(0, y) - (0, 0)$ , is labeled as explained in the case 1 (i.e. for the vertex  $(0, y)$ ).
- Case 3: For a vertex  $(x, y)$  with  $x \geq 0, 0 < y < k+1$ , we do not use a shortest path. It goes first vertically to  $(x, k+1)$  then goes horizontally and vertically:  $(x, y) - (x, k+1) - (0, k+1) - (0, 0)$ .  
We use the chain  $C^-$  to label the part  $(x, y) - (x, k+1)$  of the path. If  $x \geq k$  the chain starts from  $y = 1$  till  $y = k+1$  so edge  $(x, k-i)(x, k-i+1)$  is labeled  $c_i$ ; If  $0 \leq x < k$ , edge  $(x, i)(x, i+1)$  is labeled  $c_i$ , for  $k+1-x \leq i \leq k$ . The rest of the path, that is  $(x, k+1) - (0, k+1) - (0, 0)$ , is labeled as explained in the case 2 (i.e. for the vertex  $(x, k+1)$ ).

It remains to show that, two edges with the same label in different paths (in  $R_A, R_B, R_C$  and  $R_D$ ) are also at distance  $\geq d$ . As the labels are obtained by rotation, it suffices to verify the property for the edges labeled, for example,  $a_i$ . They are of four types of edges labeled  $a_i$ :

- Type 1 (in region  $R_A$ ):  $(0, i-1+\lambda_1(d+1))(0, i+\lambda_1(d+1))$ ;
- Type 2 (in region  $R_B$ , that is  $m(d_i)$ ):  $(-y, -i+(\lambda+1)(d+1))(-y, -(i-1)+(\lambda+1)(d+1))$  with  $y \geq 0, x \leq -(k+1)$ ;
- Type 3 (in region  $R_C$ , that is  $m^2(c_i)$ ):  
-With  $x \leq 0, -(k+1) < y < 0$ ,  
If  $x \leq -k$ , the edges are  $(-x, i-k)(-x, i-k-1)$ . If  $-k < x \leq 0$ , the edges are  $(-x, -i)(-x, -i-1)$  that we call type 3'.  
-With  $x = 0$ , the edges are in the axis. They are  $(0, i-(\lambda+1)(d+1))(0, i-1-(\lambda+1)(d+1))$ .
- Type 4 (in region  $R_D$ , that is  $m^3(b_i)$ ):  $(y, 1-i-\lambda(d+1))(y, -i-\lambda(d+1))$  with  $y \leq 0, x \geq k+1$ .

In all the cases the distance between two edges of different type is  $\geq d$ . The distance between an edge type 1 and 2 is  $\geq x + k - 1 \geq d$  as  $x \geq k + 1$  (in the case  $x = k + 1$  it corresponds to the proposition 1). For type 1 and 3 the distance is  $\geq 2d$  with  $\lambda_1 = 1$ , the distance is clearly  $\geq d$  for an edge of type 3. For the type 3' the distance is  $\geq x + k$ , if  $x_1 \geq k$  and  $x \geq 2d$  or exactly  $2d$  when  $x_1 < k$ . For type 4 it is  $\geq x_2 + k - 1 \geq d$  as  $x_2 \geq k + 1$ .

Edges of type 2, 3 or 4 are far apart (at least  $2d$ ). For the type 2 and 3 the distance is  $\geq |x - x_r| + 2d$ . For the type 3 and 4 the distance is more than  $x_2 + d - 1 \geq d$  as  $x_2 \geq k + 1$  (the case  $x_2 = k + 1$  corresponds to the proposition 1). Finally the distance between an edge of type 3' and 4 is  $\geq 2d$  ( $2d$  when  $x_1 \leq k$ ).

In summary, if we take 4 vertices one in each region and use the described paths and labeling, these 4 paths are  $(4k + 1)$ -labeled and, we can satisfy a demand of 1 in each vertex with  $4k + 1$  rounds. ■

If  $\sum_{v \in R_1} b(v) = \sum_{v \in R_2} b(v)$ ,  $\forall R_1, R_2 \in \{R_A, R_B, R_C, R_D\}$ , we say the regions are *balanced*.

**Corollary 10** *Let  $G$  be a 2-dimensional grid with gateway  $g$  in the middle, and let  $d$  be even. If the regions are balanced,  $W_{min} = \frac{2d+1}{4} \sum_{v \notin K_0} b(v)$ .*

Let  $Z'_R$  be the union of the nodes in  $Z_B, Z_C$  and  $Z_D$  of the region  $R$ .

**Theorem 13** *Let  $G$  be a 2-dimensional grid with gateway  $g$  in the middle, and let  $d$  be even. If  $\sum_{v \notin K_0} b(v)$  is multiple of 4 and  $\sum_{v \in Z'_R} b(v) \leq \frac{1}{4} \sum_{v \in R} b(v)$ ,  $\forall R \in \{R_A, R_B, R_C, R_D\}$ , there exist 4 paths from 4 nodes to  $g$  that can be  $(2d + 1)$ -labeled. Therefore,  $W_{min} = \frac{2d+1}{4} \sum_{v \notin K_0} b(v)$  is solution for IRWP.*

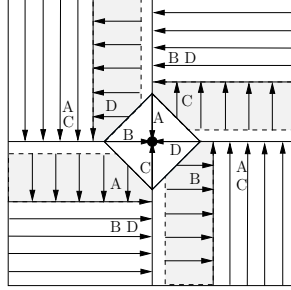
**Proof:** We can always route together a remaining demand of four nodes as  $\sum_{v \in Z'_R} b(v) \leq \frac{1}{4} \sum_{v \in R} b(v)$ . For that, we need four interference free paths, that exist as proved in theorem 12. ■

### 6.2.3 Gateway in the corner: routing the demand of a single node

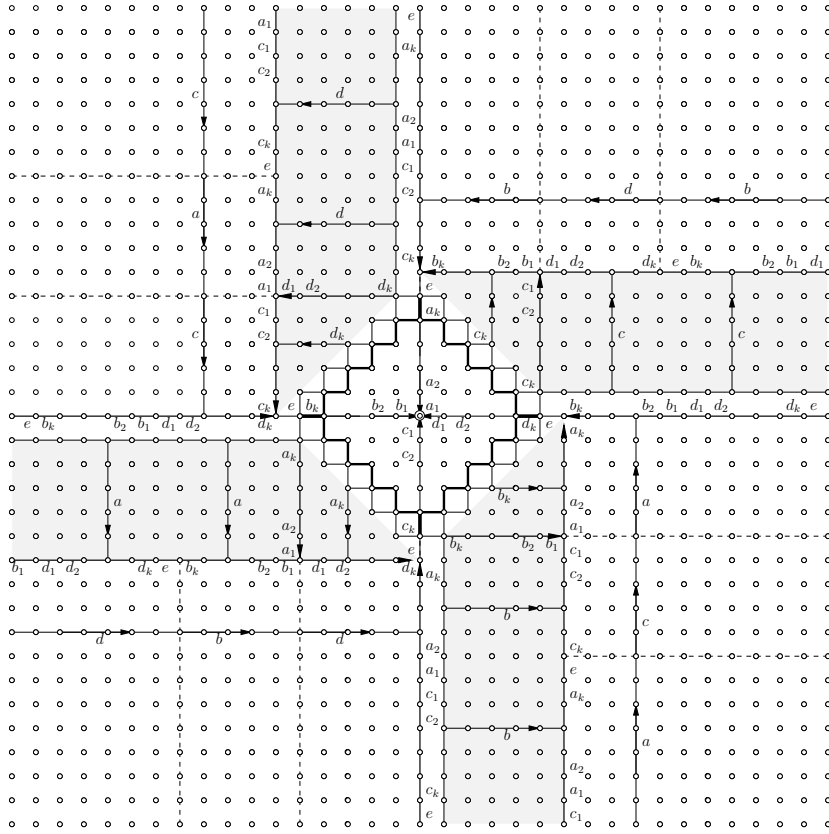
Now we consider the case where the sink is in the corner. Recall that we suppose that the gateway  $g$  is placed at vertex  $(0, 0)$  and we consider a  $p \times q$  grid with vertices  $(x, y)$  where  $0 \leq x \leq p$  and  $0 \leq y \leq q$ . In view of the example in Figure 5.4 and 5.5 (vertex  $(3, 2)$  for  $d = 4$ ), determining  $W_{min}$  when the demand is concentrated in a node can be very difficult for specific vertices.

In the next proposition, we show that for the vertices of the axis plus those vertices of the square  $\{0, d - 1\} \times \{0, d - 1\}$  the lower bound  $\frac{d+1}{2}$  is attained.

**Theorem 14** *Let  $G$  be a 2-dimensional grid  $p \times q$  with  $p \geq 3(d+1)$ ,  $q \geq 2(d+1)$  and gateway  $g$  in the corner. Considering the demand  $b(v)$  of a single node  $v = (x, y)$  with  $x = 0, y = 0$ , or  $\{x \geq d \text{ and } y \geq d\}$ , we have  $W_{min} = d(v, g)b(v)$  if  $v \in K_0$  using one shortest path from  $v$  to  $g$ . If  $v \notin K_0$ , there exist 2 paths from  $v$  to  $g$  that can be  $(d + 1)$ -labeled, therefore  $W_{min} = kb(v)$  for the single routing of  $v$ .*



(a) The regions start and finish labels.



(b) Labels re-usability with  $d = 10$ .

Figure 6.16: Routing 4 nodes (one in each region) with 4 paths.



**Proof:** If  $v \in K_0$ , by corollary 6,  $W_{min} = b(v)d(v, g)$ . If  $v \notin K_0$ , the lower bound is  $S_0 = kb(v)$ . To prove the theorem we will construct for any  $x$  a generic cycle containing all the vertices of the column  $x, y \geq 0$  and satisfying the hypothesis of the theorem 5 (length multiple of  $d + 1$  and width  $\geq d$ ). So  $W_{min} \leq kb(v)$  for all nodes  $(x, y)$ . The cycle consists of the following subdipaths (we indicate the vertices where there is a change of direction).

$$(0, 0) \text{---}(x, 0) \text{---}(x, q_2) \text{---}(-d, q_2) \text{---}(-d, -\alpha) \text{---}(0, -\alpha) \text{---}(0, 0).$$

Let  $p'$  be the largest integer  $p' \leq p$  such that  $p' + q$  is a multiple of  $d + 1$ . Let  $C$  be the following cycle of width  $d$  and length  $2(p' + q)$  (multiple of  $d + 1$ ):

$$(0, 0) \text{---}(p', 0) \text{---}(p', q) \text{---}(0, q) \text{---}(0, 0).$$

Note that  $C$  already contains all the vertices of the vertical lines  $x = 0$  and  $x = p'$ . We will use variant of this cycle, all of length multiple of  $d + 1$  and with width  $d$ , to deal with all vertices  $(x, y)$ . We distinguish five cases according to the node position:

- case 1:  $d \geq x \geq p'$  and  $y \geq d$ . We use the cycle:

$$(0, 0) \text{---}(p', 0) \text{---}(p', y) \text{---}(x, y) \text{---}(x, q) \text{---}(0, q) \text{---}(0, 0).$$

- case 2:  $x < d$  and  $y \geq d$ . We use the cycle:

$$(0, 0) \text{---}(p', 0) \text{---}(p', q) \text{---}(x, q) \text{---}(x, y) \text{---}(0, y) \text{---}(0, 0).$$

- case 3:  $x \geq d$  and  $y < d$ . We use the cycle:

$$(0, 0) \text{---}(x, 0) \text{---}(x, y) \text{---}(p', y) \text{---}(p', q) \text{---}(0, q) \text{---}(0, 0).$$

- case 4:  $p' < x \leq p$  and  $p \geq y \geq q - (x - p')$ . We use the cycle:

$$(0, 0) \text{---}(x, 0) \text{---}(x, q - (x - p')) \text{---}(0, q - (x - p')) \text{---}(0, 0).$$

- case 5:  $p' < x \leq p$  and  $y > q - (x - p')$ . this is the most difficult case as we have to add a detour. We use the cycle:

$$(0, 0) \text{---}(x, 0) \text{---}(x, q) \text{---}(2d, q) \text{---}(2d, q - \beta) \text{---}(d, q - \beta) \text{---}(d, q) \text{---}(0, q) \text{---}(0, 0).$$

See Figure 6.17 for an example with  $d = 3$  ( $k = 2$ ),  $x = 14$  and  $q = 8$ . The length of the cycle is  $2(x + q + \beta)$ . We chose  $\beta$  such that  $2(x + q + \beta) \equiv 0 \pmod{(d + 1)}$ . As  $p' + q \equiv 0 \pmod{(d + 1)}$  it suffices to chose  $\beta = d + 1 - (x - p')$ . In the example in Figure 6.17, the length of the cycle is  $40 + 2(d + 1 - \alpha) + 2\alpha$  so length  $40 + 2(d + 1) \equiv 0 \pmod{4}$ . As  $d + 1 - \alpha \leq d + 1$  and  $q \geq 2(d + 1)$  the horizontal paths are at distance  $\geq d$  and, as we chose the vertical line at  $d + 1$  and  $2(d + 1)$  (choice possible, as we have  $p \geq 3(d + 1)$ ), the vertical paths are also at distance  $\geq d$ .

■

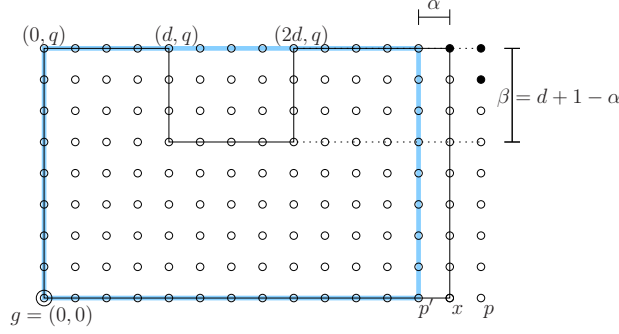


Figure 6.17: Routing method for a node  $v$  with a cycle and  $d$  odd ( $d=3$ ).

#### 6.2.4 Gateway in the corner: routing the demand of a combination of nodes

We start with the following remark.

**Remark 1** *In order to attain the lower bound given in 5.3.2, there are some nodes of which demand cannot be routed independently. Then, its demand must be routed together (sharing rounds) with the demand of some other nodes.*

An example of one of these nodes is shown in figure 5.5 for the case of the gateway in the corner.

We present a solution which takes into account these nodes in order to attain the lower bound in 5.3.2.

In the following, we will suppose that the demand is uniform, it means that  $b(v) = c > 0$  for all  $v \neq g$ . We will consider  $c = 1$ , however the following routing can be directly applied for any  $c > 0$ .

We define the *individual lower bound* of a node  $v$ , denoted by  $lb(v)$ , as the lower bound given in 5.3.2 considering the demand as  $b(v) = 1$  and  $b(u) = 0$  for all  $u \neq v$ . In other words,  $lb(v)$  is the contribution of  $v$  to the lower bound given in 5.3.2.

We will suppose that the grid is large enough to construct the routings presented below.

We will show a way of routing the demand which attains the lower bound given in 5.3.2. We will route the demand by different methods depending on the position in the grid. In figure 6.18, we can see a scheme of how the nodes are grouped according to the method of routing proposed.

For the case where a node is routed independently, the idea is to obtain a routing such that the total weight of the rounds would be equal to the individual lower bound of this node. But, as seen in remark 1, there are zones of the grid whose demand cannot be routed independently. In this case, the idea is to route a group of nodes in such a way that the sum of their individual lower bounds would be equal to the total weight of the rounds involved.

We will define  $1_{\text{odd}}(d)$  or simply  $1_{\text{odd}}$  as the function with value 1 when  $d$  is odd and 0 when  $d$  is even. In the same way, we define  $1_{\text{even}}$  the function which is 1 when  $d$  is even and 0 if  $d$  is odd.

Let us define the set of nodes  $Z_{SP}$  as the nodes  $v$  such that  $d(v, g) = lb(v)$ . Note that  $Z_{SP}$  corresponds to  $\{v = (x, y) \in V \mid x, y \leq \lceil \frac{d}{2} \rceil \text{ and } d(v, g) \leq \lfloor \frac{3(d+1_{\text{odd}})}{4} \rfloor\}$ . For a node  $v$  in  $Z_{SP}$  such that  $x \leq y$  we will route its demand by the path  $v - (0, y) - g$ . Inversely, if  $y < x$  we will use the path  $v - (x, 0) - g$ . In both cases, each path have  $d(v, g)$  edges, with  $d(v, g) \leq d$ . Then, for any node  $v$  in  $Z_{SP}$  we use a path covered with  $d(v, g)$  rounds with weight  $b(v) = 1$  each. Then, the total weight for routing each node  $v$  in  $Z_{SP}$  is  $d(v, g) = lb(v)$ .

Moreover, it is possible to move the demand due to the nodes in  $Z_D$  sharing the same rounds used to route the demand of  $Z_{SP}$ . An scheme of that is presented in figure 6.19. We can see that the rounds needed to route the nodes  $(0, i), (i, 0)$  and  $(\lceil \frac{d}{2} \rceil, j), (j, \lceil \frac{d}{2} \rceil)$  with  $i \leq \lceil \frac{d}{2} \rceil$  and  $j \leq \lfloor \frac{d}{4} \rfloor$  are enough to move the all demand due to the zone  $Z_D$ . In this way, the displaced demand is moved to nodes located out of the zone  $\{1, d-1\} \times \{1, d-1\}$ . We will see after that each unit of relocated demand can be routed with cost  $\frac{d+1}{2}$ . Thus, each node  $v$  of  $Z_D$  is routed using a weight of  $lb(v)$ .

The nodes in  $Z_C$  are the nodes  $v$  in  $\{0, v^*(d)\} \times \{0, v^*(d)\}$  such that  $lb(v) > d(v, g)$ . Then,  $Z_C$  corresponds to  $\{v = (x, y) \in V \mid x, y \leq \lceil \frac{d}{2} \rceil \text{ and } d(v, g) > \lfloor \frac{3(d+1_{\text{odd}})}{4} \rfloor\}$ . In this zone, nodes satisfy that  $lb(v) = d(v, v^*(d)) + \frac{d+1}{2}$ . The routing will be done in two parts. The first part is to move the demand from the node  $v$  to the  $v^*(d)$  with cost  $d(v, v^*(d))$ . The second part is to move the demand from  $v^*(d)$  to the gateway with cost  $\frac{d+1}{2}$ . For the first part, we will route the demand via a shortest path between  $v$  and  $v^*(d)$ . We will use  $d(v, v^*(d))$  rounds, therefore it costs  $d(v, v^*(d))$ . For the second part, as the demand is already in  $Z_E$ , we will route the normal routing of  $Z_E$  which attains a cost of  $\frac{d+1}{2}$  as we will see later.

The nodes in  $Z_B$  correspond to the nodes in  $\{v = (x, y) \mid d(v, g) \leq d \text{ with } x > \lceil \frac{d}{2} \rceil \text{ and } y \geq \lfloor \frac{d+2}{4} \rfloor\} \cup \{v = (x, y) \mid d(v, g) \leq d \text{ with } y > \lceil \frac{d}{2} \rceil \text{ and } x \geq \lfloor \frac{d+2}{4} \rfloor\}$ . Note that, for any node  $v$  in  $Z_B$ , the  $lb(v)$  is determined by  $\frac{d+1}{2} + l + 1_{\text{even}} = \lfloor \frac{d+2}{2} \rfloor + l$ , with  $l$  the distance between  $v$  and the zone  $Z_D$ . We will route the nodes by pairs: each node of  $Z_B$  will be routed together with one node of  $Z_{Ext}$ . Let us suppose that  $v = (x, y)$  is such that  $x > y$ . The path to do that is shown in figure 6.23(a). Note that the node chosen in  $Z_{Ext}$  must be a node that does not interfere with the current path (For example, any node in  $Z_{Ext}$  placed in the upper border of the grid). Now, we will route the node obtained by swapping the coordinates of  $v$ , i.e, the node  $(y, x)$ . This node will be also routed together with a node in  $Z_{Ext}$ . We will use a path as shown in figure 6.23(b). Now, we can see that it is possible to reuse some rounds of the path that routes  $v = (x, y)$ . In fact, the reused rounds are the  $l + 1_{\text{even}}$  rounds needed to move the demand out of the zone  $Z_B$ . Now, in total,  $2(d + 1 + l + 1_{\text{even}})$  rounds have been used in these two paths there is been routed the demand due to 4 nodes. Two of these nodes, the nodes in  $Z_{Ext}$ , have a lb of  $\frac{d+1}{2}$ . The two nodes in  $Z_B$  have a lb of  $\frac{d+1}{2} + l + 1_{\text{even}}$  each. Therefore, the group of 4 nodes attains a cost equivalent

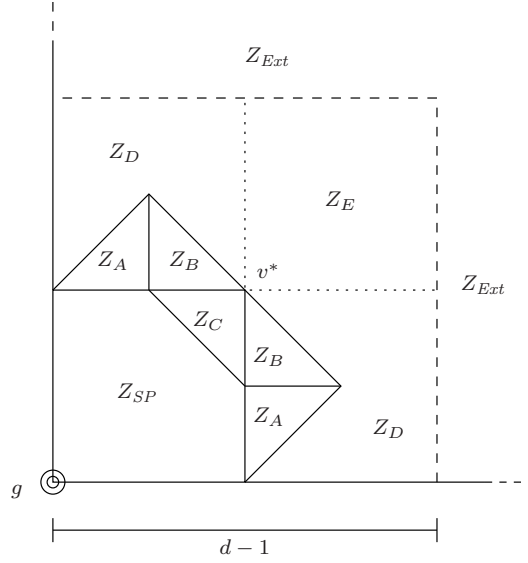


Figure 6.18: Scheme of the grid separated by method of routing.

to the sum of their 4 lb.

The nodes in  $Z_A$  correspond to the nodes in  $\{v = (x, y) \mid d(v^*, v) \leq \lfloor \frac{d}{2} \rfloor + 1_{\text{even}} \text{ and } x > \lceil \frac{d}{2} \rceil \text{ and } y \leq \lfloor \frac{d+2}{4} \rfloor\} \cup \{v = (x, y) \mid d(v^*, v) \leq \lfloor \frac{d}{2} \rfloor + 1_{\text{even}} \text{ and } y > \lceil \frac{d}{2} \rceil \text{ and } x \leq \lfloor \frac{d+2}{4} \rfloor\}$ . Each node  $(x, y)$  in  $Z_A$  with  $x > y$  will be routed together with the node  $(y, d + 1_{\text{odd}} + y - x)$ , also in  $Z_A$ . Note that  $\text{lb}(x, y) = d + 1_{\text{odd}} + y - x$  and  $\text{lb}(y, d + 1_{\text{odd}} + y - x) = x$ . The path used is constructed in the same way that the path shown in figure 6.22. To route the demand through the path,  $d + 1_{\text{odd}} + y$  rounds are needed which is exactly  $\text{lb}(x, y) + \text{lb}(y, d + 1_{\text{odd}} + y - x)$ .

The nodes in  $Z_E$  are the nodes contained in the square delimited by the nodes  $v^*$  and  $(d - 1, d - 1)$ . Each node will be routed using 2 cycles following the idea depicted in figure 6.24. Each cycle routes half of the demand and it shares  $\lfloor \frac{d+1}{2} \rfloor$  rounds with the second cycle. The total number of rounds used is  $2(d + 1)$  and each round has a capacity of  $1/4$ . Then, the weight needed for routing the demand of each node  $v$  in  $Z_E$  is  $\frac{d+1}{2} = \text{lb}(v)$ .

The remaining nodes  $v$  with non-zero demand are all placed outside the zone  $\{1, d - 1\} \times \{1, d - 1\}$ . Applying theorem 14, each node can be routed independently with cost  $\frac{d+1}{2}$  which is the value of  $\text{lb}(v)$ .

As the sum of  $\text{lb}(v)$  over all the nodes in the grid attains the lower bound given in 5.3.2, we conclude the result.

### 6.3 Conclusion

In this chapter, it is proved that for  $d$  odd our lower bound is tight for grids considering fractional round weights (the demand of each node can always be

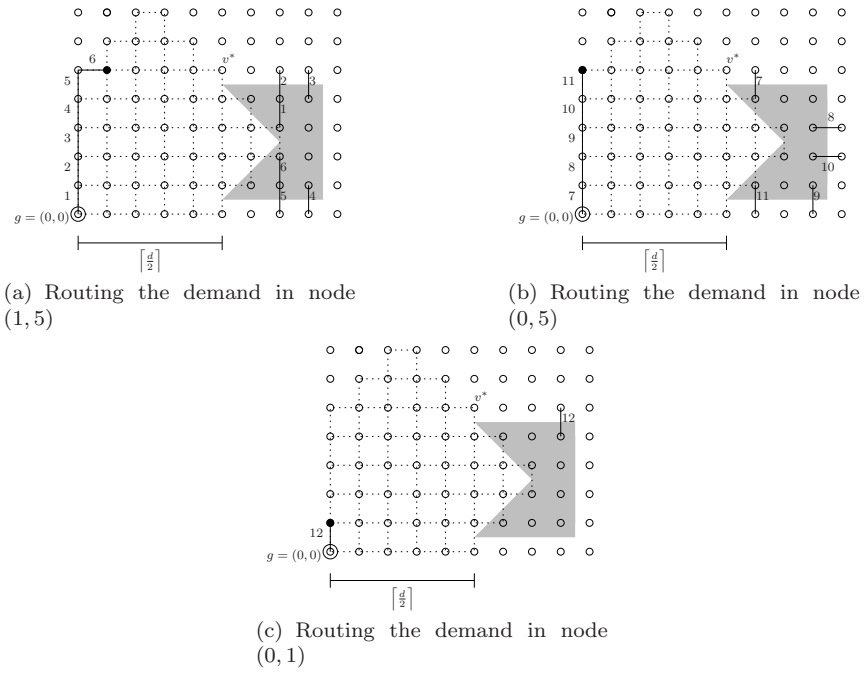


Figure 6.19: Example of moving the demand in  $Z_D$  using the routing of  $Z_B$ . In this example,  $d = 9$ .

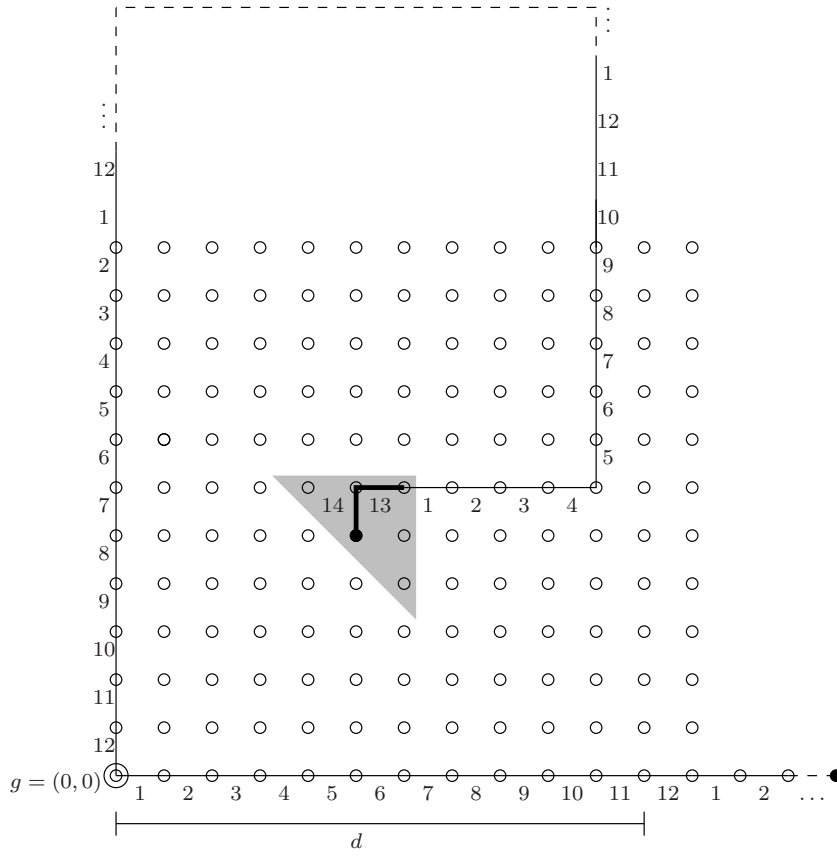


Figure 6.20: Example for  $Z_C$  with  $d$  odd. In this example, the demand.

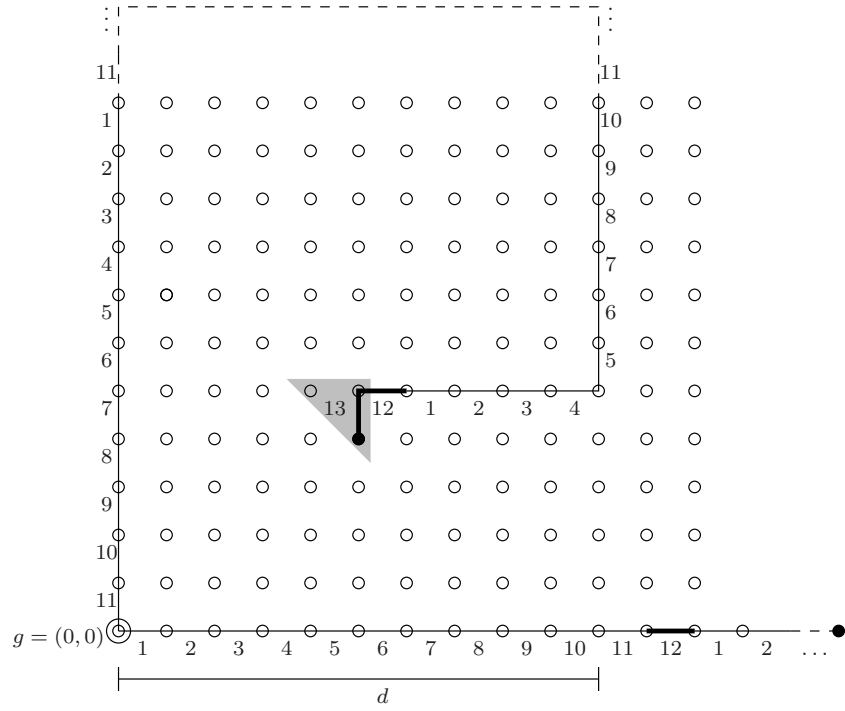


Figure 6.21: Example for  $Z_C$  with  $d$  even. In this example, the demand...

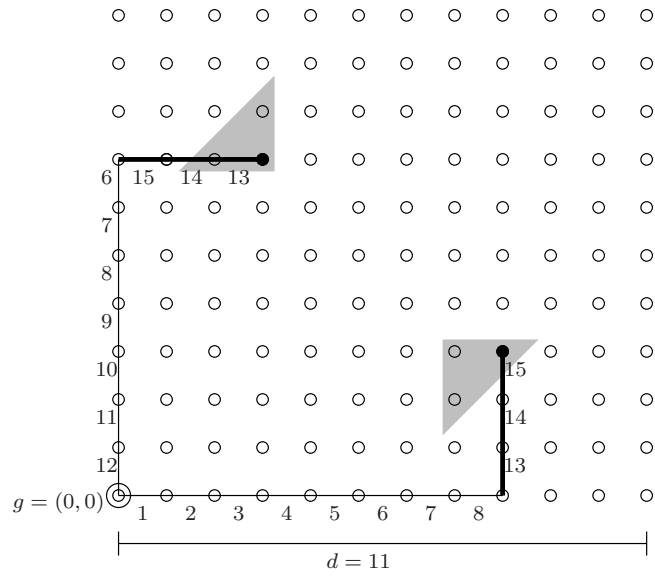


Figure 6.22: Example for  $Z_A$  with  $d$  odd. In this example, the demand. The even case is similar.







## Chapter 7

# Bottleneck region and physical interference model

Motivated by the results of the existence of a limited (bottleneck) region capable to represent the entire network, in this chapter we consider a variant of the RWP, that also deal with bandwidth allocation but using the interference model with SINR (Signal to Interference plus Noise Ratio) conditions. In this case, we do not attempt to allocate a separate slot to each link. Instead the links are allowed to communicate at the same time consequently, the rate of the communication is limited by the others. The power transmitted by each user is defined to maintain the SINR above a given threshold. The model presented here is valid for UMTS and other systems that tolerate interferences (see chapter 1.1).

We give sufficient conditions to the multi-hop problem to be reduced to a single-hop problem by only changing the utility functions. These conditions are represented by our description of utility functions. This work was published in [6]. We present the problem in a multi-hop cellular network but it could be a radio mesh network as well.

Multi-hop Cellular Network preserves the benefit of conventional single-hop cellular networks where the service infrastructure is provided by fixed bases, and also incorporates the flexibility of ad-hoc networks where wireless transmissions through mobile stations in multiple hops is allowed [42].

In ad-hoc networks, nodes communicate with each other in a peer-to-peer way and no infrastructure is required. If direct communication is not feasible, the simplest solution is to replace a single long-range link with a chain of short range links by using a series of nodes between the source and the destination: this is known as multi-hop communication [43]. The cooperation between these two networks can be interesting as ad-hoc networks can expand the covered area without the high cost of cellular networks infrastructure.

We address in this chapter a bottleneck problem that summarizes the situation of many multi-hop cellular networks, as illustrated in Figure 7.1. In our work, a gateway - or *base station (BS)* - has entire access to the rest of the world

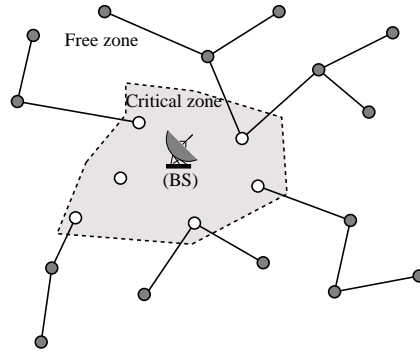


Figure 7.1: Multi-hop cellular network.

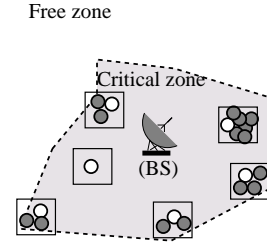


Figure 7.2: Multi-hop cellular network reduced in single-hop.

and provides this service in a more privileged way to some specific nodes, the *relay nodes* (the white ones in Figure 7.1). Those nodes are themselves relaying the service to the nodes in the free zone, called the *terminal nodes* (the gray ones in Figure 7.1). It can happen that nodes in the free zone relay one another to get the final service. In our model, the relay nodes and the gateway form a single-hop cellular network (the critical zone) that constrains the system.

Each node has a utility function representing its degree of satisfaction based on the assigned rate transmission. The whole system is governed by the optimization of the sum of utility functions over all the nodes, as in [44, 45]. We give a model that allows to transform the multi-hop network (Figure 7.1) into a single-hop network (Figure 7.2), by eventually modifying the utility functions on the relay nodes, as depicted in Figure 7.1.

The rest of this chapter is organized as follows. In the next section we discuss the related works. In section 7.1, we define the problem, the adopted notation and the considered hypotheses. The section 7.2 shows how the problem can be reduced to a single-hop network. That is how the complete network utility functions can be replaced by a small set of different functions assigned to the relay nodes, in the context of a fair and optimal optimization. We push forward our results in section 7.3 by applying them to specific cases of fairness.

This described scenario occurs in multi-hop networks as considered in [42, 43]. Indeed, it is observed that often in these networks the bandwidth is constrained specifically by a bottleneck around the gateway [46], confirming the fact that it is a representative area. Many real networks deal with this situation. For instance, using UMTS technology for the single-hop network [47, 45, 48], while the free zone is covered by WiFi or Bluetooth systems.

We show that there exists a set of utility functions that can be assigned to the relay nodes replacing the complete set of utility functions. It is due to the fact that the problem is convex under some conditions that are often met.

Convex optimization techniques are important in engineering applications because a local optimum is also a global optimum in a convex problem. Rigorous optimality conditions and a duality theory also exist to check the solution

optimality. Consequently, when a problem is cast into a convex form, the structure of the optimal solution, which often reveals design insights, can often be identified. Furthermore, powerful numerical algorithms exist to solve convex problems efficiently.

We are interested in Pareto-optimal solutions, that is solutions where the utility of an individual cannot be improved without decreasing the utility of one or more other nodes. The fairness is a key issue in wireless networks, since the medium is shared among the nodes. In our problem, it implies that each flow going through a bottleneck receives a fair share of the available bandwidth. Our work admits the generalized fairness criterion as defined in [44] that can assume several criteria (see section 7.3 for more details), for example, the proportional fairness one.

The proportional fairness has been studied in the context of the Internet flow due the similarity to the congestion control mechanism of the TCP/IP protocols, where each TCP's throughput is adapted as a function of the congestion. The work in [49] addresses the question of how the available bandwidth within the network should be shared between competing streams of elastic traffic<sup>1</sup>.

## 7.1 Model definition

We distinguish here three main types of nodes. The BS that is unique in our case, the relay nodes in  $\mathcal{R}$  that have a limited link to the BS and the terminal nodes in  $\mathcal{T}_r$  that are connected to the BS through a unique relay node  $r$  at the single-hop network. Note that multi-hops are allowed as long as connections between terminal nodes are given for free, that is the relay node has bandwidth enough for itself and its relayed terminals. The terminal nodes are considered sparsely distributed around the cell, thus interference is not a problem at the free zone.

We focus on the downlink channel (from the BS to the relay nodes) considering a given fixed bandwidth. Let  $\alpha_r$  be the rate of the downlink channel from the BS to relay node  $r$ . Let  $\rho_t$  be the downlink rate at each node  $t \in \mathcal{T}_r$ . We consider the following hypotheses.

**Hypothesis 1** *All terminal nodes in  $\mathcal{T}_r$  use a unique relay node  $r \in \mathcal{R}$ .*

**Hypothesis 2** *We only consider interferences between the relay nodes in  $\mathcal{R}$ .*

Note that the first hypothesis allows multiple hops and routes in the free zone but imposes to gather all the traffic of an individual node to a unique relay in the critical zone. The second hypothesis means also that the bandwidth is not limited in the free zone.

We summarize the important definitions below.

---

<sup>1</sup>The elastic traffic tolerates packet delays and losses and permits the nodes to adjust their rates in order to fill available bandwidth.

### Nodes Set

- $BS$ : node representing the base station. We consider a unique  $BS$ .
- $\mathcal{R}$ : set of relay nodes, directly connected to the  $BS$  by a limited link.
- $\mathcal{T}_r$ : set containing the node  $r \in \mathcal{R}$  and the set of terminal nodes relayed by  $r$ .

### Variables

- $p_{b,r}$ : power of the signal emitted by the  $BS$  to the router  $r$ .
- $\rho_t$ : downlink rate for each node  $t \in \mathcal{T}_r$ .
- $\alpha_r$ : downlink rate for each node  $r \in \mathcal{R}$ , enough to attend all nodes  $t \in \mathcal{T}_r$ .  
 $\alpha_r = \sum_{t \in \mathcal{T}_r} \rho_t$ .

### Utility Functions

- $U_t(\rho_t)$ : utility function at the node  $t \in \mathcal{T}_r$  representing its degree of satisfaction. This function is non-decreasing with  $\rho_t$ .
- $\mathcal{U}_r(\alpha_r)$ : cumulative utility function at the node  $r \in \mathcal{R}$  representing the maximum degree of satisfaction of the nodes in  $\mathcal{T}_r$ . Given that the bandwidth is  $\alpha_r$ , it is defined  $\mathcal{U}_r(\alpha_r) = \max\{\sum_{t \in \mathcal{T}_r} U_t(\rho_t); \sum_{t \in \mathcal{T}_r} \rho_t = \alpha_r\}$ .

We deal with the **optimal and fair transmission rate allocation problem** (problem (P)), we have to find a vector of the relay rates  $\alpha$  that maximizes  $\sum_{r \in \mathcal{R}} \mathcal{U}_r(\alpha_r)$  with a fair sharing among the terminals, guarantying the existence of a vector of transmissions powers  $p = (p_{b,1}, p_{b,2}, \dots, p_{b,|\mathcal{R}|})$ .

In order to model interference in the critical zone, we focus on a commonly used definition of *feasible rates* which depends on both a target  $\gamma$  and a target interference level  $K$ . The packet sent by the  $BS$  is received by the relay node if the SINR (Signal to Interference plus Noise Ratio) is above a given threshold  $\gamma$ . The constants  $N_o$  and  $g_{b,r}$  are given considering the network environment. Let  $N_o$  be the thermal noise and  $g_{b,r}$  is the channel gain between the  $BS$  and the relay  $r$ . The variables  $p_{b,r}$  represent the power of the signal emitted by the  $BS$  to the relay  $r$ .

A vector of rates  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_{|\mathcal{R}|})$  is considered a feasible solution if there exists a vector of transmissions powers  $p = (p_{b,1}, p_{b,2}, \dots, p_{b,|\mathcal{R}|})$  that satisfies the following conditions for the SINR that a node connected to the  $BS$  experiences:  $\alpha_r \gamma \leq \frac{p_{b,r} g_{b,r}}{N_o + g_{b,r} \sum_{s \neq r} p_{b,s}} = SINR_r, \forall r \in \mathcal{R}$  and  $\sum_{r \in \mathcal{R}} p_{b,r} \leq K N_o$ . A vector of rates  $\alpha$  is an *optimal rate allocation* if it is a solution to the following model on variables  $\alpha$  and  $p$ :

**Problem (P')**

$$\max \sum_{r \in \mathcal{R}} \mathcal{U}_r(\alpha_r) \quad (7.1)$$

subject to

$$\alpha_r \gamma \leq \frac{p_{b,r} g_{b,r}}{N_o + g_{b,r} \sum_{s \neq r} p_{b,s}}, \forall r \in \mathcal{R} \quad (7.2)$$

$$\sum_{r \in \mathcal{R}} p_{b,r} \leq K N_o. \quad (7.3)$$

Since utility functions are non-decreasing, an optimal solution verifies:

$$\alpha_r \gamma = \frac{p_{b,r} g_{b,r}}{N_o + g_{b,r} \sum_{s \neq r} p_{b,s}}, \forall r \in \mathcal{R}$$

which gives  $p_{b,r} = \frac{\alpha_r \gamma}{g_{b,r}} (N_o + g_{b,r} \sum_{s \neq r} p_{b,s})$ , thus

$$p_{b,r} = \frac{\alpha_r \gamma}{g_{b,r}} (N_o + g_{b,r} \sum_{s \in \mathcal{R}} p_{b,s} - g_{b,r} p_{b,r}), \forall r \in \mathcal{R}. \quad (7.4)$$

Moreover, increasing all the powers by the same factor allows to tighten constraints (7.3) while relaxing constraints (7.2). By optimality we have

$$\sum_{r \in \mathcal{R}} p_{b,r} = K N_o$$

which, put into equation (7.4) gives  $p_{b,r} = \frac{\alpha_r \gamma}{g_{b,r}} (N_o (1 + g_{b,r} K) - g_{b,r} p_{b,r})$  and we obtain:

$$p_{b,r} = \frac{\alpha_r \gamma N_o (1 + K g_{b,r})}{g_{b,r} (1 + \alpha_r \gamma)}, \forall r \in \mathcal{R}. \quad (7.5)$$

Like in [47], we use the substitution  $\frac{\alpha_r \gamma (1 + K g_{b,r})}{g_{b,r} (1 + \alpha_r \gamma)} = d_r$ . As  $\sum_{r \in \mathcal{R}} d_r \leq \sum_{r \in \mathcal{R}} \frac{p_{b,r}}{N_o} \leq K$ , we can say:

$$\alpha_r = \frac{d_r g_{b,r}}{\gamma (1 + g_{b,r} (K - d_r))}, \forall r \in \mathcal{R}.$$

So, we obtain the following equivalent problem on variables  $d_r$ :

**Problem (P)**

$$\max \sum_{r \in \mathcal{R}} \mathcal{U}_r \left( \frac{d_r g_{b,r}}{\gamma (1 + g_{b,r} (K - d_r))} \right) \quad (7.6)$$

subject to

$$\begin{cases} \sum_{r \in \mathcal{R}} d_r \leq K \\ d_r \geq 0, \forall r \in \mathcal{R}. \end{cases} \quad (7.7)$$

## 7.2 Theoretical approach with fairness and optimality

In this section, the problem is how to define the cumulative utility functions  $\mathcal{U}_r(\alpha_r)$  for each relay node  $r \in \mathcal{R}$  in a way to represent the utility functions  $U_t(\rho_t)$  of all nodes  $t \in \mathcal{T}_r$ . Moreover, the available bandwidth of each relay node has to be shared with fairness among the nodes in  $\mathcal{T}_r$ .

Indeed, we prove that there exists a set of utility functions  $\mathcal{U}_r(\alpha_r)$  (cumulative functions) that can be assigned to the relay nodes replacing the complete set of utility functions and, it can be expressed analytically in most cases. Moreover, we show that for any fixed available bandwidth  $\alpha_r$  at each relay node, if the sum  $\sum_{t \in \mathcal{T}_r} U_t(\beta_t \alpha_r)$  is maximized it converges to a fairness equilibrium. Thus, it is always possible to share  $\alpha_r$  fairly among the nodes in  $\mathcal{T}_r$ . The fairness equilibrium point is defined by the utility function adopted. We consider the following technical assumption.

**Technical Assumption 1** *The nodes' utility functions  $U_t(\cdot)$  are assumed to be strictly increasing concave functions and satisfy the condition  $U_t''(x) \leq \frac{-1}{x^2}$ .*

As said before the particular utility function  $\mathcal{U}_r(\alpha_r)$  is defined as follows:

**Problem ( $P_r$ )**

$$\mathcal{U}_r(\alpha_r) = \max \sum_{t \in \mathcal{T}_r} U_t(\rho_t) \quad (7.8)$$

subject to

$$\alpha_r = \sum_{t \in \mathcal{T}_r} \rho_t, \forall r \in \mathcal{R}. \quad (7.9)$$

We need the following lemma regarding how the rate  $\alpha_r$  assigned to a relay  $r \in \mathcal{R}$  can be shared by all terminals it relays. Our objective is that given  $\alpha_r$  we can assign a fraction  $\beta_t$  of  $\alpha_r$  for each terminal  $t \in \mathcal{T}_r$  in a fair way.

**Lemma 8** *Given the vector  $\rho^* = (\rho'_1, \dots, \rho'_{|\mathcal{T}_r|})$  being the optimal solution for the problem  $P_r$ , we consider a variable  $\beta'_t \in [0, 1]$ , a fixed feasible relay rate  $\alpha_r$  and we define  $\rho'_t = \alpha_r \beta'_t$ . We obtain*

$$U'_{t_1}(\beta'_{t_1} \alpha_r) = U'_{t_2}(\beta'_{t_2} \alpha_r), \forall t_1, t_2 \in \mathcal{T}_r.$$

**Proof:** Letting  $\beta_t = \frac{\rho_t}{\alpha_r}$ , we consider the following subproblem with a fixed  $\alpha_r$  for a  $r \in \mathcal{R}$ :

**Problem ( $P'_r$ )**

$$\max \sum_{t \in \mathcal{T}_r} U_t(\beta_t \alpha_r) \quad (7.10)$$

subject to

$$\begin{cases} \beta_t \geq 0, \forall t \in \mathcal{T}_r \\ \sum_{t \in \mathcal{T}_r} \beta_t = 1. \end{cases} \quad (7.11)$$

We can say that it is a local version of the problem  $P_r$ , in a way that an optimal solution for  $P'_r$  considering the optimal value for  $\alpha_r^* = \arg \max_{\alpha_r} \mathcal{U}_r(\alpha_r)$  can be translated into a locally optimal solution for  $P_r$ . We can rewrite the constraints as:

$$\begin{cases} -\beta_t \leq 0, \forall t \in \mathcal{T}_r \\ \sum_{t \in \mathcal{T}_r} \beta_t \leq 1 \\ \sum_{t \in \mathcal{T}_r} -\beta_t \leq -1 \end{cases} \quad (7.12)$$

Based on [50], the Lagrangian of this subproblem can be written as follow:

$$L(\beta) = \sum_{t \in \mathcal{T}_r} U_t(\beta_t \alpha_r) - \sum_{t \in \mathcal{T}_r} \lambda_t (-\beta_t) - \mu \left( \sum_{t \in \mathcal{T}_r} \beta_t - 1 \right) - \nu \left( \sum_{t \in \mathcal{T}_r} -\beta_t + 1 \right)$$

with the Lagrange multipliers  $\lambda_i \geq 0$ ,  $\mu \geq 0$  and  $\nu \geq 0$ . As  $\beta^* = (\frac{\rho'_t}{\alpha_r}, \dots, \frac{\rho'_{|\mathcal{T}_r|}}{\alpha_r})$  is necessarily a vector of optimal solutions for the Lagrangian. So it verifies KKT's optimality conditions:  $\frac{\partial L}{\partial \beta_t} = 0$  in  $\beta^*$ ,  $\forall t \in \mathcal{T}_r$ , which gives

$$\alpha_r U'_t(\beta'_t \alpha_r) + \lambda_t - \mu + \nu = 0, \forall t \in \mathcal{T}_r.$$

Moreover, by KKT complementary slackness conditions  $\lambda_t \beta'_t = 0, \forall t \in \mathcal{T}_r$ . Note that under technical assumption 1, we have  $U'_t(\rho_t) \rightarrow -\infty$  when  $\rho_t \rightarrow 0_+$  for all  $t$ . We can deduce that  $U'_t(\rho_t) \rightarrow +\infty$  and  $U_t(\rho_t) \rightarrow -\infty$  when  $\rho_t \rightarrow 0_+$  for all  $t$ , making impossible the case where  $\rho_t = 0$ . So  $\beta_t \neq 0$ , that gives  $\lambda_t = 0$ . Hence  $\alpha_r U'_t(\beta'_t \alpha_r) - \mu + \nu = 0, \forall t \in \mathcal{T}_r$ . We obtain

$$U'_{t_1}(\beta'_{t_1} \alpha_r) = U'_{t_2}(\beta'_{t_2} \alpha_r), \forall t_1, t_2 \in \mathcal{T}_r$$

As neither  $-\mu$  nor  $\nu$  depends on  $t$ , it means that  $U'_t(\rho'_t) = C, \forall t \in \mathcal{T}_r$  where  $C$  is a constant.  $\blacksquare$

We have then the following theorem regarding the maximum point of the function  $\mathcal{U}_r$ .

**Theorem 15** *The function  $\mathcal{U}_r(\alpha_r)$  for each  $r \in \mathcal{R}$  is obtained as follows. Let  $h_t = (U'_t)^{-1}$  and  $h_r = \sum_{t \in \mathcal{T}_r} h_t$ , then  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t \circ h_t \circ h_r^{-1}(\alpha_r), \forall r \in \mathcal{R}$ .*

**Proof:** By Technical Assumption 1, we have  $U''_t(x) < 0, \forall t \in \mathcal{T}_r$  then  $U'_t, \forall t \in \mathcal{T}_r$  are strictly monotonic decreasing functions. So these inverse functions  $h_t, \forall t \in \mathcal{T}_r$  exist. We set  $h_r = \sum_{t \in \mathcal{T}_r} h_t$ . By Lemma 8 and reusing the notation for  $C$ , we have

$$h_r(C) = \beta'_1 \alpha_r + \dots + \beta'_{|\mathcal{T}_r|} \alpha_r = \alpha_r. \quad (7.13)$$

Now, we can have  $\mathcal{U}_r$  expressed by functions  $U_t$ . Indeed, we have from equation (7.13),  $C = h_r^{-1}(\alpha_r)$  and  $\rho'_t = h_t(C) = h_t \circ h_r^{-1}(\alpha_r)$ . We derive  $U_t(\rho'_t) = U_t \circ h_t \circ h_r^{-1}(\alpha_r)$  and  $\sum_{t \in \mathcal{T}_r} U_t(\rho'_t) = \sum_{t \in \mathcal{T}_r} U_t \circ h_t \circ h_r^{-1}(\alpha_r)$ . So, we can consider

$$\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t \circ h_t \circ h_r^{-1}(\alpha_r), \forall r \in \mathcal{R}.$$

Making  $\alpha_r = \alpha_r^*$  we obtain  $h_r(C) = \beta'_1 \alpha_r^* + \dots + \beta'_{|\mathcal{T}_r|} \alpha_r^* = \alpha_r^*$  and the solution for  $P'_r$  is already optimal for  $P_r$ .  $\blacksquare$



### 7.3 Method usage example

In this section we show some results using our model for the problem  $P$ . To solve the model, we use a software library for nonlinear optimization of continuous systems, the Interior Point OPTimizer (IPOPT) that is part of the COIN-OR project. We used the modeling environment AMPL (A Mathematical Programming Language).

We show examples of utility functions and their cumulative representations. For sake of simplicity of presentation, our examples consider a small network with 5 nodes: 2 relays and 3 terminals, as shown in Figure 7.3. Of course our method can be applied to more complex networks.

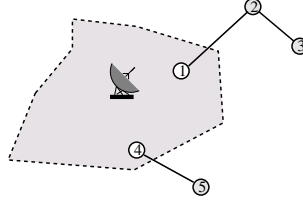


Figure 7.3: Example with 5 nodes.

We consider the utility functions described below. The graphs in Figures 7.4, 7.6 and 7.8 show the cumulative functions, that is using  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t \circ h_t \circ h_r^{-1}(\alpha_r)$ ,  $\forall r \in \mathcal{R}$ . Figures in 7.5, 7.7 and 7.9 consider all utility functions and  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t(\rho_t)$ .

We study the obtained rate  $\rho_t$  varying gain  $g_{b,r}$  (with fixed target interference level  $K = 2$ ). The graphs below show the evolution of the node rates as we increase the gain of the nodes. We consider the same gain for all relay nodes. Recall that  $h_t = (U'_t)^{-1}$ ,  $h_r = \sum_{t \in \mathcal{T}_r} h_t$  and  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t \circ h_t \circ h_r^{-1}(\alpha_r)$ . Consider  $\rho_t > 0$  and  $\rho_t < 1$ .

- $U_t(\rho_t) = c_t \ln(\rho_t)$

$U'_t = \frac{c_t}{\rho_t} = y_t$ ,  $\rho_t = \frac{c_t}{U'_t} = \frac{c_t}{y_t}$ . So,  $h_t(y_t) = \rho_t = \frac{c_t}{y_t}$ . Consider  $x = h_r(y) = \frac{1}{y} \sum_{t \in \mathcal{T}_r} c_t$  that implies  $y = \frac{1}{x} \sum_{t \in \mathcal{T}_r} c_t = h_r^{-1}(x)$ . It gives the cumulative utility function:

$$\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} c_t \ln \left( \frac{1}{\alpha_r} \frac{c_t}{\sum_{t \in \mathcal{T}_r} c_t} \right) = \sum_{t \in \mathcal{T}_r} c_t \ln(\alpha_r) + \sum_{t \in \mathcal{T}_r} c_t \ln \left( \frac{c_t}{\sum_{t \in \mathcal{T}_r} c_t} \right).$$

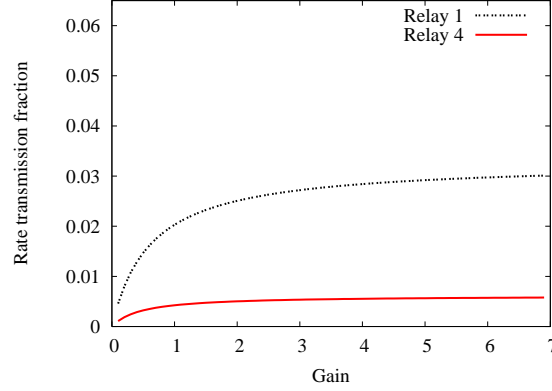


Figure 7.4: Aggregated function  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} c_t \ln(\alpha_r) + \sum_{t \in \mathcal{T}_r} c_t \ln\left(\frac{c_t}{\sum_{t \in \mathcal{T}_r} c_t}\right)$ .

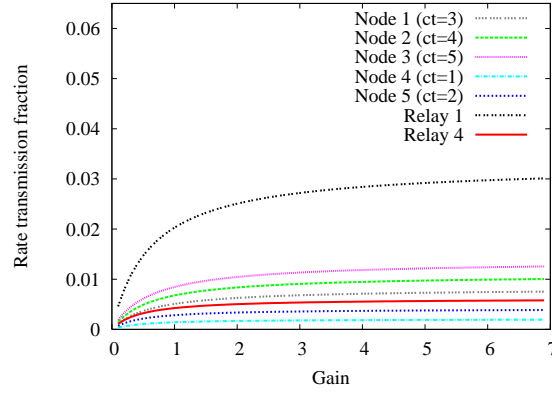


Figure 7.5: Considering multi-hop with  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t(\rho_t)$  and  $U_t(\rho_t) = c_t \ln(\rho_t)$ .

- $U_t(\rho_t) = c_t \sqrt{\rho_t}$

$U'_t(\rho_t) = -\frac{c_t}{2\sqrt{\rho_t}} = y_t$ ,  $\rho_t = \frac{c_t^2}{4y_t^2}$ . So,  $h_t(y_t) = \rho_t = \frac{c_t^2}{4y_t^2}$ . Consider  $x = h_r(y) = \frac{1}{4y^2} \sum_{t \in \mathcal{T}_r} c_t^2$  that implies  $y = \frac{1}{2\sqrt{x}} \sqrt{\sum_{t \in \mathcal{T}_r} c_t^2} = h_r^{-1}(x)$ . It gives the cumulative utility function:

$$\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} c_t \sqrt{\frac{c_t^2}{4\left(\frac{1}{2\sqrt{\alpha_r}} \sqrt{\sum_{t \in \mathcal{T}_r} c_t^2}\right)^2}} = \sqrt{\sum_{t \in \mathcal{T}_r} c_t^2} \sqrt{\alpha_r}.$$

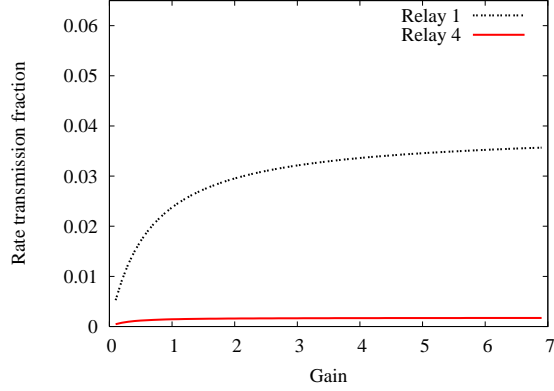


Figure 7.6: Aggregated function  $\mathcal{U}_r(\alpha_r) = \sqrt{\sum_{t \in \mathcal{T}_r} c_t^2} \sqrt{\alpha_r}$ .

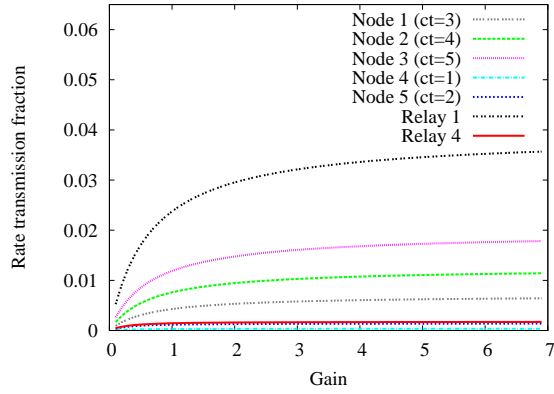


Figure 7.7: Considering multi-hop with  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t(\rho_t)$  and  $U_t(\rho_t) = c_t \sqrt{\rho_t}$ .

- $U_t(\rho_t) = \frac{-c_t}{\rho_t}$

$U'_t(\rho_t) = \frac{c_t}{\rho_t^2} = y_t$ ,  $\rho_t = \sqrt{\frac{c_t}{y_t}}$ . So,  $h_t(y_t) = \rho_t = \sqrt{\frac{c_t}{y_t}}$ . Consider  $x = h_r(y) = \frac{1}{\sqrt{y}} \sum_{t \in \mathcal{T}_r} \sqrt{c_t}$  that implies  $y = \left( \frac{\sum_{t \in \mathcal{T}_r} \sqrt{c_t}}{x} \right)^2 = h_r^{-1}(x)$ . It gives the cumulative utility function:

$$\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} \frac{-c_t}{\sqrt{\left( \frac{\sum_{t \in \mathcal{T}_r} \sqrt{c_t}}{\alpha_r} \right)^2}} = -(\sum_{t \in \mathcal{T}_r} \sqrt{c_t})^2 \frac{1}{\alpha_r}.$$

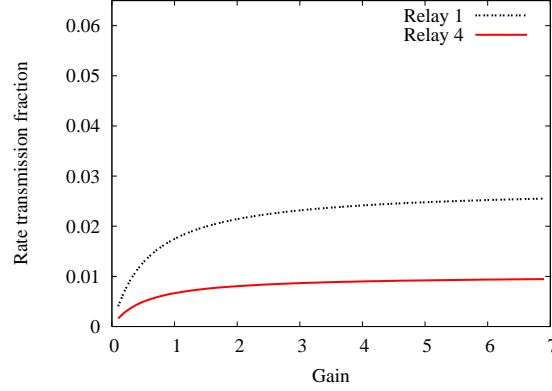


Figure 7.8: Aggregated function  $\mathcal{U}_r(\alpha_r) = -(\sum_{t \in \mathcal{T}_r} \sqrt{c_t})^2 \frac{1}{\alpha_r}$ .

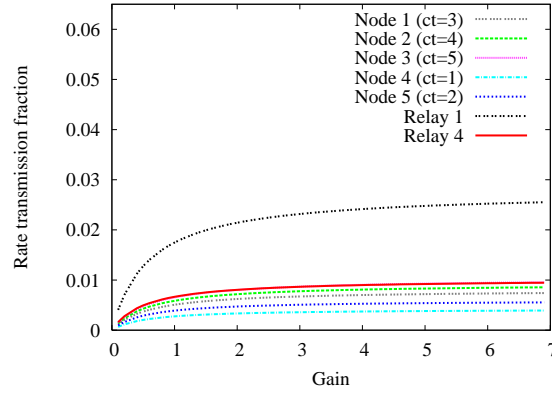


Figure 7.9: Considering multi-hop with  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t(\rho_t)$  and  $U_t(\rho_t) = \frac{-c_t}{\rho_t}$ .

## Generalized fairness utility function

Previously we saw some options of utility function respecting technical assumption 1. An interesting function was proposed by [44]:

$$U_t(\rho_t) = c_t \frac{\rho_t^{1-\kappa}}{1-\kappa} \quad (7.14)$$

This function is interesting because it generalizes all the following important cases of fairness:

- The *globally optimal allocation*: when  $\kappa = 0$ , that is  $\max \sum_{t \in \mathcal{T}_r} \rho_t$ .

- The *harmonic mean fairness*: when  $\kappa = 2$ .
- The *MaxMin fairness*: when  $\kappa \rightarrow \infty$ ,  $\max \min_{t \in \mathcal{T}_r} \rho_t$ .
- The *proportional fairness*: when  $\kappa \rightarrow 1$ ,  $\max \sum_{t \in \mathcal{T}_r} \lim_{\kappa \rightarrow 1} \frac{\rho_t^{1-\kappa}}{1-\kappa} = \sum_{t \in \mathcal{T}_r} \lim_{\kappa \rightarrow 1} \frac{e^{(1-\kappa)\ln(\rho_t)}}{1-\kappa} \sim \sum_{t \in \mathcal{T}_r} \frac{1+(1-\kappa)\ln(\rho_t)}{1-\kappa}$ , that is  $\max \sum_{t \in \mathcal{T}_r} \ln(\rho_t)$ . It is equivalent to  $\max \prod_{t \in \mathcal{T}_r} \rho_t$  that in a convex framework represents the *Nash Equilibrium*.

The previous utility functions are in fact the utility function in (7.14) with a given  $\kappa$  (respectively  $\kappa = 1$ ,  $\kappa = \frac{1}{2}$  and  $\kappa = 2$ ). Considering directly the function  $U_t(\rho_t) = c_t \frac{\rho_t^{1-\kappa}}{1-\kappa}$ , we have:  $U'_t(\rho_t) = c_t \rho_t^{-\kappa} = y_t$ ,  $\rho_t = \frac{y_t}{c_t}^{-\frac{1}{\kappa}}$ . So,  $h_t(y_t) = \rho_t = \frac{c_t}{y_t}^{\frac{1}{\kappa}}$ . Consider  $x = h_r(y) = \sum_{t \in \mathcal{T}_r} h_t = \frac{\sum_{t \in \mathcal{T}_r} c_t^{\frac{1}{\kappa}}}{y^{\frac{1}{\kappa}}}$  that implies  $h_r^{-1}(x) = \left( \frac{\sum_{t \in \mathcal{T}_r} c_t^{\frac{1}{\kappa}}}{x} \right)^{\kappa}$ . Thus  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} \frac{c_t}{1-\kappa} \left( \frac{\frac{c_t^{\frac{1}{\kappa}}}{\sum_{t \in \mathcal{T}_r} c_t^{\frac{1}{\kappa}}}}{x} \right)^{1-\kappa} = \frac{1}{\left( \sum_{t \in \mathcal{T}_r} c_t^{\frac{1}{\kappa}} \right)^{1-\kappa}} \sum_{t \in \mathcal{T}_r} \frac{c_t}{1-\kappa} c_t^{\frac{1-\kappa}{\kappa}} x^{1-\kappa}$ , therefore we can derive a generalized fairness utility function:

$$\mathcal{U}_r(\alpha_r) = \left( \sum_{t \in \mathcal{T}_r} c_t^{\frac{1}{\kappa}} \right)^{\kappa} \frac{x^{1-\kappa}}{1-\kappa}. \quad (7.15)$$

We show an example of our problem using the utility function in (7.14) for all nodes considering different values of  $\kappa$ .

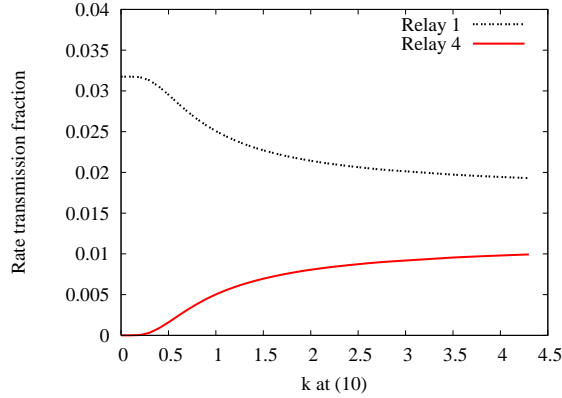


Figure 7.10: Cumulative function  $\mathcal{U}_r(\alpha_r) = \left( \sum_{t \in \mathcal{T}_r} c_t^{\frac{1}{\kappa}} \right)^{\kappa} \frac{x^{1-\kappa}}{1-\kappa}$ .

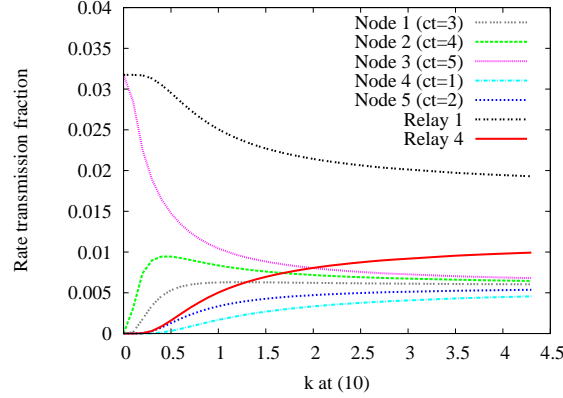


Figure 7.11: Considering multi-hop with  $\mathcal{U}_r(\alpha_r) = \sum_{t \in \mathcal{T}_r} U_t(\rho_t)$  and  $U_t(\rho_t) = c_t \frac{\rho_t^{1-\kappa}}{1-\kappa}$ .

Figures 7.10 and 7.11 show the obtained rate  $\rho_t$  varying  $\kappa$  (with fixed channel gain  $g_{b,r} = 2, \forall r$ ). Figure 7.10 shows the rates considering the cumulative function in (7.15). Figure 7.11 shows the rates of all the nodes using the utility function in (7.14), note that each node has a different value for the constant  $c_t$ . The figures in this section show that we obtain the same graph for the relay nodes in both approaches as we proved.

## 7.4 Conclusion

We have considered in this chapter the transmission rate allocation problem for multi-hop cellular networks in a way to reach an optimal and fair solution. We show that it can be reduced to a single-hop problem by only changing the utility functions.

Reducing multi-hop problems into problems with a unique cell (single-hop) has many advantages for the optimization problem. First we can reuse techniques that were designed basically for the one-cell case [47, 45, 51]. Second, we can identify bottlenecks, and in particular see if a congestion is due to the particular situation of a relay node, or to the specific utility function of the terminals it relays.

Of course the question on implementing distributed algorithms based on those results remains open. We can wonder if a pricing strategy is achievable. Another question is what kind of intermediate capacity restrictions on the second (or more) hop(s) can be added if we want to keep the same good properties.



## Chapter 8

# Conclusion and perspectives

We address a special case of the Round Weighting Problem (RWP) in Radio Mesh Networks as the source nodes are not associated with a specific destination (single-commodity). The RWP is composed of two sub-problems: the routing and the slot assignment problems. The objective is to minimize the overall period of slot activations providing enough capacity to satisfy the routers requirements of bandwidth.

In *Chapter 2*, we present a cross-layer formulation for the problem. Since the number of rounds is exponential, a column generation (CG) algorithm was used to avoid dealing with the complete set of rounds.

We present a multi-objective study for the RWP in *Chapter 3*. The first objective is to balance the load in the routers (*MinMaxLoad*). The second objective is to minimize the communication time (*MinTime*), that is the original objective of the RWP. This multi-objective formulation uses the model of *Chapter 2*.

We show that these two objectives are contradictory as minimizing the time increases the maximum load of the routers. The experimental results of our tests indicate that the relation between the maximum load and the transmission time seems to be convex and piecewise linear. It would be interesting to prove that the curves relating these two specific objectives are indeed convex and piecewise linear.

It will also be interesting to study other objectives, for example minimizing the network utilization, that is important in resource restricted networks (e.g. sensor networks). We make several tests with grid graphs showing that usually the Pareto frontier has one or two points indicating that both objectives are almost the same in grids (with non-sparse demand).

Then, we identify situations that are responsible for the difference between these two objectives considering a node close to the gateway, as shown in *Figure 8.1(a)*. In fact, the demand has to go backwards (to avoid interference with the gateway) and then use the central axes to reach the gateway with minimum time. The fact of going backwards increases the utilization of the network. We



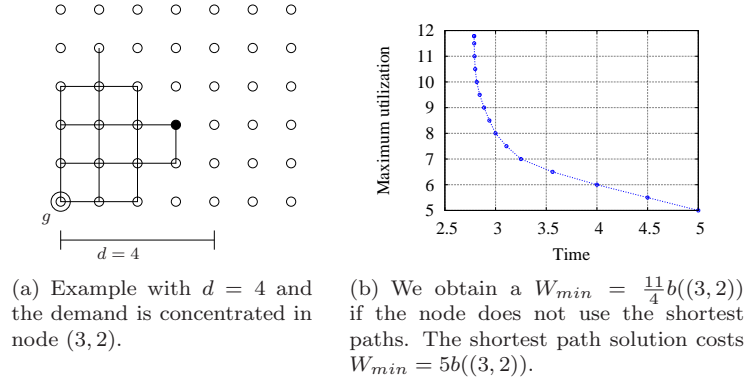


Figure 8.1: Trade-off between minimum network utilization and minimum time.

gives the curve describing the pareto frontier for this node in Figure 8.1(b) (see subsection 5.2.2 for more details).

Some ideas to find lower bounds for RWP derived from a probable bottleneck region for a general graph and using a general binary interference model are presented in *Chapter 4*. We use the model of Chapter 2 to run experiments on networks from the literature, with different numbers of gateways. Our tests show that the optimal solution for the RWP is usually equal to the weight of a min-max clique (see definition in 4) around the gateway, that is our lower bound. However, we give an example that shows that the  $W_{min}$  can be greater than that.

Although the solution for IRWP seems to be the round-up of the solutions for RWP according to our tests, we give a counter-example to that. We explain the usual frequency of these round-up results with dual studies. In fact, the links out of this bottleneck region have slacks of activation. An edge has a slack when it has several possible options to get activated forming a round with edges on the bottleneck region.

Table 8.1 summarizes the known results and our main contributions presented according the following categories:

- **Problem:** 1- The RWP; 2- IRWP (consider integer round weights);
- **Traffic:** 1- *Multi*-commodity flow; 2- *Single*-commodity flow.
- **Demand:** 1- *Non-uniform*; 2- *Uniform* (i.e. every node has the same demand); 3- *Balanced*, when the “partitions” contain the same amount of demands;
- **Interference:** 1- *Binary*, representing any binary interference model; 2- Asymmetrical interference model (see section 2.1); 3-  $d$  (any, even or odd), representing the distance- $d$  interference model.

Table 8.1: Results.

	Problem	Traffic	Demand	Interference	Graph	Complexity	Reference
01	RWP	Multi	Non-uniform	Binary	General	NP-hard	[2]
02	RWP	Single	Non-uniform	Asymmetrical	General	4-approx.	[2]
03	RWP	Multi	Non-uniform	Binary	P-pricing <sup>1</sup>	Polynomial	Corollary 1
04	IRWP	Single	Non-uniform	d Odd	General	2-approx.	Theorem 4
05	IRWP	Single	Non-uniform	d Even	General	3-approx.	Theorem 4
06	RWP	Multi	Non-uniform	d=1	General	Polynomial	Corollary 1
07	IRWP	Single	Non-uniform <sup>2</sup>	d=1	2Connected <sup>3</sup>	Polynomial	Theorem 6
08	RWP	Single	Non-uniform	d Odd	Grid <sup>4</sup>	Polynomial	Theorem 7
09	IRWP	Single	Non-uniform <sup>5</sup>	d Odd	Grid <sup>4</sup>	Polynomial	Theorem 11
10	RWP	Single	$Z_A$ Non-unif.	d Even	Grid <sup>4</sup>	Polynomial	Theorem 9
11	IRWP	Single	Balanced	d Even	Grid <sup>4</sup>	Polynomial	Theorem 12
12	RWP	Single	Uniform	d Any	Grid	Closed Form.	Section 5.3

Recall that the RWP was introduced in [2], where they show that the problem is NP-hard for single-commodity, called gathering (Table 8.1, line 01). Furthermore, they give a 4-approximation algorithm for general topologies and asymmetrical interference model (Table 8.1, line 02) and show that RWP is polynomial for paths. They asked about finding simple efficient algorithms and the complexity of the problem for grids. They also asked about purely combinatorial approximation algorithms that do not use linear programming. In this thesis, we answer these questions considering the distance- $d$  interference model.

We prove that if the pricing-problem (or sub-problem) of the CG algorithm can be solved in polynomial time then RWP can be solved in polynomial time (Table 8.1, line 03). An example of that is the RWP with  $d = 1$  in which the pricing-problem is to find a maximal edge matching that is known polynomial (Table 8.1, line 06).

Methods to obtain lower bounds (inspired by Chapter 4) for general graphs are presented formally in *Chapter 5*. We present several ways to obtain lower bounds considering the distance- $d$  model. For example, we derive a lower bound for any  $d$  using one or many call-cliques (a set of pairwise interfering edges). Our methods are applied to grid graphs (with gateway in the middle or corner).

Upper bounds (given by feasible routings) for the RWP are presented in *Chapter 6*. We use two main routing strategies. Either we route the total demand of a vertex  $v$  by finding interference free paths from  $v$  to  $g$ ; or we combine paths issued from  $v$  with paths issued from other nodes at each iteration of the period. We might have to do different combinations to be able to route all the demands at the end of the period.

<sup>1</sup>Valid for a graph  $G$  having the maximum weighted independent set problem (pricing-problem for RWP) solved in polynomial time for its  $C(G)$ .

<sup>2</sup>If  $\sum_{v \notin K_0} b(v)$  is even and  $b(v) \leq \frac{1}{2} \sum_{v \notin K_0} b(v)$ ,  $\forall v \notin K_0$ .

<sup>3</sup>Valid for a 2-connected graph.

<sup>4</sup>Consider the gateway in the middle.

<sup>5</sup>If  $\sum_{v \notin K_0} b(v)$  is even.

With the first strategy, we prove that the nodes in  $K_0$  (the set of edges in  $G$  at distance at most  $\lceil \frac{d}{2} \rceil$  of the gateway  $g$ ) have optimal routing if they are individually routed using simply a shortest path at each iteration. For the other nodes  $v \notin K_0$ , shortest path routing gives a  $\frac{d+1}{\lceil \frac{d}{2} \rceil}$  approximation, that is a 2-approximation for the case with  $d$  odd (Table 8.1, line 04) and a 3-approximation for the case with  $d$  even (Table 8.1, line 05).

In the case of a grid with  $d$  odd, we are able to find two interference free paths  $(d+1)$ -labeled from  $v$  to  $g$ , proving that the demand  $b(v)$  of a node  $v \notin K_0$  can be optimally satisfied with a weight  $W \leq \frac{d+1}{2}b(v)$  (Table 8.1, line 07).

With the second strategy, we first use two interference free  $(d+1)$ -labeled paths issued from two different vertices. It enables us to find optimal solutions for IRWP in a 2-connected planar graph with  $d = 1$  (Table 8.1, line 08) with some constraints on the demand and in grids with  $d$  odd, when the total demand is even (Table 8.1, line 09).

For grids with  $d$  even, we prove that the flow from the majority of the nodes in the grid ( $v \in Z_A$ ) can be routed using 4 paths at each iteration, with optimal time equals to our lower bound (Table 8.1, line 10). For the other regions  $\neq Z_A$ , we know it is not possible to define 4 paths. A challenge will be to solve completely the case  $d$  even by founding the exact values for the routing of each node; but that appears to be difficult.

Using the second strategy, we also consider four interference free  $(d+1)$ -labeled paths issued from four vertices, each one from different quadrants (called simply regions). Therefore, if the regions are balanced, our lower bound is also optimal for IRWP (Table 8.1, line 11). We provide closed formulae for grid graphs considering uniform demand and any  $d$  (Table 8.1, line 12).

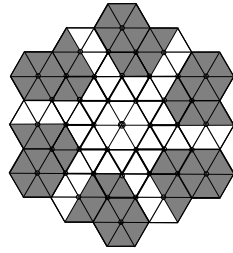
An attractive challenge will be to consider multiple gateways. Our methods can be applied if they are far enough and evenly distributed; but if the gateways are near the problem becomes very difficult.

As future work, we intend to consider an extension of our methods to other graphs (e.g. grid-like graphs). For that, we are interested in graph partitionings that separates the graph in several non-interfering partitions. Thus, we can define interference free paths according the partition containing it. We consider two types of partitioning: the pizza partitioning (see examples in figure 8.2(a) and 8.2(b)) and the block partitioning (see examples in figure 8.2(c) and 8.2(d)).

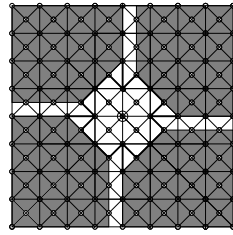
With a pizza partitioning, it is possible to route a flow of  $\pi$  from a combination of  $\pi$  nodes from non-interfering partitions. We can use the same path that guarantees *mono-routing*. If each slice contains a shortest path for all nodes of the slice, we can guarantee also minimum energy routing.

With a block partitioning, we can have more flexibility for the demand distribution as we have more (smaller) interference free regions. It is easy to define blocks for regular graphs because the blocks will consist of a copy of the same subgraph.

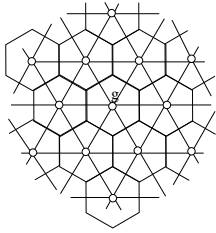
To a partition of  $G$ , we associate a graph  $H$ , where vertices represent induced subgraphs of  $G$  (blocks or slices), two vertices being connected if the corresponding subgraphs are at distance  $< d$ . The problem reduces to finding



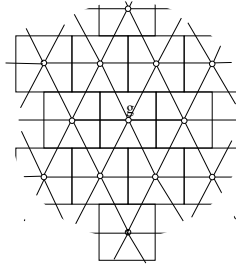
(a) Pizza partitioning with 6 slices.



(b) Pizza partitioning with 4 slices.



(c) Simple example of new graph partitioned in hexagonal blocks of side  $d = 3$  for the graph in figure 8.2(a).



(d) Simple example of new graph partitioned in square blocks of side  $2d$  for the graph in figure 8.2(b).

Figure 8.2: Reduction to the critical region. In these examples  $d = 3$ .

only paths at distance  $> 2$  (in number of edges) in  $H$ , and it is easy if  $H$  is planar (see work in [52]).

For that, we define a  $\pi$ -graph  $G$  as a graph that can be partitioned in induced subgraphs with diameter enough to admit the definition of a new graph  $H$ . The graph  $H$  has to be planar, with the nodes representing the subgraphs of  $G$  and an edge connects two nodes representing two subgraphs at distance  $< d$  (that contain two edges from different regions that are at distance  $< d$ ).

The critical region corresponds to the subgraph that contains  $g$ . The idea is given a lower bound of  $\frac{\lambda}{\pi}B$  (based on the critical region) we can find  $\pi$  paths that can be  $\gamma$ -labeled to send at each iteration a flow of  $\pi$  to the critical region. The larger the number of regions the less uniform can be the traffic because we have more non-interfering regions. The demand distribution conditions guarantees the fact that the interference free combinations exist (e.g. balanced regions).

Motivated by the results of the existence of a limited (bottleneck) region capable to represent the whole network, in *Chapter 7* we consider a variant of the RWP that also deals with bandwidth allocation, but using the interference model with SINR (Signal to Interference plus Noise Ratio) conditions. In this case, we do not attempt to allocate a separate slot to each link. Instead the links are allowed to communicate at the same time; consequently the rate of the communication is limited by the others. The power transmitted by each user is defined to maintain the SINR above a given threshold. The model presented here is valid for UMTS and other systems that tolerate interferences (see Chapter 1.1).

We give sufficient conditions to the multi-hop problem to be reduced to a single-hop problem by only changing the utility functions. These conditions are represented by our description of utility functions. In fact, we present an analytic formula for the aggregated utility function for each relay node, that represents all its terminals. We give results in the case of the generalized fairness criterion of Mo and Walrand [44].

Reducing multi-hop problems into problems with a unique cell (single-hop) has many advantages for the optimization problem. For example, we can reuse techniques that were designed basically for the one-cell case. The question on implementing distributed algorithms based on those results remains open. We can wonder if a pricing strategy is achievable. Another question is what kind of intermediate capacity restrictions on the second (or more) hop(s) can be added if we want to keep the same good properties.

# Bibliography

- [1] W. Wang, X. Li, O. Frieder, Y. Wang, and W. Song, “Efficient interference-aware TDMA link scheduling for static wireless networks,” in *International Conference on Mobile Computing and Networking (Mobicom)*. ACM Press, 2006, pp. 262–273.
- [2] R. Klasing, N. Morales, and S. Pérennes, “On the complexity of bandwidth allocation in radio networks,” *Theoretical Computer Science*, vol. 406, no. 3, pp. 225–239, Oct. 2008. [Online]. Available: <http://dx.doi.org/10.1016/j.tcs.2008.06.048>
- [3] Y. Shi, Y. T. Hou, J. Liu, and S. Kompella, “How to correctly use the protocol interference model for multi-hop wireless networks,” in *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM, 2009, pp. 239–248.
- [4] C. Gomes and G. Huiban, “Multiobjective analysis in wireless mesh networks,” in *15th Annual Meeting of the IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Istanbul, Turkey, 2007.
- [5] C. Gomes, S. Perennes, P. Reyes, and H. Rivano, “Bandwidth allocation in radio grid networks,” in *10èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications (AlgoTel'08)*, Saint-Malo, May 2008.
- [6] C. Gomes and J. Galtier, “Optimal and fair transmission rate allocation problem in multi-hop cellular networks,” in *8th International Conference on AD-HOC Networks & Wireless (ADHOC NOW)*, Murcia, Spain, Sept 2009.
- [7] C. Florens, M. Franceschetti, and R. McEliece, “Lower bounds on data collection time in sensory networks,” *Selected Areas in Communications, IEEE Journal on*, vol. 22, no. 6, pp. 1110–1120, 2004.
- [8] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Pérennes, “Gathering in specific radio networks,” in *Huitièmes Rencontres*

- Francophones sur les Aspects Algorithmiques des Télécommunications (AlgoTel'06)*, Trégastel, France, May 2006, pp. 85–88. [Online]. Available: <http://algotel2006.lip6.fr/>
- [9] V. Bonifaci, P. Korteweg, A. Marchetti-Spaccamela, and L. Stougie, “An approximation algorithm for the wireless gathering problem,” *Operations Research Letters*, vol. 36, no. 5, pp. 605 – 608, 2008. [Online]. Available: <http://www.sciencedirect.com/science/article/B6V8M-4SPJ1PW-1/2/ab4b7362e69a1d9335454aa926c39784>
  - [10] V. Bonifaci, R. Klasing, P. Korteweg, L. Stougie, and A. Marchetti-Spaccamela, *Graphs and Algorithms in Communication Networks*. Springer Monograph Springer-Verlag, 2009, ch. Data Gathering in Wireless Networks.
  - [11] J.-C. Bermond and J. Peters, “Efficient gathering in radio grids with interference,” in *Septièmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications (AlgoTel'05)*, Presqu’île de Giens, May 2005, pp. 103–106.
  - [12] J.-C. Bermond and J. G. Peters, “Optimal gathering in radio grids with interference,” in *manuscript*, 2009.
  - [13] S. Kompella, J. Wieselthier, and A. Ephremides, “Multi-hop routing and scheduling in wireless networks subject to sinr constraints,” in *46th IEEE Conference on Decision and Control*, December 2007, pp. 5690–5695.
  - [14] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Perennes, “Hardness and approximation of gathering in static radio networks,” *Parallel Processing Letters*, vol. 16(2), pp. 165–183, June 2006.
  - [15] R. Mazumdar, G. Sharma, and N. Shroff, “Maximum weighted matching with interference constraints,” *FAWN, Pisa, Italy*, 2006.
  - [16] V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan, “End-to-end packet-scheduling in wireless ad-hoc networks,” in *SODA '04: Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2004, pp. 1021–1030.
  - [17] V. Bonifaci, R. Klasing, P. Korteweg, L. Stougie, and A. Marchetti-Spaccamela, *Graphs and Algorithms in Communication Networks*. Springer Monograph Springer-Verlag. A. Koster and X. Munoz, editors, 2009, ch. Data Gathering in Wireless Networks.
  - [18] P.-J. Wan, “Multiflows in multihop wireless networks,” in *MobiHoc '09: Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*. New York, NY, USA: ACM, 2009, pp. 85–94.

- [19] Y. Wang, W. Wang, X.-Y. Li, and W.-Z. Song, "Interference-aware joint routing and tdma link scheduling for static wireless networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 19, no. 12, pp. 1709–1726, 2008.
- [20] M. Demange, D. de Werra, J. Monnot, and V. T. Paschos, "Weighted node coloring: when stable sets are expensive," in *Proceedings of WG'02, 28th International workshop on graph theoretic concepts in compute science, Lecture Notes in Computer Science 2573*. Springer Verlag, 2002, pp. 113–125.
- [21] M. Grötschel, L. Lovász, and A. Schrijver, "The ellipsoid method and its consequences in combinatorial optimization," *Combinatorica*, vol. 1, no. 2, pp. 169–197, 1981.
- [22] E. R. Scheinerman and D. H. Ullman, *Fractional Graph Theory: A Rational Approach to the Theory of Graphs*. New York: Wiley-Interscience, 1997.
- [23] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 2nd ed. The MIT Press, 2001.
- [24] A. J. Hoffman and J. Kruskal, *Integral Boundary Points of Convex Polyhedra*, ser. Linear Inequalities and Related Systems. Princeton University Press, 1965.
- [25] L. Lovász, *An Algorithmic Theory of Numbers, Graphs and Convexity*, ser. SIAM CBMS-NSF Regional Conference Series in Applied Mathematics. Philadelphia: SIAM, 1986, no. 50.
- [26] S. Mukherjee and H. Viswanathan, "Throughput-range tradeoff of wireless mesh backhaul networks," *Selected Areas in Communications, IEEE Journal on*, vol. 24, no. 3, pp. 593–602, 2006.
- [27] C. Gomes, C. Molle, and P. Reyes, "Optimal design of wireless mesh networks," in *9èmes Journées Doctorales en Informatique et Réseaux (JDIR)*, 2008.
- [28] C. Molle, F. Peix, S. Pérennes, and H. Rivano, "Formulation en coupe/rounds pour le routage dans les réseaux radio maillés," in *10èmes Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel'08)*, May 2008. [Online]. Available: <http://algotel2008.irisa.fr/index.php>
- [29] J. Zhang, H. Wu, Q. Zhang, and B. Li, "Joint routing and scheduling in multi-radio multi-channel multi-hop wireless networks," in *International Conference on Broadband Networks (Broadnets)*, vol. 1. IEEE Press, October 2005, pp. 631–640.
- [30] P. Varbrand, D. Yuan, and P. Bjorklund, "Resource optimization of spatial tdma in ad hoc radio networks: a column generation approach," in *INFOCOM*, vol. 2. IEEE, March-April 2003, pp. 814–824.



- [31] A. Mehrotra and M. A. Trick, “A column generation approach for graph coloring,” *INFORMS Journal on Computing*, vol. 8, no. 4, pp. 344–354, 1996.
- [32] J. A. Kelner and D. A. Spielman, “A randomized polynomial-time simplex algorithm for linear programming,” in *Annual ACM Symposium on Theory of Computing*, 2006, pp. 51–60.
- [33] M. Garey and D. Johnson, *Computers and intractability*. New York: W. H. Freeman and Company, 1979.
- [34] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, second corrected edition ed., ser. Algorithms and Combinatorics. Springer, 1993, vol. 2.
- [35] V. Pareto, *Cours d’économie politique*. Lausanne: F. Rouge, 1896.
- [36] V. Chankong and Y. Haimes, *Multiobjective decision making: Theory and methodology*. New York: Elsevier, 1983.
- [37] K. Miettinen, “Some methods for nonlinear multi-objective optimization,” in *First International Conference on Evolutionary Multi-Criterion Optimization (EMO01)*. London, UK: Springer-Verlag, 2001, pp. 1–20.
- [38] S. Orlowski, M. Pióro, A. Tomaszewski, and R. Wessaly, “SNDlib 1.0-Survivable Network Design Library,” in *Proceedings of the Third International Network Optimization Conference (INOC 2007), Belgium*, April 2007.
- [39] J. Mycielski, “Sur le coloriage des graphes,” in *Colloquim Mathematiques*, 1955, pp. 3:161–162.
- [40] M. Larsen, J. Propp, and D. Ullman, “The fractional chromatic number of a graph and a construction of mycielski,” in *preprint*, 1994.
- [41] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, and P. H. Vance, “Branch-and-price: Column generation for solving huge integer programs,” *Oper. Res.*, vol. 46, no. 3, pp. 316–329, 1998.
- [42] Y.-D. J. Lin and Y.-C. Hsu, “Multihop cellular: A new architecture for wireless communications,” in *INFOCOM*, 2000, pp. 1273–1282. [Online]. Available: [citeseer.ist.psu.edu/lin00multihop.html](http://citeseer.ist.psu.edu/lin00multihop.html).
- [43] F. Fitzek and M. Katz, Eds., *Cooperation in Wireless Networks: Principles and Applications – Real Egoistic Behavior is to Cooperate!*, ser. ISBN 1-4020-4710-X. Springer, Apr. 2006.
- [44] J. Mo and J. Walrand, “Fair end-to-end window-based congestion control,” *IEEE/ACM Transactions on Networking*, vol. 8, pp. 556–567, 2000.

- [45] E. Altman, J. Galtier, and C. Touati, "Fair power and transmission rate control in wireless networks," in *Proc. of IEEE GlobeCom'02*, Taipei, Taiwan, Nov. 2002.
- [46] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Perennes, "Hardness and approximation of gathering in static radio networks," *Parallel Processing Letters*, vol. 16, no. 2, pp. 165–184, June 2006.
- [47] L. Yun and D. Messerschmitt, "Power control for variable QoS on a CDMA channel," in *IEEE MILCOM 1*, 1994, pp. 178–182.
- [48] J. Price and T. Javidi, "Leveraging downlink for regulation of distributed uplink cdma," in *GLOBECOM*, 2006.
- [49] F. P. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *Journal of the Operational Research Society, Statistical Laboratory, University of Cambridge*, vol. 49, pp. 237–252, 1998.
- [50] J.-B. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of Convex Analysis*. Springer-Verlag New York, LLC, 2001.
- [51] J. Galtier, "Adaptive power and transmission rate control in cellular CDMA networks," in *Globecom*, 2006.
- [52] Y. Kobayashi, "Induced disjoint paths problem in a planar digraph," *Discrete Appl. Math.*, vol. 157, no. 15, pp. 3231–3238, 2009.