



Data Gathering in Radio Networks

Patricio Reyes

► To cite this version:

Patricio Reyes. Data Gathering in Radio Networks. Networking and Internet Architecture [cs.NI]. Université Nice Sophia Antipolis, 2009. English. NNT: . tel-00418297

HAL Id: tel-00418297

<https://theses.hal.science/tel-00418297>

Submitted on 17 Sep 2009

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

Patricio Alejandro REYES VALENZUELA

Collecte d'Information dans les Réseaux Radio

Thèse dirigée par **Jean-Claude BERMOND** et **Hervé RIVANO**

et préparée au sein du projet MASCOTTE (I3S(CNRS/UNSA)-INRIA)

soutenue le **5 août 2009**

Jury:

M.	Jean-Claude	Bermond	Directeur de Recherche CNRS	Directeur
Mme.	Luisa	Gargano	Professeur U. Salerno	Rapporteur
M.	Cyril	Gavoille	Professeur U. Bordeaux	Rapporteur
M.	Ralf	Klasing	Chargé de Recherche CNRS	Co-Rapporteur
M.	Stéphanne	Pérennes	Chargé de Recherche CNRS	Examineur
M.	Hervé	Rivano	Chargé de Recherche CNRS	Co-Directeur

Contents

Résumé	vii
Abstract	ix
1 Introduction	1
1.1 Communication Models	3
1.1.1 General Models	4
1.1.2 Models studied in this thesis	5
1.2 Minimum Time Gathering Problem	9
1.2.1 The model	9
1.2.2 Hardness results	10
1.2.3 A 4-approximation	11
1.2.4 Specific topologies	13
1.3 The Round Weighting and Gathering Problem	15
1.3.1 Model	15
1.3.2 Related Work	18
1.3.3 Hardness of Round Weighting	18
1.3.4 Relationship between gathering and round weighting	20
2 Summary of the results	23
2.1 Gathering radio messages in the path	25
2.1.1 Interference	25
2.1.2 Demand	25
2.1.3 Methodology	25

2.1.4	Results	25
2.2	Minimum delay Data Gathering in Radio Networks	27
2.2.1	Interference	27
2.2.2	Demand	27
2.2.3	Methodology	27
2.2.4	Results	27
2.3	Round weighting in the primary node interference model	29
2.3.1	Interference Model	29
2.3.2	Demand	29
2.3.3	Methodology	29
2.3.4	Results	30
2.4	Round Weighting Problem in wireless networks with symmetrical interference	31
2.4.1	Interference	31
2.4.2	Methodology	31
2.4.3	Results	32
2.5	Asymptotic Congestion Wireless Ad-Hoc and Sensor Networks	33
2.5.1	Methodology	33
2.5.2	Results	34
I	Gathering	35
3	Gathering radio messages in the Path	37
3.1	Introduction	37
3.1.1	Background and motivation	37
3.1.2	Modeling aspects	38
3.1.3	Related work	40
3.1.4	Structure of the paper	41
3.2	Preliminaries	41
3.2.1	The model: definitions and notation	41
3.2.2	The Minimum Time Gathering Problem	42
3.3	Lower Bounds and Simple Protocols for the sink as an end-vertex	43

3.3.1	A first lower bound	43
3.3.2	Optimal Protocols for P_n , $n \leq (p+1)d_T + 1$	43
3.3.3	A simple gathering protocol	44
3.3.4	Case $q = d_T - 1$ ($d_I = pd_T + d_T - 1$)	45
3.4	Another lower bound and a 1^+ -approximation	46
3.4.1	Another lower bound	46
3.4.2	A 1^+ -approximation	47
3.5	Incremental Protocols	48
3.5.1	Construction of the Incremental Protocol	48
3.5.2	A gathering protocol when $q = 0$	51
3.5.3	Upper bound for incremental protocol	52
3.5.4	Case $q + 1$ and d_T are relatively prime	54
3.5.5	Simple protocols	56
3.5.6	Existence of optimal incremental protocols	58
3.6	Gathering into an arbitrary vertex of the path	60
3.7	Conclusions	62
4	Minimum delay data gathering in radio networks	63
4.1	Introduction	63
4.1.1	Related Work	64
4.1.2	Our results	64
4.2	Preliminaries	65
4.2.1	Notations	65
4.2.2	Lower bound	66
4.3	Personalized Broadcasting Algorithms	66
4.3.1	Horizontal-Vertical broadcasting	66
4.3.2	$+2$ approximation	68
4.3.3	$+1$ approximation	69
4.4	Distributed Algorithm	75
4.4.1	Distributed Model	75
4.4.2	Basic Description of Distributed Algorithm	76

4.4.3	Formal Description of Distributed Algorithm	76
4.5	Conclusion and Further Works	79
II	Round Weighting	81
5	Round weighting in the primary node interference model	83
5.1	Introduction	83
5.2	Model	84
5.2.1	Interference Model: Matchings	85
5.3	One node with demand	85
5.3.1	Paths	86
5.3.2	Even Cycles	86
5.3.3	Odd Cycles	87
5.3.4	Special Cases	91
5.4	Multiple nodes with demands	101
5.5	Conclusions	103
6	Round Weighting Problem in wireless networks with symmetrical interference	105
6.1	Introduction	105
6.2	Related Work	106
6.3	Problem Statement	107
6.3.1	Interference model	108
6.3.2	Definitions	108
6.4	Lower Bounds	110
6.4.1	Lower Bounds using one call-clique	110
6.4.2	Lower Bounds using many call-cliques	111
6.4.3	Lower Bounds using Critical Edges	115
6.4.4	Relationship with Duality	116
6.5	Application in grids	116
6.5.1	Gateway in the middle: A lower bound	119
6.5.2	Gateway in the middle: An upper bound	121

6.5.3	Gateway in the corner: A lower bound	129
6.5.4	Gateway in the corner: An upper bound	131
6.6	Conclusion and perspectives	143
A	Asymptotic Congestion in Wireless Ad-Hoc and Sensor Networks	145
A.1	Introduction	145
A.2	Number of Paths of Length k and Congestion on the Network	146
A.2.1	Discussion	148
A.3	Analysis Tools	148
A.3.1	Random Graphs	149
A.3.2	Random Matrix Theory	151
A.4	Conclusions and Future Work	154

Résumé

Cette thèse concerne l'étude de l'algorithmique et de la complexité des communications dans les réseaux radio. En particulier, nous nous sommes intéressés au problème de rassembler les informations des sommets d'un réseau radio en un noeud central.

Ce problème est motivé par une question de France Telecom (Orange Labs) “comment amener Internet dans les villages”. Les sommets représentent les maisons des villages qui communiquent entre elles par radio, le but étant d'atteindre une passerelle connectée à Internet par une liaison satellite. Le même problème se rencontre dans les réseaux de senseurs où il s'agit de collecter les informations des senseurs dans une station de base.

Une particularité des réseaux radio est que la distance de transmission est limitée et que les transmissions interfèrent entre elles (phénomènes d'interférences). Nous modélisons ces contraintes en disant que deux sommets (équipements radio) peuvent communiquer s'ils sont à distance au plus d_T et qu'un noeud interfère avec un autre si leur distance est au plus d_I . Les distances sont considérées dans un graphe représentant le réseau. Une étape de communication consistera donc en un ensemble de transmissions compatibles (n'interférant pas).

Notre objectif est de trouver le nombre minimum d'étapes nécessaires pour réaliser un tel rassemblement et de concevoir des algorithmes réalisant ce minimum. Pour des topologies particulières comme le chemin et la grille, nous avons établi des résultats optimaux ou quasi optimaux.

Nous avons aussi considéré le cas systolique (ou continu) où on veut maximiser le débit offert à chaque noeud.

Abstract

This thesis concerns the study of the algorithmic and the complexity of the communications in radio networks. In particular, we were interested in the problem of gathering information from the nodes of a radio network in a central node.

This problem is motivated by a question of France Telecom (Orange Labs) “How to bring Internet in villages”. Nodes represent the houses of the villages which communicate between them by radio, the goal being to reach a gateway connected to Internet by a satellite link. The same problem can be found in sensor networks where the question is to collect data from sensors to a base station.

A peculiarity of radio networks is that the transmission distance is limited and that the transmissions interfere between them (interference phenomena). We model these constraints by saying that two nodes (radio devices) can communicate if they are at distance at most d_T and a node interferes with another one if their distance is at most d_I . The distances are considered in a graph representing the network. Thus, a communication step will consist in a compatible (non interfering) set of transmissions.

Our goal is to find the minimum number of steps needed to achieve such a gathering and design algorithms achieving this minimum. For special topologies such as the path and the grid, we have proposed optimal or near optimal solutions.

We also considered the systolic (or continuous) case where we want to maximize the throughput (bandwidth) offered to each node.

Chapter 1

Introduction

In this thesis, we study problems related to routing informations in a communication network where the links between nodes are established by radio transmissions.

When one thinks of wireless networking, arising examples are often 1-hop applications, for example mobile cellular phones or WiFi access points. In these settings there is a radio link between mobile terminals (mobiles, laptops, PDAs, etc.) and the antenna of a router connected to a backhaul network, most likely wired, which bridges the routers and a backbone infrastructure.

There are however many settings in which a wired backhaul network is not available, because it is technically impossible (e.g. randomly spread sensor networks) or unaffordable (very dense urban networks or underdeveloped countries). This case arises in particular in rural areas where an operator wants to *bring Internet to villages* [BBS05]. This is indeed the motivation of the research environment of this thesis, proposed by FRANCE TELECOM (now Orange Labs). As an Internet provider, they wish a planning to bring high speed Internet to a central antenna in each village and share this connectivity through some fixed multi-hop infrastructure of wireless routers installed in the houses of the village (see fig 1.1).

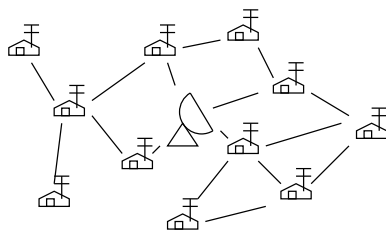


Figure 1.1: A group of clients accessing Internet through a central gateway or antenna.

A broader view of this kind of *fixed ad-hoc* network is known as *wireless mesh networks* (WMNs). In this type of networks, there are basically three kinds of devices: the clients, the routers and the gateways. The clients are often laptops, cell phones and other wireless devices while the routers forward traffic to and from the gateways which connect to the Internet. Wireless mesh networks

can be implemented with various wireless technologies including 802.11, 802.16 and micro-waves technologies.

Using radio transmissions is cheaper than deploying a wired network. Nevertheless, radio transmission causes interference between them: when a device transmits, its radio signal is propagated to a region surrounding this node. This radio signal may interfere with a second radio signal performed at the same time, causing data loss. Thus, if many transmissions are performed concurrently, many of them may fail, degrading the overall communication process.

The study of wireless networks performance has motivated many research works. WMNs deployment in operational situations such as urban areas requires quality of service (QoS) criteria that are challenging to guarantee. Indeed, recent works have pointed out fundamental issues with capacity and scalability. A seminal work by Gupta and Kumar has shown that, under specific topological, physical, traffic and routing assumptions, the transport capacity of wireless networks degrade by a factor of order $\mathcal{O}(1/\sqrt{n})$ with the number of nodes, n [GK00]. Many other studies have then refined and confirmed this result under more generic assumptions such as hierarchical or geometric routing, non uniform emitting power, etc. [DFTT04, MPR06].

These works indicate that a theoretical limit to the transport capacity exists because of interferences and lack of coordination. One consequence has been to motivate studies about the theoretical bounds on transport capacity induced by routing protocols, in order to complement simulation-based evaluations. Generic capacity evaluation frameworks have been proposed using linear programming, providing a measure that is independent of the routing protocols studied in [RTV09, JPPQ03].

One objective was to evaluate the cost, in terms of maximum achievable transport capacity, of a self-organization of the nodes. In particular, it has been highlighted that self-organization protocols can have a negligible impact on the network capacity with a traffic pattern any-to-any. However, efforts have to be done for self-organizations based routing in a many-to-one traffic pattern. This traffic pattern is precisely the most relevant one for WMNs since it corresponds to backhauling usage.

Another particular example of radio networks is known as *sensor networks*. Sensor networks consist in dense wireless networks of devices used usually to collect and disseminate data, often environmental information. Each device, called sensor node, is equipped of an antenna and a processor with a limited battery. Because of these limitations, an important question is how to use the minimum number of transmissions in each task, and avoid interfering transmissions.

This thesis focuses on two fundamental optimization problems arising in WMNs, and sensor networks. MINIMUM TIME GATHERING consists in gathering pieces of data sent by the nodes of the network to a central gateway or *sink* or *base station* while minimizing the total makespan. This problem has a straightforward application in sensor networks. It can also be interpreted as the initialization phase of the network. ROUND WEIGHTING PROBLEM tackles the permanent regime of the network: Messages are periodically created in their sources and then, a permanent communi-

cation between the nodes is sent over time. One seeks to optimize the transport capacity by means of optimizing the transmissions made during a period.

Both problems are detailed in sections 1.2 and 1.3, but we first precise in section 1.1 the models of transmission and interference we use.

1.1 Communication Models

The purpose of communication networks is that of exchanging information between their members. In the communication model it is needed to determine which communications can be performed or not and which of those can be done simultaneously.

In what follows we use the term *call* to refer to a transmission from one device to another. In a call we distinguish the sender and the receiver. We denote a call from the sender u to the receiver v as $u \rightarrow v$. We also denote V as the set of all the devices of the network.

Time plays an important role on wireless communication. If multiple nodes send data at the same time to a single node, a collision occurs. Also data communication requires time and minimizing the communication time or delay time is an important issue. There are two main distinctions in time models: the distinction between continuous and discrete time and between synchronous and asynchronous time.

In a continuous time model, nodes can start communication at any given time. This feature is commonly used to model the case where devices do not have a clock, hence they do not have the notion of time. In a discrete time model time is divided into time-slots. This model is used in cases when devices have a time clock and the devices have notions of the starting and ending time of each time-slot.

Time synchronization is an important issue in wireless networks. We suppose that there is a physical time which is a reference for the clocks of the devices in the network. In the synchronous time model, devices are considered to be synchronous with the physical time. Typically, in centralized models it is common to assume that devices have access to a single clock which indicates physical time.

We consider that time is synchronous and discrete: time is divided into *time-slots* of fixed length. The length of the time-slots corresponds to the time needed to transmit one call. During a time-slot, a number of compatible calls can be performed. When a call is performed, we consider that one message is transmitted from the sender to the receiver.

In wireless mesh networks there are two important notions: the *transmission range* or *connectivity* and the interference. They depend on many factors. One of them is the type of antennas used.

Types of antennas. There are two types of transmitter devices, based on the antenna being either unidirectional or omnidirectional (see [BKK⁺09a]). In the omnidirectional case, the signal is broadcasted in every direction. According to the signal level, we distinguish two regions. The

broadcast region is the area where devices successfully receive the information transmitted by the signal. The *interference region* is the area where the signal level is not enough to receive the information but it is enough to avoid devices to receive any other transmission. In this case, the *broadcast region* is usually modeled as a *ball* centered at the sender and the *interference region* is also modeled as another *ball* centered at the sender with a larger radius. In the unidirectional antennas case, the beam of the antenna can be pointed to a specific direction and then, the broadcast region is modeled as a narrow cone centered at the sender. Interference is also localized in a larger narrow cone, but there is also some interference in the back of the antenna.

In all cases, we will suppose that we are in *half-duplex* mode: a node cannot send and receive during the same slot (contrarily to wired networks when *full-duplex* mode is used.)

Another parameter is the possibility or not of buffering in intermediate nodes

Buffering. In some cases, in particular for sensor networks with limited storage, the gathering problem has been studied when no buffering is allowed at intermediary nodes (see [Gar07], [RS07], [FFM04a]). This implies that, if a message is received by a node at a certain time-step, then it must be sent at the following time-step. This assumption is also known in wired networks as *hot-potato* or *deflection* routing [BHW00, GG93]. In particular, hot-potato routing algorithms are well-suited for optical networks because it is difficult to buffer optical messages [BHW00].

1.1.1 General Models

1.1.1.1 The connectivity model

The connectivity model determines whether a device is able to send a signal to another device or not. Connectivity does not need to be a permanent property: devices can be turned off or a device may be moved far. Similarly, connectivity is not necessarily a symmetric property: device u may be able to transmit to v but conversely, v may not be able to perform a call to u . Connectivity also depends on the power of the transmission (which itself might depend on many factors: distance, weather, obstacles, etc.).

There are different ways to represent the possible transmissions. One, which represents the balls of transmissions, consists in a geometrical model with a transmission radius: a device is able to transmit to all the devices which are within some geometrical distance of it (that distance depending on the transmission power). Here, we will use a model proposed by FRANCE TELECOM (now ORANGE LABS) based on distances in a graph which represents the topology of the network (see section 1.1.2 for precise definitions). An edge between two nodes means that they are *neighbors*. In particular, if there are obstacles (hill, walls, etc) there is no edge between two nodes which might be geometrically near.

1.1.1.2 The interference model

Even if some devices may be connected permanently, it may occur that two calls cannot be performed simultaneously. It is the case of radio networks, where a call may not succeed if other calls are performed at the same time due to interferences. If such a situation occurs, what happens depends on the model. It may happen that, for example, only one of the two transmissions is received, a part of the transmissions is received, or any of the two transmissions is received.

In general, interference is modeled with a *Signal-to-Noise Ratio (SNR)* [Rap96] model to take into account interference from all the neighbors. The SNR model compares the level of the transmission signal with the level of the background noise. In this way, transmissions are received with different intensities. However, in this thesis we only consider the binary interference model described below.

1.1.1.3 Binary interference model

In a binary conflict model, if two calls are performed in the same slot and if they interfere, then both transmissions fail and, consequently, no information is transmitted. We will say that two calls are *compatible* if they do not interfere. The interference model is then obtained by specifying the groups of compatible calls which can be done during a slot. A set of calls is denoted a *round* if the calls are pairwise compatible.

A conflict model consists in defining the set of all the possible rounds. This set is denoted by \mathcal{R} . Notice that the size of \mathcal{R} might be exponential in the number of nodes. For this reason, the set \mathcal{R} will be usually defined by a *rule* determining whether a group of calls is a (valid) round or not.

1.1.1.4 The interference graph

We say that a binary interference model is *one-to-one* if any two calls are either compatible or they interfere and this does not depend on other possible calls being performed at the same time. In this case, the interference model can be entirely described by an *interference graph*.

The interference graph $G_I = (\mathcal{C}, \mathcal{I})$ is the graph whose vertices are the calls and there exists an edge between two calls if they are interfering. Notice that in this case, each round corresponds to an independent set of the conflict graph G_I .

Let us now precise the transmission and interference models we will use in this thesis.

1.1.2 Models studied in this thesis

Our models are based on distance on graphs. In a connected graph $G(V, E)$, we define the distance between two nodes u and v as the length of a shortest directed path between them. The distance is denoted $d_G(u, v)$ or simply $d(u, v)$.

1.1.2.1 Transmission model

The transmission range is represented by an integer d_T . A sender is able to transmit directly to any node at distance at most d_T . In other words, a transmission $u \rightarrow v$ may occur if $d_G(u, v) \leq d_T$. The particular case $d_T = 1$ is interesting as in this case the edges of G represents exactly the possible calls and G is called the *transmission graph*. This model is used in many articles (see, for example, references of table 1.4 in section 1.3.3).

1.1.2.2 Interference model

We present two binary interference models which are considered in this thesis. These models are suitable for different transmission and interference distances.

First of all, recall that we consider half-duplex radio devices. Then, a node of the graph cannot transmit and receive at the same time. In terms of rounds, it implies that a round is always a matching, in other words, a set of edges without common vertices. In many articles on radio networks only this constraint is taken into account and this model is often called *primary node interference* or *node-exclusive interference* model [MSS06].

Asymmetrical Interference model

We consider the so-called (d_I, d_T) -interference model [BGK⁺06b, BKMS08a, BKK⁺09a, Gar07]. The parameter $d_T \in \mathbb{N}$ denotes the transmission distance and $d_I \in \mathbb{N}$, with $d_I \geq d_T$, denotes the interference distance.

In terms of avoiding collisions, two calls $u \rightarrow v$ and $u' \rightarrow v'$ are interfering if either $d_G(u, v') \leq d_I$ or $d_G(u', v) \leq d_I$; otherwise the calls are *compatible*. Recall that $d_G(u, v)$ corresponds to the distance in the graph G between the nodes u and v . An example of compatible and interfering calls with $d_I = d_T = 1$ and $d_I = 2, d_T = 1$ are shown in figure 1.2.

We call this model an *asymmetrical* interference model because the calls interfering with $u \rightarrow v$ are not the same calls which interfere with $v \rightarrow u$.

Symmetrical Interference Model

Notice that if device u calls device v , it is desirable that v had a way to let u know that the transmission has been successful (*acknowledgment* or *ACK*). Such feedback is performed as a transmission from v to u . Furthermore, that is the model considered in the protocol 802.11 and is named in some papers as the *802.11 interference model* [Wan09a]. For this reason most applications and consequently models assume that transmission as well as interference are symmetrical. Therefore, we also use a symmetrical version of the interference model.

In this version, two calls interfere if one call has one of its end vertices in the interference range of some end vertex of the other call. More precisely, a call between u and v interfere with a call

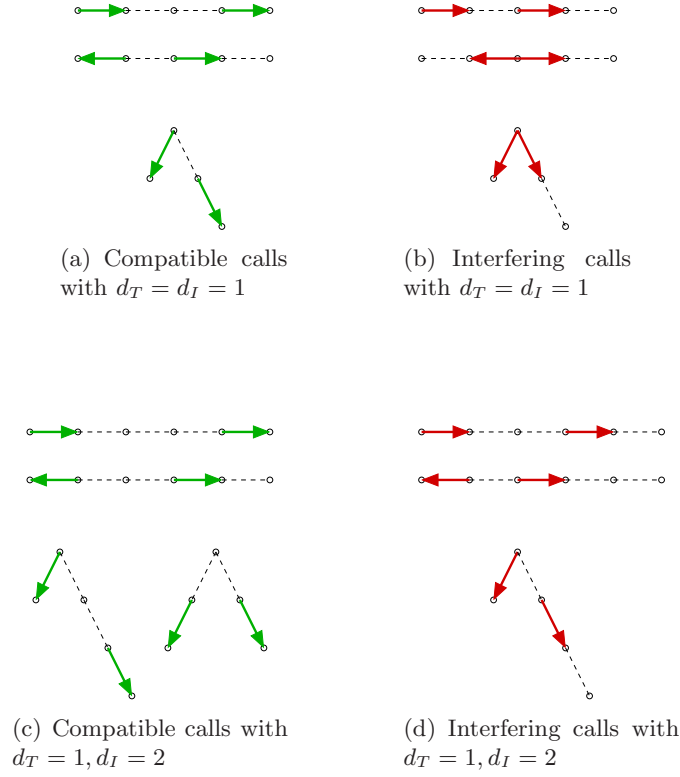


Figure 1.2: Examples of compatible and interfering calls for the asymmetrical interference model.

between u' and v' if $\min_{x \in \{u,v\}, y \in \{u',v'\}} d_G(x, y) < d_I^s$, for $d_I^s \in \mathbb{N}$ and $d_I^s \geq d_T$. Notice that this interference model makes no difference between the sender and the receiver of a call. So, calls in this interference model are said to be *symmetrical* and a call between u and v is denoted $u \leftrightarrow v$. The particular case $d_I^s = d_T = 1$ is nothing else than the *primary node interference model*, a round being a matching. In the case $d_I^s = 2$ and $d_T = 1$ we get the so called *distance-2 interference model* [KMPS04, BKK⁺09a, Wan09a, WWLS08]. In this case, a round is an induced matching.

The conflict graph in the case $d_I^s = d_T = 1$ corresponds to 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_I^s and $d_T = 1$, the conflict graph is the d_I^s -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).

Unlike asymmetrical interference models, the interfering calls produced by $u \rightarrow v$ are the same as $v \rightarrow u$.

1.1.2.3 Relationship between Asymmetrical and Symmetrical models

As depicted in Figure 1.3, we observe the following relation between both models: a round in the symmetrical ($d_I^s = 2, d_T = 1$)–interference model is a round in the asymmetrical ($d_I = 1, d_T = 1$)–

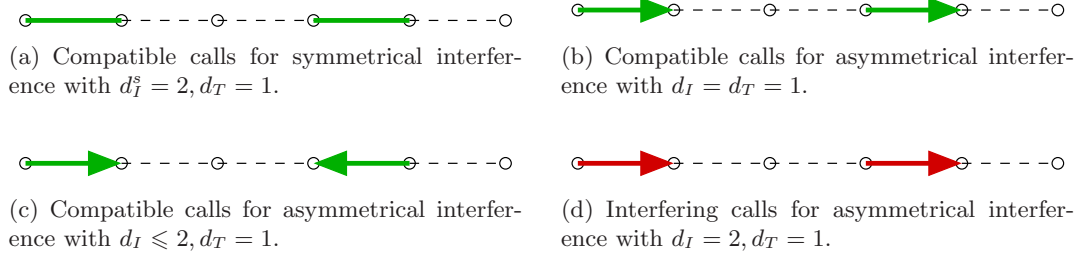


Figure 1.3: Comparison between symmetrical and asymmetrical interference for different values of d_I .

interference model. Conversely, each round in the asymmetrical $(d_I = 2, d_T = 1)$ -interference model is a round in the symmetrical $(d_T = 1, d_I^s = 2)$ -interference model. In general, a round in the (d, d_T) -symmetrical model is a round in the $(d - 1, d_T)$ -asymmetrical model; conversely, a round in the (d, d_T) -asymmetrical model is a round in the (d, d_T) -symmetrical model.

1.1.2.4 Topologies

In this thesis we will focus on the path which corresponds in particular to directional antennas. We will also consider grids as they model well cities. Furthermore, grids (in particular hexagonal grids) are a good approximation of balls and many problems are more tractable on grids than in a geometrical model enabling us to obtain exact results or good approximations.

1.2 Minimum Time Gathering Problem

Recall that the problem asked originally by FRANCE TELECOM consisted of sending (or receiving) data to (from) a specific node (connected via a high speed access to Internet) called *base station*. In the case where the node wants to send information to the base station, the problem is known as *gathering* or *data collection*. This problem appears also in sensor networks where the sensors have to send some data to a base station. In the discrete version case, where there is only one sending, it is important to minimize the total delay of transmission or the minimum completion time (*makespan*) of the gathering protocol. In our model it corresponds to minimize the total number of rounds. We will call this problem MINIMUM TIME GATHERING and denote it MTG. We will see in the next section another version of the gathering problem (the *continuous* one) where the criteria is to reserve enough bandwidth to the nodes. Another criterion of optimization considered in [BKMS08b, BKK⁺09a] is the delay between the sending of a message and its reception to the base station, called *flow time* of the message. The objective consists in minimizing the maximum *flow time* over all the messages.

The problem where the base station (central node) wants to send information to all the nodes of the network is the inverse of the MTG problem and it is called *personalized broadcasting*. It differs from the classical broadcasting problem where a source sends the same message to all the nodes; here the source sends different messages to different nodes. We will see after that these two problems are equivalent (see section 2.2 and chapter 4). The solution of one gives the solution to the other.

1.2.1 The model

The base station where all the messages are gathered will be denoted BS. In some cases, it is called a *sink* [KMP08, BGK⁺06b, BKK⁺09a] or *gateway* [BP05] and also source, when one considers the personalized broadcasting problem [BNRR09a].

We will denote by $w(u)$ the number of unitary messages a node has to transmit to the base station; the *unity* being the amount of information which can be transmitted during a call (that depends on the technology). The interference model will be either (d_I, d_T) -asymmetrical or (d_I^s, d_T) -symmetrical model (see section 1.1.2).

A gathering protocol consists of a sequence of rounds, such that the rounds in the protocol are executed according to the order of the sequence and exactly $w(u)$ messages will be gathered from node $u \in V$ into the base station BS.

The goal of the MINIMUM TIME GATHERING problem consists in finding a gathering protocol which requires the minimum number of rounds.

An example of a gathering protocol is presented in figure 1.4. Here, $w(u) = 1$ for any u , $1 \leq u \leq 6$. We consider the asymmetric interference model with $d_I = 2$, $d_T = 1$. This protocol is optimal and the number of rounds in the protocol is 18. Indeed the 4 calls $1 \rightarrow 0$, $2 \rightarrow 1$, $3 \rightarrow 2$, $4 \rightarrow 3$ are

interfering and so at most one of them can be done during one round. Therefore, we need at least 4 calls to transmit the message from each node 4, 5, 6 and i calls for the message of node i , $1 \leq i \leq 3$. All together we need at least 18 calls.

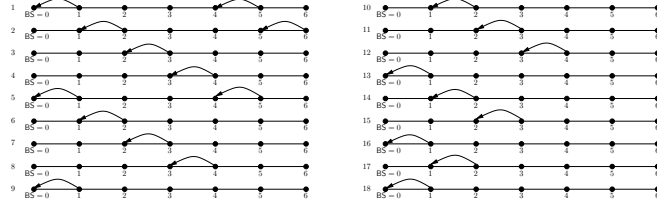


Figure 1.4: A protocol for gathering on a path with 7 nodes when $d_I = 2, d_T = 1$. Nodes from 1 to 6 have one message each. The protocol gathers all the messages into BS in 18 time-steps.

Uniform case

Each node of the graph (except the base station) may have messages to be gathered. When each node of the graph has the same number of messages $w > 0$ to be gathered, the demand is called *uniform*. In particular, if $w = 1$ (i.e, each node has one message to be gathered), the demand is called *unitary*. In all the other cases, the demand will be denoted *non-uniform* or *general* demand. As we will see later, the gathering problem with uniform demand has been studied in different cases and, depending on the case, good approximations may be given. However, for general demand, the problem becomes more difficult even for obtaining good bounds.

1.2.2 Hardness results

In the following we discuss the hardness of the MTG problem and then we present some known approximation algorithms.

It has been proved in [BGK⁺06b] that MTG with the (d_I, d_T) -interference model is NP-HARD for arbitrary graphs. The problem remains NP-HARD even for uniform demand (See [BGK⁺06b] for the case $d_I > d_T$ and [Kor08] for the case $d_I = d_T$). An interesting open question is to find the complexity of MTG for general graphs when buffering is not allowed in intermediary nodes [BGR08].

In the following, we present the idea of the proof of NP-hardness for the MTG based on the reduction of the well-known NP-HARD Problem of determining the chromatic number of a graph (see [Col02] and [BKK⁺09a]). The chromatic number of a graph is the minimum number of colors needed to color all the nodes of the graph so that no two adjacent nodes have the same color.

Theorem 1.1 ([BKK⁺09a]) *The problem MTG with the (d_I, d_T) -interference model is NP-HARD for any $d_I \geq d_T$.*

Idea: We present the reduction only for the case $d_I = 2$ and $d_T = 1$. Let G be the graph which is the instance of the CHROMATIC NUMBER. We will construct a graph G' such that G' will be an instance of MTG. Let $V(G) = \{v_1, \dots, v_n\}$ be the nodes of the graph G . Let H the graph consisting in the isolated vertices u_1, \dots, u_n .

Let $V(G') = V(H) \cup V(G) \cup \text{BS}$ be the nodes of the graph G' . The edges of G' are defined as follows: There is an edge between u_i and v_i , for each i . There is an edge between each node of G and BS. Finally, all the edges in G are also edges in G' . In other words, $E(G') = E(G) \cup \{(u_i, v_i) \mid i = 1, \dots, n\} \cup \{(v_j, \text{BS}) \mid j = 1, \dots, n\}$.

Now, we consider the problem of MTG over the graph G' where each node of H has one message to be gathered into BS. Recall that the interference model considered is $d_I = 2$ and $d_T = 1$. For this case, note that two calls $u_i \rightarrow v_i$ and $u_j \rightarrow v_j$ can be performed simultaneously iff there is no edge in G between v_i and v_j . Moreover, note that if a call occurs between a node in G and BS, then no other call can occur at the same time-step. Both remarks lead us to conclude that any gathering protocol collects all the messages into BS in at least $\chi(G) + n$ time-steps, where $\chi(G)$ is the chromatic number of G . The idea is depicted in figure 1.5. \square

1.2.3 A 4-approximation

A protocol for general graphs with an approximation factor of at most 4 is presented in [BGK⁺06b]. An extension of the problem where messages can be released over time is considered in [BKMS08a] and a 4-approximation algorithm is also shown. Moreover, in [BKMS08a] the authors study the quality of this approximation when a *shortest path following algorithm* is used. A shortest path following algorithm is an algorithm where each message is sent over some shortest path towards the sink. They have shown that, for example, for $d_I = 2$, $d_T = 1$, the best approximation one can obtain with a shortest path following algorithm has a ratio tending to 4. The idea of the proof is depicted in figure 1.6. The figure represents a graph such that there are m messages to gather into BS. These messages are located in nodes u_1, \dots, u_m . Note that, if the algorithm routes the messages via the shortest paths (i.e., passing through node u), then any protocol needs at least $4m$ time-steps. However, there is a solution such that no message is routed via the node u . It suffices to route any message from u_i to the BS via the node v_i . Thus, in 4 steps, all the messages arrive to nodes v_i . Then, we send the message stored in v_i to BS in m rounds. So, messages can be gathered in at most $m + 4$ time-steps. The approximation ratio is $\frac{4m}{m+4} = 4 - \frac{16}{m+4}$ which tends to 4 when m tends to infinity.

For the case $d_I = d_T$ the best approximation ratio for any shortest path following algorithm is 3 [Kor08].

Table 1.1: Complexity results for general graphs

Interference	Demand	buffer	problem	topology	result	reference
asymmetric $d_I = d_T$	general	buffer	MGT	general	NP-HARD	[Kor08]
asymmetric d_I, d_T	general	buffer	MGT	general	NP-HARD,	[BGK ⁺ 06b]
asymmetric $d_I = d_T$	uniform	–	MGT	general	NP-HARD	[Kor08],
asymmetric $d_I > d_T$	uniform	–	MGT	general	NP-HARD	[BGK ⁺ 06b]
symmetric $d_I^s = 2, d_T = 1$	general	–	MGT	general	NP-HARD	[BKK ⁺ 09a]
symmetric $d_I^s = d_T = 1$	uniform	–	MGT	general	POLY	[GR06a]
asymmetric d_I, d_T	general	no-buffer	MGT	general	?? (open)	[GR06a]

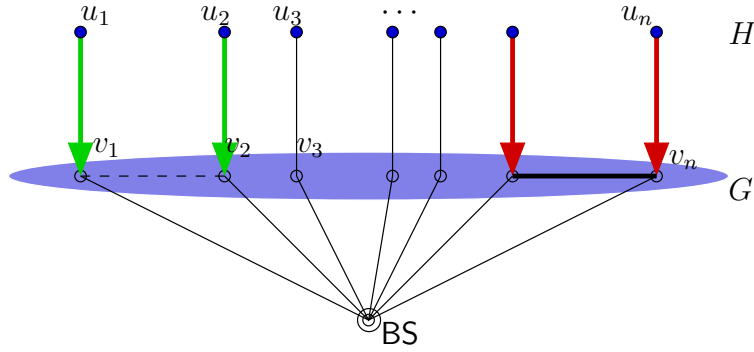


Figure 1.5: Reduction from the chromatic number problem. Instance for MGT where each node of H has one message to be gathered into BS. The (d_T, d_I) -interference model considered is $d_I = 2$ and $d_T = 1$. Green arrows indicate compatible calls, red arrows indicate two calls that interfere between them.

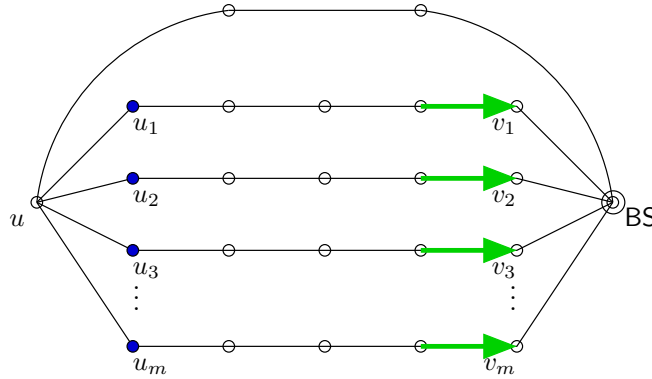


Figure 1.6: Any shortest-path following algorithm is no better than a 4-approximation. In this case, $d_I = 2, d_T = 1$

1.2.4 Specific topologies

The uniform case has been considered for specific topologies, mainly for the asymmetrical interference model. The unitary case (each node has one message to transmit to the base station) in the path with $d_T = 1$ and arbitrary d_I is studied in [BCY06] and in [BCY09]. The authors give protocols and lower bounds on the minimum number of rounds when BS is either at one end or at the center of the path. The protocols are shown to be optimal for any d_I in the first case, and for $1 \leq d_I \leq 4$, in the second case. In chapter 3, we generalize these results for arbitrary d_T using a different approach, proving 1-approximation and giving good protocols when the BS is at the end. For the two-dimensional square grid, the problem with unitary demand is studied in [BP05] for $d_T = 1$ and when the BS is placed in the center of the grid. In [BP05], the algorithms attain the optimal and near-optimal solution for d_I odd and d_I even respectively. In [BP09] optimal solutions are given for d_I even and also for hexagonal grids.

For trees, in the case $d_I = d_T = 1$ with unitary demand, an optimal solution is presented in [BY08] when buffering is allowed. The method consists in classifying the subtrees and applying specific sub-algorithms for each type. The same case but without buffering in intermediary nodes is solved in [BGR08]. In this case, the method is different and consists in studying the related problem of collision free labelings.

In [BGK⁺06a], a 1-approximation is given for stars.

The symmetric model interference has been studied mainly in the case $d_I^s = d_T = 1$ (primary node interference model). In [FFM04a] exact formulas are given for paths and for general demands. That has been extended to trees in [GR06a, GR09]. In [GR06a, GR09] a polynomial algorithm is given for any graph but with unitary demand. A related gathering problem has been studied in [BNRR09a]. In this case, the BS is placed in one of the corners of the grid and the interference is symmetrical with $d_I^s = d_T = 1$. For this settings, a +1-approximation algorithm is presented as well as a distributed +2-approximation version. These results are presented in detail in chapter 4.

Table 1.2: Algorithms for the MTG Problem

Interference	Demand	buffer	topology	result	reference
asymmetric d_I, d_T	general	buffer	general	4-approx	[BGK ⁺ 06b]
asymmetric $d_T = 1$, any d_I	uniform	buffer	path BS end	optimal	[BCY06],[BCY09]
$d_T = 1, 1 \leq d_I \leq 4$	uniform	buffer	path BS center	optimal	
asymmetric $d_T = 1, d_I$ odd	uniform	buffer	grid	optimal	[BP05, BP09]
asymmetric $d_T = 1, d_I$ even	uniform	buffer	grid	optimal	
asymmetric $d_I = d_T = 1$	uniform	buffer	tree	optimal	[BY08]
asymmetric $d_I = d_T = 1$	uniform	no-buffer	tree	optimal	[BGR08]
asymmetric d_I, d_T	uniform	buffer	path	1-approx	chapter 3
asymmetric d_I, d_T	uniform	buffer	grid	1-approx	[BGK ⁺ 06a] [BGK ⁺ 06a]
asymmetric d_I, d_T	uniform	buffer	stars	1-approx	
symmetric $d_I^s = d_T = 1$	general	no-buffer	path	optimal	[FFM04a]
symmetric $d_I^s = d_T = 1$	general	no-buffer	tree	optimal	[GR09]
symmetric $d_I^s = d_T = 1$	uniform	no-buffer	general	optimal	[GR09]
symmetric $d_I^s = d_T = 1$	general	no-buffer	grid	+1-approx	[BNRR09a]

1.3 The Round Weighting and Gathering Problem

The MINIMUM TIME GATHERING focus on the problem of routing a given demand from each node to the base station. Thus, the goal is completed when the last message reaches the BS and the protocol ends. This may not be the case for certain applications, where a permanent communication between the nodes is sent over time, i.e., messages are permanently created in their sources. This introduces a new type of model where traffic has to be allocated permanently, in a periodic way.

When the network is in a steady state, the relevant time-scale to focus on is the period. In our case, such a period is a sequence of round activations providing enough capacity for routing the flow of messages.

In other words, there is enough capacity for each node v to send $b(v)$ messages, each message to be forwarded toward the sink, and the sink to receive $\sum_v b(v)$ messages at each period. In particular, a given message may need several periods to reach its destination and the data rate provided to node v equals $\frac{b(v)}{W}$, where W is the length of the period. A fair optimization of the transport capacity is therefore equivalent to minimizing W such that there is enough capacity to send a given demand at each node.

In this thesis, we address the relaxation of this problem in which a solution is no longer a sequence of rounds, but a continuous (no longer discrete) weight function $w : \mathcal{R} \rightarrow \mathbb{R}^+$ on the set of rounds \mathcal{R} . In this case, $w(R)$ represents how often round R should be activated in such a way that $b(v)$ messages are sent by each node v during each period, and $W = \sum_{R \in \mathcal{R}} w(R)$.

The ROUND WEIGHTING PROBLEM (RWP) has been formalized in [KMP08], that jointly considers the *multi-commodity flow problem* and the *weighted fractional coloring problem*. In fact, they consider the problem for general demands $b(u, v)$ for any ordered pair (u, v) . Here we restrict ourselves to the gathering instances.

Now, we formally introduce the round weighting problem.

PROBLEM:	Round weighting for gathering instances
INPUT:	a graph $G = (V, E)$, a base station $BS \in V$, a set of possible rounds $\mathcal{R} \subset 2^E$ (whose size may be exponential), and a flow demand function $b : V \rightarrow \mathbb{R}^+$, corresponding to the demand from $v \in V$ to the BS.
SOLUTION:	A round weight function w defined over \mathcal{R} that satisfies the traffic demand b .
GOAL:	Minimize the overall weight of w , i.e. $W = \sum_{R \in \mathcal{R}} w(R)$.

1.3.1 Model

In the following we present the round weighting problem as a linear program and we study it through its dual for the case of gathering.

We consider a traffic gathering where the demand $b : V \rightarrow \mathbb{R}^+$ represents the flow $b(v)$ needed to be sent from v to the base station BS.

We say that a round weight function w satisfies the traffic demand b if there exists a flow ϕ such that

- it satisfies the traffic demand b

$$(\forall v \in V) \quad \sum_{P \in P_{v, \text{BS}}} \phi(P) \geq b(v),$$

where $P_{v, \text{BS}}$ denotes the set of paths between v and BS, and

- it respects the capacity c_w induced by w :

$$(\forall e \in E) \quad \sum_{P \in \mathcal{P}: e \in P} \phi(P) \leq c_w(e) = \sum_{R \in \mathcal{R}: e \in R} w(R).$$

Summarizing, the round weighting problem can be written as:

$$\begin{aligned} \min_{w, \phi} \quad & \sum_{R \in \mathcal{R}} w(R) \\ & - \sum_{P \in \mathcal{P}_{v, \text{BS}}} \phi(P) \leq -b(v) \quad (\forall v \in V) \end{aligned} \quad (1.1)$$

$$\sum_{P \in \mathcal{P}: e \in P} \phi(P) - \sum_{R \in \mathcal{R}: e \in R} w(R) \leq 0 \quad (\forall e \in E) \quad (1.2)$$

$$w, \phi \geq 0$$

Now, we derive the dual using positive multipliers. Let λ_v be the multiplier associated to (1.1) and $l(e)$ to that of (1.2). Thus, the dual formulation consists in:

$$\begin{aligned} \max_{\lambda, l} \quad & \sum_{v \in V} \lambda_v b(v) \\ & \sum_{e \in P} l(e) \geq \lambda_v \quad (\forall v \in V)(\forall P \in \mathcal{P}_{v, \text{BS}}) \end{aligned} \quad (1.3)$$

$$\sum_{e \in R} l(e) \leq 1 \quad (\forall R \in \mathcal{R}) \quad (1.4)$$

$$\lambda, l \geq 0$$

Notice that $l(e)$ can be seen as the *length* of a call e . Therefore $l(e)$ induces a metric $d_l(u, v)$ which corresponds to $d_l(u, v) = \min_{P \in \mathcal{P}_{u, v}} l(e)$, the path with shortest *length* (in terms of l) between two

1.3.2 Related Work

The use of a duality in problems of routing flow in wireless ad-hoc networks has been also presented in [CLD05]. However, the approach proposed is different. In particular, they propose a lagrangian relaxation and then, they prove the convergence of their method towards the optimal solution.

It has been shown in [KMP08] that the RWP is NP-HARD in general (reduction in the case $d_I = 2, d_T = 1$ to the fractional coloring problem). They also showed that for gathering instances, RWP remains NP-HARD even when restricted to unitary demands. The authors present a 4-approximation algorithm in this case. Finally, they conjecture that it is always possible to get a PTAS for gathering instances.

The complexity of the problem has been also studied for specific topologies. For paths and trees, under gathering instances, RWP can be solved in polynomial time on the length of the path. (The idea is presented in [KMP08]). For general instances in trees, the problem admits a FPTAS. For grids, RWP remains NP-HARD, but it admits a PTAS [KMP08]. However, the PTAS proposed is purely theoretical. Moreover they posed the following question: Is it possible to get simple and efficient algorithms for the 2-dimensional grids? Is it possible to give purely combinatorial algorithms that would not use linear programming? In chapter 6 we answer these questions by presenting optimal solutions for d_I, d_T arbitrary and uniform demands. This work is a generalization of [GPRR08] where the case $d_I^s = 2, d_T = 1$ is solved.

A related problem consists in finding the maximum round (in terms of number of calls) satisfying the distance-2 interference model. This problem is also called *maximum induced matching* [SV82] and *maximum distance 2 matching* (D2EMIS) [BBK⁺04]. D2EMIS is known to be APX-COMPLETE for regular graphs [Mah00], but admits a PTAS for disk graphs [BBK⁺04]. The problem is generalized for arbitrary d_I in [MSS06] by considering different weights to the edges. The problem is called the *maximum weighted K-valid matching problem* (MWKVMP) for $K = d_I^s$ in our case. The MWKVMP is polynomial when $d_I^s = 1$. However, if $d_I^s \geq 2$ then the problem is NP-HARD and not approximable. The authors also provide a PTAS for unit disk graphs.

Interference	Demand	problem	topology	result	reference
asymmetric d_I, d_T	general	RWP	general	NP-HARD	[KMP08]
	uniform	RWP-GATHERING		NP-HARD	
	uniform	RWP-GATHERING		4-approx	

Table 1.3: Complexity Results for RWP in general topologies. RWP-GATHERING indicates the RWP restricted to gathering instances.

1.3.3 Hardness of Round Weighting

As seen in section 1.3.2 the ROUND WEIGHTING PROBLEM is NP-HARD. Now, we show that the RWP remains NP-HARD even for gathering instances.

Interference	Demand	problem	topology	result	reference
asymmetric d_I, d_T	general	RWP-GATHERING RWP-GATHERING	path tree	POLY FPTAS	[KMP08]
asymmetric d_I, d_T	general	RWP RWP	grid grid	NP-HARD PTAS	[BGK ⁺ 06c]
asymmetric d_I, d_T	general	RWP-GATHERING	grid	NP-HARD ?? (open)	

Table 1.4: Approximation algorithms for for RWP in specific topologies. RWP-GATHERING indicates the RWP restricted to gathering instances.

Theorem 1.2 ([KMP08]) *The RWP in graphs with asymmetrical interference $d_I = 2$, $d_T = 1$ is NP-HARD even for gathering instances.*

Idea: The reduction is from MAXIMUM INDEPENDENT SET. Let G the graph which is an instance of MAXIMUM INDEPENDENT SET, we will construct a new graph H such that H will be an instance of RWP. Let α the size of the maximum independent set in $G = (V, E)$. Since checking whether $\alpha = 1$ is polynomial, we will assume that $\alpha \geq 2$. Let $G' = (V', E')$ a copy of G . Now we define the graph H . The set of nodes in H is the set $V \cup V' \cup \{s, \text{BS}\}$. The edges of H are defined as follows: the edges of G and G' are also edges in H . There exists one edge between s and each node of G . There exists one edge between each node of G' and BS . There is an edge between each node v in G and its copy v' in G' . In other words, the set of edges in H is the set $\{(s, v) \mid v \in V\} \cup \{(v, v') \mid v \in V, v' \text{ copy of } v \text{ in } G'\} \cup \{(v', \text{BS}) \mid v' \in V'\} \cup E \cup E'$. The graph H is depicted in figure 1.8. We define the associated demand function as $f(s, \text{BS}) = b > 0$ and 0 otherwise.

Since the interference parameters are $d_T = 1, d_I = 2$ the construction assures that two calls (u, u') and (v, v') , $u, v \in V$, $u', v' \in V'$ are compatible if and only if u and v are independent nodes in G . Note also that rounds containing calls of type (v, v') cannot contain calls of type (s, v) or (v', BS) . Moreover, if a call occurs between s and a node v in V , the only possible compatible call corresponds to an edge (v', BS) such that (v, v') is not an edge in H . Let w^* the optimal weight function and W^* the corresponding weight solution. Let R^* be the largest round composed by calls of type (v, v') and such that $w^*(R^*) > 0$. Notice that this round induces an independent set in G . Indeed, the number of calls in the round corresponds to the size, denoted α^* , of this independent set. It follows that $W^* \geq \frac{b}{\alpha^*} + b$.

Now, let us propose a weight function w which is solution for the round weighting problem. We define $I = (v_i)_{i=0}^{\alpha-1}$ as a maximum independent set of G . The weight function proposed is $w(\{(s, v_i), (v'_{(i+1) \bmod \alpha}, \text{BS})\}) = \frac{b}{\alpha}$, for $i = 0, \dots, \alpha - 1$; $w(\{(v_i, v'_i)_{i=0}^{\alpha-1}\}) = \frac{b}{\alpha}$; and 0 for all the other rounds. The overall weight W of this solution satisfies $W^* \leq W = \frac{b}{\alpha} + b$. From both, lower and upper bound of W^* combined, we conclude that $\alpha^* \geq \alpha$. Therefore, the problem of determining W^* is as difficult as the maximum independent set problem. \square

The proof can be also extended to gathering instances with unitary demand [KMP08].

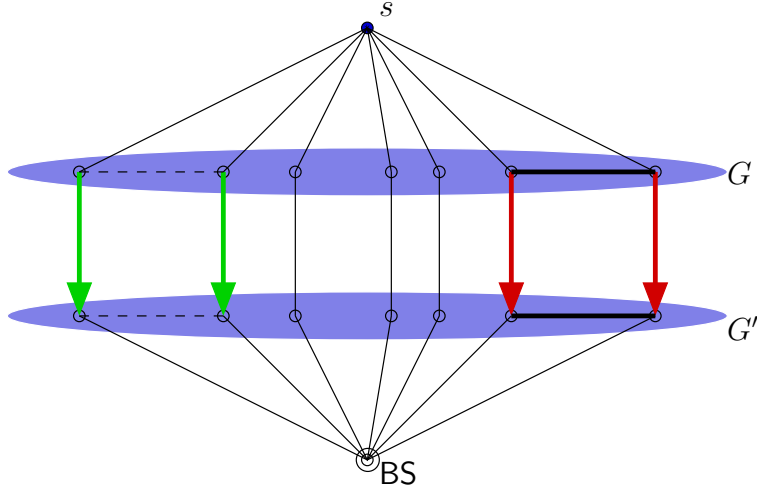


Figure 1.8: Reduction from the maximum independent set problem. Instance for RWP where node s has b unit-demand to be gathered into BS. The (d_I, d_T) -interference model considered is $d_I = 2$ and $d_T = 1$. Green arrows indicate compatible calls, red arrows indicate two calls that interfere between them.

1.3.4 Relationship between gathering and round weighting

To get some idea of the relationship between the ROUND WEIGHTING and the MINIMUM TIME GATHERING problems we present the example of figure 1.9. In this figure, we present the case of a network which is a path with n nodes numbered from 0 to $n - 1$. The base station is placed in node 0. The demand consists in one message placed in node $n - 1$. We suppose an asymmetrical interference model with parameters $d_T = d_I = 1$.

For the case of MINIMUM TIME GATHERING(see 1.9(a)) having 1 unit of demand in node $n - 1$ means that the message must be transmitted from node $n - 1$ to node $BS = 0$. The solution presented requires $n - 1$ time-steps. Indeed, we can observe that this solution is optimal. The solution consists of $n - 1$ rounds where the round i consists in one call transmitting the message between nodes $n - i$ and $n - i - 1$.

For the case of the ROUND WEIGHTING PROBLEM(see 1.9(b)) having 1 unit of demand in node $n - 1$ means that we look for the minimum period W such that 1 message must be routed from node $n - 1$ to BS in a period of length W . The solution proposed consists in a function of weights over the rounds w which is non-zero for three rounds denoted R_1 , R_2 and R_3 . These rounds are defined as $R_i = \{j \rightarrow j - 1 \mid j = i \bmod 3, i = 1, 2, 3\}$, where $j \rightarrow j - 1$ denotes a call between node j and node $j - 1$. The weight of each round is defined as $R_i = 1$ for $i = 1, 2, 3$ and 0 for all the other rounds. In this way, the associated capacity c_w of each edge of the path is 1. Then, there exists a flow function satisfying the capacity c_w of each node and transmits 1 unit of demand from $n - 1$ to BS . The total weight of the solution is $W(w) = w(R_1) + w(R_2) + w(R_3) = 3$. Hence, we have that the solution routes $1/3$ units per time-step from node $n - 1$ to BS . As a remark, we can observe that this solution is optimal.

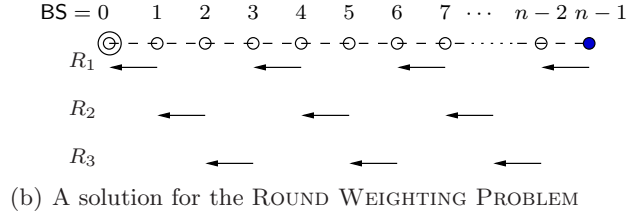
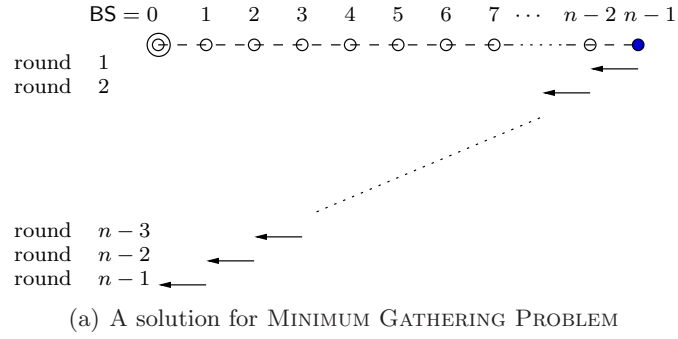


Figure 1.9: Comparison between solutions for MINIMUM TIME GATHERING and ROUND WEIGHTING. In both cases, the instance is the same: A path with n nodes, BS placed in node 0, and 1 unit of demand in node $n - 1$ to be gathered into BS.

As an interesting remark, note that the length of the optimal gathering protocol increases in the length of the path. However, the solution of the round weighting problem remains constant. This constant depends on the parameters of the interference model d_I and d_T .

Chapter 2

Summary of the results

In this chapter, we give a brief overview of the results we have obtained. Full versions are given in chapters 3,4, 5, 6 and appendix A.

In chapters 3 and 4 we study the MINIMUM TIME GATHERING PROBLEM. Chapter 3 is focused on the complexity of the problem in the path. Chapter 4 studies the MTG in the grid when no buffering is allowed in intermediate nodes.

Chapter 5 and 6 study the ROUND WEIGHTING PROBLEM. Chapter 5 studies the case where the demand may be routed by means of a cycle between each demanding node and the base station. In chapter 6 we present lower bounds and, for the case of the grid, we present upper bounds which are optimal for unitary demand, and optimal in many cases for general demand.

Table 2.1 summarizes the problems MTG and RWP studied in this thesis and the main results obtained.

Interference	Demand	buffer	problem	topology	result	reference
asymmetric $d_T = 2, 3, 5$, any d_I	unitary	buffer	MTG	path BS at end	optimal	chapter 3
asymmetric d_I, d_T	unitary	buffer	MTG	path	1-approx	
symmetric $d_I^s = d_T = 1$	general	no-buffer	MTG	grid	+1-approx, distributed +2-approx	chapter 4, [BNRR09a], [BNRR09b]
symmetric $d_I^s = d_T = 1$	general	–	RWP (routing)	general	near optimal	chapter 5
symmetric $d_T = 1$, any d_I^s	uniform general	– –	RWP RWP	grid grid	optimal approx	chapter 6, [GPRR08]

Table 2.1: Results of this thesis related to MTG and RWP.

As an appendix, and not related to gathering problems, we study we deal with the question of how to provide measures of congestion in networks. Using random graph theory we studied the number of paths and the connectivity of the network (see appendix A).

In the following we explain the different problems studied in this thesis. We explain briefly each

model, its settings, important results and notations.

2.1 Gathering radio messages in the path

In chapter 3¹, we address the problem of MTG in the path. First, we discuss lower and upper bounds when the base station is placed in an arbitrary vertex. After that, we focus the study on the complexity of the problem when the base station is an end-vertex of the path. To do that, the question to solve is how to construct a protocol for a certain path starting with a protocol for a path of smaller size.

The base station is called *sink* and it is denoted t .

2.1.1 Interference

In terms of interference, we consider the **asymmetrical** (d_I, d_T) -**interference** model. Notice that in this chapter we consider arbitrary values of d_I and d_T (with $d_I \geq d_T$).

2.1.2 Demand

We deal with **uniform demand**. More precisely, we consider unitary demand: each node has one message to be sent into the sink. However, some results are also valid for general demand.

2.1.3 Methodology

For a path with n nodes, denoted P_n , we devise a procedure that, given a gathering protocol S_n for P_n , allows us to construct a protocol S_{n+1} for P_{n+1} . The protocol S_{n+1} is constructed *incrementally* on the protocol P_n . That is: the rounds of S_{n+1} correspond to the same rounds of S_n plus some extra (compatible calls), and some additional rounds. An example is depicted in figure 2.1 We show that this procedure allows us to construct optimal gathering protocols for P_{n+1} , i.e., that there always exists a gathering protocol S_n that we can increment, and that is optimal.

2.1.4 Results

When the location of the sink is arbitrary, we have calculated lower bounds for the minimum number of rounds. These bounds improve over the lower bounds of previous works (valid for the non-unitary case). We have also presented specific protocols for the unitary case and shown that the number of rounds required for these protocols is greater than our lower bounds, but only by a constant number of rounds.

When the sink is located in an end-vertex of the path, we conjecture that the problem of calculating a protocol using a minimum number of rounds is polynomial in the length of the path (for the unitary case). We give an explicit way to implement the incremental procedure. That gives optimal

¹Joint work with Jean-Claude Bermond, Ralf Klasing, Nelson Morales and Stephane Pérennes.

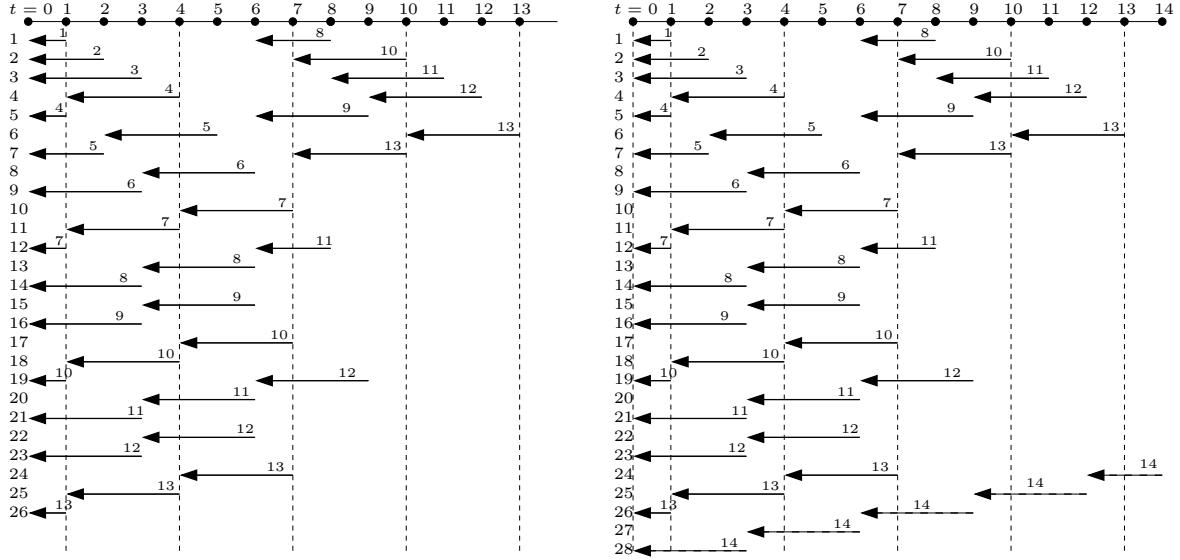


Figure 2.1: An example of an incremental protocol. On the left, the starting protocol for P_{14} . On the right, the incremental protocol for P_{15} . The *sink* denoted t corresponds to node 0. The interference considered is $d_I = 4$, $d_T = 3$. (Numbers over calls indicate the original vertex of the message transmitted in the call. New calls are dashed.)

protocols for $d_T = 2, 3, 5$. We have also optimal protocols for other values of d_T, d_I according to the congruence classes, namely if $d_I = pd_T + q$, $0 \leq q < d_T$, we have optimal protocols if $q = d_T - 1$; $q = 0$ and $d_T \leq 7$; $q + 1$ and d_T relatively primes; and $d_T - q - 1 \leq p + 4$.

It will be interesting to find polynomial algorithms for paths with the BS anywhere and in particular in the center of the path for any d_I, d_T . However in view of the complexity of the solution for the case $d_T = 1$ where the answer is known only for $d_I \leq 4$, that appears as a difficult task.

Results are summarized in table 2.2.

Table 2.2: Settings of the problem studied in chapter 3

Interference		Demand	buffer	problem	topology	result
asymmetric	with $d_I = pd_T + q$	unitary	buffer	MGT	path BS at end	optimal
	$d_T = 2, 3, 5$, any d_I	unitary	buffer	MGT	path BS at end	optimal
	any d_T , $q = d_T - 1$	unitary	buffer	MGT	path BS at end	optimal
	any $d_T \leq 7$, $q = 0$	unitary	buffer	MGT	path BS at end	optimal
	$q + 1$ and d_T rel primes	unitary	buffer	MGT	path BS at end	optimal
	$d_T - q - 1 \leq p + 4$	unitary	buffer	MGT	path BS at end	optimal
asymmetric	d_I, d_T arbitrary	unitary	buffer	MGT	path	1-approx

2.2 Minimum delay Data Gathering in Radio Networks

The aim of chapter 4² is to design efficient gathering algorithms for the MTG over the grid where the base station is placed in one of the corners and interference constraints are present.

2.2.1 Interference

In terms of the interference models presented above, it corresponds to a **symmetrical interference** model with $d_I^s = d_T = 1$.

2.2.2 Demand

We suppose a **non-uniform demand**. It means that nodes may have different numbers of messages to be gathered. Notice that for the case with uniform demand (all the nodes have the same number of messages to be gathered) the problem becomes easy.

2.2.3 Methodology

Formally, we consider the grid as a graph where the nodes are represented by their coordinates. The base station, denoted BS is placed at $(0, 0)$ and called *source*. We considered a set of messages \mathcal{M} to be gathered.

We consider an equivalent formulation of the gathering problem which is the *personalized broadcasting*. In this way, we suppose that the base station BS must send the \mathcal{M} messages to the corresponding nodes.

The idea of the solution consists in finding a protocol such that, at each time-slot, the BS sends one message. Due to the interference, the paths of two consecutive messages (i.e, at two consecutive time-steps) must be disjoint (see figure 2.2).

In the solution proposed, the base station sends each message by a path following the direction of either horizontal-vertical or vertical-horizontal as shown in figure 2.2. Therefore, and due to the interference, the solution consists in a protocol such that the base station sends alternately a message horizontally and vertically at each time-slot. The problem is reduced to provide BS with a good delivery order of the messages.

2.2.4 Results

We propose a very simple algorithm that achieves makespan plus two, as well as a more involved +1-approximation algorithm. In addition, we show a distributed version of the +2-approximation algo-

²Joint work with Jean-Claude Bermond, Nicolas Nisse and Hervé Rivano

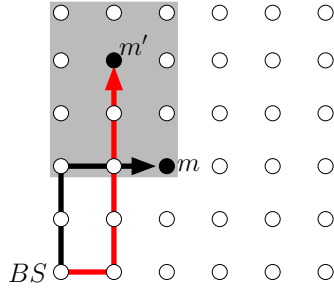


Figure 2.2: Configuration when two consecutive messages interfere.

rithm. The best known algorithm for grids was a multiplicative 1.5-approximation algorithm [RS07]. Furthermore our algorithms need **no buffering**. Results are summarized in table 2.3.

This work is published in [BNRR09a]. An extended version of this work has been accepted in the conference *AdHocNow'09* [BNRR09b].

Table 2.3: Settings of the problem studied in chapter 4

Interference	Demand	buffer	problem	topology	result	reference
symmetric $d_I^s = d_T = 1$	general	no-buffer	MGT	grid	+1-approx, +2-approx distributed	[BNRR09b]

2.3 Round weighting in the primary node interference model

In chapter 5³ we discuss mainly the problem of RWP for routing the demand. In this case, we deal with the problem of routing the demand by means of two paths between each demanding node and the base station.

In this chapter the base station is called *gateway*.

2.3.1 Interference Model

In this chapter we consider a **symmetrical interference** with $d_I^s = d_T = 1$. Then, a round corresponds to a matching over the graph.

2.3.2 Demand

We consider instances of **general demand**. Indeed, in most of our results, the demand of each node is routed independently to the other nodes and then, different nodes use different rounds to route the demand. Hence, the methodology to tackle with uniform or the general demand is the same.

2.3.3 Methodology

The idea consists in covering the cycle which consists in two paths between a demanding node and the base station. Let us suppose that we have to sent b units of demand from one node to the base station. If there exists a cycle of even length containing the demanding node and the gateway, then it is possible to present a solution for the RWP with cost $W = b$. In this case, $\frac{b}{2}$ units of flow are sent by one path and $\frac{b}{2}$ by the other. Two rounds labeled 1 and 2 are used. The weight of each round is $\frac{b}{2}$. The solution is depicted in figure 2.3.

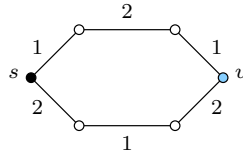


Figure 2.3: Routing b units in the even cycle C_6 . Each round has a cost of $b/2$, thus $W = b$. Label over each edge represent the round covering the edge. Black and cyan nodes represent the demanding node v and the base station.

However, if there is no even cycle between the demanding node and the base station, more than two rounds are needed. Therefore, the total weight of the solution depends inversely on the length of the cycle.

³Joint work with Jean-Claude Bermond, Hervé Rivano, Stephane Pérennes and Joseph Yu.

For each type of cycle considered, we will propose protocols and we will check that in some cases protocols are optimal by presenting lower bounds derived from dual solutions.

2.3.4 Results

We propose solutions for routing the demand of each node by means of a simple cycle, a cycle with ears and cycles with chords. For the case of the cycle with ears, we have shown that the solution proposed is optimal by means of studying the lower bounds provided by the dual formulation. Results are summarized in table 2.4.

Table 2.4: Settings of the problem studied in chapter 5

Interference	demand	buffer	problem	topology	results	reference
symmetric $d_I^s = d_T = 1$	general	–	RWP (routing)	general	near optimal	chapter 5

2.4 Round Weighting Problem and Gathering in wireless networks with symmetrical interference

In chapter 6⁴ we considered the RWP for gathering instances. We present methods to obtain lower bounds for general topologies using cliques of calls. However, explicit lower bounds and optimal results are presented for grids with the base station placed either at the center or at the corner. A preliminary version has been presented in [GPRR08] for the case $d_I^s = 2$, $d_T = 1$.

In this chapter, the base station is denoted g (as a reference of *gateway*).

2.4.1 Interference

The interference considered corresponds to the **symmetrical interference** model for $d_T = 1$ but $d_I^s \geq 1$ arbitrary.

2.4.2 Methodology

We deal with this problem by studying the sets of edges defined by the fact that all the edges in the set are pairwise interfering. This set of edges are called *call-cliques*. Let us see as an example in the grid where the *call-clique* considered is placed around the base station as shown in figure 2.4. In this case, the interference considered corresponds to $d_I^s = 3$ and $d_T = 1$. We can check that if one of the edges depicted belongs to a round, due to the interference, no other edge in the figure may be in the same round. Notice that the *call-clique* depicted covers two levels of edges: the first level consist of the edges adjacent to the base station; and the second level which are the edges adjacent to the edges in the first level. The lower bounds that we will propose depend on the number of levels covered.

The technique of covering the edges using one *call-clique* sometimes will not be enough to attain the optimal solution. In some cases we will need to combine different *call-cliques* and even using *call-cliques* which are not around the base station.

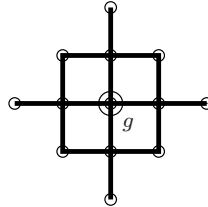


Figure 2.4: Each depicted edge around the base station must be activated in a different round. In this scheme, $d_I^s = 3$, $d_T = 1$.

⁴Joint work with Jean-Claude Bermond, Cristiana Gomes and Hervé Rivano.

2.4.3 Results

We present lower bounds for general topologies. For the case of the grid in the uniform demand, we present upper and lower bounds which attain the optimal solution. The results are valid when the base station is placed either in the center or in a corner of the grid. For general demand, we determine the zones of the grid where the demand is crucial. Results are summarized in table 2.5.

Table 2.5: Settings of the problem studied in chapter 6

Interference		demand	buffer	problem	topology	result	reference
symmetric	d_I^s, d_T	general	–	RWP	general	LB	chapter 6
symmetric	d_I^s, d_T	general	–	RWP	grid	UB, LB	chapter 6
symmetric	d_I^s, d_T	uniform	–	RWP	grid	optimal	chapter 6
symmetric	$d_I^s = 2$, any d_T	uniform	–	RWP	grid	optimal	[GPRR08]

2.5 Asymptotic Congestion Wireless Ad-Hoc and Sensor Networks

In appendix A⁵, we study a measure of link-level congestion in static wireless ad-hoc and sensor networks randomly deployed over an area. The measure of congestion considered is the inverse of the greatest eigenvalue of the adjacency matrix of the random graph. This measure gives an approximation of the average quantity of wireless links of a certain length in the network.

2.5.1 Methodology

We use concepts of spectral graph theory and we analyze the asymptotic behavior of the number of paths of length k .

The ADJACENCY MATRIX of a graph G , denoted A , is the matrix with rows and columns labelled by graph vertices, defined as

$$A_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are connected by an link,} \\ 0 & \text{otherwise.} \end{cases}$$

We will denote A^k the k -th power of the adjacency matrix. In this way, the coordinate (i, j) of A^k will represent the number of paths of length exactly k between nodes i and j .

Furthermore, the SPECTRAL RADIUS of a graph G , denoted λ_A , is the size of the largest eigenvalue of the adjacency matrix of the graph that can be written as

$$\lambda_A = \sup_{x \in \mathbb{R}^d \setminus \{0\}} \frac{\langle x, Ax \rangle}{\langle x, x \rangle}.$$

where $\langle x, y \rangle$ denotes the vector multiplication. Using the Perron-Frobenius' theorem, [HJ90], we can combine both definitions achieving that

$$\lim_{k \rightarrow +\infty} (\vec{1}^* A^k \vec{1})^{1/k} = \lambda_A.$$

where $\vec{1}$ corresponds to the vector in which all of the elements are ones, and $\vec{1}^*$ is its transpose.

From this result we obtain that the number of paths of length k in G is approximately λ_A^k , for k large enough.

In this way we define the CONGESTION NUMBER as the inverse of the spectral radius of the graph λ_A^{-1} .

The intuitive explanation to this definition is that while we have more paths of a fixed length in order to send information, we can split the information on these paths and coordinate it to arrive with the same number of hops at the receiver.

⁵Joint work with Alonso Silva and M  rouane Debbah (Supelec).

2.5.2 Results

We have provided a model to deal with congestion of randomly deployed wireless nodes. For Bernoulli random graphs and Geometric random graphs we have provided, in the case of large networks, the congestion number which is linked to the number of connected paths of some given length. Quite remarkably, the mean congestion number can be explicitly derived using asymptotic results of random matrix theory and the results hold even for a not so large number of nodes.

Part I

Gathering

Chapter 3

Gathering radio messages in the Path

In this chapter, we address the problem of gathering information in one node (sink) of a radio network where interference constraints are present: when a node transmits, it produces interference in an area bigger than the area in which its message can actually be received. The network is modeled by a graph; a node is able to transmit one unit of information to the set of vertices at distance at most d_T in the graph, but when doing so it generates interferences that do not allow nodes at distance up to d_I ($d_I \geq d_T$) to listen to other transmissions.

We are interested in finding a gathering protocol, that is an ordered sequence of rounds (each round consists of non-interfering transmissions) such that $w(u)$ messages are transmitted from node u to a fixed node t (the sink). We focus on the specific case where the network is a path and where the traffic is unitary ($w(u) = 1$ for all u). Our aim is to find the minimum number of rounds called gathering time.

We give lower and upper bounds that differ only by a constant (independent of the length of the path) so obtained a 1^+ -approximation.

When the sink is an end vertex of the path, we give a way to construct incremental protocols. An incremental protocol for the path on $n + 1$ vertices is obtained from a protocol for n vertices by adding new rounds and new calls to some rounds but without changing the calls of the original rounds. We conjecture that this incremental construction gives optimal protocols and prove it in special cases (small d_T in particular).

3.1 Introduction

3.1.1 Background and motivation

In radio networks a set of radio devices communicate by using radio transmissions which, depending on the technology used, are subject to different interference constraints (see for instance for 802.11 [Bia00, Gal04, Müh02]). This means that only certain transmissions can be performed simultane-

ously, therefore the devices have to act in a cooperative manner in order to achieve an effective flow of information in the network. In this context, we study a problem proposed by FRANCE TELECOM, about “how to provide Internet to villages” (see [BBS05]).

The houses of the village are equipped with radio devices and they want to access the *rest of the world* via Internet. For that purpose they have to send (and receive) information via a gateway where there is a *central* antenna. This creates a special *many-to-one* information flow demand in which the access to the gateway must be provided. Therefore, we will consider a specific traffic pattern, similar to a single commodity flow with a distinguished node representing the gateway, called *sink* and denoted t .

Unlike in wired networks, when a node u transmits a message it does not use a resource as simple as some capacity on a link; instead it produces a signal that may prevent other transmissions to occur. The set of possible concurrent transmissions follows from a complex n -ary interference relation which properly models the idea that the noise intensity must be small enough compared to the signal intensity. In order to get tractable models, a widely used simplification consists of associating to each node a *transmission area* in which it can transmit a message and an *interference area* in which it produces a strong noise (see [SW06]). Then, the communication from a node u to a node v is possible if v is in the transmission area of u , and no third node transmitting has v in its interference area. Note that, by doing so, we replace the n -ary relation with a binary relation: two (possible) transmissions (that we will denote calls) can be performed concurrently when they do not interfere.

3.1.2 Modeling aspects

One possible way of modeling would be to represent the houses (radio devices) as nodes in the plane with Euclidean distance (the areas of transmission and interference being disks). Here, we choose to model the network by an undirected graph $G = (V, E)$, where V is the set of devices in the network and to use as distance the distance between nodes in the graph. Firstly, it simplifies the analysis and enables us to give tractable gathering algorithms. Secondly, for some graphs like grids or hexagonal grids the distance in the graphs is a good approximation for the Euclidean distance. Finally, some nodes which are close to each other in the plane might not be able to communicate due to different reasons like obstacles, hills, social relations, security. So, there is an edge if two houses are neighbors and able to communicate.

We model the transmission area and the interference area as balls in the graph by introducing two parameters: d_T , the transmission radius and d_I , the interference radius and we suppose that $d_I \geq d_T$. The transmission area (resp. interference area) is then the ball of radius d_T (resp. d_I).

The information transmitted by a node becomes available to all the nodes that are in its transmission area if they are listening, and if they are not in the interference area of a third transmitting node. We will denote the fact that node s (like sender) is transmitting a message to node r (like receiver)

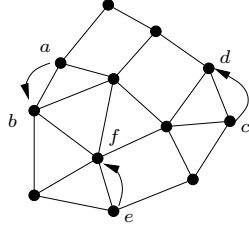


Figure 3.1: Interfering/compatible calls.

by saying there is a *call* $s \rightarrow r$. We will say that two calls $s \rightarrow r$ and $s' \rightarrow r'$ with $s \neq s'$ are *compatible* if s does not interfere with r' and s' does not interfere with r .

Figure 3.1 shows a set of 3 calls, which are represented by the arrows over the edges of the graph. If $d_T = d_I = 1$, all these calls are compatible. However, if $d_T = 1, d_I = 2$, vertex b is under the interference of vertex e , and vertex f is under the interference of vertices a and c . In this case, a round could either consist of one single call ($a \rightarrow b$ or $c \rightarrow d$ or $e \rightarrow f$), or of the two calls $a \rightarrow b, c \rightarrow d$.

Under this model, the problem raised by France Telecom consists of gathering information from each node of the network into the central node (the sink t). We will suppose that each node u has to transmit an integer $w(u) \geq 0$ number of units of information to the sink.

Time is considered discrete, i.e., divided into time-steps of fixed length, so during a time-step, a number of compatible calls (a round) is performed. We look then for an optimal protocol (sequence of rounds), such that if the rounds of the protocol are performed, then exactly $w(u)$ messages travel from u to the sink t . Our measure for optimality is the completion time (i.e., the number of rounds) needed to achieve gathering, hence our objective is to study the *minimum time gathering problem* (called also *makespan*). For example, Figure 3.2 shows an optimal gathering protocol using 18 rounds for a path with 7 vertices (each having one piece of information), with $d_T = 1, d_I = 2$ and sink $t = 0$.

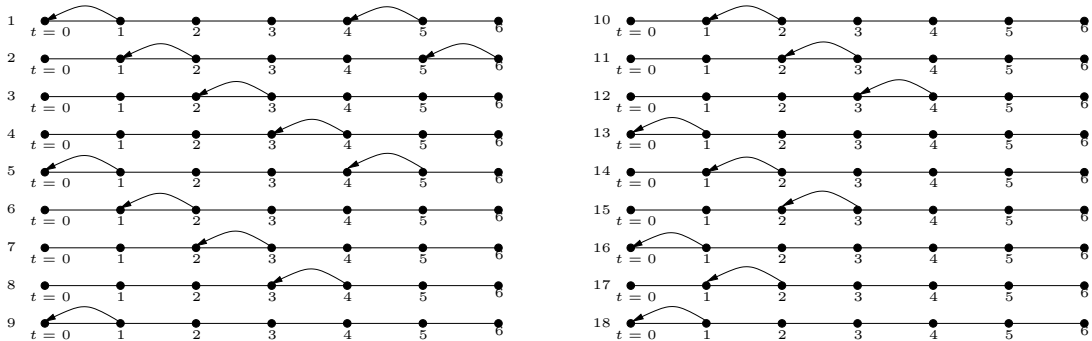


Figure 3.2: A gathering protocol in the path when $d_T = 1, d_I = 2$ and every vertex has one message to send to the sink $t = 0$.

Note that we may as well study the converse problem called (personalized broadcast) for which we need to send personalized information from the central node to each node. Indeed, like in many other communication models, we can simply reverse the order of the communication steps and the direction of the calls to get that gathering and personalized broadcast are formally equivalent. Due to this equivalence, all the results (algorithms, complexity, bounds) that we give are also valid for personalized broadcast. Here, we focus on gathering issues.

3.1.3 Related work

Basic communication problems for the dissemination of information (like gathering, broadcasting, gossiping) have been widely studied in classical interconnection networks (see the book [HKP⁺05]).

The broadcasting and gossiping problems in radio networks with $d_T = d_I = 1$ are studied in [CW91, EK04, GP02] and [CGR02, GM03, CGL02, BGP98, BGRV98] respectively. Note that broadcasting is different to our problem because in a broadcast the same information has to be transmitted to all the other nodes and therefore flooding techniques can be used.

With respect to the gathering problem (see [BKK⁺09a] for a survey) different cases have been studied. In [BGK⁺06b] a protocol for general graphs with an approximation factor of at most 4 is presented. An extension of this problem where messages can be released over time is studied in [BKMS08a] and a 4-approximation is presented. Using the same interference model, a relaxed approach has been studied in [KMP08] where the problem is studied in terms of collecting the flow demands. The unitary case (where each node has one unit of information to transmit) has been considered under different topologies. The unitary case in the path with $d_T = 1$ and arbitrary d_I is studied in [BCY06] and in [BCY09]. For the two-dimensional square grid, optimal solutions are provided in [BP05] and [BP09]. For trees, in the case $d_T = d_I = 1$, an optimal solution is presented in [BY08], and in [BGR08] if no buffering is allowed in intermediate nodes.

The fact that no buffering is allowed come from the application in sensor networks. In sensor networks (see [Gar07] for a survey), a model closer to ours is considered in [FFM04a]. Here, they consider mainly one directional antenna and the so called *primary node interference model* where a node cannot receive and transmit during the same time slot. In [FFM04a] they give optimal gathering protocols for paths and trees. The results have been extended to general graphs in the unitary case in [GR06a, GR09] where a polynomial algorithm is given.

Finally, note that some articles consider symmetric interference models, due to the fact that in the protocol 802.11 when a message is transmitted, acknowledgments need to be transmitted as well. Indeed, in some papers this model is named the *802.11 interference model* (see [Wan09b]).

3.1.4 Structure of the paper

In Section 3.2 we introduce the notation and precise the problem to solve. In sections 3.3, 3.4 and 3.5 we consider the case where the sink is an end-vertex of the path.

In section 3.3, we present the classical lower bound and an algorithm which turns out to be optimal when $d_I = pd_T + d_T - 1$.

In section 3.4, we give a new lower bound for the asn we show that the preceding algorithm gives a 1^+ -approximation (the number of rounds differ from the lower bound by a constant independent of the length of the path).

In section 3.5 we give a procedure such that, given a gathering protocol for the path of length n , it produces a solution for the path of length $n + 1$. We call this procedure *incrementing* as it does not modify the solution for n , but it only adds extra calls and rounds to gather the additional message (for the new vertex). We next show that this procedure can be used to obtain better protocols. In particular, it gives optimal protocols for small d_T and d_I such that if $d_I = pd_T + q$ then $q + 1$ is relatively prime with d_T . That gives optimal solutions for $d_T = 2, 3, 5$. We conjecture that an optimal solution can be obtained by the incremental procedure; if true that will give an optimal protocol polynomially in the length of the path.

In section 3.6 we extend the protocol and lower bounds of section 3.4 for arbitrary positions of the sink obtaining a 1^+ -approximation.

3.2 Preliminaries

In this section we introduce the model and main notation, and we state the problem to solve.

3.2.1 The model: definitions and notation

In the whole paper, we are given a graph $G = (V, E)$ with n vertices and with a distinguished vertex $t \in V$, called the *sink*, and two integers $d_I, d_T \in \mathbb{N}$, such that $d_I \geq d_T > 0$, where d_I is the *interference distance* and d_T is the *transmission distance*. The distance between two vertices u and v is the length of the shortest path from u to v and is denoted $d_G(u, v)$.

In the *gathering problem*, every node $u \in V$ has $w(u)$ unitary pieces of information (called shortly *messages*) which have to reach the sink t , where $w(u)$ is a nonnegative integer. We denote by $m(u)$ any of the $w(u)$ messages originated in the node u .

A *call* is a couple (s, r) with $s, r \in V$, $0 < d(s, r) \leq d_T$, and where s is the *sender* and r the *receiver*. We denote the call (s, r) as $s \rightarrow r$. Call $s \rightarrow r$ *interferes* with call $s' \rightarrow r'$ if $d(s, r') \leq d_I$ or $d(s', r) \leq d_I$. We say that the two calls $s \rightarrow r$ and $s' \rightarrow r'$ are *compatible* if they do not interfere, that is both $d(s, r') > d_I$ and $d(s', r) > d_I$. During one unit of time only one (unitary) message can

be transmitted.

A *round* is a set of compatible calls. If R is a round and $s \rightarrow r \in R$ is a call, we say that $s \rightarrow r$ is performed during round R , and this corresponds to the sender s transmitting a message to receiver r if there is one message available.

A *gathering protocol* is an ordered sequence of rounds that allows to gather the information of the nodes in the sink.

We will often specify protocols by giving simply the sequence of rounds, without specifying which message is sent, indeed that is irrelevant as long as each vertex can forward something new. Also, observe that when gathering it is not useful to have multiples copies of a message in different vertices: it suffices to keep the copy that arrives first to the sink. This allows us to consider simply calls of the type $s \rightarrow r$, meaning that the sender can select a unique receiver between the potential ones.

3.2.2 The Minimum Time Gathering Problem

Let us now precise the problem to solve. We call it the Minimum Time Gathering problem. The input of the problem is given by a tuple (G, w, t, d_I, d_T) with

1. A base graph $G = (V, E)$.
2. A weight function $w : V \rightarrow \mathbb{N} \cup \{0\}$, $w(u)$ being the number of messages to gather from vertex u into the sink t .
3. A *sink* $t \in V$.
4. A transmission distance $d_T \in \mathbb{N}$, $d_T \geq 1$.
5. An interference distance $d_I \in \mathbb{N}$, $d_I \geq d_T$ ($d_I = pd_T + q$, with $0 \leq q < d_T$).

We will often write simply (G, w, t) for an instance of Minimum Time Gathering.

Definition 3.1 (Gathering protocol) *A gathering protocol (or simply protocol) is an (ordered) sequence of rounds such that, once all the rounds in the protocol are executed, exactly $w(u)$ messages have been gathered from vertex $u \in V$ into the sink t .*

The goal of Minimum Time Gathering Problem is to find a protocol that requires a minimum number of rounds, called *gathering number*.

Definition 3.2 (Gathering number) *Given an instance (G, w, t, d_I, d_T) of gathering, the minimum number of rounds for any gathering protocol for the instance will be called the gathering number and will be denoted as $g_{d_I, d_T}(G, w, t)$, or simply $g(G, w, t)$ if d_I, d_T are clear from the context.*

In this article, we restrict ourselves to the case where $G = P_n$, the path with n vertices. Formally, P_n is the graph with vertex set $\{0, 1, \dots, n-1\}$ and edges between vertices i and j if and only if $|i-j| = 1$. We consider protocols only for Unitary Minimum Time Gathering which is the the unitary case where $w(u) = 1$ for all $u \neq t$. However, lower bounds are given for general values of w .

In the next sections (except the last one), we suppose the sink is the end vertex of the path $t = 0$. We will use the simplified notation $g_{d_I, d_T}(P_n, w)$ for $g_{d_I, d_T}(P_n, w, t = 0)$ and $g_{d_I, d_T}(P_n)$ for $g_{d_I, d_T}(P_n, w = 1, t = 0)$. We will also denote $A(P_n)$ for a gathering protocol that gathers one message from each vertex $i \neq 0$ into the sink $t = 0$.

3.3 Lower Bounds and Simple Protocols for the sink as an end-vertex

In the rest of the paper, we suppose d_T and d_I are given. Let $\mathbf{d_I} = p\mathbf{d_T} + \mathbf{q}$ with p and q integers, $p \geq 1$ (or $d_I \geq d_T$) and $0 \leq q < d_T$. We will also use intensively the notation $\mathbf{D} = \mathbf{d_I} + \mathbf{d_T} + 1$.

3.3.1 A first lower bound

In [BGK⁺06b] the authors give a general lower bound which is presented in the following proposition for the path P_n with the sink at vertex 0 and general weights w .

Proposition 3.1 ([BGK⁺06b]) *We have $g_{d_I, d_T}(P_n, w) \geq \text{LB}_0(P_n, w)$, where*

$$\text{LB}_0(P_n, w) = \sum_{i \leq d_I+1} w(i) \left\lceil \frac{i}{d_T} \right\rceil + \left\lceil \frac{d_I+2}{d_T} \right\rceil \sum_{i > d_I+1} w(i)$$

Note that the bound can be easily obtained in that case by noting that there is at most one call (s, r) with $r \leq d_I + 1$.

3.3.2 Optimal Protocols for P_n , $n \leq (p+1)d_T + 1$

Using a greedy protocol we can obtain the value of $g_{d_I, d_T}(P_n)$ for small n , specifically for $n \leq (p+1)d_T + 1$.

Proposition 3.2 *Let $d_I = pd_T + q$, $0 \leq q < d_T$. For $n \leq (p+1)d_T + 1$, $g_{d_I, d_T}(P_n, w) = \sum_{i \leq n-1} w(i) \left\lceil \frac{i}{d_T} \right\rceil$*

Proof: From proposition 3.1, by noting that $\lceil \frac{d_I+2}{d_T} \rceil \geq p+1$ and that for $d_I+2 \leq i \leq (p+1)d_T$ we have $\lceil \frac{i}{d_T} \rceil = p+1$, we get $\text{LB}_0(P_n, w) \geq \sum_{i \leq n-1} w(i) \lceil \frac{i}{d_T} \rceil$.

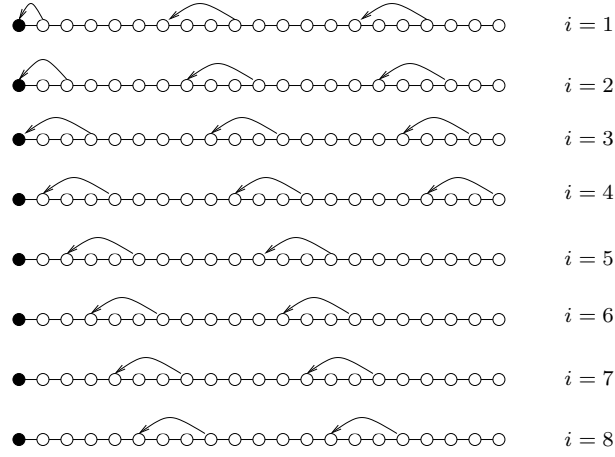


Figure 3.3: The rounds $\{i + kD \rightarrow \max[0, i + kD - d_T] : k \geq 0, i + kD \leq n - 1\}$ for $1 \leq i \leq D$ in P_{21} when $t = 0, d_I = 4, d_T = 3$ and hence $D = 8$.

Now the bound is attained by considering the greedy protocol consisting of single rounds of length d_T if possible. More precisely, for a message located at a vertex $i = \alpha d_T + \beta$ with $1 \leq \beta \leq d_T$, the protocol performs $\alpha + 1 = \lceil \frac{i}{d_T} \rceil$ rounds which are $i - j d_T \rightarrow i - (j + 1) d_T$ for $0 \leq j \leq \alpha - 1$ and $(i - \alpha d_T = \beta) \rightarrow 0$. ■

3.3.3 A simple gathering protocol

The algorithm we describe is very similar to the general algorithm of [BGK⁺06b] which gives a $\frac{3}{2}$ -approximation in the particular case of P_n . But as we consider only the unitary case ($w(u) = 1, \forall u \neq 0$), it is very simple. However, it will be sufficient to solve completely the case $q = d_T - 1$ ($d_I = p d_T + d_T - 1$) and to give in general a $+1$ -approximation. It can also be viewed as an extension of the algorithm given in [BCY09] for $d_T = 1$. Recall $D = d_T + d_I + 1$.

We will use the rounds $\{i + kD \rightarrow \max[0, i + kD - d_T] : k \geq 0, i + kD \leq n - 1\}$ that we define for $i = 1, \dots, D$ (see Figure 3.3 for an example on P_{21}). We observe that the rounds are well-defined, because the distance between two consecutive transmitters is D .

The algorithm consists of 2 phases: a loop that reduces the instance into an instance of P_k , where $k \leq D$ and a simple greedy gathering for P_k .

<p>Input: n, d_I, d_T</p> <pre> 1 while $n - 1 \geq D$ do 2 for $i = 1$ to D do Apply the round $\{i + kD \rightarrow \max[0, i + kD - d_T] : k \geq 0, i + kD \leq n - 1\}$ end $n \leftarrow n - d_T$ end 3 Gather each message independently, using a shortest path to the sink. </pre>
--

Algorithm 1: Algorithm A_1 solves gathering in P_n for $t = 0$

Proposition 3.3 *Algorithm 1 gathers in $|A_1(P_n)|$ rounds, where*

$$|A_1(P_n)| = \begin{cases} |A_1(P_{n-d_T})| + D & \text{if } n - 1 \geq D, \\ \sum_{i=1}^{n-1} \left\lceil \frac{i}{d_T} \right\rceil & \text{if } n - 1 < D \end{cases}$$

Proof: Clearly, the result holds if $n - 1 < D$, thus we focus on the case $n - 1 \geq D$. For $n - 1 \geq D$, we have that each iteration of the inner **for** loop (Step 2) requires D rounds and transforms the instance $(P_n, 0)$ into instance $(P_{n-d_T}, 0)$, hence the claim. ■

3.3.4 Case $q = d_T - 1$ ($d_I = pd_T + d_T - 1$)

In the case of $q = d_T - 1$, we can give exact values as we will see that $\text{LB}_0(P_n)$ and $|A_1(P_n)|$ are equal. This case can be viewed as an extension of $d_T = 1$ (see [BCY09]) as $q < d_T$ implies $q = 0 = d_T - 1$ for $d_T = 1$.

Proposition 3.4 *If $d_I = pd_T + d_T - 1$ ($q = d_T - 1$), then $g_{d_I, d_T}(P_n) = \sum_{i \leq d_I+1} \left\lceil \frac{i}{d_T} \right\rceil + (p+2)(n - d_I - 2)$*

Proof: In that case $d_I + 2 = (p+1)d_T + 1$ and so $\left\lceil \frac{d_I+2}{d_T} \right\rceil = p+2$ and so by proposition 3.1 (with $w(i) = 1$)

$$\text{LB}_0(P_n) = \sum_{i \leq d_I+1} \left\lceil \frac{i}{d_T} \right\rceil + (p+2)(n - d_I - 2)$$

We also have $D = d_I + d_T + 1 = (p+2)d_T$ and so $|A_1(P_n)| = \sum_{i \leq d_I+1} \left\lceil \frac{i}{d_T} \right\rceil + (p+2)(n - d_I - 2)$. ■

For the other cases $q \neq d_T - 1$, we get that for $n > (p+1)d_T + 1$, $\text{LB}_0(P_n)$ and $|A_1(P_n)|$ are different. Indeed when n increases 1, $\text{LB}_0(P_n)$ increases $p+1$ (as $q < d_T - 1$, then $\left\lceil \frac{d_I+2}{d_T} \right\rceil = p+1$) and so when n increases d_T , $\text{LB}_0(P_n)$ increases $(p+1)d_T$, but $|A_1(P_n)|$ increases $D > (p+1)d_T$.

In the next subsection we give another lower bound which increases D when n increases d_T and hence, we will deduce an 1^+ -approximation result.

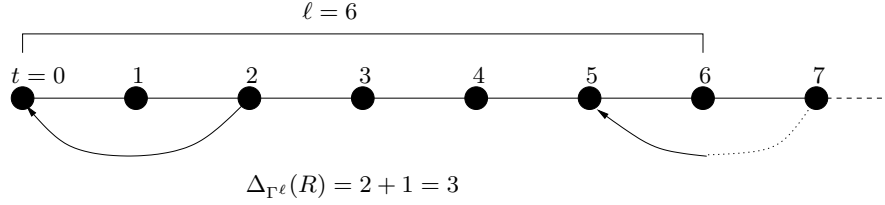


Figure 3.4: An example of the distance contribution in a specific neighborhood of the sink. Round R contains the calls $2 \rightarrow 0$ and $7 \rightarrow 5$.

3.4 Another lower bound and a 1^+ -approximation

3.4.1 Another lower bound

Let us define the distance contribution $\Delta_D(R)$ of a round R in the interval $[0, D]$ as the distance that the message transmitted during round R advances towards the sink $t = 0$ inside the interval $[0, D]$ (see figure 3.4 for an example). More precisely

$$\Delta_D(R) = \sum_{s \rightarrow r \in R} \max[0, \min[d_G(s, 0) - d_G(r, 0), D - d_G(r, 0)]]$$

Note that if r is not in $[0, D - 1]$, then $D - d_G(r, 0) \leq 0$ and hence, the call contributes 0 in $\Delta_D(R)$. If a call is backwards $s < r$, $d_G(s, 0) - d_G(r, 0) < 0$ and then such a call also contributes 0 in $\Delta_D(R)$.

If $\mathcal{R} = (R_j)_{j \in J}$ is a sequence of rounds, we define its contribution as the sum of the contribution of its rounds $\Delta_D(\mathcal{R}) = \sum_{j \in J} \Delta_D(R_j)$.

These definitions are useful to prove the following lower bound. We give it for general w although we will use only for the unitary case.

Proposition 3.5 $g_{d_I, d_T}(P_n, w) \geq \text{LB}_1(P_n, w)$, where

$$\text{LB}_1(P_n, w) = \frac{1}{d_T} \left(\sum_{i=1}^{D-1} iw(i) + D \sum_{i \geq D} w(i) \right)$$

In particular for the unitary case

$$g_{d_I, d_T}(P_n) \geq \text{LB}_1(P_n) = \frac{D(n - D)}{d_T} + \frac{D(D - 1)}{2d_T}$$

Proof: Let $R = (R_j)_{j=1}^{|R|}$ be a gathering protocol. We observe that, even when two receptions can be performed inside the interval $[0, D]$ during the same round (because the distance between a

vertex receiving a message and a vertex transmitting another is at least $d_I + 1$), then

$$(\forall j = 1, \dots, |R|), \quad \Delta_D(R_j) \leq d_T \quad (3.1)$$

We also observe that

- if $i \geq D$, a message from node i has to travel at least a distance D inside $[0, D]$ to reach the sink and there are $\sum_{i \geq D} w(i)$ such messages; and
- if $i < D$, a message from node i needs to travel a distance i inside $[0, D]$ to reach the sink and in the beginning there are $w(i)$ messages at vertex i , thus overall these messages need to travel a distance $iw(i)$ towards the sink.

Adding these values for $i = 1, \dots, n - 1$, it follows that

$$\Delta_D(R) \geq \sum_{i=1}^{D-1} iw(i) + D \sum_{i \geq D} w(i) \quad (3.2)$$

but from the definition of distance contribution and (3.1)

$$\Delta_D(R) = \sum_{j=1}^{|R|} \Delta_D(R_j) \leq d_T |R|. \quad (3.3)$$

Using (3.2) and (3.3), we have that for any gathering protocol $d_T |R| \geq \sum_{i=1}^{D-1} iw(i) + D \sum_{i \geq D} w(i)$, which corresponds to the first claim.

Now, for the second claim, we distinguish two cases.

If $n \geq D$, then $\sum_{i=1}^{D-1} i = \frac{(D-1)D}{2}$ and $\sum_{i \geq D} 1 = n - D$, hence

$$g_{d_I, d_T}(P_n) \geq \frac{D}{d_T}(n - D) + \frac{D(D-1)}{2d_T}.$$

If $n < D$, $\sum_{i=1}^{n-1} i = \sum_{i=1}^{D-1} i - \sum_{i=n}^{D-1} i \geq \frac{D(D-1)}{2} + D(n - D)$ as $i \leq D - 1$. ■

3.4.2 A 1^+ -approximation

Recall that an algorithm A calculates a 1^+ -approximation for the UNITARY MINIMUM GATHERING TIME if there exists a constant $C = C(d_I, d_T)$ independent of n such that $|A(P_n)| \leq g_{d_I, d_T}(P_n) + C$. That means that the gap between the number of rounds of algorithm A and the optimum value is an additive constant which does not increase with the size of the path.

Theorem 3.1 *Algorithm A_1 gives a 1^+ -approximation for $g_{d_I, d_T}(P_n)$.*

Proof: Let $n \geq (p+1)d_T + 1$, $n = D - \gamma + kd_T$, where $0 \leq \gamma < d_T$ then

$$\begin{aligned} |A_1(P_n)| &= kD + \sum_{i=1}^{D-\gamma-1} \left\lceil \frac{i}{d_T} \right\rceil = \frac{(n-D+\gamma)}{d_T} D + \sum_{i=1}^{D-\gamma-1} \left\lceil \frac{i}{d_T} \right\rceil \\ &= \frac{D(n-D)}{d_T} + C_1(d_I, d_T) \end{aligned}$$

By proposition 3.5

$$\text{LB}_1(P_n) = \frac{D(n-D)}{d_T} + \frac{D(D-1)}{2d_T} = \frac{D(n-D)}{d_T} + C_2(d_I, d_T)$$

and so $\frac{|A_1(P_n)|}{\text{LB}_1(P_n)} \rightarrow 1$ as $n \rightarrow \infty$. Said more precisely,

$$|A_1(P_n)| - g_{d_I, d_T}(P_n) \leq C(d_I, d_T) = C_1(d_I, d_T) - C_2(d_I, d_T).$$

■

3.5 Incremental Protocols

In what follows, it will be convenient to define $X = \{1, 2, \dots, d_T\}$, the set of possible transmission lengths and consider the translation function $f : X \rightarrow X, x \mapsto f(x) = [(x+q) \bmod d_T] + 1$.

3.5.1 Construction of the Incremental Protocol

In this section, we are interested to construct protocols incrementally from n to $n+1$ by adding new calls (without changing the former calls). More formally, protocol $R^+ = (R_j)_{j \leq |R^+|}$ for the path P_{n+1} is an increment of $R = (R_j)_{j \leq |R|}$ for the path P_n , if $R_j \subset R_j^+$, for all $1 \leq j \leq |R|$. We show how to construct a specific increment of a gathering protocol R for P_n , using a single round $d \rightarrow 0$ of R . We will call it R^+ or $\text{Inc}(R, d)$ if we want to precise the call $d \rightarrow 0$ used.

We show that the protocol $\text{Inc}(R, d)$ satisfies the following properties:

Lemma 3.1 *Let $n \geq D+1$ (the contribution works also for $(p+1)d_T + 1 \leq n \leq D$ if $d \leq n-2-d_I$). Let R be a gathering protocol for P_n containing a simple round $\{d \rightarrow 0\}$. There exists an incremental protocol for P_{n+1} denoted $\text{Inc}(R, d)$ with the following properties*

(i)

$$|\text{Inc}(R, d)| = |R| + \begin{cases} p+1 & d \leq d_T - q - 1, \\ p+2 & d > d_T - q - 1 \end{cases}$$

(ii) *The family of single rounds of the form $\{s \rightarrow 0\}$ of $\text{Inc}(R, d)$ is the family of single rounds of*

R minus one round $\{d \rightarrow 0\}$ and perhaps another round plus the round $\{f(d) \rightarrow 0\}$, where f is the translation function $f(d) = [(d + q) \bmod d_T] + 1$.

Before going in the construction, let us give a simple example to show how the construction works. Let $d_T = 2$, $d_I = 2$ ($p = 1, q = 0$), so $D = 5$. For $n = 5 = (p + 1)d_T + 1$ the greedy protocol of proposition 3.2 consists of 6 single rounds $R_1 = \{4 \rightarrow 2\}$, $R_2 = \{3 \rightarrow 1\}$, $R_3 = \{2 \rightarrow 0\}$, $R_4 = \{1 \rightarrow 0\}$, $R_5 = \{2 \rightarrow 0\}$, $R_6 = \{1 \rightarrow 0\}$. Using $d = 1$ we obtain the increment $R^+ = \text{Inc}(R, 1)$ for $n = 6$ by keeping rounds R_1 to R_5 , replacing R_6 by $R_6^+ = \{1 \rightarrow 0\} \cup \{5 \rightarrow 4\}$ and adding $R_7^+ = \{4 \rightarrow 2\}$, $R_8^+ = \{2 \rightarrow 0\}$. Here $f(d) = 2$ and the number of rounds of $\text{Inc}(R, 1)$ is $8 = 6 + (p + 1)$. Starting from R^+ , as $n = 6 \geq D + 1$, we can increment it using again $d = 1$ (Round R_4) obtaining a protocol $R^{++} = \text{Inc}(R^+, 1)$ for $n = 7$ using the same rounds of R^+ except R_4^+ replaced by $R_4^{++} = \{1 \rightarrow 0\} \cup \{6 \rightarrow 4\}$ and two new rounds $R_9^{++} = \{4 \rightarrow 2\}$, $R_{10}^{++} = \{2 \rightarrow 0\}$. Note that R^{++} is optimum as $\text{LB}_1(P_7) = \frac{5 \cdot 2}{2} + \frac{5 \cdot 4}{4} = 10$. Now in R^{++} there are 4 single rounds $s \rightarrow 0$ but all of them of the form $2 \rightarrow 0$ and so an increment of R^{++} will have 3 more rounds giving a protocol for $n = 8$ with 13 rounds ($\text{LB}_1(P_8) = 12.5$) but with a new single round $1 \rightarrow 0$ which can be used to obtain an increment for P_9 with $15 = \text{LB}_1(P_9)$ rounds.

Construction Inc: The idea of the construction is that, given the gathering protocol R for the instance $(P_n, 0)$ (i.e. R gathers messages from vertices $i = 1, \dots, n - 1$ into the sink), we will show that there exist rounds in R such that $m(n)$ (recall that n is the last vertex in P_{n+1}) can be transmitted near to the sink by extending these rounds of R with some additional calls. Once message $m(n)$ is close to the sink, we will add x additional single rounds to complete gathering in P_{n+1} .

Let R_{j_0} be the round $R_{j_0} = \{d \rightarrow 0\}$, which exists by hypothesis.

For $k \in \mathbb{N}$ such that $k \geq 1$ and $d + kd_T \leq n - 1$, define j_k in such a way that the last round in R with a transmitter s , $d + (k - 1)d_T + 1 \leq s \leq d + kd_T$ is R_{j_k} . Notice that we have that $j_{k+1} < j_k$ and if s transmits during round R_{j_k} then $s \leq d + kd_T$. Let also k_d be the largest k such that $d + d_I + 1 + kd_T \leq n - 1$. (See Figure 3.5 for an example of the construction.). Note that k_d exists ($k_d \geq 0$); indeed by hypothesis either $n \geq D + 1$ or $d \leq n - 2 - d_I$.

For $j = 1, \dots, |R|$, $j \neq j_k$, $k = 0, \dots, k_d$ we set $R_j^+ = R_j$.

For $k = 0, \dots, k_d - 1$ we set $R_{j_k}^+ = R_{j_k} \cup \{d + d_I + 1 + (k + 1)d_T \rightarrow d + d_I + 1 + kd_T\}$ and obtain a valid round as by maximality of k_d for $k \leq k_d - 1$, $d + d_I + 1 + (k + 1)d_T \leq n - 1$. Indeed, any transmitter s in R_{j_k} is such that $s \leq d + kd_T$, hence the distance from the receiver of the new call to the largest transmitter in R_{j_k} is $d(s, d + d_I + 1 + kd_T) \geq d + d_I + 1 + kd_T - d - kd_T = d_I + 1$.

For $k = k_d$ we observe that the distance from vertex n to $d + d_I + 1 + k_d d_T$ is at most d_T , hence we can set $R_{j_{k_d}}^+ = R_{j_{k_d}} \cup \{n \rightarrow d + d_I + 1 + k_d d_T\}$.

The protocol we have devised consists of $|R|$ rounds, it gathers the same messages as R , and transmits message $m(n)$ from vertex n up to vertex $v_0 = d + d_I + 1$. Note that there always exists a call ending in v_0 . Indeed, either $n \geq D + 1$ and as $d \leq d_T$, $v_0 = d + d_I + 1 \leq d_T + d_I + 1 = D < n$;

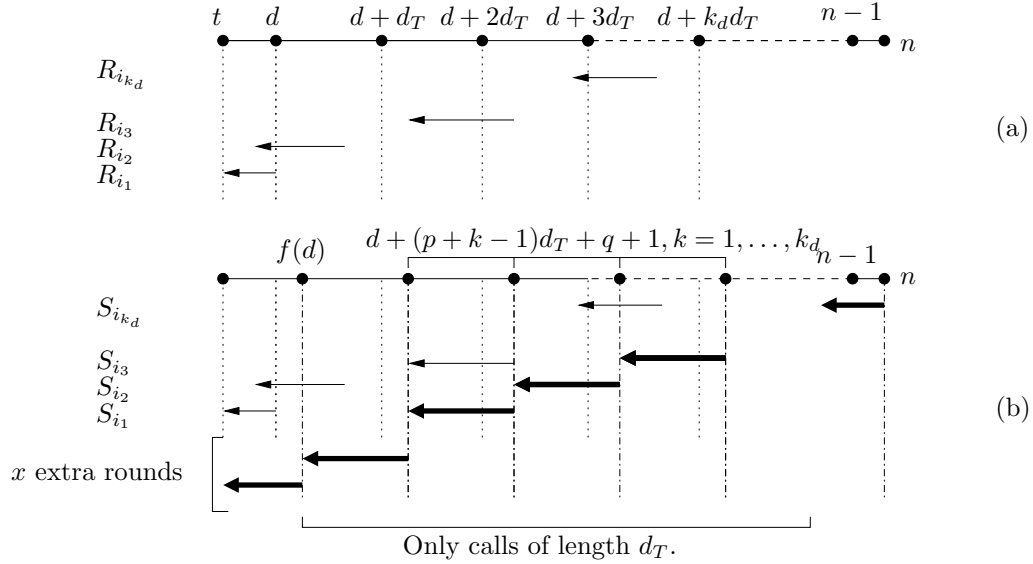


Figure 3.5: An example of the step for constructing incremental protocols. $x = p + 1$ if $d \leq d_T - q - 1$ or $x = p + 2$ otherwise.

or $(p + 1)d_T + 1 \leq n \leq D$, but in this case we choose d such that $d \leq d_T - q - 1$ and so $v_0 = d + d_I + 1 \leq (p + 1)d_T < n$.

Now we can add extra single rounds to transmit the message from vertex v_0 up to the sink $t = 0$. We do so using only calls of length d_T (excepting, maybe, the last one). Notice that properties (i) and (ii) are satisfied. Indeed we have added $x = \left\lceil \frac{d+d_I+1}{d_T} \right\rceil$ rounds, hence $x = p + \left\lceil \frac{d+q+1}{d_T} \right\rceil$ and therefore $x = p + 1$ if $d + q + 1 \leq d_T$ ($\iff d \leq d_T - q - 1$) or $x = p + 2$ if $d + q + 1 > d_T$ (notice that $d + q + 1 \leq 2d_T$ as $q < d_T$ and $d \leq d_T$). We also obtain that the very last call performed in this way, which is the only single call transmitting $m(n)$ ending in 0, is $d + q + 1 \rightarrow 0$ if $d \leq d_T - q - 1$ or $d + q + 1 - d_T$ if $d > d_T - q - 1$, that is $f(d) \rightarrow 0$. Note that for $k = 1$, R_{j_1} contains a unique call with sender s such that $d + 1 \leq s \leq d + d_T$. It might happen that $s \leq d_T$ and the call of R_{j_1} is an $s \rightarrow 0$. So, we might loose a second single call of type $s \rightarrow 0$. \square

We can repeat this incremental *construction Inc.* Let us start at some value n_0 with a protocol $A(P_{n_0})$ containing the family S_0 of single rounds of the form $\{s \rightarrow 0\}$. Let us define a sequence $(d_0, \dots, d_t, \dots, d_{n-n_0-1})$ as *admissible* if $d_t \in S_t$, where S_t is the family of values d of single rounds $d \rightarrow 0$ for the protocol at step t . By construction, $S_{t+1} \subset S_t \setminus \{d_t\} \cup \{f(d_t)\}$. Then for any admissible sequence $(d_0, \dots, d_{n-n_0-1})$ we get, by using the preceding construction at each step t , a protocol $A(P_n)$ which satisfies $|A(P_n)| = |A(P_{n_0})| + (n - n_0)(p + 1) + \delta$, where δ is the number of d_t such that $d_t > d_T - q - 1$. We will call such values *bad values*. Otherwise, the values d_t for which $d_t \leq d_T - q - 1$ are denoted *good values*.

The aim of the next section is to give examples which give optimal protocols, then to prove that the best choice of d_t is, in many cases, to choose the smallest d of S_t .

3.5.2 A gathering protocol when $q = 0$

Let us show how incremental protocols can give in some cases optimal solutions. We first deal with the case $q = 0$ that is $d_I = pd_T$ and suppose $q \neq d_T - 1$, that is $d_T \neq 1$ as we deal with this case in proposition 3.1 and the result is already known (see [BCY06, BCY09]). By proposition 3.2 we know that for $n \leq n_0 = (p+1)d_T + 1$ we have an optimal protocol with $\sum_{i \leq n-1} \left\lceil \frac{i}{d_T} \right\rceil = \frac{(p+1)(p+2)}{2}$ rounds. Furthermore $A(P_{n_0})$ contains $p+1$ single rounds $d \rightarrow 0$ for each d , $1 \leq d \leq d_T$. Said otherwise, the sequence S_0 of admissible values of d consists of $p+1$ values d for $1 \leq d \leq d_T$. Note that for each i we have also p rounds of the form $d_T + i \rightarrow i$ which can be used for R_{j_1} in the preceding construction. Note also that as $q = 0$, $f(d) = d + 1$ and for any $d < d_T$ the number of rounds in $\text{Inc}(R, d)$ is $|R| + p + 1$ and $\text{Inc}(R, d)$ contains one more single round $d + 1 \rightarrow 0$ (and at least one less round $d \rightarrow 0$). We do first increments by choosing the smallest possible value d at each time.

Starting from $A(P_{n_0})$, we use $d = 1$ for $(p+1)$ steps as there are $(p+1)$ rounds $1 \rightarrow 0$. We obtain a protocol for $n_0 \leq n \leq n_1 = n_0 + (p+1)$. The protocol $A(P_{n_1})$ contains now $2(p+1)$ single rounds $2 \rightarrow 0$ (the $(p+1)$ rounds of $A(P_{n_0})$ plus the $p+1$ created by incrementing). We have also created $p+1$ rounds of the form $d_T + 2 \rightarrow 2$ and use no rounds of the form R_{j_1} .

We apply $2(p+1)$ increments starting from $A(P_{n_1})$ and using $d = 2$ getting a protocol for $n_1 \leq n \leq n_2 = n_1 + 2(p+1)$. We have used all the calls $2 \rightarrow 0$ in rounds R_{j_0} and also $2(p+1) - 1$ calls of type $d_T + i \rightarrow i$ for $i = 1, 2$ for rounds R_{j_1} (Note that the -1 comes from the fact that for sending n_1 , there is no need of round R_{j_1}). Doing so, $A(P_{n_2})$ contains $3(p+1)$ single rounds $3 \rightarrow 0$ (the $(p+1)$ of $A(P_{n_0})$) and the $2(p+1)$ created by incrementing). We have also created $2(p+1)$ rounds of the form $d_T + 3 \rightarrow 3$. So altogether we have available for next rounds R_{j_1} $3p + (p+1) + 1$ rounds of type $d_T + i \rightarrow i$ for $1 \leq i \leq 3$ (those existing in $A(P_{n_1})$ minus $2(p+1) - 1$, plus $2(p+1)$ created).

We iterate the process until either we have used all the $d \rightarrow 0$ with $d < d_T$ in rounds R_{j_0} (case d_T small); or at some step, we have not enough rounds available for R_{j_1} . Indeed, at each step d we use $d(p+1)$ rounds of the form $d_T + i \rightarrow i$ (with $1 \leq i \leq d$) and create $d(p+1)$ of the form $d_T + i + 1 \rightarrow i + 1$. So, the total number of rounds available at step d for round R_{j_1} is the dp rounds of $A(P_{n_0})$ of the form $d_T + i \rightarrow i$, $1 \leq i \leq d$, plus the $p+1$ created at step 1, minus $2(p+1) - 1$ used at step 2 plus $2(p+1)$ created at step 2 and more generally minus those used at step d in number equal to those created at the same step. So, altogether the number of calls available at any step is $dp + p + 2$. But at some step $d \geq 3$ we need $d(p+1)$ calls for each R_{j_1} and so we are obliged to use some single rounds $s \rightarrow 0$ for R_{j_1} if $d(p+1) > dp + (p+2)$ that is $d > p + 2$. Furthermore, we can without problem use the calls $d_T \rightarrow 0$ (in number $p+1$) for rounds R_{j_1} but no other calls $s \rightarrow 0$. That works for $dp + (p+2) + (p+1) \geq d(p+1)$ that is $d \leq 2p + 3$. If the condition is not satisfied that is $d_T - 1 > 2p + 3$ we need to use a round $s \rightarrow 0$ with $s \leq d_T - 1$ and the distance contribution of the round R_{j_1} in the interval $[0, d]$ will be s and the lower bound is not attained.

In summary, we have constructed a protocol guaranteed to be optimal only if $d_T \leq 2p + 4$ getting the following proposition:

Proposition 3.6 *If $d_T \leq 2p + 4$ and $q = 0$ then we have an optimal protocol.*

Corollary 3.1 *If $d_T = 2$ then we have an optimal protocol.*

Proof: We have only two cases: $q = 0$ and $q = 1$. For the case $q = 0$, the result follows from proposition 3.6. For the case $q = 1$ (case $q = d_T - 1$), the result follows from proposition 3.4. ■

Remark 3.1 *For $d_T > 2p + 4$ for example $d_T \geq 7$ for $p = 1$ we have a gap between the number of rounds of the increment and the lower bound. We conjecture that in fact the protocol is optimal and that the lower bound should be increased. Anyway, this gap is small and better than that obtained by using Algorithm 3.3.*

3.5.3 Upper bound for incremental protocol

As we have seen in the case $q = 0$, it might be difficult to know the set of values d in S_t corresponding to single calls $d \rightarrow 0$. If we do not use single calls $s \rightarrow 0$ in round R_{j_1} we will see after (proposition 3.9) that the best choice is to select the smallest d in S_t . However we can overcome this difficulty by always choosing a value of d for which we are sure it belongs to S_t . It suffices to take the sequence $(d, f(d), f^2(d), \dots, f^t(d))$ starting with a d in S_0 as by lemma 3.1 we are sure that $f(d) \in S_1$ and more generally $f^t(d) \in S_t$. Using this sequence of values of d we will get an upper bound which have the same behaviour as $\text{LB}(1)(P_n)$. The following lemma indicates that we have interest to choose the smallest d .

Lemma 3.2 *If $d < d'$ the sequence $\{d, f(d), \dots, f^{h-1}(d)\}$ contains more good values (values such that $f^i(d) \leq d_T - (q + 1)$) than $\{d', f(d'), \dots, f^h(d')\}$*

Proof: We will prove that if we have α good values in the sequence $(d, f(d), \dots, f^h(d))$ with $d \geq 2$, we have at least α good values in the sequence $(d - 1, f(d - 1), \dots, f^{h-1}(d - 1))$. Indeed, $f^i(d - 1) = f^i(d) - 1$ (the values being taken in $[1, d_T]$). Suppose $f^i(d)$ is a good value. We consider two cases:

- Case $f^i(d) \geq 2$, which is the case for $i = 0$. As $f^i(d) \leq d_T - q - 1$, then $1 \leq f^i(d - 1) < d_T - q - 1$ and so $f^i(d - 1)$ is also a good value.
- Case $f^i(d) = 1$, with $i \geq 1$. Then, $f^{i-1}(d) = d_T - q$ and so $f^{i-1}(d)$ is a bad value. But $f^{i-1}(d - 1) = d_T - q - 1$ and $f^{i-1}(d - 1)$ is a good value. So altogether we have at least as much good values for $d - 1$ than for d (perhaps one more).

■

Proposition 3.7 *For any $n, h \in \mathbb{N}$*

$$g_{d_I, d_T}(P_{n+h}) \leq g_{d_I, d_T}(P_n) + \left\lceil \frac{hD}{d_T} \right\rceil.$$

Proof: We have to show that there exists a sequence of h increments such that the number of *bad values* δ (those values $d > d_T - q - 1$) satisfies $(p+1)h + \delta \leq \left\lceil \frac{hD}{d_T} \right\rceil$. As $D = d_I + d_T + 1 = (p+1)d_T + q + 1$ it suffices to show that $\delta \leq \left\lceil \frac{h(q+1)}{d_T} \right\rceil$. Let $\delta_h = \left\lceil \frac{h(q+1)}{d_T} \right\rceil$ that is $(\delta_h - 1)d_T < h(q+1) \leq \delta_h d_T$. By lemma 3.2 it suffices to show that the worst sequence $d_T, f(d_T), \dots, f^{h-1}(d_T)$ contains at most δ_h *bad values* (Note that there exists always a sequence $d, f(d), \dots, f^{h-1}(d)$ as in any protocol the last round is necessarily a single round.)

Fact 3.1 *The sequence $d_T, f(d_T), \dots, f^{h-1}(d_T)$ contains at most $\delta_h = \left\lceil \frac{h(q+1)}{d_T} \right\rceil$ bad values.*

Proof: By induction on h . The fact is true for $h = 1$ as $\delta_h = 1$ and the sequence has one value d_T (bad in that case). Note that $f^i(d_T) = [i(q+1) - 1] \bmod d_T + 1$. Suppose the fact is true for $h - 1$. We distinguish two cases:

- If $(h-1)(q+1) \leq (\delta_h - 1)d_T$ and so $\delta_{h-1} = \delta_h - 1$ and by induction hypothesis we have at most $\delta_h - 1$ *bad values* and so at most δ_h *bad values* for h .
- If $(h-1)(q+1) > (\delta_h - 1)d_T$. As $f^{h-1}(d_T) = [(h-1)(q+1) - 1] \bmod d_T + 1 = (h-1)(q+1) - (\delta_h - 1)d_T$ and $h(q+1) \leq \delta_h d_T$, then $f^{h-1}(d_T) \leq \delta_h d_T - (q+1) - (\delta_h - 1)d_T = d_T - (q+1)$. Therefore $f^{h-1}(d_T)$ is a good value and so the number of bad values is δ_{h-1}

■

■

Note that, taking $h = d_T$ in proposition 3.7, we find again proposition 3.3.

Proposition 3.8 *If there exists $N \in \mathbb{N}$ such that $\text{LB}_1(P_N) = |A^*(P_N)|$ where $A^*(P_N)$ is a gathering protocol for the path P_N , then there exists an optimal protocol for the instance $(P_{n+h}, 0)$ for any $h \geq 0$.*

Proof: Because $|A^*(P_N)| = \text{LB}_1(P_N)$ we have $A^*(P_N)$ is optimum. Now, notice that

$$\text{LB}_1(P_{N+h}) = \text{LB}_1(P_N) + \frac{hD}{d_T},$$

and that from proposition 3.7 we have $g_{d_I, d_T}(P_{n+h}) \leq g_{d_I, d_T}(P_n) + \left\lceil \frac{hD}{d_T} \right\rceil = |A^*(P_N)| + \left\lceil \frac{hD}{d_T} \right\rceil$. But $|A^*(P_N)| = \text{LB}_1(P_N)$ and we can write

$$|A^*(P_N)| + \frac{hD}{d_T} \leq g_{d_I, d_T}(P_{N+h}, 0) \leq |A^*(P_N)| + \left\lceil \frac{hD}{d_T} \right\rceil,$$

from where the result follows. ■

Proposition 3.9 *For $n \geq D + 1$, the protocol $A^*(P_n)$ obtained from a protocol $A(P_{n_0})$ by choosing at each incremental step the smallest d possible is optimum among all the incremental protocols obtained by incrementing $A(P_{n_0})$, if we do not use rounds $s \rightarrow 0$ for the R_{j_1} .*

Proof: The value of a sequence $S = (d_0, \dots, d_{n-1})$, denoted $\delta(S)$ is the number of *bad values* d (such that $d \geq d_T - q - 1$). So $|A(P_n)| = |A(P_{n_0})| + \delta(S) + (p+1)(n - n_0)$. Let $S^{\text{opt}} = (d_0^{\text{opt}}, \dots, d_{n-n_0-1}^{\text{opt}})$ the sequence obtained by incrementing $A(P_{n_0})$ to achieve a protocol $A(P_n)$ with the minimum number of rounds. Let $S^* = (d_0^*, \dots, d_{n-n_0-1}^*)$ defined as the sequence which increments at each time by choosing the call $d \rightarrow 0$ such that d is minimum. If $S^* = S^{\text{opt}}$, then we are done. Otherwise, let t_0 the first value where the two sequences differ, then $d_{t_0}^* < d_{t_0}^{\text{opt}}$. We distinguish two cases

- $d_{t_0}^* = d_{t'}^{\text{opt}}$ for some $t' > t_0$. In this case we simply replace in S^{opt} the value $d_{t_0}^{\text{opt}}$ by $d_{t_0}^*$ and $d_{t'}^{\text{opt}} = d_{t_0}^*$ by $d_{t_0}^{\text{opt}}$ (we exchange the values in positions t_0 and t'). The sequence obtained is also a valid one which has exactly the same value as S^{opt} , so it is also optimal but it has a large subsequence in common with S^* . That is valid only if we can use $d_{t_0}^*$ as $d_{t'}^{\text{opt}}$ which supposes that $d_{t_0}^*$ has not been used for a round R_{j_1} explaining the condition of the statement of the proposition.
- t_0^* do not appear in the rest of the sequence of S^{opt} . In this case, we replace in S^{opt} the element $d_{t_0}^{\text{opt}}$ by $d_{t_0}^*$. It might happen that in S^{opt} some $d_{t_1}^{\text{opt}} = f(d_{t_0}^{\text{opt}})$, for $t_1 \geq t_0$, then we replace $d_{t_1}^{\text{opt}}$ by $d_{t_1}' = f(d_{t_0}^*)$ and so if there exists a sequence $t_i > t_{i-1}$ such that $d_{t_i}^{\text{opt}} = f(d_{t_{i-1}}^{\text{opt}}) = f^i(d_{t_0}^{\text{opt}})$, we replace each $d_{t_i}^{\text{opt}}$ by $d_{t_i}' = f(d_{t_{i-1}}') = f^i(d_{t_0}^*)$. We obtain a new valid sequence.

Let us show now that the new sequence denoted S' satisfies that $\delta(S') \leq \delta(S^{\text{opt}})$. To do that, it suffices to prove that each time the second case is applied, the new sequence has no more *bad values* as the precedent sequence. Since $d_{t_0}^* < d_{t_0}^{\text{opt}}$, lemma 3.2 guarantees that $\{d_{t_0}^*, d_{t_1}' = f(d_{t_0}^*), \dots, d_{t_h}' = f^{h-1}(d_{t_0}^*)\}$ has at least the same number of *good values* than $\{d_{t_0}^{\text{opt}}, d_{t_1}^{\text{opt}} = f(d_{t_0}^{\text{opt}}), \dots, d_{t_h}^{\text{opt}} = f^{h-1}(d_{t_0}^{\text{opt}})\}$ and so the result follows. ■

3.5.4 Case $q + 1$ and d_T are relatively prime

According to proposition 3.8 if we are lucky enough to find a value N for which $|A(P_N)| = \text{LB}_1(P_N)$ (which in particular means $\text{LB}(1)(P_N)$ is an integer) we can conclude that for $n \geq N$ the increment R^{+h} are all optimum. That is what happens for $q = 0$ where we found such an $N = (p + 1)d_T + (p + 1)\frac{d_T - 1}{2}d_T$, when $d_T \leq 2p + 4$. In that case we started from the greedy protocol for $n = n_0 = (p + 1)d_T + 1$. But in general it is not the good protocol to start with; in contrary we might have to start from some non optimal protocol. To see what happens consider a case where $q \neq 0$ and $q \neq d_T - 1$ for example $d_T = 3, d_I = 4$ ($p = 1, q = 1$) and $D = 8$. For $n_0 = 7$ we have the

greedy optimal protocol with $|R| = 9$ rounds, containing two rounds $\{1 \rightarrow 0\}$, two $\{2 \rightarrow 0\}$ and two $\{3 \rightarrow 0\}$. Using twice $d = 1$ when possible we get a protocol for $n = 9$ with $9 + 2 + 2 = 13$ rounds and single rounds twice $\{2 \rightarrow 0\}$ and 4 times $\{3 \rightarrow 0\}$ as $f(1) = 3$. It is optimal as $\text{LB}(,0)(9) = 13$. Incrementing with $d = 2$, which give $f(2) = 1$ and then with $d = 1$ we obtain a protocol for $n = 13$ with $13 + 2 \cdot 5 = 23$ rounds. Here $\text{LB}(,0)(13) = 21$, but $\text{LB}(,1)(13) = 22 + 2/3$ and so it is optimal. But then, we are obliged to use $d = 3$ wich gives $f(d) = 2$ and then $d = 2$. So for $n = 15$ we have a protocol with $23 + 3 + 3 = 29$ rounds but $\text{LB}(,1)(15) = 28$. Table 3.5.4 gives the values of the number of rounds using the best increment and the corresponding values of $\text{LB}(,0)$ and $\text{LB}(,1)$. Note that sometimes $|A(P_n)| = \lceil \text{LB}(,1)(P_n) \rceil$ but never $|A(P_n)| = \text{LB}(,1)(P_n)$

n	7	8	9	10	11	12	13	14	15	16	17	18
$ A(P_n) $	9	11	13	16	18	21	23	26	29	31	34	37
$ A^*(P_n) $		12	14	16	18	21	23	26	28	31	34	36
$\text{LB}(,1)(P_n)$	7	$9 \frac{2}{3}$	12	$14 \frac{2}{3}$	$17 \frac{1}{3}$	20	$22 \frac{2}{3}$	$25 \frac{1}{3}$	28	$30 \frac{2}{3}$	$33 \frac{1}{3}$	36
$\text{LB}(,0)(P_n)$	9	11	13	15	17	19	21	23	25	27	29	31
Optimum	*	*	*		*		*	*	*	*	*	*

In fact we will see in the next proposition that if we start with the greedy protocol for $n = D$ containing only single rounds here $n_1 = 8$ with $|R| = 12$ (not optimal) but with 3 single rounds $\{1 \rightarrow 0\}$, $2\{2 \rightarrow 0\}$, $2\{3 \rightarrow 0\}$ and increment it we get a protocol using the 3 rounds $\{1 \rightarrow 0\}$ for $n = 11$ $|R| = 12 + 3 \cdot 2 = 18$ and for $n = 15$ using the rounds $\{2 \rightarrow 0\}$ and $\{1 \rightarrow 0\}$ created, then $|R| = 18 + 2 \cdot 5 = 28$ which attain exactly $\text{LB}(,1)(P_n)$. The values are indicated on $A^*(P_N)$ which is optimum for $n \geq 15$ by proposition 3.8 (In fact, it is also optimal for $n = 13, 14$).

Proposition 3.10 *If $q + 1$ and d_T are relatively prime and $d_T - q - 1 \leq p + 4$, then there exists N such that $A(P_n)$ is optimal, for any $n \geq N$.*

Proof: Let us start with the trivial non optimal solution R_1 for $n_1 = D = (p + 1)d_T + q + 1$ consisting of the single rounds for $i < n_1$, $i = \alpha d_T + \beta$, $1 \leq \beta \leq d_T$ $\{i \rightarrow i - d_T\}, \{i - d_T \rightarrow i - 2d_T\}, \dots, \{i - (\alpha - 1)d_T \rightarrow i - \alpha d_T\}, \{i - \alpha d_T = \beta \rightarrow 0\}$.

Starting from R_1 we increment if by always chosing if it exists a round $d \rightarrow 0$ with d minimum and $d < d_T$ and let $A(P_n)$ be the protocol obtained for n . As $q + 1$ and p are relative primes then for any d there exists an i such that $f^i(d) = d_T$. It follows that there exists some N such that $A(P_N)$ contains no round $d \rightarrow 0$ with $d < d_T$. Note that when we did an increment from some n to $n + 1$ using the round $d \rightarrow 0$ (with $d < d_T$) we add to the calls $d \rightarrow 0$ the call $d + d_I + d_T + 1 \rightarrow d + d_I + 1$ or $n \rightarrow d + d_I + 1$ if $n \leq D + d$ (In fact, the second case does not appear if we choose the d minimum as we have at least two rounds $1 \rightarrow 0$ (with $d \leq 2$) used for $D + 3$, $D + 4$ and so on). Furthermore, if the call $d \rightarrow 0$ is used only in round R_{j_0} , the distance contributions of the rounds of $A(P_N)$ in the interval $[0, D]$ are either $d_T \rightarrow 0$ or that of the 2 calls $d \rightarrow 0$ and $D = d_T + d_I + 1 \rightarrow d + d_I + 1$. The total contribution being $d + d_T - d = d_T$. So $\text{LB}(,1)(P_N)$ is exactly attained.

A computation similar to the case $q = 0$ shows that no call $d \rightarrow 0$ is used in rounds R_{j_1} if $d_T - q - 1 \leq p + 4$. ■

Corollary 3.2 *If $d_T = 3$ or 5 , we have an optimal protocol for n large enough.*

Proof: 3 and 5 being primes, and $d_T \leq 5 \leq p + 4 + q + 1$ implies by proposition 3.10 that $A(P_n)$ is optimal for $n \geq N$. ■

For $d_T = 4$, $q = 0, 2$ the result follows from proposition 3.10 but when $q = 1$, $q + 1 = 2$ is not prime with 4 and then we can not apply the proposition. We can find a protocol with one more round than $\text{LB}(1)(P_n)$ (we conjecture that this protocol is optimal which will need an improvement of the lower bound).

We have seen in this section how to obtain in some cases exact or asymptotical results by using incremental protocols. Clearly not all increments are optimal and the choice of the starting protocol is not evident as we have seen.

The aim of section 3.5.6 is to show that there exists always an optimum protocol which is increment of some protocol R . But first, in order to simplify the proofs we show that we can restrict ourselves to a specific class of protocols that we call *simple*.

3.5.5 Simple protocols

In this first part we show that optimal gathering protocols in the path with $t = 0$ can be assumed to have a certain *simple* structure. This will allow us to restrict ourselves to look for optimal solutions in this smaller class of protocols.

Lemma 3.3 *There always exists an optimal gathering protocol $R = (R_i)_{i=1}^{|R|}$ for the instance $(P_n, w, 0)$ of gathering such that:*

- (a) *If $s \rightarrow r$ is a call in the protocol R , then there exists an actual message being transmitted in the call.*
- (b) *R only performs forward calls, i.e., calls $s \rightarrow r$ such that $r < s$.*

Proof:

(a) is direct: If a protocol performs calls that transmit no message, it suffices to remove the calls that do not transmit an actual message.

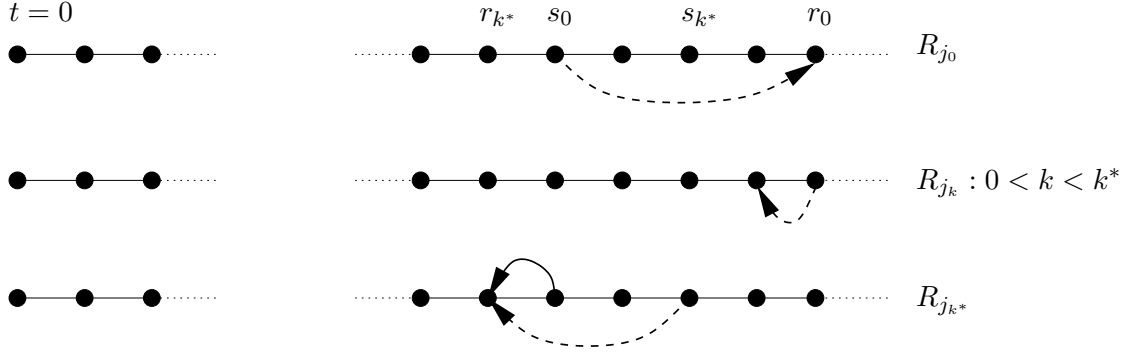


Figure 3.6: Removal of a backward call in a gathering protocol in the path. Dashed calls are deleted. Solid calls are added.

(b) To prove it, we describe a procedure that, given a gathering protocol that performs at least one backward call $s \rightarrow r, s \leq r$, it removes one backward call from the protocol and the resulting protocol has at most the same number of rounds as the original gathering protocol.

Let $R = (R_i)_{i=1}^{|R|}$ be a gathering protocol as in the statement of the lemma and let R_{j_0} be the last round such that $s_0 \rightarrow r_0 \in R_{j_0}$ with $s_0 < r_0$ and name m the message being transmitted in this call (if there are two backward calls in round R_{j_0} we pick anyone). Then there exist rounds $R_{j_1}, R_{j_2}, \dots, R_{j_{j_k}}$ and calls $s_q \rightarrow r_q \in R_{j_q}$ such that $j_q > j_{q-1}$, $s_q > r_q$ and $s_q = r_{q-1}$, for $q = 1, \dots, j_k$ where m is transmitted and with $r_k = t = 0$. (See Figure 3.6.)

It follows that there exists k^* such that $r_{k^*} < s_0 \leq s_{k^*}$ and we can remove the call $s_q \rightarrow r_q$ from round R_{j_q} for any $q \leq k^*$ and add call $s_0 \rightarrow r_{k^*}$ to round $R_{j_{k^*}}$ ($s_0 \leq s_{k^*} \Rightarrow d(s_0, r_{k^*}) \leq d(s_{k^*}, r_{k^*}) \leq d_T$ and therefore $s_0 \rightarrow r_{k^*}$ is a call) . In fact

- removing a call from the rounds $R_{j_k}, k = 1, \dots, k^*$ produces valid rounds because it does not introduce interference (at most one of these rounds may become empty, in which case the round is completely removed), and
- $s_0 \rightarrow r_{k^*}$ can be added to the round $R_{k^*} - \{s_{k^*} \rightarrow r_{k^*}\}$, because (i) R_{k^*} is a round, thus r_{k^*} is not under the interference of any other transmitter (so it is available for receiving, in our case, from s_0); and (ii) s_0 does not interfere any other receiver, as if $s \rightarrow r \in R_{k^*} - \{s_{k^*} \rightarrow r_{k^*}\}$ then either $r > s_{k^*} \geq s_0 \Rightarrow d(r, s_0) \geq d(r, s_{k^*}) > d_I$, or $r < s < r_{k^*} < s_0 \Rightarrow d(r, s_0) > d(s, r_{k^*}) > d_I$ (we have $r < s$ because $j_{k^*} > j_0$ and R_{j_0} is the last round with a backward call).

Property (b) follows from the fact that the procedure can be applied iteratively, until every backward call has been removed. ■

We will say that a protocol that satisfies (a) and (b) is *simple* and from this point on we will restrict ourselves to the class of protocols that are simple. Indeed, this subset of all possible protocols always contains an optimal solution for the path P_n , hence there is no loss of generality.

3.5.6 Existence of optimal incremental protocols

We will show how to obtain for $n \geq (p+1)d_T + 1$ from a protocol R for $n+1$ a protocol R^- for P_n such that there exists an increment S of R^- with a number of rounds at most that of R . In particular if R is optimum, then S is optimum.

Lemma 3.4 *For $n \geq D+1$, consider a simple gathering protocol $R = (R_j)_{j=1}^{|R|}$ for the path P_{n+1} . Then, there exist protocols R^-, S , such that*

- (i) R^- is a gathering protocol for the instance $(P_n, 0)$,
- (ii) S is a gathering protocol for the instance $(P_{n+1}, 0)$,
- (iii) S is an increment of R^- obtained by construction Inc of lemma 3.1.
- (iv) $|S| \leq |R|$.

Proof: The sketch of the proof is the following. First, we construct the protocol R^- for $(P_n, 0)$ based on R . Afterthat, we use Lemma 3.1 to obtain the protocol S for $(P_{n+1}, 0)$ as an incremental protocol of R^- .

In order to construct R^- , we mainly remove the calls (and potentially full rounds, if they consist of a single call) that transmit $m(n)$, that is the message corresponding to the last node in P_{n+1} , in such a way that we can guarantee not only that R^- is valid, but also that we can estimate its length. However, this has to be done carefully, because if $m(n)$ is not the very last message to reach the sink, simply removing calls may not reduce enough the length of the resulting protocol.

Let us do now the proof and construct the protocol R^- starting from R . Let R_{j_1} in R be the last round containing a call $c^* = s^* \rightarrow r^*$ such that $s^* > (p+1)d_T$ and $r^* \leq (p+1)d_T$. Let u^* be the node such that $m(u^*)$ is the message transmitted in $s^* \rightarrow r^*$.

After rounds R_{j_1} we are left with an instance such that $w(u) = 0$ for any $u > (p+1)d_T$ and so we have now an instance corresponding to a path of length $n' \leq (p+1)d_T + 1$ and some weight function w' . We replace all the rounds appearing after R_{j_1} by those of the proof of proposition 3.2 getting a new protocol R' with a number of rounds less than or equal to R . In particular we have that $m(u^*)$ is transmitted to the sink in exactly $(p+1)$ rounds all except the last one being of length d_T . Furthermore, to simplify the rest of the proof, we rearrange the rounds such that the rounds transmitting $m(u^*)$ are the last rounds of the protocol. Let us denote then $R_{j_2}, \dots, R_{j_{p+2}}$.

Now, from R' we will construct R^- . We distinguish two cases depending on the node u^* .

- If u^* corresponds to the last node n of P_{n+1} , then we simply remove the rounds $R_{j_2}, \dots, R_{j_{p+2}}$.
- On the contrary, if $u^* \neq n$, we will change the message transmitted by the last rounds in such a way that $R_{j_2}, \dots, R_{j_{p+2}}$ become rounds transmitting $m(n)$. Recall that u^* is such that $u^* > (p+1)d_T$.

As $m(u^*)$ is the last message which enters into the zone of nodes between $t = 0$ and $(p+1)d_T$, there exists a round before R_{j_1} such that $m(n)$ becomes closer to the sink than $m(u^*)$. We denote R_{j_0} this round. We denote also $s_0 \rightarrow r_0$ the call in R_{j_0} which transmits $m(n)$ and such that $m(u^*)$ is placed in one node v_0 between s_0 and r_0 .

Now we exchange the messages of n and u^* as follows. We replace the call (s_0, r_0) in R_{j_0} by the call (v_0, r_0) which will transmit $m(u^*)$. Moreover all the rounds after R_{j_0} which transmit $m(n)$ become calls transmitting $m(u^*)$. We add first after R_{j_0} a single round $R'_{j_0} = \{s_0 \rightarrow v_0\}$ transmitting $m(n)$ and then all the rounds after R_{j_0} which transmit $m(u^*)$ become a call transmitting $m(n)$. So we are in a case similar to the first one and we delete R'_{j_0} and $R_{j_2}, \dots, R_{j_{p+2}}$ (which now carries $m(n)$).

After removing the rounds, we can also remove the remaining calls (not rounds) transmitting $m(n)$, because these calls will be no longer used in R^- . Note that if R_{j_1} contains only one call, it will be completely removed.

In summary, the resulting protocol R^- is a valid protocol for the instance $(P_n, 0)$. Moreover, whatever the number of calls of R_{j_1} , protocol R^- contains at least $p+1$ rounds less than R .

Now, we can construct the new protocol S , for the instance $(P_{n+1}, 0)$ by means of incrementing R^- and using Lemma 3.1. To do that, we will set $S = \text{Inc}(R^-, d)$ and choose parameter d conveniently. We distinguish two cases:

- If round R_{j_1} was removed (that is, it contained only one call in the original protocol R), then we define d to be the sender from the call of the last round of R^- . In other words, $d = s$ where $s \rightarrow 0$ is the last call of R^- . As in this case $|R^-| = |R| - (p+2)$, then Lemma 3.1 guarantees $|S| \leq |R^-| + p + 2 = |R|$.
- If round R_{j_1} was not removed (it has exactly two calls in the original protocol R), it consists only of a call $s_1 \rightarrow r_1$. By definition of R_{j_1} , node s_1 satisfies that $s_1 \leq d_T - q - 1$ and by the choice of R' and R^- we have that $r_1 = 0$. So, we apply Lemma 3.1 with $d = s_1$ and get an increment S such that $|S| = |R^-| + p + 1 = |R|$.

In this way we have shown that the protocol S satisfies that $|S| \leq |R|$. ■

We conjecture that:

Conjecture 3.1 *For any $n = D + 1$ there exist an optimal protocol S for P_n obtained by repeated applications of construction Inc increments of some protocol R for P_{D+1} .*

If conjecture 3.1 is true, we could prove the main conjecture:

Conjecture 3.2 Unitary Minimum Time Gathering in the path P_n with $t = 0$ is polynomial in the length n of the path.

3.6 Gathering into an arbitrary vertex of the path

So far, we have discussed only the case where $t = 0$. In this section we remove this constraint and take $0 < t < n - 1$.

In [BCY06, BCY09] results are given for $d_T = 1$ and an optimal solution is given for $d_I \in \{1, 2, 3, 4\}$. So, for $d_T > 1$ there is no hope to find general optimal protocols. However, in this section we will give a 1^+ -approximation in a similar manner than section 3.4.

Let us refer to the vertices $i = 0, \dots, t - 1$ as the *left side*, and $j = t + 1, \dots, n - 1$ as the *right side*. Without loss of generality we will suppose that $t \leq n - 1 - t$.

Proposition 3.11

$$g_{d_I, d_T}(P_n, t) \geq \frac{d_I + d_T + 1}{d_T} (\max[t, n - 1 - t] - D + 1) + \frac{(d_I + d_T + 1)(d_I + d_T)}{2d_T}.$$

Proof: Recall that we assume that $\max[t, n - 1 - t] = n - 1 - t$ and consider the interval of vertices $I = \{t, t + 1, \dots, t + D - 1\}$, with $D = d_I + d_T + 1$. As in proposition 3.5, we have that for $i \in I$, a message originated in i has to travel (within I) a distance $i - t$ to the sink, and that for $j \geq t + D$, a message originated in j has to travel a distance D (within I) in order to reach the sink t . However, we observe that even if two messages can move inside I , overall these two messages cannot progress more than d_T vertices toward the sink, from where it follows that

$$g_{d_I, d_T}(P_n, t) \geq \frac{D}{d_T} (n - t - D) + \frac{D(D - 1)}{2d_T}.$$

■

Notice that this bound assumes perfect synchronization between the calls in the two sides, i.e., that gathering messages from the shortest side does not delay gathering messages from the longest side. In general, transmitting messages in one side produces interference in the other side, thus some extra rounds may be required (see [BCY06]).

Now we want to extend the 1^+ -approximation result of Section 3.4.2. Since Proposition 3.11 establishes that the number of rounds required to gather the path that has two sides is, roughly, lower bounded by the number of rounds needed to gather its longest side, the algorithm we introduce works as follows: Given the protocol that gathers the longest side, its rounds are modified by adding calls in the shorter side so messages coming from that side are gathered into the sink at the same

time. Moreover, this is done in such a way that when finished, we can guarantee that only the vertices in $\{t+1, \dots, t+D-1\}$ have messages that are still unknown for the sink, and that they have at most one message. Because these are $O(1)$ vertices, each at a distance $O(1)$ from the sink, we deduce that we can gather these messages in constant time.

Theorem 3.2 *Let $n \in \mathbb{N}$. For the Unitary Minimum Time Gathering problem where the base graph is a path P_n , the interference distance is d_I , and transmission distance is d_T , there exists a 1^+ -approximation.*

Proof: First, let $\ell_1 = t, \ell_2 = n - 1 - t$ and recall that $\ell_1 \leq \ell_2$. If $D \geq \ell_2$, the size of the network is bounded so to gather a message at a distance i from t requires at most $c(i) = \lceil i/d_T \rceil \leq \lceil D/d_T \rceil \leq 2 + d_I/d_T$ rounds. Therefore to gather the path in this case can be done in at most $2 \sum_{i=1}^D c(i) \leq 2(2 + d_I/d_T)D$ rounds, which does not depend on n .

To analyze the case $\ell_2 \geq D$, let us rename the vertices of the path in such a way that vertex i becomes $i - t$, so the sink is vertex 0, the left side consists of vertices $-\ell_1, -(\ell_1 - 1), \dots, -1$ and the right (and longer) side corresponds to vertices $1, 2, \dots, \ell_2$.

We define $B_i = \{i + kD \rightarrow \max[0, i + kD - d_T] : k \geq 0, i + kD \leq \ell_2\}$. We have that, after applying $B_i, i = 1, \dots, D$, each message at the right side of the sink either has reached the sink or it has advanced a distance d_T towards the sink. Similarly, we define $A_i = \{i - D - 1 - kD \rightarrow \min[0, i - D - 1 - kD + d_T] : k \geq 0, i - D - 1 - kD \geq -\ell_1\}$ and have that after applying $A_i, i = 1, \dots, D$, each message at the left side of the sink has reached it or it moved a distance d_T towards the sink.

The protocol we use is calculated by the following algorithm (recall that $\ell_1 \leq \ell_2$):

Input: $n, \ell_1, \ell_2, d_I, d_T$

```

1 while  $\ell_2 - 1 \geq D$  do
2   for  $i = 1$  to  $D$  do
3     | Apply  $A_i \cup B_i$ .
4   end
5    $\ell_1 \leftarrow \ell_1 - d_T, \ell_2 \leftarrow \ell_2 - d_T$ .
6 end
7 Gather each message independently, using a shortest path to the sink.
```

Algorithm 2: Solves gathering in P_n for arbitrary t .

Indeed, this algorithm is almost identical to Algorithm 1, the only difference is that we have replaced “Apply $\{i + kD \rightarrow \max[0, i + kD - d_T] : k \geq 0, i + kD \leq n - 1\}$ ” with “Apply $A_i \cup B_i$ ” (in fact, B_i is precisely the set $\{i + kD \rightarrow \max[0, i + kD - d_T] : k \geq 0, i + kD \leq n - 1\}$ where we replaced $n - 1$ with ℓ_2).

We only need to check that each call in A_i is compatible with each call in B_i . Indeed, for any i the closest calls are $i \rightarrow \max[0, i - d_T] \in B_i$ and $i - D - 1 \rightarrow \min[0, i - D - 1 + d_T] \in A_i$, and we

have $d(i, \min[0, i - D - 1 + d_T]) = D + 1 - d_T = d_I + 1$ and similarly $d(\max[0, i - d_T], i - D - 1) = D + 1 - d_T = d_I + 1$.

Because Step 3 requires $O(1)$ rounds, we focus on Step 1. This step requires at most $\max\left[0, D \left\lceil \frac{\ell_2 - D}{d_T} \right\rceil\right] \leq D \frac{\ell_2}{d_T}$ rounds. Adding up the overall number of rounds performed by the algorithm and using that $\ell_2 = \max[t, n - 1 - t]$ we get

$$g_{d_I, d_T}(P_n, t) \leq \frac{D}{d_T}(\max[t, n - 1 - t] - D + 1) + 2(2 + d_I/d_T)D.$$

But the second term does not depend on n , hence by using Proposition 3.11, we obtain that this is a 1^+ -approximation. ■

3.7 Conclusions

We studied the problem of finding the minimum number of rounds needed to gather information in a path in the unitary case. This problem appears to be a difficult one (much more difficult than we thought when starting the research).

We have obtained a $+1$ -approximation for any position of the sink. When the sink is an end vertex of the path, we have also described an incremental procedure which produces optimal protocols for small d_T .

We conjecture that this procedure always give optimal protocols. One challenging problem will be to prove that the minimum time gathering problem can be solved in polynomial time in a general or at least in the unitary case or even with the sink at the end of the path. Extending the result to other topologies like trees will be of interest, although optimal solutions might be difficult to obtain in view of the complicated proofs of the case $d_T = 1, d_I = 1$.

Chapter 4

Minimum delay data gathering in radio networks [BNRR09a, BNRR09b]

4.1 Introduction

We address here the challenging problem of gathering information in a Base Station (denoted BS) of a wireless multi hop grid network when interferences constraints are present. This problem is also known as data collection and is particularly important in sensor networks, but also in access networks.

The communication network is modeled by a graph. In this paper we focus on grid topologies as they model well both access networks and also random networks (which approximatively behave like if the nodes were on a grid [KLNP05]). We assume that the time is slotted and that during each time slot, or *step*, a transmission that is activated between two neighboring nodes can transport at most one data item (referred in what follows as a message). Each vertex of the grid may have any number of messages to transmit, including none. We also suppose that each device (sensor, station, ...) is equipped with an half duplex interface: a node cannot both receive and transmit during a step. As for an example, this is a relevant model of mono-frequency smart antennas radio system: at any step, each device can configure its antenna array to shape a beam and reach any of its neighbours without interfering with others. Nevertheless, sending a message prevents a node from receiving another one because, among other causes, of near-far effects. We refer to this model as the *smart-antennas model*.

During any step a set of pairwise non interfering transmissions can be achieved, and such a set form a matching of the grid. Our aim is to design algorithms to do a gathering under such hypotheses, which minimize the minimum number of steps needed to send all messages to BS, this *completion*

time is also denoted *makespan* of the call scheduling.

Following the work of Revah and Segal [RS07], we focus on the specific case of “*open-grid*”, that is a grid network with the base station at a corner (say lower-left w.l.g.) and no message is generated by a node on the lower and left borders of the grid (line 0 and column 0). As a matter of fact, the case of *closed-grid*, which is by the way similar to having the BS anywhere on the grid, is more complex and cannot be solved to optimality with shortest paths¹. Even though we know how to adapt our weakest algorithm, the value of the lower-bound in that case is still under active investigation [BGN⁺09].

4.1.1 Related Work

A lot of authors have studied the gathering problem under various assumptions (see the surveys [BKK⁺09b] and [Gar07]).

In [FFM04b], the smart antennas model is considered with the extra constraint that non buffering is allowed in intermediary nodes: when a node receives a message at some step, it must transmit it during the next step. In this setting, optimal polynomial-time algorithms are presented for path and tree topologies [FFM04b, RS08]. The work of [FFM04b] has been extended to general graphs in [GR06b] and [GR09] but in the uniform case where each node has exactly one message to transmit. The case of *open-grids* is considered in [RS07] where a 1.5-approximation algorithm is presented. The gathering problem has also been studied when nodes can both emit and receive a message during the same step. When no buffering is allowed, this kind of routing is known as the hot-potato routing and it is considered in [BHW00, MPS95].

The case of omnidirectional antennas has been extensively studied. In this model, nodes can transmit at any of their neighbours at distance $d_T \geq 1$ but any emission creates some interferences. More precisely, when a node v transmits, any node at distance at most $d_I \geq d_T$ of v cannot receive a message from another node than v during the same step. Moreover, any node has to transmit at least one message and buffering is allowed. In this setting, computing the makespan is NP-hard [BGK⁺06d]. A 4-approximation algorithm and lower bounds for general graphs are also provided in [BGK⁺06d]. A 4-approximation algorithm has been proposed to handle the online version [BKMS08a]. In [BP05], the case of grids is considered when $d_T = 1$: an optimal polynomial-time algorithm is provided when BS stands at the center of the grid. Gathering in grids is also considered within a continuous model in [GPRR08].

4.1.2 Our results

We focus on the gathering problem in *open-grids*. We provide a very simple algorithm that schedules all messages within a lower bound of the makespan plus two steps, and a more involved +1

¹The authors would like to thanks Prof. Frédéric Guinand who raised this question.

approximation algorithm. As a matter of fact, we prove that our algorithms delay each message by at most 1 or 2 steps (depending on which algorithm) from a given scheduling which would be optimal if there were no interference (hence lower bounding the real makespan). We also provide a linear-time (in the number of vertices of the grid) distributed algorithm for the +2-approximation algorithm. Besides, our algorithms need no buffering, which considerably improves on existing algorithms. Our algorithms are presented in the smart antennas model, even though we conjecture that they can be extended to other distance-based interference model.

One helpful idea is to actually study the related one-to-many personalized broadcast problem in which the BS wants to communicate different data items to some other nodes in the network. Using this framework, protocols may be described easier. Solving the above dissemination problem is equivalent to solve data gathering in sensor networks. Indeed, let T denote the makespan (delay), that is, the largest step used by a personalized broadcast algorithm; a gathering schedule with delay T consists in scheduling a transmission from node y to x during slot t iff the broadcasting algorithm schedules a transmission from node x to y during slot $T - t + 1$, for any t with $1 \leq t \leq T$.

4.2 Preliminaries

From now on, we consider the equivalent problem of personalized broadcasting where BS has to transmit messages to some destination nodes in the open grid.

4.2.1 Notations

In the following, we consider a $N \times N$ grid $G = (V, E)$ where vertices are given their natural coordinates. The base station BS , also called *the source*, has coordinates $(0, 0)$, and any vertex v has coordinates (x_v, y_v) . A vertex v is *above* (resp., *below*) $w \in V$ if $y_v \geq y_w$ (resp., if $y_v \leq y_w$). Similarly, v is *to the right* (resp., *to the left*) of $w \in V$ if $x_v \geq x_w$ (resp., if $x_v \leq x_w$). Finally, a vertex v is *nearer to the source* than $w \in V$ if $d(v, BS) \leq d(w, BS)$, where $d(u, v)$ denotes the classical distance between nodes u and v .

We consider a set of $M \geq 0$ messages \mathcal{M} that must be sent from the source BS to some destination nodes. Let $dest(m) \in V$ denote the destination of $m \in \mathcal{M}$. A message $m \in \mathcal{M}$ is *lower* (resp., *higher*) than $m' \in \mathcal{M}$ if $dest(m)$ is below (resp., above) $dest(m')$. A message m is *righter* (resp., *leftier*) than m' , if $dest(m)$ is to the right (resp., to the left) of $dest(m')$. We use $d(m)$ to denote $d(dest(m), BS)$, and $m \preceq m'$ if $dest(m)$ is nearer to the source than $dest(m')$, that is, if $d(m) \leq d(m')$. We suppose in what follows that the messages are ordered by non increasing distance of their destination nodes, and we note $\mathcal{M} = \{m_1, \dots, m_M\}$ where $m_i \preceq m_j$ for any $i \leq j \leq M$, so $d(m_1) \geq d(m_2) \geq \dots \geq d(m_M)$.

$S \odot S'$ denotes the sequence obtained by concatenation of two sequences S and S' .

4.2.2 Lower bound

Consider a model without interferences, i.e., any node can receive and transmit simultaneously, but where the source can only send one message per step. Whatever be the broadcasting scheme, a message m sent at step $t \geq 1$ will be received at step $t' \geq d(m) + t - 1$. A broadcasting scheme is said *greedy* if, given an ordered sequence \mathcal{S} of the messages, the source sends one message per step, in the ordering \mathcal{S} , and each message follows a shortest path toward its destination node. Note that, in the model without interferences, if the messages follow shortest paths, a vertex will never receive more than one message per step.

Definition 4.1 $LB = \max_{i \leq M} d(m_i) + i - 1$.

Lemma 4.1 *In the model without interferences, when the source emits at most one message per step, a greedy algorithm following the ordered sequence of messages (m_1, m_2, \dots, m_M) is optimal, with makespan LB .*

Proof: Clearly, sending the messages following the sequence (m_1, m_2, \dots, m_M) along shortest paths achieves such a makespan. Let us consider an optimal schedule of the messages (s_1^*, \dots, s_M^*) different from (m_1, m_2, \dots, m_M) and let $i \geq 1$ be the smallest integer such that $s_i^* \neq m_i = s_j^*$ ($j > i$). Sending the messages following the sequence $(s_1^*, \dots, s_{i-1}^*, s_j^*, s_{i+1}^*, \dots, s_{j-1}^*, s_i^*, s_{j+1}^*, \dots, s_M^*)$ does not increase the makespan: indeed, only the i^{th} and j^{th} messages differ and $\max\{d(s_j^*) + i - 1, d(s_i^*) + j - 1\} \leq d(s_j^*) + j - 1$ because $d(m_i) = d(s_j^*) \geq d(s_i^*)$ and $j > i$. By iterating this process, we get that the ordering of the sequence (m_1, m_2, \dots, m_M) is also optimal. ■

Corollary 4.1 *In the smart antennas model, no algorithm can achieve a makespan less than LB .*

4.3 Personalized Broadcasting Algorithms

In this section, we present a very simple broadcasting scheme that we prove to be sufficient to obtain a good approximation of the optimal makespan. We then refine it to obtain an almost optimal algorithm.

These algorithms use Horizontal-Vertical routing schemes, hence proving that fancier shortest path routing is worthless with respect to the minimum makespan objective.

4.3.1 Horizontal-Vertical broadcasting

Given a message whose destination node v has coordinates (x, y) , the message is sent *horizontally* to v if it follows the shortest path from BS to v passing through $(x, 0)$. The message is sent *vertically* if it follows the shortest path from BS to v passing through $(0, y)$.

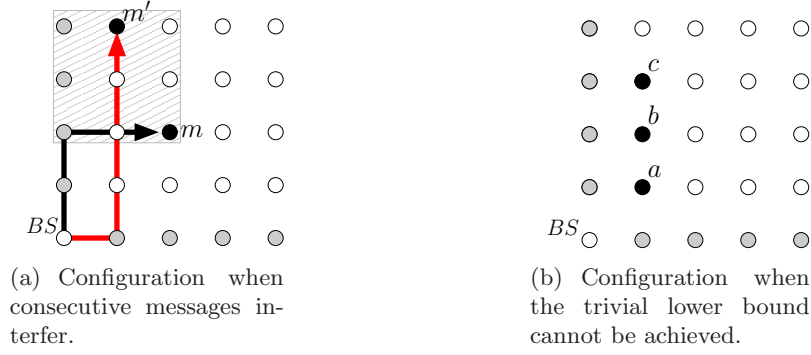


Figure 4.1: Two particular configurations

Definition 4.2 A Horizontal-Vertical broadcasting scheme, or HV-scheme, takes an ordering \mathcal{S} of \mathcal{M} as an input and proceeds as follows. A direction, horizontal or vertical, is chosen for the first message. Then, the source sends one message every step in the ordering \mathcal{S} , alternating horizontal and vertical messages.

Let us do some easy remarks about any HV-scheme. Consider two distinct messages sent by the source x time-slots apart. Since these messages follow shortest paths, while the first message has not reached its destination, both messages are separated by a distance at least x . Hence,

Claim 4.1 In a HV-scheme, only consecutive messages may interfere.

Let us characterize forbidden and acceptable configurations in HV-scheme. Assume that two messages are sent consecutively. It is possible to guess the respective positions of their destination nodes by knowing whether both messages interfere or not. In Figure 4.1(a), messages in the grey part are those higher and lefter than the message m . Figure 4.1(a) illustrates the following Fact.

Claim 4.2 Let m, m' be 2 messages sent consecutively by a HV-scheme, with m sent vertically and m' sent horizontally. Messages m and m' interfere if and only if their destinations are distinct and m' is higher and lefter than m .

Before continuing, let us remark that there exist configurations for which no gathering protocol can achieve better makespan than $LB + 1$. Figure 4.1(b) represents such a configuration. Indeed, in Figure 4.1(b), the three destinations a, b and c have coordinates $(1, 1)$, $(1, 2)$ and $(1, 3)$, and $LB = 4$. However, to achieve such a makespan, the first message must be sent to c (because c is at distance 4 from BS) and the second message must be sent to b (because the message start after the first step and must go at distance 3). To avoid collision, the only possibility is to send the first message vertically, and the second one horizontally. But then, the last message cannot reach a before step 5.

Input: $\mathcal{M} = \{m_1, \dots, m_M\}$, the set of messages ordered in non increasing distance order
Output: (s_1, \dots, s_M) an ordered sequence of \mathcal{M} satisfying (i) and (ii)
begin
 Case $M = 0$ **return** \emptyset
 Case $M = 1$ **return** (m_1)
 Case $M \geq 2$
 Let q be the lowest message in $\{m_{M-1}, m_M\}$ and let r be the other one
 if $M = 2$ **return** (q, r)
 else let $\mathcal{O} \odot p = \text{TwoApprox}(\{m_1, \dots, m_{M-2}\})$
 if p is higher than q **return** $\mathcal{O} \odot (p, q, r)$
 else **return** $\mathcal{O} \odot (m_{M-1}, p, m_M)$
end

Figure 4.2: Algorithm *TwoApprox*

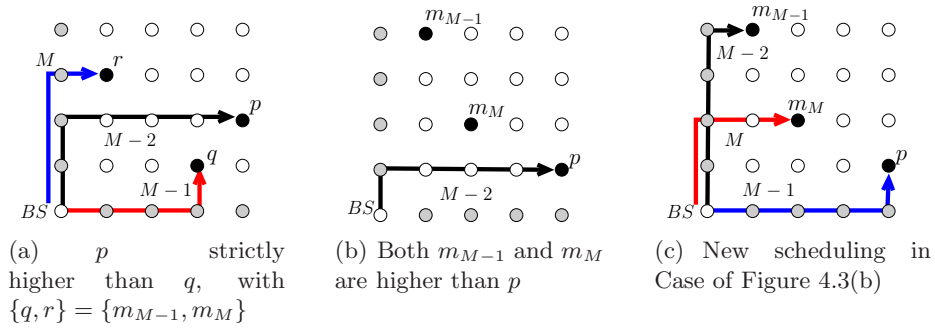


Figure 4.3: $M - 2$ messages have been scheduled, finishing with the one to $p \in \{m_{M-2}, m_{M-3}\}$. When the next two messages must be scheduled, two cases occur according to the position of m_{M-1} and m_M relatively to p . In the figures, an arrow with label i represents the route of the i^{th} message.

4.3.2 +2 approximation

Recall that (m_1, \dots, m_M) denotes the ordered sequence of the messages in the non increasing ordering of the distance to their destinations. In this section, we give the Algorithm *TwoApprox*, depicted in Figure 4.2, that computes an ordered sequence $\mathcal{S} = (s_1, \dots, s_M)$ of the messages satisfying the two following properties:

(i) HV-scheme(\mathcal{S}) broadcasts the messages without collisions, sending the last message vertically, and

(ii) $s_i \in \{m_{i-2}, m_{i-1}, m_i, m_{i+1}, m_{i+2}\}$ for any $i \leq M$, and $s_M \in \{m_{M-1}, m_M\}$

Theorem 4.1 *Algorithm TwoApprox computes an ordering \mathcal{S} of the messages satisfying properties (i) and (ii) and so HV-scheme(\mathcal{S}) achieves makespan at most $LB + 2$.*

Proof: To prove the correctness of Algorithm *TwoApprox*, we proceed by induction on M . If $M \leq 2$, the result holds obviously. Let us assume that the ordering of the sequence computed by *TwoApprox*($\{m_1, \dots, m_{M-2}\}$) satisfies properties (i) and (ii). Let p be the last message of this sequence. By the induction hypothesis, $p \in \{m_{M-3}, m_{M-2}\}$ is sent vertically. Let t be the message before p in this sequence. By Fact 4.2, p must be higher or left than t . The sequence is denoted by $\mathcal{O} \odot p = \mathcal{O}' \odot (t, p)$.

Let q be the lowest message in $\{m_{M-1}, m_M\}$ and let r be the other one. We consider two cases depending on the positions of p , q and r .

- a) **Case p is higher than q .** It is sufficient to send q horizontally at step $M - 1$, and r vertically at step M . This case is depicted in Figure 4.3(a). Indeed, by Fact 4.1 only p and q , or q and r may interfere. By Fact 4.2, there are no interferences. It is easy to check that $\mathcal{O} \odot (p, q, r)$ satisfies (i) and (ii).
- b) **Case q and r are higher than p .** Since $q, r \preceq p$ (i.e. q, r closer to BS than p), they are higher and left than p . This case is depicted in Figure 4.3(b). In this case, instead of sending p at step $M - 2$, the source sends m_{M-1} vertically at step $M - 2$, then p horizontally at step $M - 1$, and then m_M vertically at step M . The transformation is depicted in Figure 4.3(c). Clearly, $\mathcal{O} \odot (m_{M-1}, p, m_M)$ satisfies (i) and (ii). By Fact 4.1 only t and m_{M-1} , or m_{M-1} and p , or p and m_M may interfere. Since m_{M-1} is higher and left than p that is higher or left than t , by Fact 4.2, m_{M-1} interferes neither with t nor with p . Similarly, m_M is higher and left than p and these messages do not interfere.

■

4.3.3 +1 approximation

In this section, we give the Algorithm *OneApprox*, depicted in Figure 4.4, that computes an ordered sequence $\mathcal{S} = (s_1, \dots, s_m)$ of the messages satisfying:

- (i) HV-scheme(\mathcal{S}) broadcasts the messages without collisions, sending the last message vertically, and
- (iii) $s_i \in \{m_{i-1}, m_i, m_{i+1}\}$ for any $i \leq M$ (in particular, either $s_M = m_M$, or $s_M = m_{M-1}$ and $s_{M-1} = m_M$).

An ordered sequence $\mathcal{S} = (s_1, \dots, s_M)$ of \mathcal{M} satisfying (i) and (iii) is said *valid*. Clearly, for any valid sequence \mathcal{S} , HV-scheme(\mathcal{S}) achieves makespan at most $LB + 1$. To prove that Algorithm *OneApprox* computes a valid ordered sequence of \mathcal{M} , we proceed by induction on M . Roughly, starting from a valid ordered sequence of $\{m_1, \dots, m_{M-2}\}$, the algorithm includes m_{M-1} and m_M

Input: $\mathcal{M} = \{m_1, \dots, m_M\}$, the set of messages ordered in non increasing distance order
Output: (s_1, \dots, s_M) an ordered sequence of \mathcal{M} such that $m_i \in \{s_{i-1}, s_i, s_{i+1}\}$ for any $i \leq M$

```

begin
  Case  $M = 0$  return  $\emptyset$ 
  Case  $M = 1$  return  $(m_1)$ 
  Case  $M \geq 2$ 
    Let  $q$  be the lowest message in  $\{m_{M-1}, m_M\}$  and let  $r$  be the other one
    if  $M = 2$  return  $(q, r)$ 
    else let  $\mathcal{O} \odot p = OneApprox(\{m_1, \dots, m_{M-2}\})$ 
      if  $p$  is higher than  $q$  return  $\mathcal{O} \odot (p, q, r)$ 
      else if  $p = m_{M-2}$  return  $\mathcal{O} \odot (m_{M-1}, p, m_M)$ 
      else
        /* This last case may occur only if  $M > 3$  */
        Let  $(s_1, \dots, s_{M-4}) \odot (m_{M-2}, m_{M-3}) = OneApprox(\{m_1, \dots, m_{M-2}\})$ 
        return  $MakeValid((s_1, \dots, s_{M-4}) \odot (m_{M-3}, m_{M-1}, m_{M-2}, m_M), 2)$ 
end

```

Figure 4.4: Algorithm *OneApprox*

Input: A $(j-1)$ -good (see def. 4.3) sequence $\mathcal{O} = (s_1, \dots, s_M)$ of a set of messages $\{m_1, \dots, m_M\}$, and an integer j , $1 < j \leq \lfloor M/2 \rfloor$.
Output: A valid sequence of \mathcal{M}

```

if  $s_{M-2j}$  and  $s_{M-2j+1}$  do not interfere
  /* In particular, this case occurs if  $M - 2j = 0$  */
  return  $\mathcal{O}$ 
else if  $s_{M-2j} = m_{M-2j}$ 
  /* In particular, this case occurs if  $M - 2j = 1$  */
  return
   $(s_1, \dots, s_{M-2j-2}) \odot (s_{M-2j-1}, s_{M-2j+1}, s_{M-2j}, s_{M-2j+2}) \odot (s_{M-2j+3}, \dots, s_M)$ 
else return
  /* This last case may occur only if  $M - 2j \geq 2$  */
  /* Note that, in this case,  $s_{M-2j} = m_{M-2j-1}$  and  $s_{M-2j-1} = m_{M-2j}$  */
   $MakeValid((s_1, \dots, s_{M-2j-2}) \odot (s_{M-2j}, s_{M-2j+1}, s_{M-2j-1}, s_{M-2j+2})$ 
     $\odot (s_{M-2j+3}, \dots, s_M), j+1)$ 
end

```

Figure 4.5: MakeValid

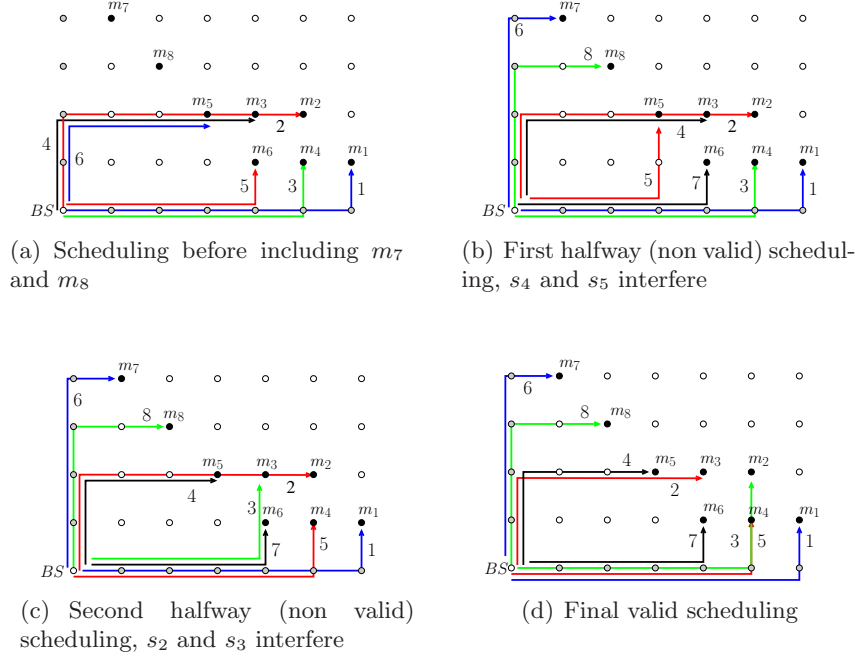


Figure 4.6: Recursive modifications of the scheduling

in this ordered sequence. Then, either the obtained sequence \mathcal{S} is valid, or it is 1-good, where the notion of *i-goodness* is defined as follows:

Definition 4.3 Let $i \in \{1, \dots, \lfloor M/2 \rfloor - 1\}$. An ordered sequence $\mathcal{S} = (s_1, \dots, s_M)$ is *i-good* if

- $s_{M-2i-1} = m_{M-2i-1}$, $s_{M-2i} = m_{M-2i+1}$ and $s_M = m_M$, and
- \mathcal{S} satisfies properties (i) and (iii) but s_{M-2i-2} may interfere with s_{M-2i-1} .

In the latter case, subprocedure *MakeValid* (see Figure 4.5) is recursively applied to \mathcal{S} increasing the parameter of goodness until either a valid sequence is obtained or we arrive to an $(\lfloor M/2 \rfloor - 1)$ -good sequence. But, by definition, a $(\lfloor M/2 \rfloor - 1)$ -good sequence is always valid.

We now detail the execution of Algorithm *OneApprox* on the example depicted in Figure 4.6. In this example, BS must send 8 messages $\{m_1, \dots, m_8\}$ to distinct vertices in a 6×6 -grid. Algorithm *OneApprox* first computes a valid ordered sequence $(s_1, \dots, s_6) = (m_1, m_2, m_4, m_3, m_6, m_5)$ of first 6 messages. This scheduling is depicted in Figure 4.6(a). Then, the positions of m_7, m_8 and $s_6 = m_5$ are compared. In the example, m_8 is the lowest message among m_7 and m_8 , and it is higher than s_6 . Moreover, $s_6 \neq m_6$. Hence, Algorithm *OneApprox* applies Subprocedure *MakeValid* to integer $j = 2$ together with the ordered sequence $(s_1, \dots, s_4) \odot (m_5, m_7, m_6, m_8) = (m_1, m_2, m_4, m_3, m_5, m_7, m_6, m_8)$. The scheduling corresponding to this sequence is depicted in Figure 4.6(b). It is easy to check that this sequence is 1-good: in particular, it is valid except for the interference between m_3 and m_5 . Note that the integer variable j in the input of Subprocedure

MakeValid simply indicates that the interference may appear between the $M - 2j^{th}$ and the $M - 2j + 1^{th}$ messages of the given sequence. The goal of Subprocedure *MakeValid* is to locally modify the sequence in order to remove interference between the $M - 2j^{th}$ and the $M - 2j + 1^{th}$ messages. However, a new interference may appear between the $M - 2(j + 1)^{th}$ and the $M - 2(j + 1) + 1^{th}$ messages of the obtained sequence, in which case Subprocedure *MakeValid* is recall recursively. Such a situation occurs in the example. Indeed, in the sequence $(m_1, m_2, m_4, m_3, m_5, m_7, m_6, m_8)$, m_3 and m_5 interfere and the fourth message of this sequence is not m_4 . Then, Subprocedure *MakeValid* is applied to the sequence $(m_1, m_2) \odot (m_3, m_5, m_4, m_7) \odot (m_6, m_8)$ with $j = 3$. This sequence is depicted in Figure 4.6(c) and is 2-good since m_2 and m_3 interfere. Note that the second message of this sequence interferes and that this message is actually m_2 . Therefore, the next call to Subprocedure *MakeValid* only exchanges m_2 and m_3 (Case 2 of the subprocedure) and returns the ordered sequence $(m_1, m_3, m_2, m_5, m_4, m_7, m_6, m_8)$. The scheduling corresponding to this sequence is depicted in Figure 4.6(d) and it is easy to check that it is valid.

We now prove the correctness of Algorithm *OneApprox* and Subprocedure *MakeValid*.

Theorem 4.2 *Algorithm OneApprox computes an ordering \mathcal{S} of the messages satisfying properties (i) and (iii) and so HV-scheme(\mathcal{S}) achieves makespan at most $LB + 1$.*

Proof: We prove that Algorithm *OneApprox* computes an ordered sequence of messages satisfying the properties (i) and (iii). We proceed by induction on M . If $M \leq 2$, the result holds obviously. Let us assume that the sequence *OneApprox* ($\{m_1, \dots, m_{M-2}\}$) satisfies (i) and (iii). Let p be the last message of this sequence. By the induction hypothesis, $p \in \{m_{M-3}, m_{M-2}\}$ is sent vertically. The sequence is denoted by (s_1, \dots, s_{M-2}) .

For any $i < M - 2$, \mathcal{O}_i denotes (s_1, \dots, s_i) , i.e., $(s_1, \dots, s_{M-2}) = \mathcal{O}_i \odot (s_{i+1}, \dots, s_{M-2})$. Recall that by induction hypothesis, $p = s_{M-2} \in \{m_{M-3}, m_{M-2}\}$ and has been sent vertically. Let q be the lowest message in $\{m_{M-1}, m_M\}$ and let r be the other one. We consider the 3 cases of Algorithm *OneApprox* (when $M \geq 2$).

- a) **Case p is higher than q .** We proceed like in case (a) of the proof of Theorem 4.1. In this way it is easy to check that $\mathcal{O}_1 \odot (p, q, r)$ is valid.
- b) **Case q and r are higher than p , and $p = m_{M-2}$.** We proceed like in case (b) of the proof of Theorem 4.1. The transformation is depicted in Figure 4.3(c). The sequence $\mathcal{O}_1 \odot (m_{M-1}, p, m_M)$ is valid.
- c) **Case q and r are higher than p , and $p \neq m_{M-2}$** (i.e., $s_{M-3} = m_{M-2}$ and $p = s_{M-2} = m_{M-3}$). Moreover, p is sent vertically. This case is depicted in Figure 4.3.3 (with $M = 8$).

In this case, Algorithm *OneApprox* returns the result of *MakeValid*($\mathcal{O}_{M-4} \odot (m_{M-3}, m_{M-1}, m_{M-2}, m_M), 2$). We now prove that the computed sequence is valid.

By Claim 4.2 and because $s_{M-3} \preceq s_{M-2} = p$, s_{M-3} is lower than $s_{M-2} = p$. Indeed, these messages do not interfere in the ordered sequence (s_1, \dots, s_{M-2}) computed by Algorithm *OneApprox*. Then, it is possible to send messages $m_{M-3}, m_{M-1}, m_{M-2}$ and m_M , alternatively horizontal and vertical (starting horizontally) without any interference between these four messages. Therefore, the scheduling $\mathcal{O}_{M-4} \odot (m_{M-3}, m_{M-1}, m_{M-2}, m_M)$ is either valid or at least 1-good.

Actually, the scheduling $\mathcal{O}_{M-4} \odot (m_{M-3}, m_{M-1}, m_{M-2}, m_M)$ has some more useful properties. More precisely, it is 1-good, where a i -good sequence ($i \geq 2$) is defined as follows:

Let us do the following easy remarks related to the i -goodness that will be useful later.

1. In a $(i-1)$ -good sequence \mathcal{S} , if s_{M-2i} does not interfere with s_{M-2i+1} then \mathcal{S} is valid.
2. In particular, if $2i = M$, a $(i-1)$ -good sequence is valid.

Let i , $1 < i \leq \lfloor M/2 \rfloor$ and let \mathcal{S} be a $(i-1)$ -good sequence. By reverse induction on i , we prove that $\text{MakeValid}(\mathcal{S}, i)$ eventually computes a valid sequence. More precisely, we prove that $\text{MakeValid}(\mathcal{S}, i)$ directly returns a valid sequence (first two cases of Subprocedure *MakeValid*), or it returns the result of $\text{MakeValid}(\mathcal{S}', i+1)$ where \mathcal{S}' is an i -good sequence, and so the result holds by induction.

By definition of an $(i-1)$ -good sequence,

$$\begin{aligned} \mathcal{S} &= (s_1, \dots, s_{M-2i-2}) \odot (s_{M-2i-1}, s_{M-2i}, s_{M-2i+1}, s_{M-2i+2}) \odot (s_{M-2i+3}, \dots, s_M) \\ &= (s_1, \dots, s_{M-2i-2}) \odot (s_{M-2i-1}, s_{M-2i}, m_{M-2i+1}, m_{M-2i+3}) \odot (s_{M-2i+3}, \dots, s_M) \end{aligned}$$

First we prove the initialization of the induction, i.e., we consider the case when $2i \in \{M-1, M\}$.

If $2i = M$, then s_{M-2i} does not interfere with s_{M-2i+1} (because s_{M-2i} is not defined). $\text{MakeValid}(\mathcal{S}, i)$ returns \mathcal{S} which is valid by Remark 2.

If $2i = M-1$ and s_1 and s_2 do not interfere, $\text{MakeValid}(\mathcal{S}, i)$ returns \mathcal{S} that is valid by Remark 1. Otherwise, if $2i = M-1$ and s_1 and s_2 interfere, then $\mathcal{S} = (s_1, m_2, m_4) \odot (s_4, \dots, s_{2i+1})$. This implies that $s_1 = m_1$. Moreover, by parity and because s_{2i+1} is sent vertically, m_2 is sent horizontally. By Claim 4.2 and because $m_1 \preceq m_2 \preceq m_4$, m_1 must be lower than m_2 that is lower than m_4 . In this case, $\text{MakeValid}(\mathcal{S}, i)$ returns $(m_2, m_1, m_4) \odot (s_4, \dots, s_{2i+1})$. By Claim 4.1, the only possible interferences in the resulting scheduling may occur between m_1 and m_2 , or m_2 and m_4 . The respective positions of the destination nodes of m_1, m_2, m_4 imply that this sequence is valid.

Now, let us consider the case $2i < M-1$. For purpose of induction, let us assume that, for any $j > i$, if \mathcal{S} is $(j-1)$ -good, $\text{MakeValid}(\mathcal{S}, j)$ returns a valid sequence. We prove that, for any $(i-1)$ -good sequence \mathcal{S} , $\text{MakeValid}(\mathcal{S}, i)$ returns a valid sequence.

c.1 s_{M-2i} and s_{M-2i+1} do not interfere.

In this case, $MakeValid(\mathcal{S}, i)$ returns \mathcal{S} that is valid by Remark 1.

Let us then consider the case when s_{M-2i} and s_{M-2i+1} interfere.

We first do general remarks on the relative positions of some destination's nodes. By parity, in \mathcal{S} , s_{M-2i} is sent vertically and s_{M-2i+1} horizontally. By Claim 4.2, s_{M-2i+1} is higher and lefter than s_{M-2i} . Moreover, since s_{M-2i+1} and s_{M-2i+2} do not interfere, then s_{M-2i+2} must be higher or lefter than s_{M-2i+1} . Similarly, s_{M-2i-1} must be either lower or righter than s_{M-2i} .

There are two cases to be considered according to the value of s_{M-2i} . Recall that, because $s_{M-2i+1} = m_{M-2i+1}$, $s_{M-2i} \in \{m_{M-2i}, m_{M-2i-1}\}$.

c.2 s_{M-2i} and s_{M-2i+1} do interfere and $s_{M-2i} = m_{M-2i}$. In this case, $MakeValid(\mathcal{S}, i)$ returns the sequence $(s_1, \dots, s_{M-2i-2}) \odot (s_{M-2i-1}, s_{M-2i+1}, s_{M-2i}, s_{M-2i+2}) \odot (s_{M-2i+3}, \dots, s_M)$.

Because of their respective positions, no interferences are created between messages s_{M-2i-1} , s_{M-2i+1} , s_{M-2i} and s_{M-2i+2} , when sending them alternatively horizontal and vertical (starting horizontally). Moreover, only $s_{M-2i+1} = m_{M-2i+1}$ and $s_{M-2i} = m_{M-2i}$ have been switched. Therefore, it is easy to check that it is valid.

c.3 s_{M-2i} and s_{M-2i+1} do interfere and $s_{M-2i} = m_{M-2i-1}$. Note that this implies $s_{M-2i-1} = m_{M-2i}$.

In this case, $MakeValid(\mathcal{S}, i)$ returns $MakeValid(\mathcal{S}', i+1)$ where

$$\begin{aligned} \mathcal{S}' &= (s_1, \dots, s_{M-2i-2}) \odot (s_{M-2i}, s_{M-2i+1}, s_{M-2i-1}, s_{M-2i+2}) \odot (s_{M-2i+3}, \dots, s_M) \\ &= (s_1, \dots, s_{M-2i-2}) \odot (m_{M-2i-1}, m_{M-2i+1}, m_{M-2i}, s_{M-2i+2}) \odot (s_{M-2i+3}, \dots, s_M). \end{aligned}$$

Because of their respective positions, no interferences are created between messages s_{M-2i} , s_{M-2i+1} , s_{M-2i-1} and s_{M-2i+2} , when sending them alternatively horizontal and vertical (starting horizontally). Hence, the only possible interference in the sequence \mathcal{S}' is between s_{M-2i-2} and to s_{M-2i} : It is easy to check that \mathcal{S}' is i -good. Thus, the result is valid by the induction assumption.

■

We prove that Algorithm *OneApprox* performs in linear time, with respect to the number of messages.

Theorem 4.3 *The time complexity of Algorithm OneApprox is $O(M)$.*

Proof: We note $\chi(i)$ the time-complexity of $OneApprox(\{m_1, \dots, m_i\})$. We prove by induction on i that $\chi(i) \leq 2i \cdot O(1)$. Let $S = (s_1, \dots, s_M)$ be the ordered sequence computed by $OneApprox(\mathcal{M})$.

The pivot s_{M-2P} of this sequence is the message such that $s_{M-2P} = m_{M-2P}$ and minimizing P . More precisely, we prove that $\chi(M) \leq O(1) \cdot (2(M - 2P) + P)$. If $M \leq 1$, the result is trivial. Let us assume $M \geq 2$. By induction, the computation of $OneApprox(\{m_1, \dots, m_{M-2}\})$ takes time at most $O(1) \cdot (2(M - 2 - 2P') + P')$ where P' is the pivot of the obtained sequence. There are three possible cases corresponding to the three cases the algorithm (when $M \geq 2$).

- a. Clearly, in this case, $\chi(M) = \chi(M - 2) + O(1)$. Moreover, either $P = P' + 1$ (if $r = m_{M-1}$) or $P = 0$ (if $r = m_M$). In both cases, we get $\chi(M) \leq O(1) \cdot (2(M - 2P) + P)$.
- b. $\chi(M) = \chi(M - 2) + O(1)$ and $P = 0$, thus $\chi(M) \leq O(1) \cdot 2M$.
- c. Let $\chi'(M)$ be the complexity of $Makevalid((s_1, \dots, s_M), 2)$. In this case, $\chi(M) = \chi(M - 2) + \chi'(M)$ and $P = 0$ (because in the computed sequence, $s_M = m_M$). Finally, when executed $Makevalid((s_1, \dots, s_M), 2)$, the same subprocedure $MakeValid$ is recursively executed until s_{M-2j} and s_{M-2j+1} do not interfere, or $s_{M-2j} = m_{M-2j}$. Therefore, it is executed at most P' times and each execution takes $O(1)$. Hence, $\chi(M) \leq O(1) \cdot 2(M - 2P') + O(1) \cdot P' \leq 2M$.

■

4.4 Distributed Algorithm

We present a synchronous distributed algorithm for the gathering in a N^2 -node grid, which is based on the Algorithm *TwoApprox*, for personalized broadcasting presented in section 4.3.2. This algorithm uses control messages of size $O(\log N)$ bits and it performs in $O(N^2)$ steps (with similar complexity in terms of number of control messages), i.e., its time-complexity is linear in the size of the grid.

4.4.1 Distributed Model

The network is assumed to be synchronous. Each node has only a local view of the network. However, it has access to the following global information: its position (x, y) in the grid, the position of BS (for sake of simplicity, we assume that BS has coordinates $(0, 0)$), and the size $N \times N$ of the grid (an upper bound on N is sufficient). Finally, any node v has $m(v) \geq 0$ messages that it must send to BS . At every step, a node can send or (exclusive) receive a control message, or *signaling*, of size $O(\log N)$ to (from) one of its neighbours. In the following, for any $i \leq 2N$, $Diag(i)$ denotes the set of vertices at distance i from BS . We refer to $Diag(i)$ as the *diagonal* i . The *central node* $c(2a)$ (resp., $c(2a + 1)$) of $Diag(2a)$ (resp., $Diag(2a + 1)$) is the node with coordinates (a, a) (resp., $(a + 1, a)$). Finally, let *AntiDiag* be the set that consists of the vertices $c(i)$ for all $i \leq 2N$. The algorithm consists of four phases that we describe now.

4.4.2 Basic Description of Distributed Algorithm

Our algorithm aims at giving to any message m its position in the ordering \mathcal{S} computed by Algorithm *TwoApprox* (in terms of personalized broadcasting) and the makespan. This is performed in $Y = O(N^2)$ steps (Y will be specified below) using $O(N^2)$ signalings. Then, with this information, any message can compute its starting time, given that the first message will be sent at step $Y + 1$.

Let us give a rough description of the four phases of the distributed algorithm. First two phases consist in giving to any message m its position in the non increasing order of their distance to BS such that nodes in the same diagonal are ordered up to down (the ordering of messages hosted at a same node is arbitrary). Moreover, each message m_{2a+1} with $a \geq 0$, resp., m_{2a+2} , (actually, the node hosting this message) will learn the position(s) of messages $m_{2a+2}, m_{2a+3}, m_{2a+4}$, resp., $m_{2a+1}, m_{2a+3}, m_{2a+4}$. Then the third phase starts. With the information previously learnt, according to Algorithm *TwoApprox*, message m_1 can decide the ordering in \mathcal{S} of the first three messages: s_1, s_2, s_3 . Two of these three positions are occupied by m_1 and m_2 . The remaining place is occupied by m_3 or m_4 (This comes from the definition of the *TwoApprox* algorithm). Then, at some step, the message s_{2a+3} is fixed. With this information, we prove that message m_{2a+3} can extend the ordering to s_{2a+4} and s_{2a+5} using the *TwoAlgo* algorithm. At the end of this phase, any node knows its position in \mathcal{S} and BS knows the makespan. During the last phase, BS broadcasts the makespan to any node. With this information, each node can compute its starting time for the gathering process.

4.4.3 Formal Description of Distributed Algorithm

Phase 1. It is divided into 2 processes executed “almost” simultaneously.

- The first one is executed in parallel by all diagonals. For any $i \leq 2N$, it aims at collecting some information in $c(i)$, the central node of $Diag(i)$. When this process ends up at step $i + 5$, $c(i)$ has learnt
 - the number of messages l_i standing in $Diag(i)$ in nodes with greater ordinate than $c(i)$,
 - the number of messages r_i standing in $Diag(i)$ in nodes with smaller ordinate than $c(i)$,
 - the position(s) of the three messages with greatest ordinate in $Diag(i)$.

Moreover, at the end of the phase, any node v with coordinates (x, y) in $Diag(i)$ has learnt the position of the (at most 3) node(s) of $Diag(i)$ hosting the closest 3 messages that are higher (if $y \geq x$) or lower (if $y \leq x$) than v .

To do so, two signalings $D1_i$ and $D1'_i$, initiated by nodes $(i, 0)$ and $(0, i)$ respectively, are propagated toward $c(i)$. From $(i, 0)$ (resp., from $(0, i)$), $D1_i$ (resp., $D1'_i$) is transmitted to node $(i, 1)$ and then to $(i - 1, 1)$ (resp., to $(1, i)$ and then to $(1, i - 1)$), and so on until reaching

$c(i)$. To avoid interferences, $D1_i$ and $D1'_i$ are initiated at step 1 by $(i, 0)$ and $(0, i)$ if i is odd. If i is even, $D1_i$ is initiated at step 5 by $(i, 0)$, and $D1'_i$ is initiated at step 6 by $(0, i)$. It is easy to see how information can be aggregated as $D1_i$ and $D1'_i$ go along, in order to obtain the desired information. Moreover, signalings $D1_i$ (resp. $D1'_i$) have size $O(\log N)$ since they contain: the number of messages they met, the position(s) of the first three messages they met, and the position(s) of the last three messages they met.

- At step 7, a signaling $A1$ is initiated in BS and is propagated along *AntiDiag* towards $(N-1, N-1)$. When $c(i)$ receives $A1$ at step $i+6$, it learns the total number of messages hosting by nodes in $\bigcup_{j < i} \text{Diag}(j)$ and the position(s) of the three messages in $\bigcup_{j < i} \text{Diag}(j)$ that are further to BS and with greatest ordinate. Then, using the information propagated by messages $D1_i$ and $D1'_i$, $c(i)$ updates message $A1$ and sends it to $c(i+1)$ during the next step.

The signaling $A1$ arrives to $(N-1, N-1)$ at step $2N+5$ which concludes this phase.

Phase 2. The second phase is divided into three successive processes.

- At step $2N+6$, a signaling $A2$ is initiated in $(N-1, N-1)$ and is propagated along *AntiDiag* towards $(0, 0)$. When $c(i)$ receives $A2$ at step $4N+6-i$, it learns the total number of messages M and the position(s) of the three messages in $\bigcup_{j > i} \text{Diag}(j)$ that are closest to BS and with smallest ordinate.

Note that after step $4N+6-i$, $c(i)$ knows the interval of the positions occupied by messages in $\text{Diag}(i)$, i.e., from $M - \bigcup_{j \leq i} \text{Diag}(j) + 1 = \bigcup_{j > i} \text{Diag}(j) + 1$ to $\bigcup_{j > i} \text{Diag}(j) + l_i + r_i + m(c(i))$.

The two last processes ensure that any message m_{2a} (resp., m_{2a+1}) knows its position in the non increasing order of their distance to BS , i.e., its position in the ordered sequence \mathcal{M} , and the position(s) of messages $m_{2a+1}, m_{2a+2}, m_{2a+3}$ (resp. $m_{2a}, m_{2a+2}, m_{2a+3}$).

- At step $4N+6-i+3$ if i is odd and $4N+6-i+5$ if i is even, a signaling $D2_i$ is initiated in $c(i)$ and is propagated toward $(i, 0)$. $D2_i$ transmits: the next position (in \mathcal{M}) to be attributed to the messages in $\text{Diag}(i)$ with smaller ordinates than $c(i)$, i.e., from $\bigcup_{j > i} \text{Diag}(j) + l_i + m(c(i))$ to $\bigcup_{j > i} \text{Diag}(j) + l_i + r_i + m(c(i))$ (in such a way that any message lower than $c(i)$ in $\text{Diag}(i)$ learns its number in the ordering when it meets the signaling $D2_i$), the position(s) of the last three messages met by this signaling, and the position(s) of the three messages in $\bigcup_{j < i} \text{Diag}(j)$ furthest to BS and with greatest ordinates.
- At step $4N+6-i+2$ if i is odd and $4N+6-i+6$ if i is even, a signaling $D2'_i$ is initiated in $c(i)$ and is propagated toward $(0, i)$. $D2'_i$ transmits: the next position (in the ordering) to be attributed to the messages in $\text{Diag}(i)$ with greater ordinates than $c(i)$, i.e., from $\bigcup_{j > i} \text{Diag}(j)$ to $\bigcup_{j > i} \text{Diag}(j) + l_i$ (in such a way that any message higher than $c(i)$ in $\text{Diag}(i)$ learns its

number in the ordering when it meets the signaling $D2'_i$), the position(s) of the last three messages met by this signaling, and the position(s) of the three messages in $\bigcup_{j < i} \text{Diag}(j)$ furthest to BS and with greatest ordinates.

This phase ends at slot $4N + 12$.

Phase 3. Any message will learn its position in the final ordering \mathcal{S} .

We define the start of this phase at slot $4N + 13$ after finishing phase 2.

At the beginning of this phase, message m_{2a+1} ($a \geq 0$) knows its position in the ordered sequence \mathcal{M} and the position(s) of m_{2a+2} , m_{2a+3} , and m_{2a+4} .

The procedure starts as follows. Node m_1 knows m_2, m_3, m_4 . Using *TwoApprox* algorithm with input (m_1, m_2, m_3, m_4) , it computes the ordering of the first three positions of \mathcal{S} . According to the algorithm the possible configurations for the first three messages in \mathcal{S} are (m_1, m_2, m_3) , (m_1, m_3, m_2) , (m_1, m_2, m_4) , (m_2, m_1, m_3) , (m_2, m_3, m_1) , (m_2, m_1, m_4) . Note that, although the algorithm returns also a message for the fourth position, it is not definitive because it could be modified when the next pair of messages (m_5, m_6) is included. The first message s_1 is decided arbitrarily to be vertical.

Then, m_1 computes the current makespan, i.e., $\max_{j \in \{1,2,3\}} d(BS, s_j) + m_j - 1$ and propagates the information to m_2 and m_3 . That is, m_1 sends them the ordering of the first three messages (s_1, s_2, s_3) of \mathcal{S} (again, $\{s_1, s_2, s_3\} \subset \{m_1, \dots, m_4\}$) and the current makespan. The corresponding signaling is sent at step $4N + 13$ to m_3 and at step $4N + 15$ to m_2 . The signaling reaches m_3 at step $4N + 12 + t$ where t is the distance between m_1 and m_3 .

The process continues iteratively until m_{2a+3} receives a signaling from m_{2a+1} at step $4N + 12 + t$, for $t = \sum_{0 \leq k \leq p} \text{dist}(m_{2k+1}, m_{2k+3})$. This signaling contains the positions of messages $s_{2a+1}, s_{2a+2}, s_{2a+3}$, and the current makespan, i.e., the makespan restricted to messages s_1 to s_{2a+3} . At this step, m_{2a+3} must decide which messages will occupy positions s_{2a+4} and s_{2a+5} in \mathcal{S} . This decision is taken according to Algorithm *TwoApprox*. Note that, Algorithm *TwoApprox* requires as input the next pair of messages m_{2a+5}, m_{2a+6} and the message $m^* \in \{m_1, \dots, m_{2a+4}\}$ whose position in \mathcal{S} has not been decided yet. By property of Algorithm *TwoApprox*, $m^* \in \{m_{2a+3}, m_{2a+4}\}$.

Thus, m_{2a+3} is able to decide which messages will occupy positions s_{2a+4} and s_{2a+5} in \mathcal{S} , and then it can update the current makespan. Finally, at step $4N + 12 + t + 1$ (resp., at step $4N + 12 + t + 3$), message m_{2a+3} sends a signaling to m_{2a+5} (resp., to m_{2a+4}). This signaling contains the current makespan, s_{2a+3}, s_{2a+4} and s_{2a+5} . The signaling is received by m_{2a+5} at step $4N + 12 + t + t'$ where t' is the distance between m_{2a+3} and m_{2a+5} .

The end of this phase is upper bounded by step $4N + 12 + 2N^2$.

Phase 4. At the end of previous phase, BS learns the makespan of a HV-scheme realizing the computed ordering and starts broadcasting it to any node at step $4N + 13 + 2N^2$.

This is done thanks to a signaling through *AntiDiag*, and signalings from $c(i)$ to $(i, 0)$ and $(0, i)$ ($i \leq 2N$) in a similar way as Phase 2. This process ends at step $6N + 19 + 2N^2$.

Defining $Y = 6N + 19 + 2N^2$, each node knows the step when it has to send the message given that the starting step is $Y + 1$. Moreover, the message s_j is sent horizontally or vertically according to the parity of j .

4.5 Conclusion and Further Works

In this paper, we have presented almost optimal centralized and distributed algorithms for the minimum makespan personalized broadcasting in *open-grid* networks. In these settings, the problem is strictly equivalent to the data gathering problem.

The next step is obviously to provide algorithms for the closed grid case. As a matter of fact, one can check that the lower bound is weaker then and one cannot restrict the routing to shortest paths anymore. The +2-approximation algorithm can be fixed to handle this case even though its actual approximation gap is still under investigation [BGN⁺09].

Besides, one can note that our network model assumes that an optimal MAC layer is available. It would be interesting to investigate on the behaviour of the problems under weaker assumptions, such as faulty transmissions and/or intermittent nodes.

Another direction to investigate is the online version of the problems. It is worth pointing out that, in this case, personalized broadcasting and gathering are no longer equivalent. Last but not least, the time complexity of the gathering problem in (open) grids is an open problem.

Part II

Round Weighting

Chapter 5

Round weighting in the primary node interference model

5.1 Introduction

In this chapter, we address the problem of allocating bandwidth efficiently in a radio network, in such a way that given traffic demands are satisfied. Due to the sharing property of the radio bandwidth, one has to schedule radio transmissions in the network in order to avoid concurrent interfering transmissions. We consider traffic gathering where the nodes of the network have a bandwidth requirement to send to a sink node called *gateway*. The problem is to find sets of compatible communication links in the network, called *rounds*, such that the node bandwidth requirements are achieved.

This problem can be seen as a relaxation of the MINIMUM TIME GATHERING PROBLEM. In the MINIMUM TIME GATHERING PROBLEM the demand is a discrete quantity of messages and the goal is to provide a sequence of rounds such that all the messages are collected. In our case, a solution is no longer a sequence of rounds, but a continuous weight function on the rounds, and the objective function is to provide enough capacity for a flow to fit the bandwidth requirements.

This problem has been formalized into the *Round Weighting Problem (RWP)* by [KMP08], that jointly considers the *multi-commodity flow problem* and the *weighted fractional coloring problem*. The radio network is modeled by a topology graph G , in which each node v is a router of the network having a bandwidth demand $b(v)$ to send to the sink node s called also the *gateway*. The goal of the RWP is to assign *weights* to a set of rounds satisfying the demand and minimizing the total weight.

To avoid transmission collisions, one has to define an interference model. Here, we consider a binary interference model known as *primary node interference* or *node-exclusive interference* model [MSS06], where each round corresponds to a matching over the graph.

First, the method proposed consists in routing the demand of each node through a cycle. After, we propose a routing by pairing the nodes. In both cases, we provide lower and upper bounds which in some cases are optimal. Lower bounds are obtained by providing dual solutions.

In section 5.2 we introduce the Round Weighting Problem. In section 5.3 we discuss the case where the demand is concentrated in only one node v , i.e, when $b(u) = 0$ for all node $u \neq v$. In section 5.4 we remove this constraint and we discuss the case with more general demand functions.

5.2 Model

In this section we present the round weighting problem as a linear program and study it through its dual for the case of gathering. For more details, see section 1.3.

We considered a traffic gathering where the demand $b : V \rightarrow \mathbb{R}^+$ represents the flow $b(v)$ needed to be sent from v to the gateway s .

We say that a round weight function w satisfies the traffic demand b if there exists a flow ϕ such that

- it satisfies the traffic demand b

$$(\forall v \in V) \quad \sum_{P \in P_{sv}} \phi(P) \geq b(v),$$

where P_{sv} is the set of paths between the gateway and v .

- and respects the capacity c_w induced by w :

$$(\forall e \in E) \quad \sum_{P \in \mathcal{P}: e \in P} \phi(P) \leq c_w(e) = \sum_{R \in \mathcal{R}: e \in R} w(R).$$

Summarizing, the round weighting problem can be seen as follows.

PROBLEM:	Round weighting
INPUT:	a graph $G = (V, E)$, a set of possible rounds $\mathcal{R} \subset 2^E$ (whose size may be exponential), and a flow demand function $b : V \rightarrow \mathbb{R}^+$ between each node v and the gateway s .
SOLUTION:	A round weight function w defined over \mathcal{R} that satisfies the traffic demand b .
GOAL:	Minimize the overall weight of w , i.e. $W = \sum_{R \in \mathcal{R}} w(R)$.

In order to obtain good lower bounds, we will use the dual formulation discussed in section 1.3. In this case, we will only use the following property:

Fact 5.1 ([KMP08]) *The dual problem of round weighting consists of finding a metric $l : E \rightarrow \mathbb{R}^+$ onto the call set maximizing the total distance that the traffic needs to travel ($W = \sum_{v \in V} d_l(v, s)b(v)$, where d_l is the distance induced by metric l) and such that the maximum length of a round is 1 ($\forall R \in \mathcal{R}, w(R) = \sum_{e \in R} l(e) \leq 1$).*

This fact will allow us to find a feasible dual solution which will give a lower bound for the primal problem.

In the following, we will denote as *cost* the weight of the rounds. We will denote the *price* of an edge, the dual value $l(e)$ associated to each edge e . The *price of the solution* is the value of the objective function of a dual problem solution ($\sum_{v \in V} d_l(v, s)b(v)$).

5.2.1 Interference Model: Matchings

We suppose that any node is able to perform a transmission to any of its neighbours. We consider a basic interference model given by the following main assumption: a node cannot transmit and receive at the same time. Thus a round is a set of edges which can be performed simultaneously, therefore, in fact a matching. This model of interference is also known as *primary node interference* or *node-exclusive interference* [MSS06].

5.3 One node with demand

In the following, we assume that the graph is *simple*. A *simple graph* is a graph containing no loops or multiple edges. Moreover, all the paths are considered are *simple*, meaning that no vertices (and thus no edges) are repeated.

We consider the case when the demand is concentrated over only one node. The demand of this node v is denoted by $b(v)$. We will present lower bounds via dual solutions and upper bounds by providing protocols.

First, we present the following simple property.

Property 5.1 *For an instance of RWP where the total demand is b , any protocol which attains a cost of b is optimal.*

This property is true due to the fact that any feasible solution must route all the demand and then, the cost of the solution must be at least the value of the total demand. The property will be useful later in order to prove optimality when the solution attains a cost equal to the demand.

In the following, we will present different methods of routing the demand between the node v and the gateway s . We will give optimal solutions for path and cycles. Furthermore, we will show that in the case of the cycles, the parity plays an important role.

5.3.1 Paths

First, we show that routing all the demand of v through one path may not be a good solution. Indeed the cost of the solution will be twice the demand (except when the path consists in only one edge between s and v).

Lemma 5.1 *Let v be a node with demand b . Routing the demand through a path between v and s costs at most $2b$.*

Proof: As a path is 2-coloreable, we can cover the edges of the path with 2 rounds of weight b each. ■

Now we will present a lower bound which states that the solution above is optimal. The lower bound is obtained from a dual solution.

Lemma 5.2 *Let v be a node with a demand b . Routing the traffic through a path between v and s costs at least $2b$.*

Proof: Recall that a dual solution consists in a length function l as in fact 5.1. The proposed length function assigns a value (or *price*) of 1 for the two (consecutive) closest edges to the gateway, and 0 for the remaining ones. The idea is depicted in figure 5.1. Since two consecutive edges cannot be in the same round, every round R satisfies $\sum_{e \in R} l(e) \leq 1$. Therefore, by fact 5.1, the length function is a feasible dual solution. Since the demand must use all the edges of the path in order to reach s , in particular it uses the two edges with price 1. Thus, the value of the dual solution is $2b$. ■

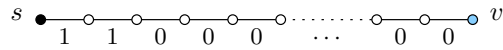


Figure 5.1: Dual solution for the path. Labels indicate the value of the length function for each edge.

From the above lemmas we have the following optimal solution.

Theorem 5.1 *Let v be a node with demand b . Routing the demand through a path between v and s costs $W = 2b$.*

We will see that the cost can be reduced using a cycle between v and s .

5.3.2 Even Cycles

We now present a simple lemma which gives a routing that attains a cost equal to the demand, therefore we have an optimal solution.

Theorem 5.2 *Given an even cycle in G containing the gateway and a node v , then we can route b units of demand from node v to the gateway with $W = b$*

Proof: As the cycle is even, we can cover its edges with two rounds alternately (See Fig 5.2). Considering a weight of $b/2$ for each round, we can route $b/2$ units of traffic for each path of the cycle between v and the gateway. ■

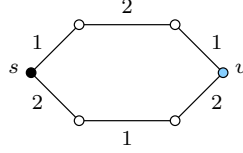


Figure 5.2: Routing the demand from v to s in the even cycle C_6 . Labels over each edge represents the round covering the edge.

Corollary 5.1 *Given a 3-connected graph in which only one node has a demand b . Then, $W = b$.*

Proof: As the graph is 3-connected, there exists an even cycle between the demanding node and the gateway. Thus, lemma 5.2 shows that using 2 rounds with cost $\frac{b}{2}$, the routing can be done with $W = b$. ■

Corollary 5.2 *Given a 2-connected bipartite graph, where only one node v has a demand b , then $W = b$.*

Proof: A bipartite graph does not contain any odd cycle and therefore if it is 2-connected, then there exists an even cycle between the demanding node and the gateway. By Lemma 5.2 the result follows. ■

Remark that a grid is a 2-connected bipartite graph, then corollary 5.2 can be applied.

Corollary 5.3 *Given a grid, where only one node has a demand b to be sent to the gateway, then $W = b$.*

We will see in section 5.4 that above results can also be extended to the case when more than one node has demand.

5.3.3 Odd Cycles

Now, we study the case when the demanding node v and the gateway s are contained in an odd cycle.

Notice that if (v, s) is an edge, then the optimal protocol consists in routing through this edge all the demand. We assume that there is no edge between v and s .

Let C_n denote a cycle of size n formed by two paths from v to s denoted P and Q . The length of these paths are p and q respectively. Therefore, $n = p + q$. We assume that $p > q > 1$.

Now, we present a protocol for routing the demand through an odd cycle. We will see later that this solution is in fact optimal.

Lemma 5.3 *Let C_n be an odd cycle. If the node v has a demand b , then the routing can be done with a cost of at most $b \frac{2p}{2p-1}$.*

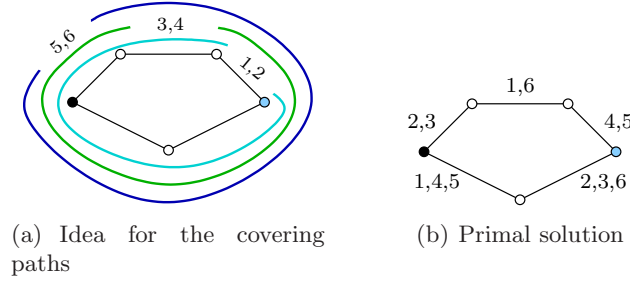


Figure 5.3: In fig. 5.3(a), the 3 lines denote the $p = 3$ paths covering $n - 1 = 4$ edges each. The labels (above each non-covered edge) indicate the labels of the rounds used in each path. In fig. 5.3(b): labels indicate the labels of the rounds used in each edge.

Proof: Notice that an even path can be covered with 2 different rounds. Hence, the idea consists in covering the cycle with p even paths of size $n - 1$.

First, for each edge e in P , we cover the edges of $C_n \setminus \{e\}$ with two rounds $R_{e,1}$ and $R_{e,2}$ alternately. Each one of this rounds has a cost of $b \frac{1}{2p-1}$. Notice that each pair of rounds leaves out a different edge of P at each time. Therefore, each edge of P is covered by $p - 1$ paths and each edge of Q is covered by p paths.

A scheme showing the covering paths ($n = 5$ and $p = 3$) is presented in figure 5.3(a).

We will check now that the assignment of rounds gives a feasible solution. Each edge in P is covered by $p - 1$ rounds and each edge in Q is covered by p rounds. Now, we will check the *induced capacity* of each path. Recall that the *induced capacity* of an edge is the sum of all the costs of rounds covering this edge. For each edge in P , the induced capacity is $b \frac{p-1}{2p-1}$ and for an edge in Q is $b \frac{p}{2p-1}$. Therefore, the flow function $\phi(P) = b \frac{p-1}{2p-1}$ and $\phi(Q) = b \frac{p}{2p-1}$ satisfies the capacity constraints and b units of demand can be routed from v to s . ■

We now use a dual approach to give a lower bound.

Lemma 5.4 *Let C_n be an odd cycle. If the node v has a demand b , then the routing need a cost of at least $b \frac{2p}{2p-1}$.*

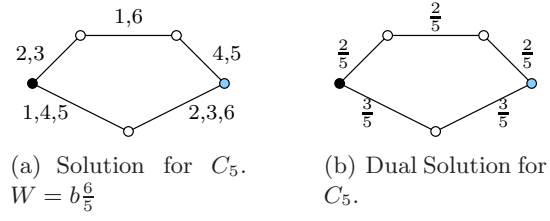


Figure 5.4: Primal and Dual solution for a C_5 with one node with a demand of b . In fig. 5.4(a) labels denote the labels of the rounds covering each edge. In fig. 5.4(b) labels denote the *price* of each edge.

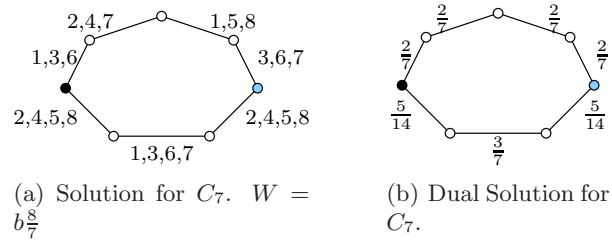


Figure 5.5: Primal and Dual solution for a C_7 with one node with demand b .

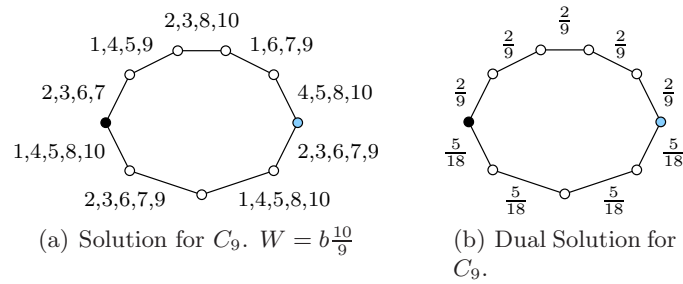


Figure 5.6: Primal and Dual solution for a C_9 with one node with demand b .

Proof: We need to find a feasible dual solution to get a lower bound. According to fact 5.1 a dual solution consists in a function l over the edges such that each round R satisfies $\sum_{e \in R} l(e) \leq 1$. Recall that we denote *price* the dual value $l(e)$ associated to each edge e . We will construct a dual solution with an objective value $b \frac{2p}{2p-1}$. Each edge in P will be given a price of $\frac{2}{2p-1}$. Depending on the parity of Q , we distinguish 2 cases: if q is even, each edge in Q will be given a price of $\frac{2p-1}{2p-1} \frac{1}{q}$ (See figures 5.4(b), 5.6(b) and 5.7(b)); and if q is odd, for the $\lceil \frac{q}{2} \rceil$ non-adjacent edges in Q we will use a price of $\frac{p+1}{2p-1} \frac{1}{\lceil \frac{q}{2} \rceil}$, and for the $\lfloor \frac{q}{2} \rfloor$ remaining edges we will use a price of $\frac{p-1}{2p-1} \frac{1}{\lfloor \frac{q}{2} \rfloor}$ (See figure 5.5(b)). Thus, the sum of the prices of the edges in a round is at most 1, which satisfies the condition of fact 5.1. Moreover, whatever the path chosen (either P or Q), the sum of the prices of all the edges in the path is $\frac{2p}{2p-1}$. Therefore, the value of the dual solution is $b \frac{2p}{2p-1}$. ■

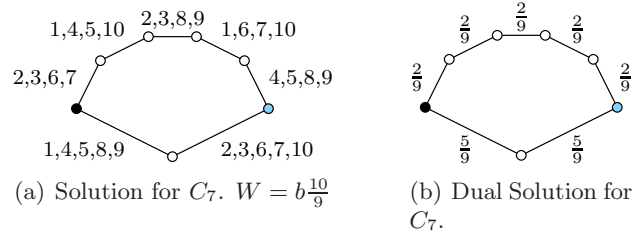


Figure 5.7: Primal and dual solution for a C_7 .

Using lemmas 5.3 and 5.4 we have found the optimal solution when routing is done over an odd-length cycle.

Theorem 5.3 *Let C_n be an odd cycle. If the node v has a demand b , then the optimal solution is $W = b \frac{2p}{2p-1}$.*

We now consider a 2-connected graph and we are able to provide an upper bound for the routing, whatever the cycle chosen between the demanding node and the gateway.

Corollary 5.4 *The routing of b units from v to the gateway s using a cycle containing v and s can be done with a cost of at most $b \frac{6}{5}$.*

Proof: First, note that the cost of routing with either even cycles or a C_3 is b . Then, we only need to show that for any odd cycle C_n with $n > 3$, the routing can be done with a cost of at most $b \frac{6}{5}$. For a given odd cycle C_n with $n > 3$, the cost of routing is $b \frac{2p}{2p-1}$ by theorem 5.3. As the cost decreases with p , the worst case is routing with a C_5 with $p = 3$ and $q = 2$ in which the cost is $W = b \frac{6}{5}$. ■

The above corollary guarantees that we can design $\frac{6}{5}$ -approximation solutions even when the demand is not concentrated in one node, as we will see in section 5.4.

Remark that if no even cycle contains the demanding node and the gateway, it does not imply that the cost of the solution is greater than the demand. In the next section we present some cases in which routing through an odd cycle with an ear may also attain a cost equal to the demand.

5.3.4 Special Cases

5.3.4.1 Odd cycles with ears

Let X be a path. We define an ear of X as a path whose inner nodes has degree 2 and such that its end nodes belong to X .

In this section, we consider the case when either P or Q has an ear and the end nodes of the ear are neither the gateway s nor v .

We denote T the set of edges of the ear. Let $t = |T|$, and x be the end node of the ear which is closest to s , and y be the other end node.

In the following, we distinguish two cases depending on whether the end nodes of the ear are joined by an edge or not.

No edge joining the end nodes of the ear

We consider the case when the end nodes of the ear are not joined by an edge of the cycle.

Lemma 5.5 *Let C_n be an odd cycle with an ear, $n \geq 5$, containing v and s , where v has a demand b . If there is no edge between the end nodes of the ear, then the routing can be done with cost $W = b$.*

Proof: Without loss of generality, we assume that the ear is over P . Let x, y be the end nodes of the ear. Let C_{xy} the cycle formed by the ear and the subpath of P between x and y . We suppose that the two paths between x and y forming C_{xy} have the same parity. Otherwise, there exists an even cycle between v and s and the result follows from theorem 5.2. By hypothesis, we suppose also that C_{xy} has at least 4 edges.

We will propose a solution using 4 rounds such that each of them has a cost of $\frac{b}{4}$. Hence, the solution proposed attains a cost of b which is optimal by property 5.1.

We cover the edges of the cycle C_n except the edges in C_{xy} . Each edge of $C_n \setminus C_{xy}$ is covered by 2 rounds. Let $\{1, 2\}$ and $\{3, 4\}$ be the labels for the two rounds. Therefore, each edge of $C_n \setminus C_{xy}$ is covered by the sets $\{1, 2\}$ and $\{3, 4\}$ alternately.

In order to cover the edges in C_{xy} , we consider two cases: (1) the end edges of $C_n \setminus C_{xy}$ are both covered by the same set of rounds or (2) they are covered by two different sets of rounds.

For the first case, suppose now that the end edges of $C_n \setminus C_{xy}$ are covered by the rounds $\{1, 2\}$. Then, as C_{xy} is an even cycle, we can cover C_{xy} using 3 and 4 alternately (figure 5.8(a)).

For the second case, suppose that the edge adjacent to x in $C_n \setminus C_{xy}$ is covered by the rounds 1 and 2. Then, we cover the two edges of C_{xy} adjacent to x using the rounds 3 and 4. The remaining edges in C_{xy} are covered with one round each using the rounds 1 and 2 (figure 5.8(b)).

Clearly, the solution proposed is feasible, because there exists a flow between v and s with a demand of b . ■

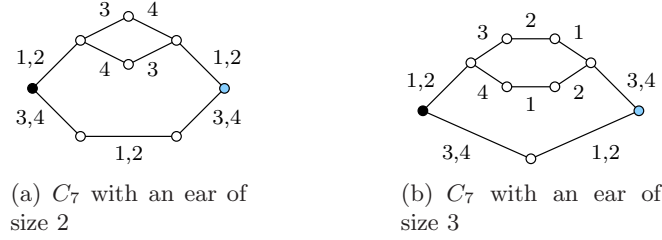


Figure 5.8: Routing b units in 2 cases of a C_7 with an ear. In both cases, each round has cost $b/4$, thus $W = b$.

End nodes of the ear joined by an edge.

In this case (x, y) corresponds to an edge of the cycle. Depending on the location of the ear, (x, y) belongs to either P or Q . Moreover, when $p > q$, the cases for the ear on P or Q are different.

We define also e_x^T to be the edge of T which is incident to x and e_y^T be the edge incident to y .

Notice also that the length of the ear is odd. We denote T' the set of $\frac{t+1}{2}$ non-adjacent edges in T . We denote T'' the remaining $\frac{t-1}{2}$ edges in T .

Lemma 5.6 *Let C_n be an odd cycle with one ear, $n \geq 5$, containing v and s , where v has a demand b . If there is an edge of C_n that joins the two ends of the ear, then the routing can be done with cost at most $b \frac{2pt}{2pt-1}$.*

Proof: We will define $2p$ rounds denoted by R_i , $i = 1, \dots, 2p$ to cover the edges of P and Q in the same way as in Lemma 5.3, but with a cost of $b \frac{1}{2pt-1}$ per edge (See also figure 5.9(a)).

Then, we cover each edge in T'' with the rounds R_i , $i = 1, \dots, 2p$ (See figure 5.9(b))

Now, we will compensate the remaining edges of T . Let e be an edge in T' . We use 2 new rounds $R_{e,1}$, $R_{e,2}$ with a cost of $b \frac{2p}{2pt-1}$ to cover alternately the edges in $P \cup Q \cup T \setminus \{(x, y), e\}$. The process is repeated for all the edges in T' (See figure 5.9(c) and the solution in 5.9(d)).

The total cost is $\sum_{i=1}^{2p} w(R_i) + \sum_{e \in T'} (w(R_{e,1}) + w(R_{e,2})) = b \frac{1}{2pt-1} + (t-1)b \frac{2p}{2pt-1} = b \frac{2pt}{2pt-1}$. Figure 5.9 shows an example of a solution for a C_5 with an ear of 7 edges over P . The rounds

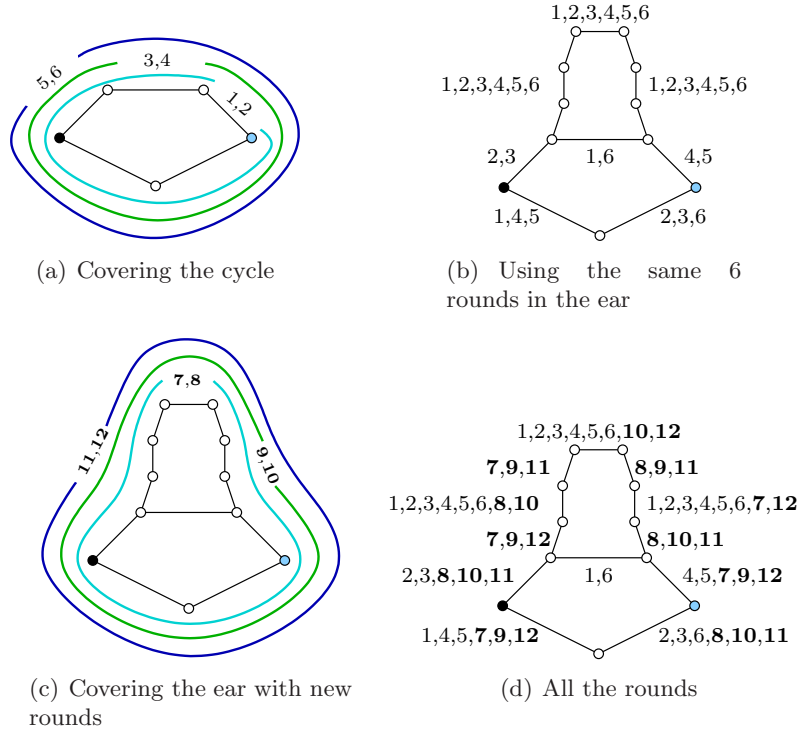


Figure 5.9: Construction of the Primal Solution for a C_5 with an ear of size 7.

labeled by 1 to 6 correspond to R_i , $i = 1, \dots, 6$ with a cost of $w(R_i) = b \frac{1}{41}$, and the rounds labeled 7 to 12 correspond to the ones with a cost of $b \frac{6}{41}$ each.

In the following, we check that the solution proposed is feasible. To do that, we need to study the induced capacity of each edge. Recall that the induced capacity of an edge is the sum of the costs of the rounds covering the edge.

The induced capacity for an edge in $P \setminus \{(x, y)\}$ is $b \left(\frac{p-1}{2pt-1} + \frac{t-1}{2} \frac{2p}{2pt-1} \right) = b \frac{pt-1}{2pt-1}$. The induced capacity for an edge in $Q \setminus \{(x, y)\}$ is $b \left(\frac{p}{2pt-1} + \frac{t-1}{2} \frac{2p}{2pt-1} \right) = b \frac{pt}{2pt-1}$.

Depending on whether (x, y) (the ear) is on P or Q , its induced capacity is $b \frac{p-1}{2pt-1}$ or $b \frac{p}{2pt-1}$ respectively.

Let us check the induced capacities for the edges in T . Each edge of T'' is covered by $2p$ rounds with a cost of $b \frac{1}{2pt-1}$ and $\frac{t-3}{2} = |T''| - 1$ rounds of cost $b \frac{2p}{2pt-1}$. Moreover, each edge in T' is covered by $\frac{t-1}{2}$ rounds with a cost of $b \frac{2p}{2pt-1}$. Thus, the induced capacity of each edge in T is $b \frac{(t-1)p}{2pt-1}$.

The schemes of figure 5.10 summarize the values of the induced capacity of each edge according to the location of the ear. The induced capacities hence shows that is possible to send a flow of b from v to the gateway s . Therefore, the solution proposed is feasible. \blacksquare

Lemma 5.7 *Given an odd cycle C_n with one ear, $n \geq 5$, containing v and s , where v has a demand b . If there is an edge of C_n that joins the two ends of the ear, then the routing cost at least $b \frac{2pt}{2pt-1}$.*

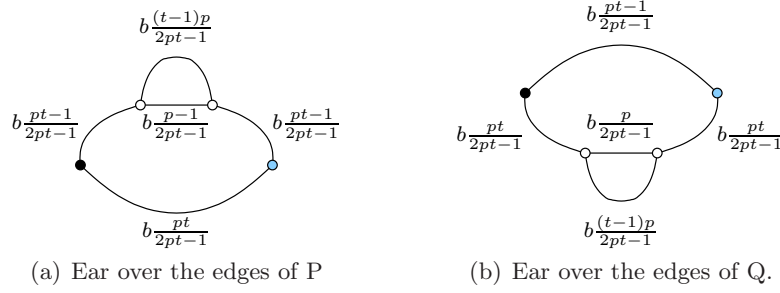


Figure 5.10: Scheme of induced capacities for the different sections in the cycle.

Proof: The dual solution proposed depends again on both the location of the ear over the cycle (on P or Q) and the parity of P and Q . In the following we describe the solution according to the different cases.

- **Case 1: Ear over P and p odd (q even).** In this case, we assign the prices for the edges as follows:

$$\begin{aligned} \frac{2}{2pt-1} & \text{ for } e \in T \\ \frac{2t}{2pt-1} & \text{ for } e \in P \\ \frac{p}{q} \frac{2t}{2pt-1} & \text{ for } e \in Q \end{aligned}$$

We now need to check that the solution proposed above is feasible. According to fact 5.1, we need to show that the price of any possible round is less or equal than 1.

First, note that $\frac{p}{q} > 1$, then $\frac{p}{q} \frac{2t}{2pt-1} > \frac{2t}{2pt-1} > \frac{2}{2pt-1}$. It means that the order in terms of price is: the edges in Q , the edges in P and finally the edges in T . We will construct an upper bound for the price of the rounds using as many edges in Q as possible.

Note also that a round is composed by at most $\frac{p+q-1}{2}$ edges in $P \cup Q$. Then, the round with the highest price is a round with $\frac{q}{2}$ edges in Q and $\frac{p-1}{2}$ edges in P . Since that the round has $\frac{p-1}{2}$ edges in P , there is always at least one edge in P which is adjacent to T . Then, the maximum number of edges in T that can be used in the round is $\frac{t-1}{2}$.

Thus, the price for a round in this case is upper bounded by:

$$\frac{q}{2} \frac{p}{q} \frac{2t}{2pt-1} + \frac{p-1}{2} \frac{2t}{2pt-1} + \frac{t-1}{2} \frac{2}{2pt-1} = 1$$

We conclude that the solution proposed for this case is feasible.

- **Case 2: Ear over P and p even (q odd).** We distinguish two sets of edges in Q . Let Q' be the matching of $\frac{q+1}{2}$ edges in Q and Q'' be $Q \setminus Q'$. We assign the prices for each edge as

follows:

$$\begin{aligned}
& \frac{2}{2pt-1} \quad \text{for } e \in T \\
& \frac{2t}{2pt-1} \quad \text{for } e \in P \\
& \frac{1}{\left(\frac{q+1}{2}\right)} \frac{(p+1)t}{2pt-1} \quad \text{for } e \in Q' \\
& \frac{1}{\left(\frac{q-1}{2}\right)} \frac{(p-1)t}{2pt-1} \quad \text{for } e \in Q''
\end{aligned}$$

To check the feasibility of this solution, note that $\frac{1}{\left(\frac{q-1}{2}\right)} \frac{(p-1)t}{2pt-1} > \frac{1}{\left(\frac{q+1}{2}\right)} \frac{(p+1)t}{2pt-1} > \frac{2t}{2pt-1} > \frac{2}{2pt-1}$. Thus, the order of edges in terms of price is given by: the edges in Q'' , Q' , P and finally the edges in T . Note also that a round has at most $\frac{p+q-1}{2}$ edges in $P \cup Q$. Then, in order to have the rounds with the higher prices, we have 2 possibilities: to use either $\frac{q+1}{2}$ in Q and $\frac{p-2}{2}$ edges in P , or to use $\frac{q-1}{2}$ edges in Q and $\frac{p}{2}$ edges in P .

If we use $\frac{q+1}{2}$ in Q , the highest price is attained using the $\frac{q+1}{2}$ edges in Q' and $\frac{p-2}{2}$ edges in P . Similar to case 1, according to this assignment, the maximum number of edges that can be used in T is $\frac{t-1}{2}$.

Thus, the highest price attained by a round with $\frac{q+1}{2}$ edges in Q is:

$$\frac{q+1}{2} \frac{1}{\left(\frac{q+1}{2}\right)} \frac{(p+1)t}{2pt-1} + \frac{p-2}{2} \frac{2t}{2pt-1} + \frac{t-1}{2} \frac{2}{2pt-1} = 1$$

If we use $\frac{q-1}{2}$ in Q , the highest price is attained using the $\frac{q-1}{2}$ edges in Q'' , $\frac{p}{2}$ edges in P and $\frac{t-1}{2}$ in T .

Thus, the highest price attained by a round with $\frac{q-1}{2}$ edges in Q is:

$$\frac{q-1}{2} \frac{1}{\left(\frac{q-1}{2}\right)} \frac{(p-1)t}{2pt-1} + \frac{p}{2} \frac{2t}{2pt-1} + \frac{t-1}{2} \frac{2}{2pt-1} = 1$$

Hence, we conclude that the solution proposed for this case is feasible.

- **Case 3: Ear over Q and p odd (q even).** In this case, different prices are given to the end edges of T . We denote $d_Q(s, x)$ the distance over Q between s and x . The prices given

are the following:

$$\begin{aligned}
& \frac{2t}{2pt-1} \quad \text{for } e \in P \\
& \frac{p}{q} \frac{2t}{2pt-1} \quad \text{for } e \in Q \\
& \frac{2}{2pt-1} \quad \text{for } e \in T \setminus \{e_x^T, e_y^T\} \\
& \frac{2}{2pt-1} \quad \text{for } e_x^T, \text{ and} \\
& \left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} \quad \text{for } e_y^T, \text{ if } d_Q(s, x) \text{ is even} \\
& \frac{2}{2pt-1} \quad \text{for } e_y^T, \text{ and} \\
& \left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} \quad \text{for } e_x^T, \text{ if } d_Q(s, x) \text{ is odd}
\end{aligned}$$

First, note that the main difference with the solutions for the cases above is that one of the end edges of T has a different price. This is due to the fact that the price of the path between s and v crossing through T must be at least $\frac{2pt}{2pt-1}$. In fact, the price assigned in the proposed solution for this path is:

$$(q-1) \frac{p}{q} \frac{2t}{2pt-1} + (t-1) \frac{2}{2pt-1} + \left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} = \frac{2pt}{2pt-1}$$

Let us now suppose that the round does not use any of the two end edges of T . Thus, similarly to case 1, a round with the highest price is a round that uses $\frac{q}{2}$ edges in Q , $\frac{p-1}{2}$ edges in P and $\frac{t-1}{2}$ edges in T . As seen also in case 1, this assignment of round attains a price of 1.

Let us now consider the rounds using some of the end edges of T . Note that rounds using both end edges do not attain the maximum price. In this case, the round loses 1 edge in Q , but it obtains 1 extra edge in T . However, as the highest price for an edge in T is $\left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1}$ and is always less than the price of an edge in Q , this configuration will not attain the maximum price for a round.

Then, let us consider that the round contains the end edge e_x^T and $d_Q(s, x)$ is even. It means that e_x^T has price $\frac{2}{2pt-1}$. In this case, e_x^T interferes with its 2 incident edges in Q . Therefore, the round may be composed by at most $\frac{d_Q(s, x)}{2}$ edges between s and x and $\frac{d_Q(x, v)}{2}$ edges between x and v . An example showing this case is presented in figure 5.11(a)

Then, the round uses at most $\frac{q}{2}$ edges of Q , and uses the two end edges of Q which are incident to s and v respectively. Thus, we are able to use $\frac{p-1}{2}$ edges in P .

Hence, this solution uses $\frac{q}{2}$ edges in Q , $\frac{p-1}{2}$ edges in P and $\frac{t-1}{2}$ edges in T with a price of $\frac{2}{2pt-1}$. Therefore, the maximum price attained by this assignment is the same as that of case 1 which is 1.

Let us now check the rounds using the end edge e_y^T when $d_Q(s, x)$ is even. In this case, the

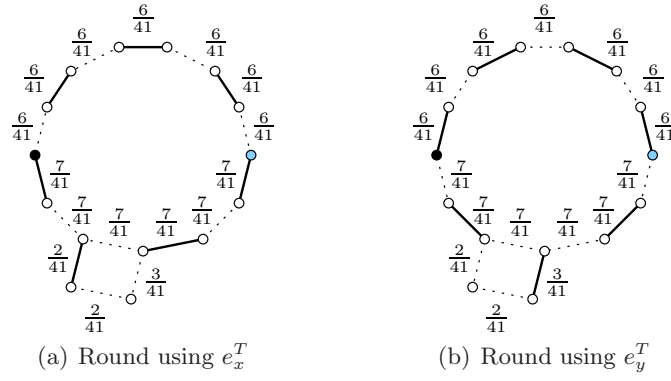


Figure 5.11: Two types of rounds using a different end edge of T .

edge e_y^T has a price of $\left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1}$. Note that there is an odd number of edges between s and e_y^T . Then, the round must use at most $\frac{d_Q(s,y)-1}{2}$ edges between s and y , and $\frac{d_Q(y,v)-1}{2}$ edges between y and v . Thus, the round may use at most $\frac{q}{2} - 1$ edges in Q . Moreover, as we can choose edges in Q which are not adjacent to the end edges in P , the round may have $\frac{p+1}{2}$ edges in P . Adding the $\frac{t-1}{2} - 1$ edges of T with a price of $\frac{2}{2pt-1}$ and e_y^T which has a price of $\left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1}$, the highest price for this assignment is:

$$\left(\frac{q}{2} - 1\right) \frac{p}{q} \frac{2t}{2pt-1} + \left(\frac{p+1}{2}\right) \frac{2t}{2pt-1} + \left(\frac{t-1}{2} - 1\right) \frac{2}{2pt-1} + \left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} = 1$$

An example of this assignment is presented in figure 5.11(b). Note that the difference between the price of e_x^T and the other edges of T is $\left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1}$ which is exactly the difference between an edge in Q and an edge in P .

Let us now check the case $d_Q(s, x)$ odd. Notice that, as in case $d_Q(s, x)$ even, the prices of the end edges of T are $\frac{2}{2pt-1}$ and $\left(\frac{p}{q} - 1\right) \frac{2t}{2pt-1} + \frac{2}{2pt-1}$. Hence, the price of a round in case $d_Q(s, x)$ odd is the same price than in case $d_Q(s, x)$ even. Therefore, each round in the case $d_Q(s, x)$ odd also satisfy that its total price is less or equal than 1.

We conclude that the dual solution proposed for this case is feasible.

- **Case 4: Ear over Q and p even (q odd).** The prices given are the following:

$$\begin{aligned}
& \frac{2t}{2pt-1} \quad \text{for } e \in P \\
& \frac{1}{\left(\frac{q+1}{2}\right)} \frac{(p+1)t}{2pt-1} \quad \text{for } e \in Q' \\
& \frac{1}{\left(\frac{q-1}{2}\right)} \frac{(p-1)t}{2pt-1} \quad \text{for } e \in Q'' \\
& \frac{2}{2pt-1} \quad \text{for } e \in T \setminus \{e_x^T, e_y^T\} \\
& \frac{d_Q(s,x)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} \quad \text{for } e \in e_x^T, \text{ and} \\
& \frac{d_Q(y,v)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} \quad \text{for } e \in e_y^T, \text{ if } d_Q(s,x) \text{ is even} \\
& \frac{d_Q(x,v)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} \quad \text{for } e \in e_x^T, \text{ and} \\
& \frac{d_Q(s,y)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} \quad \text{for } e \in e_y^T, \text{ if } d_Q(s,x) \text{ is odd}
\end{aligned}$$

First, note that the two end edges of T have a different price than the rest of the edges. In the same way that case 3, the idea is that the price of the path from s to v through T must have a price of at least $\frac{2pt}{2pt-1}$.

Suppose that the ear T is such that $d_Q(s,x)$ is even. Then $d_Q(s,x) + d_Q(y,v) = q-1$ and the sum of the prices of all the edges of T is:

$$\frac{q-1}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{4}{2pt-1} + (t-2) \frac{2}{2pt-1} = \frac{p+1}{q+1} \frac{2t}{2pt-1}$$

which is the price of (x,y) because of $(x,y) \in Q'$.

Suppose now that the ear is such that $d_Q(s,x)$ is odd. Then $d_Q(x,v) + d_Q(s,y) = q+1$ and the sum of the prices of all the edges in T is:

$$\frac{q+1}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{4}{2pt-1} + (t-2) \frac{2}{2pt-1} = \frac{p-1}{q-1} \frac{2t}{2pt-1}$$

which is the price of (x,y) when $(x,y) \in Q''$.

In both cases, the total price of T is equal to the price of (x,y) . Hence, the price of the path from s to v through T is the same price than the path using only the edges of Q .

Let us consider the rounds such that no end edge of T is used. As the edges in Q have a higher price than the edges in P , there are 2 assignments to attain the maximum price: either using $\frac{q+1}{2}$ edges of Q' and $\frac{p-2}{2}$ edges of P , or using $\frac{q-1}{2}$ edges of Q and $\frac{p}{2}$ edges of P . Note that in both assignments we are using $\frac{t-1}{2}$ edges from T .

For the first assignment, the price attained is:

$$\frac{q+1}{2} \frac{1}{\left(\frac{q+1}{2}\right)} \frac{(p+1)t}{2pt-1} + \frac{p-2}{2} \frac{2t}{2pt-1} + \frac{t-1}{2} \frac{2}{2pt-1} = 1$$

and for the second configuration, the price is:

$$\frac{q-1}{2} \frac{1}{\left(\frac{q-1}{2}\right)} \frac{(p-1)t}{2pt-1} + \frac{p}{2} \frac{2t}{2pt-1} + \frac{t-1}{2} \frac{2}{2pt-1} = 1$$

Note that, in terms of edges used, these assignments behave similarly as case 2.

Let us now consider the rounds using at least one end edge of T . First of all, note that rounds using both end edges of T will not attain the maximum price. This is because of using the 2 end edges blocks 3 edges of Q of being used. Let us now consider the rounds using only one end edge of T .

Assume that $d_Q(s, x)$ is even and e_x^T is used. Since e_x^T belongs to some round, its two adjacent edges in Q cannot be in the round. Hence, the round contains at most $\frac{d_Q(s, x)}{2}$ edges in Q' and $\frac{d_Q(y, v)}{2}$ edges in Q'' . Thus, the round has $\frac{q-1}{2}$ edges in Q , $\frac{p}{2}$ edges in P and $\frac{t-1}{2}$ edges in T . The highest price for this assignment is:

$$\begin{aligned} & \frac{d_Q(s, x)}{2} \frac{p+1}{q+1} \frac{2t}{2pt-1} + \frac{d_Q(y, v)}{2} \frac{p-1}{q-1} \frac{2t}{2pt-1} + \frac{p}{2} \frac{2t}{2pt-1} \\ & + \frac{d_Q(s, x)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} + \frac{t-3}{2} \frac{2}{2pt-1} \end{aligned}$$

As $d_Q(s, x) + d_Q(y, v) = q-1$ the expression above is reduced to:

$$\frac{q-1}{2} \frac{p-1}{q-1} \frac{2t}{2pt-1} + \frac{p}{2} \frac{2t}{2pt-1} + \frac{t-1}{2} \frac{2}{2pt-1} = 1$$

therefore, the condition is satisfied.

Assume now that $d_Q(s, x)$ is even and e_y^T is used. The round has at most $\frac{d_Q(s, x)}{2}$ from Q'' and $\frac{d_Q(y, v)}{2}$ from Q' . Thus, the round has $\frac{q-1}{2}$ edges in Q , $\frac{p}{2}$ edges in P and $\frac{t-1}{2}$ edges in T . The highest price for this assignment is:

$$\begin{aligned} & \frac{d_Q(s, x)}{2} \frac{1}{\left(\frac{q-1}{2}\right)} \frac{(p-1)t}{2pt-1} + \frac{d_Q(y, v)}{2} \frac{1}{\left(\frac{q+1}{2}\right)} \frac{(p+1)t}{2pt-1} + \frac{p}{2} \frac{2t}{2pt-1} \\ & + \frac{d_Q(y, v)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1} + \frac{t-3}{2} \frac{2}{2pt-1} \end{aligned}$$

As $d_Q(s, x) + d_Q(y, v) = q-1$, the sum above equivalent to:

$$\frac{q-1}{2} \frac{p-1}{q-1} \frac{2t}{2pt-1} + \frac{p}{2} \frac{2t}{2pt-1} + \frac{t-1}{2} \frac{2}{2pt-1} = 1$$

An example for the case $d_Q(s, x)$ even is presented in figure 5.12. In this case the cycle is a C_{15} , $p = 8$, $q = 7$ with an ear over Q with $t = 3$. Figures 5.12(a) and 5.12(a) show the rounds with highest price using e_x^T and e_y^T respectively.

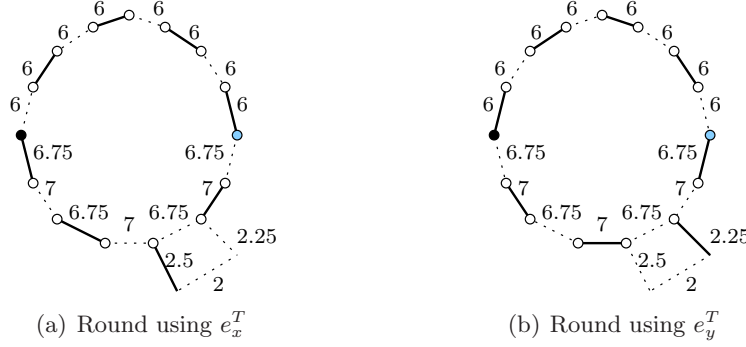


Figure 5.12: Rounds with highest prices using either e_x^T or e_y^T . Edges e_x^T and e_y^T correspond to the edges with labels 2.5 and 2.25 respectively. The cycle is a C_{15} , $p = 8$, $q = 7$ with an ear over Q and $t = 3$. Solid lines represent the edges of the rounds, and the labels indicate the price of each edge. By simplicity of notation, the prices per edge are all multiplied by $2pt - 1 = 47$.

Let us now check the case $d_Q(s, x)$ odd. Let a be a node in Q and let e_a be the end edge of the ear incident to a . The price of e_a is assigned $\frac{d_Q(v,a)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1}$ if $d_Q(s, a)$ is odd; and $\frac{d_Q(s,a)}{2} \left(\frac{p-1}{q-1} - \frac{p+1}{q+1} \right) \frac{2t}{2pt-1} + \frac{2}{2pt-1}$ if $d_Q(s, a)$ is even. Hence the price of e_a is independent of the definition of the nodes x and y . Therefore, a round using e_a has the same price whether $d_Q(s, x)$ is odd or even. We conclude that all the rounds satisfy the condition and then, the dual solution proposed for this case is feasible. ■

We now conclude that the optimal solution of routing the demand b from one node to the gateway over a cycle with an ear has a cost of $\frac{2pt}{2pt-1}$ and we have the following theorem.

Theorem 5.4 *Given an odd cycle C_n with one ear, $n \geq 5$, containing v and s , where v has a demand b . If there is an edge of C_n that joins the two ends of the ear, then the optimal routing solution has a cost of $b \frac{2pt}{2pt-1}$.*

As an example, figure 5.13 shows the primal and dual optimal solutions for a C_5 with an ear when $t = 3$, $p = 3$, $q = 2$ and $b = 1$.

Theorem 5.4 can be considered as a generalization of theorem 5.3. The case of theorem 5.3 in which the cycle has no ears may be interpreted as an *ear* of length $t = 1$ in theorem 5.4.

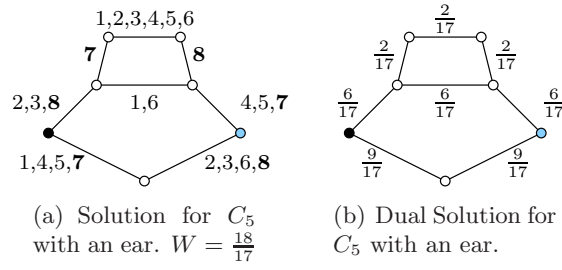


Figure 5.13: Solution for C_5 with an ear. $w(R_i) = \frac{1}{17}$, $i = \{1, \dots, 6\}$, $w(R_7) = w(R_8) = \frac{6}{17}$, $W = \frac{18}{17}$

As a remark, there are some configurations in which it is possible to obtain a solution with $W = b$ in an odd cycle with 2 ears. In figure 5.14 we present two examples.

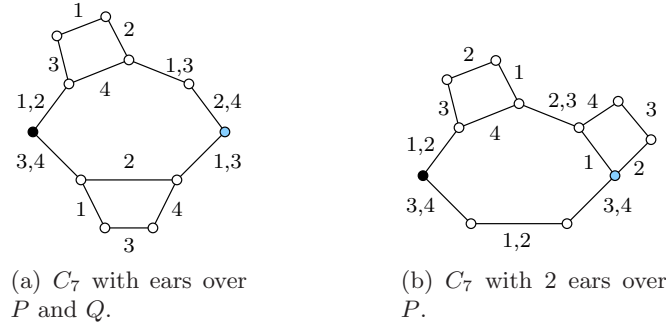


Figure 5.14: Examples of solutions for a C_7 with 2 ears. In both examples, $W = b$.

5.4 Multiple nodes with demands

Now we study the case of general demand, i.e. when the demand is not concentrated over one node. We denote $B = \sum_v b(v)$ the total demand. Here again any routing with cost B is optimal due to the fact that the cost of the solution attains the value of the total demand (see property 5.1).

First, we consider the problem of routing the demand of each node through a cycle. In this way, the results of section 5.3 can be applied directly to design solutions for general demand. If each demanding node can be routed over some even cycle, then lemma 5.2 guarantees that the optimal solution with a cost B is attained. Thus, corollaries 5.1, 5.2 and 5.3 state that this is the case for 3-connected graphs, 2-connected bipartite graphs and grids. In general, if every demand can be routed over a cycle, corollary 5.4 guarantees a $\frac{6}{5}$ -approximation.

Recall that we suppose that the paths considered are *simple*, i.e. they do not have repeated vertices.

Lemma 5.8 *Suppose that there exists a path from u to v containing the gateway s . Then there exists a way of routing a demand of $b_{\min} = \min\{b(u), b(v)\}$ from each node with total cost $W = 2b_{\min}$.*

Proof: We use two rounds of weight b_{\min} each to cover all the edges of this path alternately. ■

We now present a technical lemma.

Lemma 5.9 *In a 2-connected graph, for any three nodes u, s, v , there exists a path containing the 3 nodes starting in u and ending in v .*

Proof: We construct a new graph G' which is obtained by adding a new node x and edges (x, v) and (x, u) to G . It is clear that the graph G' is also 2-connected, therefore, there exist two node-disjoint paths between x and s . Since x is joined to G only by the edges (x, v) and (x, u) , the result follows. ■

Using lemma 5.9, we have the following upper bound.

Theorem 5.5 *Given a 2-connected graph, and V_1 and V_2 form a partition of the nodes set, then there exists a solution with a cost at most $B + \frac{1}{5} |\sum_{v \in V_1} b(v) - \sum_{v \in V_2} b(v)|$*

Proof: The idea is to route the demand by pairing the nodes. Each pair formed by any node in V_1 and any node in V_2 . Let $\{v_1, v_2\}$ be a pair such that $v_1 \in V_1$ and $v_2 \in V_2$. We assume that $b(v_1) > b(v_2)$. Lemma 5.9 guarantees that there exists a (simple) path containing s , starting at v_1 and ending at v_2 . Then we route a demand of $b(v_2)$ from both v_1 and v_2 ($b(v_2)$ units from v_1 and $b(v_2)$ from v_2). Therefore, by lemmas 5.8 and 5.9 a demand of $2b(v_2)$ is routed optimally with cost $2b(v_2)$. Then, the remaining demand of node v_1 is $b(v_1) - b(v_2)$ and node v_2 has no more demand. The process is repeated until there are no two nodes such that both have non-zero demand. At this step $B - |\sum_{v \in V_1} b(v) - \sum_{v \in V_2} b(v)|$ of demand has been routed optimally with a cost of $B - |\sum_{v \in V_1} b(v) - \sum_{v \in V_2} b(v)|$.

Now, there is one node with a demand of $|\sum_{v \in V_1} b(v) - \sum_{v \in V_2} b(v)|$. Corollary 5.4 guarantees that this demand can be routed through a cycle with a cost of at most $\frac{6}{5} |\sum_{v \in V_1} b(v) - \sum_{v \in V_2} b(v)|$. ■

In the above theorem, the quality of the solution depends on the partition chosen. We illustrate that with an example. Let $\{8, 7, 5, 4\}$ be the demands of the nodes in the graph. If the partition is $\{8, 5\}$ and $\{7, 4\}$, the difference between the sum of the demands of each subset is 2. Therefore, theorem 5.5 guarantees a solution with cost at most $24 + \frac{2}{5}$. However, if we take the partition $\{8, 4\}$ and $\{7, 5\}$ the total demand of each subset is 12 and therefore, the difference is 0. In this way, the solution from this partition has a cost of at most 24, which is optimal.

Finding the optimal partition is not easy. In fact the problem is known as NUMBER PARTITIONING PROBLEM, which is NP-HARD.

Using the partition $\{v_{\max}\}$ and $V \setminus \{v_{\max}\}$ we have the following corollary.

Corollary 5.5 *Given a 2-connected graph G , then there exists a solution with cost at most $B + \frac{1}{5} |2b_{\max} - B|$, with $b_{\max} = \max_{v \in V} b(v)$.*

Notice that, for the case when v_{\max} has more than the half of the total demand, the partition used is the best one.

5.5 Conclusions

In this chapter we deal with the ROUND WEIGHTING PROBLEM for gathering instances considering the primary node interference model.

We present upper bounds by providing feasible routings and show that in some cases the solutions are also optimal. The method used to prove optimality consists in providing good feasible solutions for the dual formulation following the ideas in [KMP08].

We first study the case when the demand of each node is routed independently. We discuss extensively the case of routing through a cycle and show that routing through an even cycle attains a cost equal to the demand. Therefore it is optimal. However, routing through an odd cycle the optimal solution gives a higher cost than the demand.

The case when routing through a cycle containing an ear is also considered. Optimal solution is attained as well here. We remark that solutions with costs equal to the demands may be also attained with odd cycles including ears.

In this chapter, we have always considered a primary node interference model. It means that rounds correspond to matchings over the set of edges. An interesting extension consists in using more general interference models, for example, for the case of distance-2 interference model when the rounds correspond to induced matchings. It is clear that we need some different techniques to deal with the general case. Even for routing over a cycle, the problem become much more complicated.

Chapter 6

Round Weighting Problem and Gathering in wireless networks with symmetrical interference [GPRR08]

6.1 Introduction

In this chapter we address the problem of allocating bandwidth efficiently in a radio network, in such a way that the traffic demands are satisfied. Due to the sharing property of the radio bandwidth, one has to schedule radio transmissions in the network in order to avoid concurrent interfering transmissions. We consider traffic gathering where the nodes of the network have a bandwidth requirement to send to a sink node called *gateway*. The problem is to find sets of compatible communication links in the network, called *rounds*, such that the node bandwidth requirements are achieved.

This problem can be seen as a relaxation of the MINIMUM TIME GATHERING PROBLEM. In the MINIMUM TIME GATHERING PROBLEM the demand is a discrete quantity of messages and the goal is to provide a sequence of rounds such that all the messages are collected. In our case, a solution is no longer a sequence of rounds, but a continuous weight function on the rounds, and the objective function is to provide enough capacity for a flow to fit the bandwidth requirements.

This problem has been formalized into the *Round Weighting Problem (RWP)* by [KMP08], that jointly considers the *multi-commodity flow problem* and the *weighted fractional coloring problem*. The radio network is modeled by a topology graph G , in which each node v is a router of the network having a bandwidth demand $b(v)$ to send to the sink node s called also *gateway*. The goal of the RWP is to assign *weights* on a set of rounds satisfying the demand and minimizing the total weight.

The values obtained depend on the interference model. Here we choose a binary interference model

and for the general case we suppose we are given the set of all the possible rounds. Notice that the size of \mathcal{R} might be exponential in the number of nodes. For precise results we will define the set of rounds by a rule depending on the distance between the calls.

6.2 Related Work

The routing problem of steady traffic demands in a radio network has been studied extensively in the literature. In [KMP08] it was proved that if traffic demands are sufficiently steady the problem can be expressed in an independent form of the interference model as the *Round Weighting Problem (RWP)*.

In the case of a general transmission graph with an arbitrary traffic pattern the problem is very difficult to approximate, indeed, to approximate the RWP within $n^{1-\varepsilon}$ is NP-Hard [KMP08]. An important case is the *Gathering* (or personalized broadcasting): the traffic pattern corresponds then to a simple flow, i.e. all demands are directed toward a single node called the gateway. Gathering is easier to approximate since a simple 4-approximation does exist, but the problem remains NP-Hard. Instances on specific graphs with symmetrical traffic distribution are tractable mainly due to the local structure of these graphs, we give particular attention to grid graphs.

In [BP05], a similar problem, the *Round Scheduling Problem (RSP)* was treated. The relation with the RWP is the following: if one must repeat rounds scheduling many times then the problem is equivalent to the RWP. The RSP is quite harder to solve than our problem which can be considered either as a limited case or relaxation. Not surprisingly we obtain not only simpler formulae than Bermond and Peters, but they are valid for a larger class of traffic patterns. Note that, in [BP05] $d > 1$ and it is not symmetric because they deal with the exact case of gathering (directed interference).

The work in [GPR08] presents a lower bound for RWP based on the graph coloring problem, and several experiments in which this lower bound was tight. They present a mixed integer linear programming model and a branch-and-price algorithm to solve the RWP considering mono-routing. The work presented in [GPRR08] has given optimal bounds to the problem considering grid graphs.

In [GH07], a column generation algorithm is used to avoid dealing with the exponential set of rounds of the RWP. They present a multi-objective study relating the minimum transmission time (in number of rounds) and the maximum load, observing that the worst bottleneck was located around the gateway in the instances of test.

The present chapter extends the work in [GPR08, GPRR08], we present a polynomial algorithm to solve RWP in special graphs. We have a particular interest in the network graphs where the clique number represents already the optimal solution of our problem.

6.3 Problem Statement

Suppose that time is synchronous and divided in time-slots. Each node can either transmit or receive at the same time-slot but not both simultaneously. We denote a *call* to a transmission between two nodes of the graph. Due to the interference (see section 6.3.1) two calls may be performed simultaneously or not. Therefore we define a *round* R as a set of pairwise non-interfering calls. In other words, R is a set of calls which can be done at the same time-slot. Each round is represented by a set of edges. Let \mathcal{R} be the set of all the rounds.

The ROUND WEIGHTING PROBLEM problem has been formalized in [KMP08] for general demands $b(u, v)$ from any demanding node u to any destination v . Here we restrict ourselves to the gathering instances. Therefore, we consider a traffic gathering where the demand $b : V \rightarrow \mathbb{R}^+$ represents the flow $b(v)$ needed to be sent from v to the gateway g .

Given a graph $G = (V, E)$ and the demand function, the problem consists in providing a weight w to the rounds. Therefore, the function w will induce a capacity over the each edge e given by the sum of the weights of the rounds containing the edge e . We denote $c_w(e)$ the induced capacity of the edge e . In this way $c_w(e) = \sum_{R \in \mathcal{R}: e \in R} w(R)$. Therefore, a solution w is admissible if there exists a flow ϕ such that:

- it satisfies the traffic demand b

$$(\forall v \in V) \quad \sum_{P \in P_{v,g}} \phi(P) \geq b(v),$$

where $P_{v,g}$ denotes the set of paths between v and BS.

- and respects the capacity c_w induced by w :

$$(\forall e \in E) \quad \sum_{P \in \mathcal{P}: e \in P} \phi(P) \leq c_w(e) = \sum_{R \in \mathcal{R}: e \in R} w(R).$$

Then, the ROUND WEIGHTING PROBLEM may be summarized as follows.

PROBLEM:	Round weighting for gathering instances
INPUT:	a graph $G = (V, E)$, a set of possible rounds $\mathcal{R} \subset 2^E$ (whose size may be exponential), and a flow demand function $b : V \rightarrow \mathbb{R}^+$.
SOLUTION:	A round weight function w defined over \mathcal{R} that satisfies the traffic demand b .
GOAL:	Minimize the overall weight of w , i.e. $W = \sum_{R \in \mathcal{R}} w(R)$.

6.3.1 Interference model

We use a model of interference base on the distance of graph and that we call d (*symmetrical*) *interference* model. This model can be considered as a symmetric variant of the interference model presented in [KMP08].

Notice that if device u calls device v , it is desirable that v has a way to let u know that the transmission has been successful (acknowledgment or ACK). Such feedback is performed as a transmission from v to u . Furthermore, that is the model considered in the protocol 802.11 and is named in some papers as the 802.11 interference model [Wan09a]. For this reason most applications and consequently models assume that transmission as well as interference are symmetrical. Therefore, we also use a symmetrical version of the interference model. In this version, two symmetrical calls $u \leftrightarrow v$ and $u' \leftrightarrow v'$ interfere if a call has one of its end vertex in the interference range of distance d of some end vertex of the other call. More precisely, the two symmetrical calls (we will also say the two edges (u, v) and (u', v')) interfere if $\min_{x \in \{u, v\}, y \in \{u', v'\}} d_G(x, y) < d$. Said otherwise two calls are compatible if there are at least d edges between them. The particular case $d = 1$ is nothing else than the primary node interference model or node-exclusive interference model [MSS06], used in many articles. In that case, a round is a matching. In the case $d = 2$ we get the so called distance-2 interference model [KMPS04, BKK⁺09a, Wan09a, WWLS08]. In this case, a round is an induced matching. The conflict graph in the case $d = 1$ corresponds to 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).

6.3.2 Definitions

We will present some definitions that will be useful in future sections. The definitions are the following:

Definitions related to the edges of G

- $G(V, E)$: graph with nodes in V and edges in E .
- $L(G)$: 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).
- 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, $E_l = \{e = (u, v) \in E \mid d(g, u) = l \text{ and } d(g, v) = l - 1\}$. 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 , denoted $C(G)$, is the graph whose vertices represent the edges of G , two vertices are joined if the corresponding edges (which represent calls) interfere. In the d -interference model we consider, the conflict graph is the d -th power of the line graph $L(G)$.
- R (Round): set of non-interfering edges. It corresponds to an independent set in $C(G)$.
- call-clique: set of edges pairwise interfering. The corresponding vertices form a clique in $C(G)$. For example in the d -interference model, K_0 is a call-clique.

Definitions related to the flow

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

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

- $\phi_v(e)$: flow sourced at node v traversing the edge e .
- $\phi(e)$: flow traversing the edge e . $\phi(e) = \sum_{v \in V} \phi_v(e)$.
- \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 ϕ if

$$c_w(e) \geq \phi(e) \quad \forall e \tag{6.1}$$

- $\phi(E')$: $\sum_{e \in E'} \phi(e)$. 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.

6.4 Lower Bounds

In this section, we present lower bounds for the problem of RWP. In all the subsections we first present lower bounds independent of the interference models. Then we give more precise lower bounds for the d -interference model and for the grid.

6.4.1 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 6.1 *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 6.2 $c_w(F) \geq \sum_{v \in V} b(v) \text{LB}(v, F)$.

Proof: For any flow ϕ 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 $d(v, g) \geq l$, $\text{LB}(v, E_l) \geq 1$ and we have the following corollary.

Corollary 6.1 $c_w(E_l) \geq \sum_{v; d(v,g) \geq l} b(v)$.

We will use the corollary 6.1 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 6.1 $S_0 = \sum_{v \in V_{K_0}} d(v, g) b(v) + \lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v)$.

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

Lemma 6.3 $c_w(K_0) \geq S_0$.

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 6.1 In the d -interference model $W_{\min} \geq S_0$.

In some cases, the lower bound S_0 is attained. We will see after that it happens for the grid with the gateway at the center and unitary traffic for d odd. In some other cases we use lemma 6.2 with a maximum call-clique K containing K_0 . For example, for the grid with d odd and the gateway in the corner, the maximum call-clique is larger than K_0 (see figure 6.19) and gives a better bound than S_0 (Theorem 6.2). We will show that the bound is attained for unitary demand. However, using only one call-clique does not necessary give a tight bound.

6.4.2 Lower Bounds using many call-cliques

We present a result similar to lemma 6.2 but improved for multiple sets of edges. We denote $\mathcal{P}_{v,g}$ the set of all the paths between v and g .

Lemma 6.4 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 ϕ and any node v , $\sum_{i=1}^q \phi_v(F_i) \geq b(v) \min_{P_{v,g}} \sum_{i=1}^q \text{LB}(P_{v,g}, F_i)$. ■

Consider the example of a grid with the gateway at the corner and $d = 2$ depicted in figure 6.1. 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 6.4, 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.

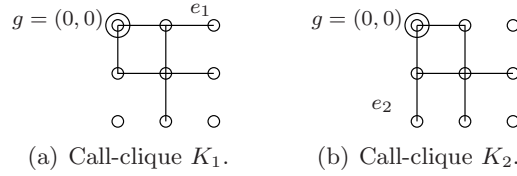


Figure 6.1: Two maximum call-cliques K_1 and K_2 for the case $d = 2$.

Proposition 6.2 *For the grid with the gateway in the corner and $d = 2$*

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

In general we have the following lemma.

Lemma 6.5 *Let K_1, \dots, K_q be a family of call-cliques. Then one of the call-cliques K^* satisfy $c_w(K^*) \geq \frac{1}{q} \sum_{v \in V} b(v) \min_{P_{v,g}} \sum_{i=1}^q \text{LB}(P_{v,g}, K_i)$*

Proof: By lemma 6.4, $\sum_i c_w(K_i) \geq \sum_{v \in V} b(v) \min_{P_{v,g}} \sum_{i=1}^q \text{LB}(P_{v,g}, K_i)$ and so one of the call-cliques, denoted K^* , has value $c_w(K^*)$ greater than or equal to the mean. ■

Corollary 6.2 *Let K_1, \dots, K_q be a family of call-cliques such that each edge of E_l appears at least λ_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 6.3 *Let G the grid with the gateway at the center 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 6.2 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.

■

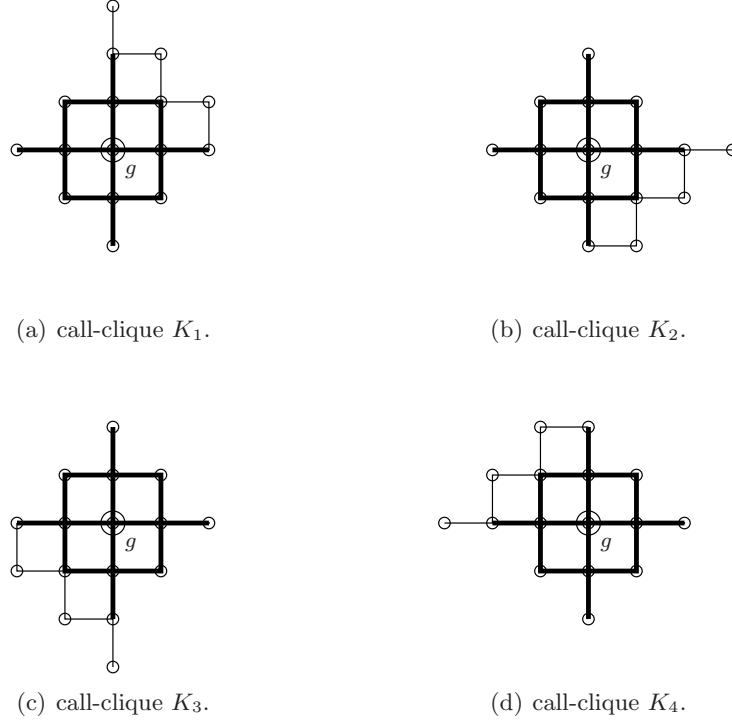


Figure 6.2: Case d even and g in the middle. The 4 call-cliques combined covers E_i , $1 \leq i \leq k + 1$ for $d = 2k$. In this scheme, $d = 4$.

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 necessarily 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} (see Proposition ?? where it is proven that one of the call-cliques K^* satisfies $c_w(K^*) \geq \frac{5}{2}b((3, 2))$).

A better lower bound for the same example consists in using call-cliques which not necessarily cover the gateway. The new lower bound uses the call-cliques depicted in figure 6.4. The call-clique K_2 (see figure 6.4(b)) is used twice and K_1 (see figure 6.4(a)) and K_3 (see figure 6.4(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

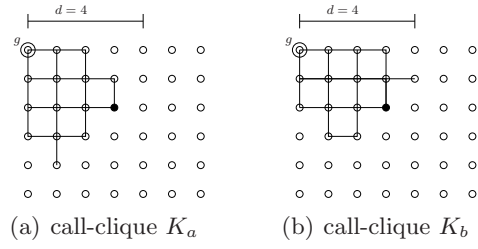


Figure 6.3: Example of a specific lower bound when the demand is concentrated in one node. In this example, $d = 4$ and the demand is concentrated in node $(3, 2)$. A lower bound of $\frac{5}{2}b((3, 2))$ is attained using the two call-cliques K_a and K_b .

Table 6.1: Possible Paths							
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)	-	5	3	-	11
-	(1, 3)	(1, 2)	(1, 1)	4	3	2	12
-	(1, 3)	(1, 2)	(0, 2)	4	2	3	11
-	(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

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 6.4(d) that we do not have path from $(3, 2)$ to the gateway $(0, 0)$ that costs less than 11. 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 $c_w(K^*) \geq \frac{11}{4}b((3, 2))$.

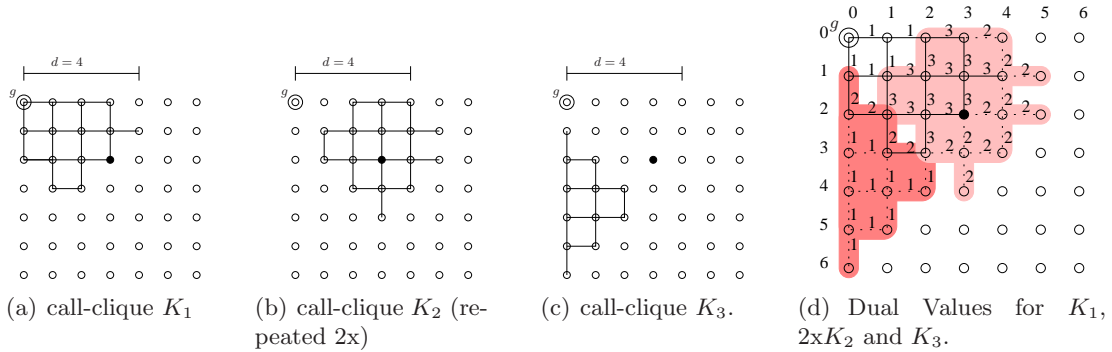


Figure 6.4: Example with $d = 4$ and the demand is concentrated in node $(3, 2)$. Four call-cliques are needed to obtain a tight lower bound of $\frac{11}{4}b((3, 2))$ which is higher than $\frac{5}{2}b((3, 2))$.

6.4.3 Lower Bounds using Critical Edges

Lemma 6.5 is not sufficient in all cases. Consider the example of figure 6.5 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 6.2 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 6.5(c) shows an integer solution with $W_{\min} = 13$ and figure 6.5(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 $W \geq 10 + 2.5 = 12.5$.

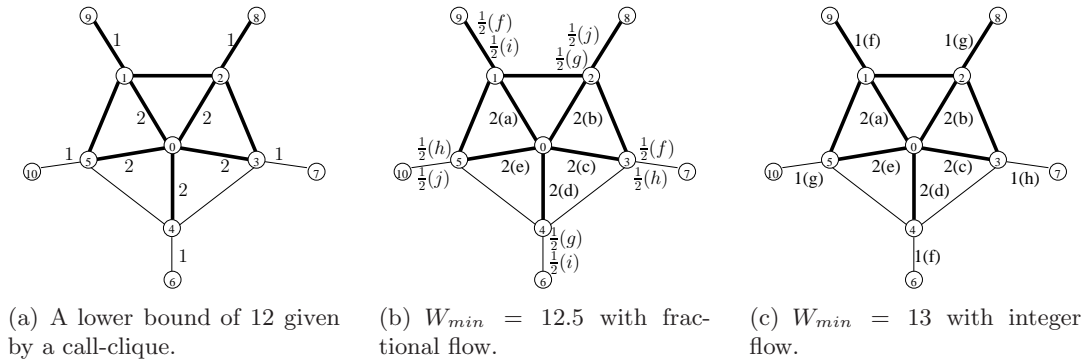


Figure 6.5: Example of lower bound calculation. In this case $d = 3$

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 colors implying in the integer case a lower bound of 3 and so $W_{\text{integer}} \geq 13$. In the fractional case, it is known that we can use a fractional coloring with $\frac{5}{2}$ colors.

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 conflict graph generated by F .

Definition 6.2 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 6.6 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

$$W = \sum_{R \in \mathcal{R}} w(R) = \sum_{R; R \cap K \neq \emptyset} w(R) + \sum_{R; R \cap F \neq \emptyset} w(R) \quad (6.2)$$

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 (6.2), we have that $W \geq c_w(K) + \frac{c_w(F)}{\alpha(F)}$. \blacksquare

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 6.3 *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 the corollary 6.3 for the grid with the gateway at the center 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 6.3.

6.4.4 Relationship with Duality

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

The dual formulation of RWP has been studied in [KMP08]. A dual solution for the RWP for gathering instances can be described with the following property.

Property 6.1 ([KMP08]) *The dual problem of round weighting consists of finding a metric $m : E \rightarrow \mathbb{Q}^+$ 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 $\bigcup \mathcal{K}$ we define $\mathcal{K}_e = \{K \in \mathcal{K} \mid e \text{ is an edge of } K\}$. Thus, the dual solution proposed $m : E \rightarrow \mathbb{Q}^+$ 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, $\sum_{e \in E'} m(e) = \sum_{e \in E'} \frac{|\mathcal{K}_e|}{|\mathcal{K}|} \leq 1$.

6.5 Application in grids

In this section¹, we apply the lower bound given in section 6.4 to a particular case of the grid. Notice that the formulas are indeed optimal as we show in subsection 6.5.2. We present lower

¹The section *Application in grids* will be updated for the final version.

bounds for any given interference parameter d and a given demand B . In subsection 6.5.2, we give protocols (upper bound) which are optimal in some cases. To understand what we want to show, we present several examples of grid graphs with the gateway placed at several places considering uniform unitary demand. Figures 6.6(e) and 6.6(f) show call-cliques with $d = 2$ for a grid with the gateway in the middle. Figures 6.7(c) and 6.7(d) show similar results for the gateway at the corner. Figures 6.8(a)-6.8(d) show one of the call-cliques for a grid considering $d \in \{3, 4, 5, 6\}$. The dotted edges are only to illustrate the routing direction of the demand for some critical nodes. Figures 6.9(a)- 6.9(d) shows the gateway placed in different parts of the grid.

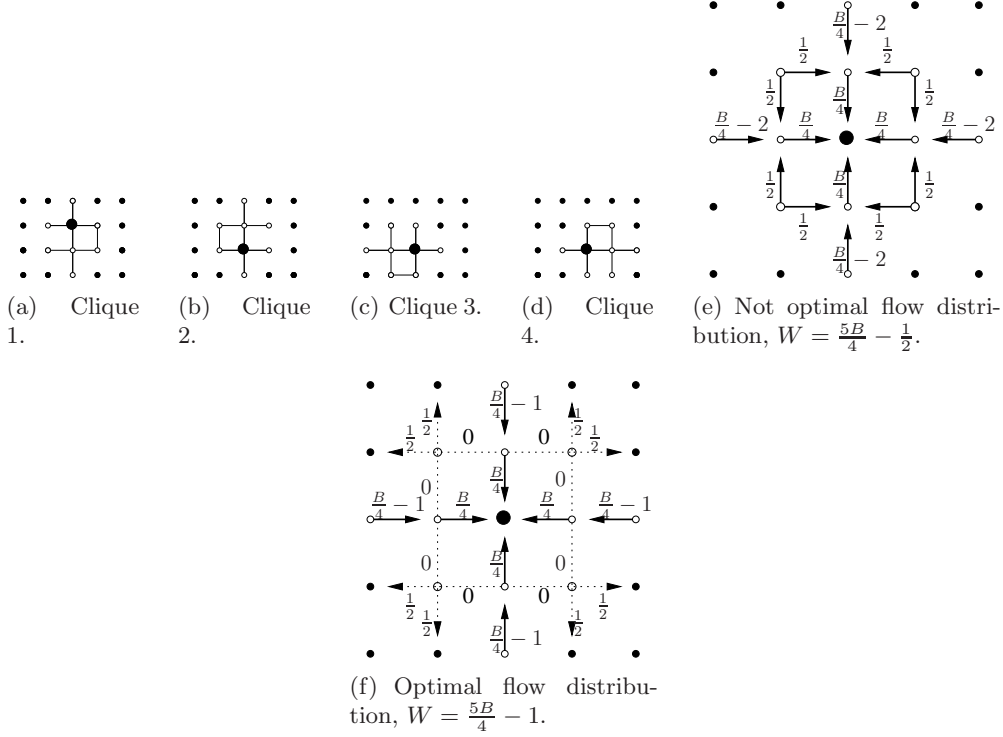


Figure 6.6: Looking for the best routing in a grid graph with the gateway in the middle with uniform demand ($d=2$).

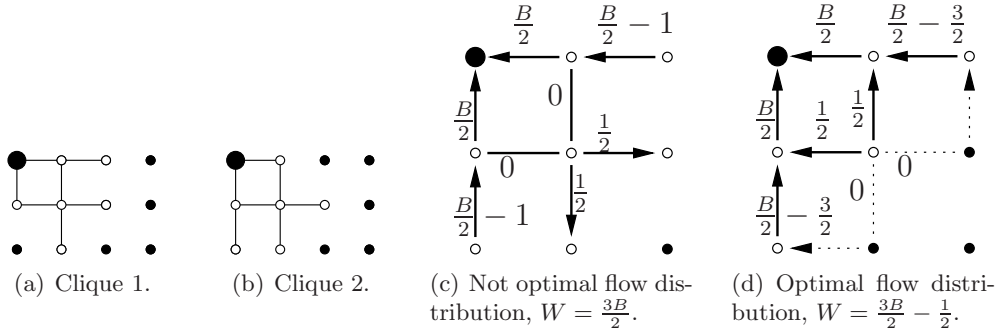


Figure 6.7: Looking for the best routing in a grid graph with the gateway at the corner with uniform demand ($d=2$).

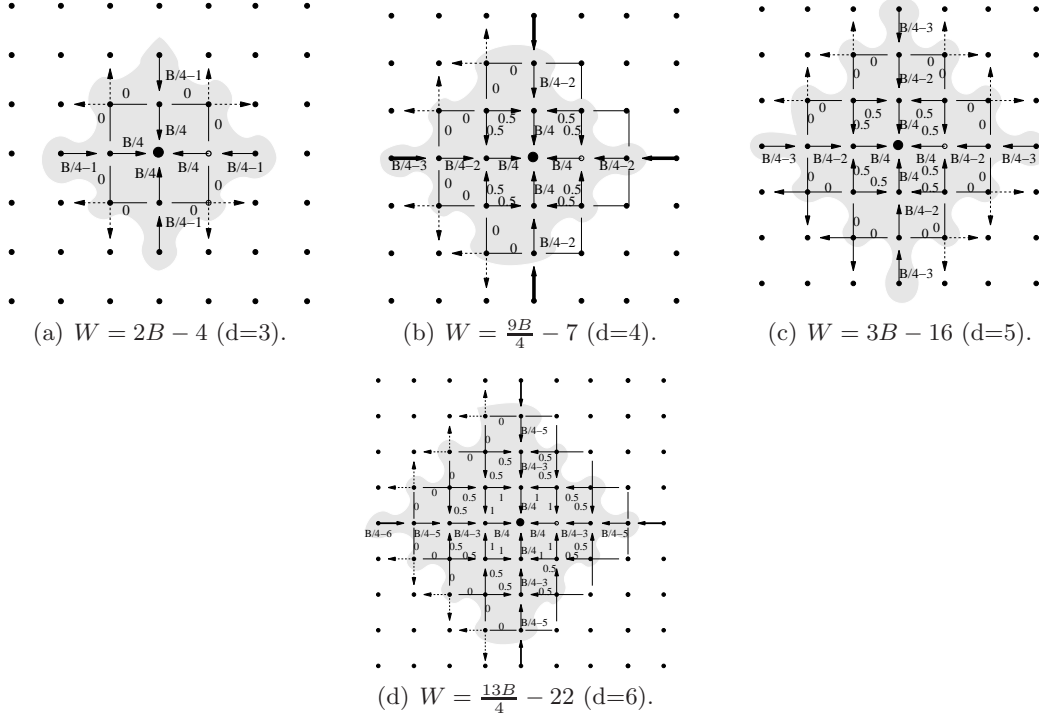


Figure 6.8: Call-cliques considering other interference distances.

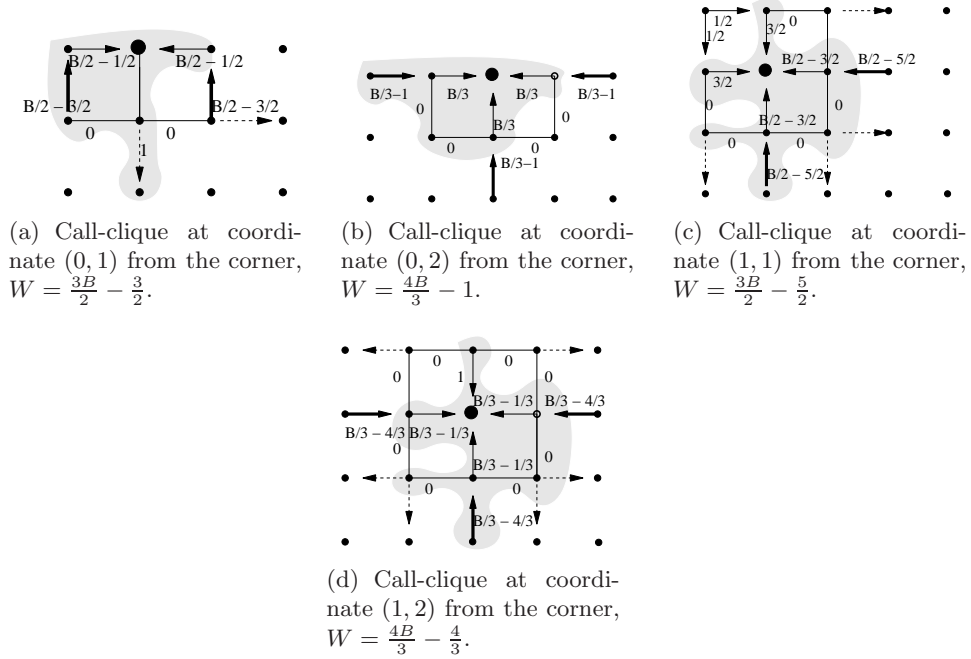


Figure 6.9: Call-cliques for the grid graph with the gateway in other positions with uniform demand ($d=2$).

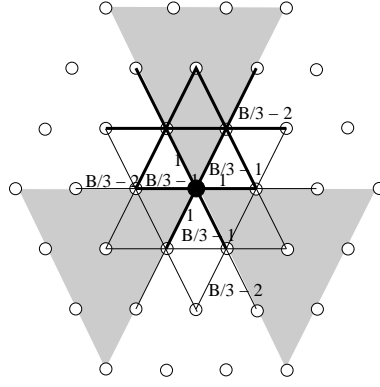


Figure 6.10: Hexagonal graphs ($d=2$) at the center, $W = \frac{4B}{3} - 2$

An example in hexagonal grids can be seen in figure 6.10.

6.5.1 Gateway in the middle: A lower bound

Now we consider the case when g is placed at a distance at least $\lceil \frac{d+1}{2} \rceil$ from the borders of the grid. For d odd, we only consider a lower bound using the call-clique given by K_0 . Using proposition 6.1, we obtain the lower bound $\sum_{v \in V_{K_0}} d(v, g)b(v) + \lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v)$. The figure 6.11 shows an example of a clique in this case.

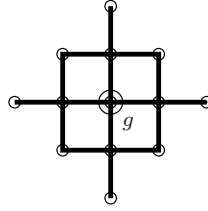


Figure 6.11: Clique for d odd with g in the middle. In this scheme, $d = 3$.

For the case when d is even, we will use 4 cliques to cover the edges of $K_0 \cup E_{\frac{d}{2}+1}$. Figure 6.12(a) shows one of the call-cliques. Note that the set of edges depicted is a call-clique because each edge is at distance at most d to any other edge. The remaining call-cliques are constructed by rotating the first one as shown in Figure 6.12(b).

Using these 4 cliques we obtain the lower bound given by property 6.3.

Lower bound formulas for grids

In the following, we consider the case of unitary demand ($b(v) = 1, \forall v$) in a square grid of size $N \times N$. In this case, the total demand is $N^2 - 1$. We derive formulas only in function of the d that compute a lower bound for grid graphs. In subsection 6.5.2, we prove that these formulas give the optimal solution.

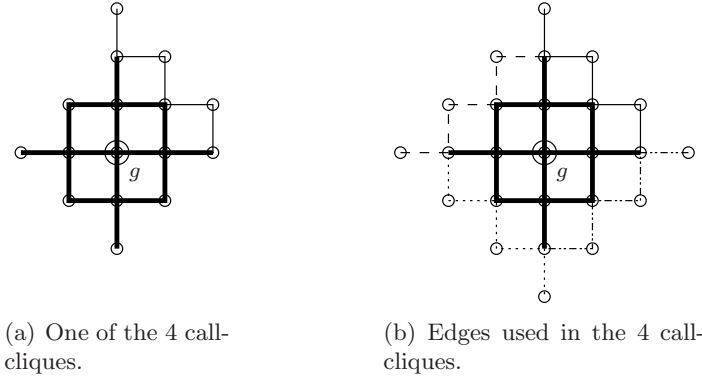


Figure 6.12: Case d even and g in the middle. In this scheme, $d = 4$.

In the following, we consider N_i as the number of nodes at distance i to the gateway, hence $N_i = 4_i$.

Lemma 6.7 *Given a grid $N \times N$ with the gateway in the middle. Considering unitary demand and $d = 2k - 1 \leq N - 2$ odd, then $W_{\min} \geq k(N^2 - 1) - \frac{4}{6}(k+1)(k)(k+1)$.*

Proof: By proposition 6.1, $W_{\min} \geq S_0 = \sum_{v \in V_{K_0}} d(v, g)b(v) + \lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v)$. As $b(v) = 1$ 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 $\lceil \frac{d}{2} \rceil \sum_{v \notin V_{K_0}} b(v) = k(B - \sum_{i \leq k} N_i)$ where $B = N^2 - 1$. Then we have:

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

■

Lemma 6.8 *Given a grid with the gateway in the middle. Considering unitary demand B and $d = 2k$ even, then $W_{\min} \geq (k + \frac{1}{4})B - \frac{k(k+1)(4k-1)}{6}$.*

Proof: As $b(v) = 1$ for all v , by proposition 6.3, $W_{\min} \geq S_0 + \frac{1}{4} \sum_{v; d(v, g) > k} b(v) = S_0 + \frac{1}{4}(B - \sum_{i \leq k} N_i)$. From lemma 6.7, $S_0 = kB - \frac{4}{6}k(k-1)(k+1)$ and

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

■

In section 6.5.2, we will prove that these formulas give the optimal solution.

Method of routing with gateway in the middle

We study the routing of a unique node with flow $b_v = 1$. Figures 6.13 and ?? show different zones where can be localized the node v considering an even d and odd d respectively. Depending of the node position it has a different W_{min} . Figure 6.13 shows the zones considering an even d , in this case we have just the kernel and the zone Z_A . The zone Z_A is composed by all nodes that are at distance of g greater than $\frac{d+1}{2}$. Considering a grid big enough, we can always make a unique cycle using the lower bound, thus $W_{min} = \frac{d+1}{2}$. Figure 6.14 shows some examples of routing. Notice that we do not have problems with the border due to the fact that the routing uses a unique cycle.

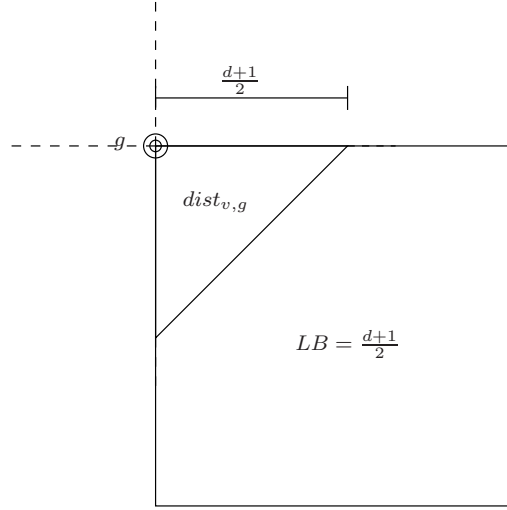


Figure 6.13: Scheme of the grid separated by method of routing considering odd d .

6.5.2 Gateway in the middle: An upper bound

In this section, we present a routing strategy providing a valid solution for our problem in a grid. We consider the *case* of a grid graph with the gateway in the middle, considering any d and *balanced demand* (see definition 6.3). We prove that this upper bound reaches our lower bound for grid graphs, it means that our solution is optimal. The routing strategy splits the grid in disjoint regions. Each region has an orientation and a pre-defined sequence of labels. A path crossing a region has to follow the region orientation and it is covered by a repetition of the labels (sequence) associated to the region.

We use the coordinate system as in [BP09] to represent a node position, we consider the gateway g on the coordinate $(x, y) = (0, 0)$. The edges of the routing paths are represented by the pair of letters (u, v) , describing two different vertices and a direction. That is a transmission between the coordinate of u to the coordinate of v . We define (x, y^+) the edge (u, v) defined by $u = (x, y)$

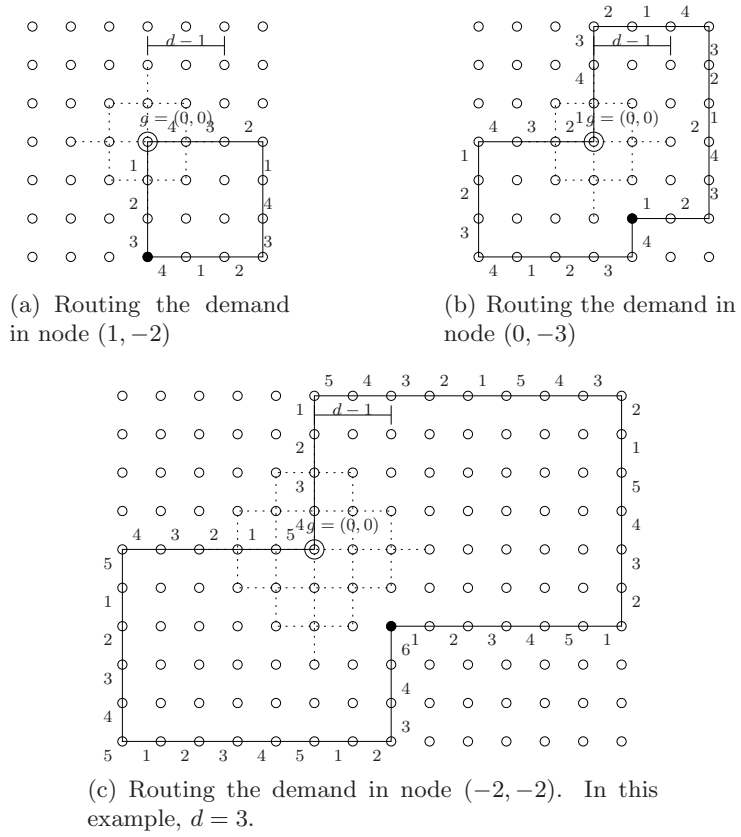


Figure 6.14: Example using a unique cycle with odd d . In this example, $d = 5$.

and $v = (x, y + 1)$. Also, (x, y^-) means the edge (u, v) defined by $u = (x, y)$ and $v = (x, y - 1)$. The signal can be applied also to x . Before presenting the routing strategy, we need the following definitions.

Definitions:

- $k: \lceil \frac{d+1}{2} \rceil$. Recall d is the distance of interference. In this way, $d = 2k - 1$ for d odd and $d = 2k$ for d even.
- P_i : a set of the edges of G . For our case, we split the grid into disjoint parts P_i .
- CH_j : A chain, that is a sequence of labels in increasing order that is allocated to a part. In grid graphs we need 4 chains: CH_A, CH_B, CH_C, CH_D and one more label e for the cases where d is even. Each chain has k different labels, for instance $CH_A = \{a_1, a_2, \dots, a_k\}$. We define CH'_j the chain in inverse order, thus $CH'_A = \{b_k, \dots, b_2, b_1\}$. We use the chains to cover the edges of the paths, each label is assigned to each edge of the path respecting the order imposed by the chain. Notice that a *round* is defined by the set of edges with the same label.
- V_I : the set of intermediary vertices in K_0 for all paths created by vertices $v \in P_i$. The vertices $(0, k)$, $(k, 0)$, $(0, -k)$ and $(-k, 0)$ (the white vertices in Figure 6.15(b)).
- ρ : corresponds to a rotation in the plane of $\frac{\pi}{2}$ around the gateway node represented by g . A rotation ρ is the node-to-node mapping $\rho((x, y)) = (-y, x)$. Similarly, $\rho^2((x, y)) = (-x, -y)$ corresponds to a rotation of π , and $\rho^3((x, y)) = (y, -x)$ corresponds to a rotation of $\frac{3\pi}{2}$.
- $\rho(e)$: rotation of an edge e , $\rho((u, v))$ corresponds to the edge $(\rho(u), \rho(v))$.
- E_{CH_j} : set of edges e that are supposed to receive a label from CH_j in our routing strategy (see Figure 6.16(a)).
- $\rho(E_{CH_j})$: rotation of a set E_{CH_j} of edges, $\rho(E_{CH_j}) = \{\rho(e) | e \in E_{CH_j}\}$.

We partition the grid into five sets of edges: K_0, P_A, P_B, P_C and P_D (see Figure 6.15(a)). The set P_A is defined as the edges of the grid joining the vertices in $\{(x, y) \mid x + y \geq k, x, y \geq 0\}$. We can define the other sets by rotation, thus the part $P_B = \rho(P_A)$ and the part $P_C = \rho(P_B) = \rho^2(P_A)$. Finally, the part $P_D = \rho(P_C) = \rho^2(P_B) = \rho^3(P_A)$. The part P_A is splitted in three different sub-parts denoted P_i^1 , P_i^2 and P_i^3 (see Figure 6.15(b)). The following routing strategy is defined in Figure 6.16(a). The chains assigned for all parts are shown in Figure 6.16(b).

Definition 6.3 (Balanced demand) *Each part P_i has the same quantity of flow to send, that is $\sum_{v \in P_i} b(v) = \sum_{u \in P_h} b(u)$ for all parts h and i . Into the Kernel we can consider any traffic because the vertices are routed individually as shown in the Lemma 6.1. An uniform demand $b(v) = 1, \forall v \in P_i$ for grid graphs is a specific case of balanced demand because the parts have the same number of nodes.*

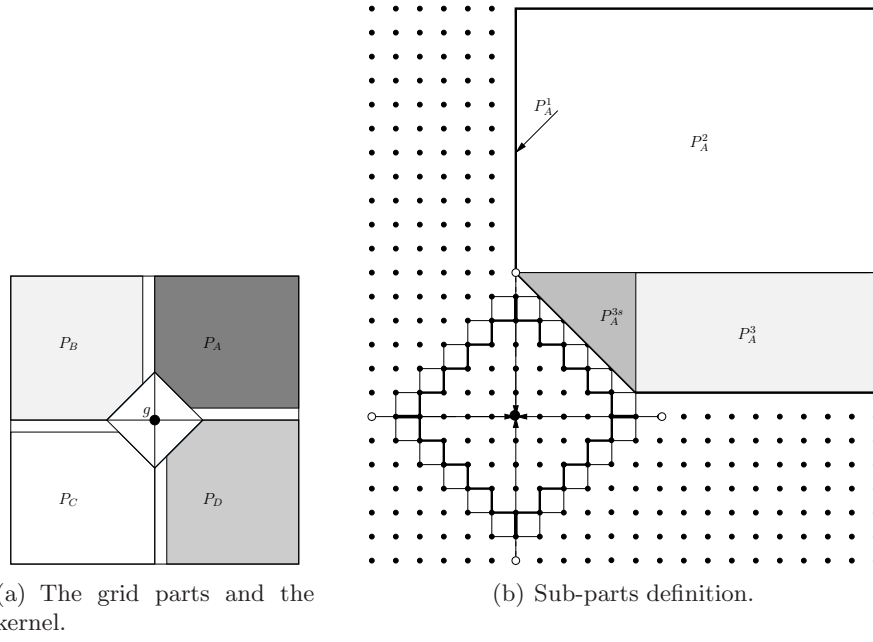


Figure 6.15: Parts Definition.

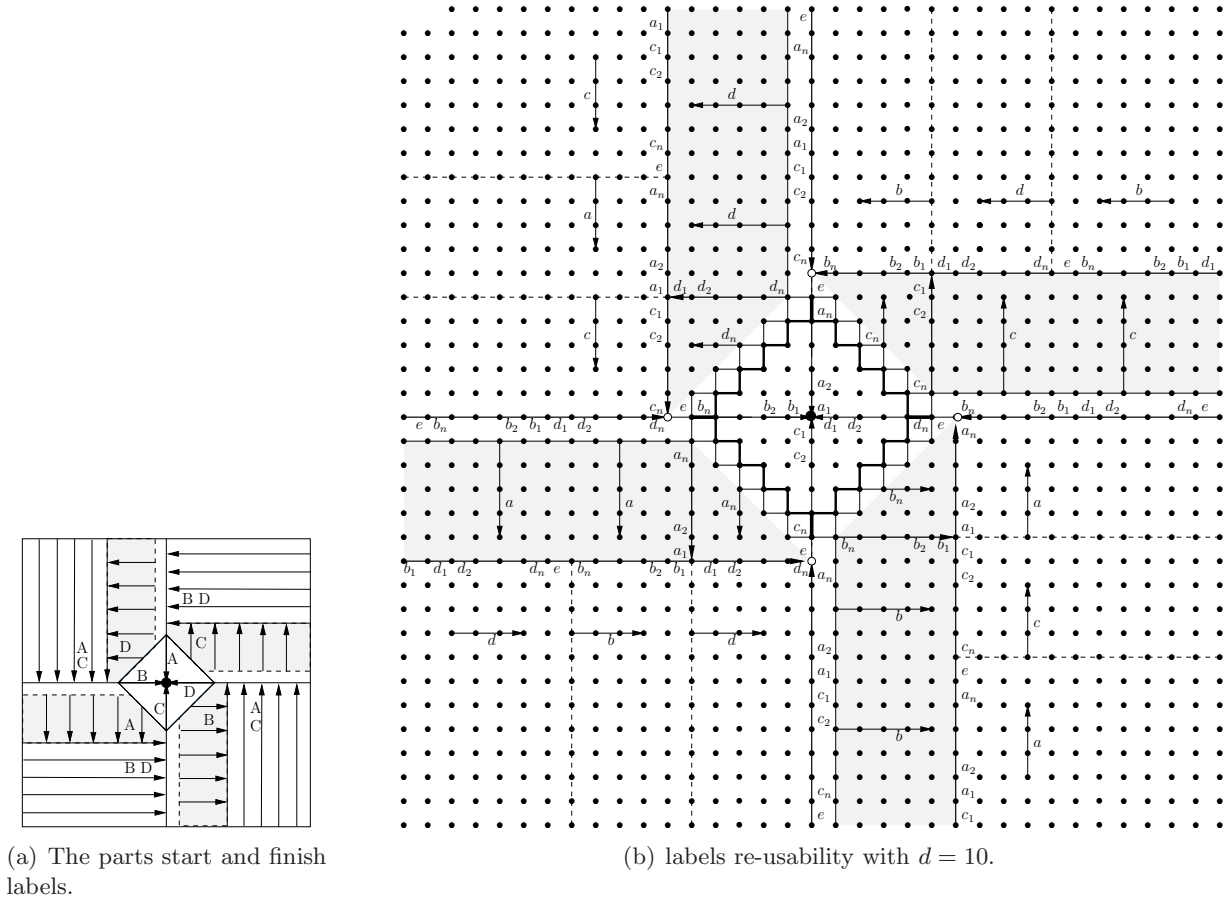


Figure 6.16: The routing strategy.

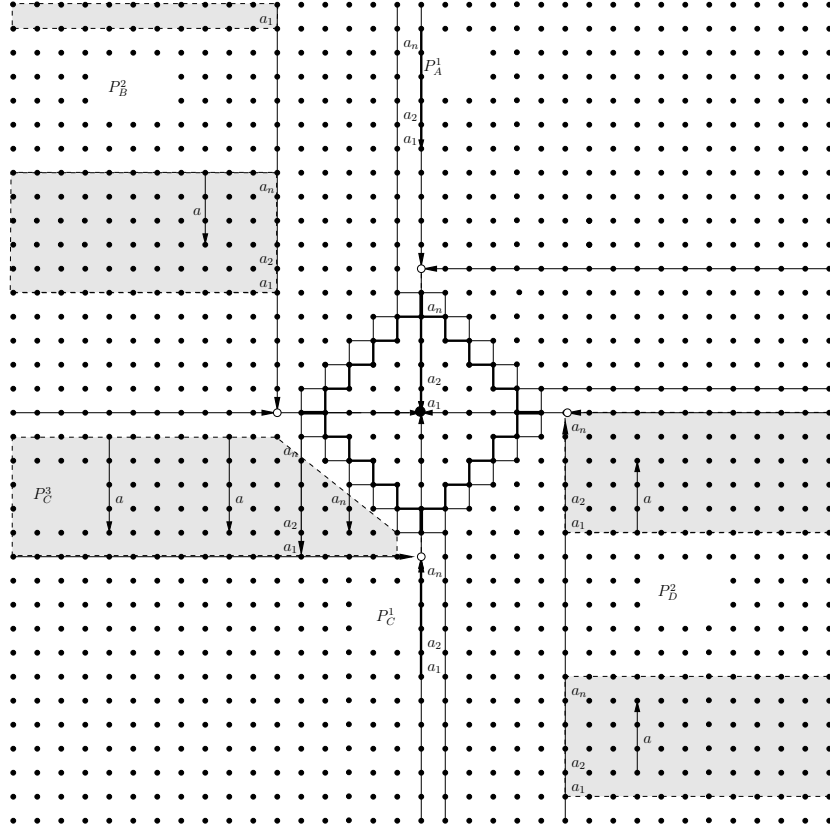


Figure 6.17: The chain CH_A .

Sub-part definition and routing strategy:

- P_A^1 : vertices with coordinates $(0, y \geq k)$ in P_A . Each path using these vertices assumes the direction $(0, y) \rightarrow (0, y - 1)$ and has its edges covered by an alternating between the chains CH_A and CH'_C . From the gateway to the source node in P_A^1 , the edges of the path receive a repetition of this ordered sequence: $\{a_1, a_2, \dots, a_k, c_k, \dots, c_2, c_1, a_1, a_2, \dots\}$, this sequence is repeated until reaching the source node in P_A^1 .
- P_A^2 : vertices with coordinates $(x > 0, y \geq k)$ in P_A . Each flow on these vertices is routed in the direction $(x, y) \rightarrow (x - 1, y)$ alternating between the chains CH'_B and CH_D until reaching a node in part P_A^1 , this node becomes the source node for the part P_A^1 (see the routing strategy for P_A^1). From this node at sub-part P_A^1 to the source node in P_A^2 , the edges of the path receive a repetition of this ordered sequence: $\{b_k, \dots, b_2, b_1, d_1, \dots, d_k, b_k, \dots\}$, this sequence is repeated until reaching the source node in P_A^2 .
- P_A^3 ($P_A^{3s} \subset P_A^3$): vertices with coordinates $(x > 0, 0 < y < k)$ in P_A . Each flow on these vertices is routed in the direction $(x, y) \rightarrow (x, y + 1)$ using only the chain CH'_C until reaching a node in part P_A^2 , this node becomes the source node for the part P_A^2 (see the routing strategy for P_A^2). From the source node in P_A^3 to the node at sub-part P_A^2 , the edges of the path receive this ordered sequence: $\{c_n, \dots, c_2, c_1\}$. The subset P_A^{3s} is defined by the vertices with coordinates $(0 < x < k, 0 < y < k)$.

Now we introduce the following definition which characterizes the interference-free paths.

Definition 6.4 (Paths d -disjoint γ -labelled) *Two paths P and Q are said to be interference free γ -labelled if we can label the edges with γ colors such that two edges with the same color do not interfere. In the model with distance of interference d we will say the paths are d -disjoint γ -labelled.*

The following lemma shows that it is possible to define at each iteration 4 paths d -disjoint γ -labelled. Each path is completely contained into a different part P_i and takes 1 unit of flow from a vertex $v \in P_i$ to the gateway.

Lemma 6.9 *Given a 2-dimensional grid graph $G = (V, E)$ with gateway g placed in the middle. Our routing strategy does not present neither interfering chains into a same part nor between a part and the kernel.*

Proof: We show for the chain CH_A and the others can be obtained by rotation. The kernel uses the chain CH_A between the edges $E_{K_0} = \{(0, k^-)_1 \dots (0, 1^-)_k\}$ considering d odd. For d even, we add one more label without problems as shown in Figure 6.16(b).

- P_A^1 : The chain CH_A is repeated in P_A^1 using the edges $E_{P_A^1}^n = \{(0, 3kn^-)_1 \dots (0, (2kn + 1)^-)_k\}$ (with $n \in \mathbb{N}$). As the distance between the two chains $d(u_i \in E_n, v_i \in E_{K_0}) \geq d$, these two

chains do not interfere. The chain CH_A is repeated in P_A^1 at each interval of k , so we have no interference because $d(u_i \in E_{P_A^1}, v_i \in E_{P_A^1}^{n+1}) = 2k > d$. So, one chain can be repeated at each interval of k .

- P_B^2 : The chain CH_A is also used in P_B^2 , it is repeated at each interval of k . The closest part to the kernel uses the edges in $E_{P_B^2} = \{(-k, 2k^-)_1 \dots (-k, (k+1)^-)_k\}$, thus $d(u_i \in E_{P_B^2}, v_i \in E_{K_0}) = 2k \geq d$.
- P_C^3 : The chain CH_A is used once in P_C^3 , notice that P_C^1 and P_C^3 are mutually exclusive in a path. The closest part to the kernel is defined by the edges $E_{P_C^3} = \{(-k, -1^-)_1, -(k-1), -2^-\}_2, \dots, (-1, -k^-)_k\}$. Any edge $u \in E_{P_C^3}$ has distance $d(u_i \in E_{P_C^3}, v_i \in E_{K_0}) = 2k \geq d$.
- P_D^2 : The chain CH_A is used in P_D^2 . The chain CH_A is repeated in P_D^2 at each interval of k . The closest part to the kernel is $E_{P_D^2} = \{(k, -k^+)_1 \dots (k, -1^+)_k\}$, thus $d(u_i \in E_{P_D^2}, v_i \in E_{K_0}) > d$.

■

Lemma 6.10 *Given a 2-dimensional grid graph G with gateway g placed in the middle. Our routing strategy does not present interfering chains between opposite parts, that is between P_A and P_C , or P_B and P_D .*

Proof: Any pair of edges $((x_a, y_a) \in P_A, (x_c, y_c) \in P_C)$ does not interfere, the same happens for the edges from P_B and P_D by rotation. It is because these pairs of parts has the kernel (diameter d) separating them. ■

Lemma 6.11 *Given a 2-dimensional grid graph $G = (V, E)$ with gateway g placed in the middle. Our routing strategy does not present interfering chains.*

Proof: By lemma 6.9 the chains does not interfere to the chains of the kernel. Now, we list all possible combinations of sub-parts using the chain CH_A , and we prove the sub-parts also do not interfere. The same arguments can be applied to other chains by rotation (the chains have the same configuration, see figure 6.16(b)).

- $P_A^1 \times P_B^2$: The chain CH_A is repeated in P_A^1 starting by the edges $E_{P_A^1} = \{(0, 3k^-)_1 \dots (0, (2k+1)^-)_k\}$. In P_B^2 , the closest part defined parallel to P_A^1 (the distance is maintained between the labels of the chain) is defined by the edges $E_{P_B^2} = \{(-k, 2k^-)_1 \dots (-k, (k+1)^-)_k\}$. As the distance between the two chains $d(u_i \in E_{P_A^1}, v_i \in E_{P_B^2}) \geq d$ and it is maintained for the sequence of repetitions of the chains, these two sub-parts do not interfere.
- $P_A^1 \times P_C^3$: Independent by the Lemma 6.10.
- $P_A^1 \times P_C^1$: Independent by the Lemma 6.10.

- $P_A^1 \times P_D^2$: The chain CH_A is repeated in P_A^1 starting by the edges $E_{P_A^1} = \{(0, 3k^-)_1 \dots (0, (2k+1)^-)_k\}$. In P_D^2 , the closest part to P_A^1 is defined by the edges $E_{P_D^2} = \{(k, -k^+) \dots (k, -1^+)\}$. As the distance between the two chains $d(u_i \in E_{P_A^1}, v_i \in E_{P_D^2}) \geq d$, these two chains do not interfere. As the other edges in P_D^2 are yet farther from the part P_A^1 , consequently these two sub-parts do not interfere.
- $P_B^2 \times P_C^3$: As the part P_C^3 goes far away from part P_B^2 using the chain CH_A after an inactive edge; and the part P_B^2 arrives in P_B^1 using the chains CH_C ($|CH_C| = k$) followed by CH_A . There is no interference, it is like a continuation of P_B^2 from one side to the other (repetitions at each interval of k).
- $P_B^2 \times P_C^1$: The same that $P_A^1 \times P_D^2$.
- $P_B^2 \times P_D^2$: Independent by the Lemma 6.10.
- $P_C^3 \times P_C^1$: There is no interference by Lemma 6.9.
- $P_C^3 \times P_D^2$: The chain CH_A is repeated in P_D^2 starting by the edges $E_{P_D^2} = \{(k, -k^+) \dots (k, -1^+)\}$ and in P_C^3 , the closest part is defined by the edges $E_{P_C^3} = \{(-k, -1^-)_1, (-(k-1), -2^-)_2, \dots, (-1, -k^-)_k\}$. The distance between the edge i in P_D^2 and the edges in $E_{P_C^3}^i$ is $d(u_i \in E_{P_D^2}, v_i \in E_{P_C^3}^i) \geq d$.
- $P_C^1 \times P_D^2$: The same that $P_A^1 \times P_B^2$.

■

The following theorem proves that the flows coming from the parts can be routed with $W = 4 \sum_{v \in P_A} b(v) (\lceil \frac{d}{2} \rceil + \frac{1}{4}((d+1) \bmod 2))$.

Theorem 6.1 *Given a 2-dimensional grid graph G with gateway g placed in the middle. Given four vertices v from different parts. We can route 1 unit from each v to the gateway with routing iteration $W = 4 \sum_{v \in P_A} b(v) (\lceil \frac{d}{2} \rceil + \frac{1}{4}((d+1) \bmod 2))$.*

Proof: By the Lemmas 6.9 and 6.11, the proposed routing does not use more chains than these ones already used by the kernel. Thus, the interval W is the same proposed by the lower bound for vertices $v \notin V_{K_0}$, that is $\sum_{v \notin V_{K_0}} b(v) (\lceil \frac{d}{2} \rceil + \frac{1}{4}((d+1) \bmod 2))$ by the Theorem ???. As we have balanced parts, $\sum_{v \notin V_{K_0}} b(v) = 4 \sum_{v \in P_A} b(v)$. ■

The following corollary states that the lower bound of the Lemma 6.1 is tight when the demand is balanced.

Corollary 6.4 $W_{\min} = \sum_{v \in V_{K_0}} d(v, g) b(v) + (\lceil \frac{d}{2} \rceil + \frac{1}{4}((d+1) \bmod 2)) \sum_{v \notin V_{K_0}} b(v)$.

Corollary 6.5 *We can route using integer flow and mono-routing with balanced parts. The balanced parts guarantees that we have always 4 paths for all iterations (routing 4 vertices), so the demand is not splitted.*

6.5.3 Gateway in the corner: A lower bound

We use a notation based on coordinates but for readability of the figures we have put the gateway labelled $(0,0)$ in the top left corner. So we use (x,y) instead of $(x,-y)$ the vertical axis being labeled with positive integers.

6.5.3.1 Case d odd

In this section, we study the case when d is odd. Let d be as $d = 2k - 1$.

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 will show that the lower bounds performed are tight.

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 $d = 9$ is depicted in figure 6.19.

Lemma 6.12 *For the grid with the gateway at the corner and $d = 2k - 1$, then*

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

where $\text{Int}(S_{\text{od}})$ is defined as $\{v \mid d(v, g) \leq 2k - 1 \text{ and } d(v, v^*) \leq k - 1\}$.

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_{od} . The number of edges used in S_{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_{od} composed by the vertices at distance $2k$ from g ($x + y = 2k$) ; or $k - d(v, v^*)$ to attain the diagonals bordering S_{od} below ($y = x + k$) or above ($x = y + k$). ■

Theorem 6.2 *For the grid with the gateway at the corner and $d = 2k - 1$,*

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

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 6.2 we can derive an explicit formula for the lower bound when the demand is uniform.

Proposition 6.4 *For the grid with $N \times N$ nodes ($N \geq k + \lceil \frac{k}{2} \rceil$) with the gateway at the corner and $d = 2k - 1$, if $b(v) = 1$ for all v , then*

$$W \geq kB + \frac{k-2}{6} \left\lfloor \frac{k}{2} \right\rfloor \left(\left\lfloor \frac{k}{2} \right\rfloor - 5 \right)$$

where $B = N^2 - 1$

Proof: We have to count $\sum_v \text{LB}(v, K_{\max})$. For all the vertices not in $V_{K_0} \cup \text{Int}(S_{\text{od}})$, $\text{LB}(v, K_{\max}) = k$ (Recall that $\text{Int}(S_{\text{od}})$ 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_{od} give an excess for those at distance $i > 0$ from one of the 4 diagonals delimiting S_{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_{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 $k = 2\lambda + 1$, $B_k - A_k = \frac{k}{6}\lambda(\lambda - 5)$. Then, in both cases $B_k - A_k$ can be written as $\frac{k-2}{6} \left\lfloor \frac{k}{2} \right\rfloor \left(\left\lfloor \frac{k}{2} \right\rfloor - 5 \right)$. ■

6.5.3.2 Case d even.

For the case when d is even we consider two cliques as shown in figure 6.20. The first clique corresponds to the solid edges and the dashed edges at the top. The second clique corresponds to the solid edges and the dashed edges at the bottom. We obtain the following lower bound.

$$\begin{aligned} & \sum_{v \in V_{K_0}} d(v, g)b(v) \\ & + \sum_{v \in S_{\text{ev}}} \left(\frac{d+1}{2} + \min \left\{ d - d(v, g) + 1, \quad d(v, g) - \frac{d+1}{2}, \quad \frac{d+1}{2} - d(v_{\text{ev}}^*, v) + 1 \right\} \right) b(v) \\ & + \frac{d+1}{2} \sum_{v \notin V_{K_0} \cup S_{\text{ev}}} b(v) \end{aligned}$$

with $S_{\text{ev}} = \{v \in V \mid \lceil \frac{d}{2} \rceil + 1 \leq d(v, g) \leq d \text{ and } d(v, v_{\text{ev}}^*) \leq \lceil \frac{d}{2} \rceil\}$.

6.5.4 Gateway in the corner: An upper bound

We start introducing the notion of the *width* of a cycle.

Definition 6.5 (Width of a Cycle. First version) *Given a 2-dimensional grid G . Let C be an cycle in the grid and we define d_C the distance over the cycle C . We define S_d the set of pairs (P, Q) of induced paths of C such that $d_C(P, Q) \geq d$. The width of a cycle is defined as $\min_{(P, Q) \in S_d} d_G(P, Q)$.*

Lemma 6.13 *Given a 2-dimensional grid G with nodes in $\{0, a\} \times \{0, b\}$, with $a \geq b$. Let v a node such that $v \notin \{1, d-1\} \times \{1, d-1\}$ with $d \geq 1$. If G satisfies one of the following conditions:*

- $b \geq 2d$ and $a \geq 3d$.
- $b = 2d - 1$ and $a \geq 5d$.
- $b = 2d - 1$ and $a \geq 3d$, for d odd.
- $b = 2d - 1$ and $a \geq 3d$, for d even and $a \neq k(d+1) + 4$, for any $k \geq 1$.

Then, a cycle of length multiple of $d+1$ can be done going through $g = (0, 0)$ and v and such that their width is at least d .

Proof: We distinguish 2 cases depending on whether or not $2a + 2b$ is multiple of $d+1$.

- **Case 1: $2a + 2b$ is multiple of $d+1$.** First, let us check the case when $2a + 2b$ is multiple of $d+1$. We distinguish 3 zones in the grid: $Z_{NE} = \{d, a\} \times \{0, d-1\}$, $Z_{SE} = \{d, a\} \times \{d, b\}$ and $Z_{SW} = \{0, d-1\} \times \{d, b\}$. (The notation follows the idea of the cardinal directions with the gateway placed in the corner at North-West.)

To define a cycle over the grid, we will use the following notation. We note $(x_1, y_1) - (x_2, y_2)$, with either $x_1 = x_2$ or $y_1 = y_2$, as the edges in the shortest path between (x_1, y_1) and (x_2, y_2) . Then, we note a cycle over the grid, for example, $(0, 0) - (x, 0) - (x, y) - (a, y) - (a, b) - (0, b) - (0, 0)$ as the cycle composed by the paths $(0, 0) - (x, 0)$, $(x, 0) - (x, y)$, etc. Note that the notation works because there is always one repeated coordinate between 2 consecutive nodes.

For a node $v = (x, y) \in Z_{NE}$ we design the cycle $(0, 0) - (x, 0) - (x, y) - (a, y) - (a, b) - (0, b) - (0, 0)$. For a node $v = (x, y) \in Z_{SE}$ we design the cycle $(0, 0) - (a, 0) - (a, y) - (x, y) - (x, b) - (0, b) - (0, 0)$. For a node $v = (x, y) \in Z_{SW}$ we design the cycle $(0, 0) - (a, 0) - (a, b) - (x, b) - (x, y) - (0, y) - (0, 0)$. All of these cycles have the same length than the cycle $(0, 0) - (a, 0) - (a, b) - (0, b) - (0, 0)$. The length is $2a + 2b$ which is multiple of $d+1$. Moreover, as $a \geq 3d$ and $b \geq 2d - 1$, all the cycles have width at least d .

Note that it is not possible to construct a cycle between g and any node in $Z_I = \{1, d-1\} \times \{1, d-1\}$ with width greater or equal than d .

Finally, note that the nodes in $\{0\} \times \{0, d-1\}$ and in $\{0, d-1\} \times \{0\}$ are covered by all the cycles presented above. Then, except Z_I , all the nodes are covered by a cycle of length multiple of $d+1$ and width at least d .

- **Case 2: $2a+2b$ is not multiple of $d+1$.** Let $a_1 = \max\{a' \mid a' < a, \quad 2a'+2b \text{ multiple of } d+1\}$. The nodes in $\{0, a_1\} \times \{0, b\} \setminus Z_I$ can be covered as presented in Case 1. We define the remaining nodes as $Z' = \{a_1 + 1, a\} \times \{0, b\}$.

First, we cover the nodes in Z' which are in the border of the grid G . These nodes are $Z'_{\text{ext}} = \{a_1 + 1, a\} \times \{0\} \cup \{a\} \times \{0, b\} \cup \{a_1 + 1, a\} \times \{b\}$. For these nodes we will present a cycle based on a prolongation of the cycle $(0, 0) \rightarrow (0, a) \rightarrow (a, b) \rightarrow (0, b) \rightarrow (0, 0)$. Second, we cover the remaining nodes of Z' , defined as $Z'_{\text{int}} = Z' \setminus Z'_{\text{ext}} = \{a_1 + 1, a-1\} \times \{1, b-1\}$. We will show that a slight modification over the cycle for Z'_{ext} works also for the nodes in Z'_{int} .

- **Covering Z'_{ext} .** Now we will present a prolongation of the cycle $C = (0, 0) \rightarrow (0, a) \rightarrow (a, b) \rightarrow (0, b) \rightarrow (0, 0)$ to cover the nodes in Z' which are in the border of the grid. Let us define $a_2 = \min\{a' \mid a' > a, \quad 2a'+2b \text{ multiple of } d+1\}$. To obtain a cycle of length multiple of $d+1$, we need to extend the cycle C with $2(a_2 - a)$ more edges, satisfying the constraint of width. Note that $a_2 - a_1 = d+1$ for d even, and $\frac{d+1}{2}$ for d odd. Then, $a_2 - a \leq d$ for d even, and $a_2 - a \leq \frac{d-1}{2}$ for d odd.

If $b \geq 2d$, we propose the cycle $C^1 = (0, 0) \rightarrow (a, 0) \rightarrow (a, b) \rightarrow (2d, b) \rightarrow (2d, b - (a_2 - a)) \rightarrow (d, b - (a_2 - a)) \rightarrow (d, b) \rightarrow (0, b) \rightarrow (0, 0)$. As $b \geq 2d$, then $b - (a_2 - a) \geq d$ satisfying the desired width. Moreover, the cycle has length multiple $d+1$.

If $b = 2d-1$ and $a_2 - a < d$, we will use the same cycle presented above. But, if $b = 2d-1$ and $a_2 - a = d$ we don't have enough space to use the same cycle. If this is the case, the above cycle does not satisfy the constraint of width, because $b - (a_2 - a) = d-1 < d$. Then, we propose the cycle $C^2 = (0, 0) \rightarrow (a, 0) \rightarrow (a, b) \rightarrow (4d, b) \rightarrow (4d, b-1) \rightarrow (3d, b-1) \rightarrow (3d, b) \rightarrow (2d, b) \rightarrow (2d, b - (d-1)) \rightarrow (d, b - (d-1)) \rightarrow (d, b) \rightarrow (0, b) \rightarrow (0, 0)$. This cycle satisfies the width constraint and the desired length also.

With this technique, the length of the cycles can be extended in $2(\lceil \frac{a-d}{2d} \rceil (b-d))$. Thus, in general, a and b must satisfy that

$$\left\lceil \frac{a-d}{2d} \right\rceil (b-d) \geq a_2 - a,$$

for $a_2 - a \leq d$ for d even, and $a_2 - a \leq \frac{d-1}{2}$ for d odd. Note that the case $a_2 - a = d$ occurs only when d is even and a can be written as $k(d+1) + 4$, for some $k \geq 1$. In this case, the inequality is satisfied for a and b such that either $b = 2d-1$ and $a \geq 5d$ or $b \geq 2d$ and $a \geq 3d$.

- **Covering Z'_{int} .** Now, we will check that a slight modification over the above cycles can cover all the nodes in Z'_{int} . First, we will split Z'_{int} in two parts: $Z'_{\text{int}N} = \{a_1 + 1, a - 1\} \times \{1, \lfloor \frac{b}{2} \rfloor\}$ and $Z'_{\text{int}S} = \{a_1 + 1, a - 1\} \times \{\lceil \frac{b}{2} \rceil, b\}$. To cover a node $v = (x, y)$ in $Z'_{\text{int}N}$, we will modify the cycle C^1 and C^2 as follows: Instead of using the paths $(0, 0) \rightarrow (a, 0) \rightarrow (a, b)$, we will use $(0, 0) \rightarrow (x, 0) \rightarrow (x, y) \rightarrow (a, y) \rightarrow (a, b)$. We will note the new cycles as C^1_{int} and C^2_{int} respectively. An example of this cycle is presented in figure 6.21. The length of the cycles have not changed. Now, we need to verify that the width is still greater or equal than d . To do that, for the cycle C^1_{int} , we need to check that $|(2d, b - (a_2 - a)) - (x, y)| \geq d$. And for C^2_{int} , we need to check that $|(4d, b - 1) - (x, y)| \geq d$. For C^1_{int} , as $a \geq 3d$, for any $(x, y) \in Z'_{\text{int}N}$,

$$\begin{aligned} \left| (2d, b - (a_2 - a)) - (x, y) \right| &\geq \left| (2d, b - (a_2 - a)) - (x, y) \right| \\ &\geq \left| (2d, b - (a_2 - a)) - (a_1 + 1, \lfloor \frac{b}{2} \rfloor) \right| \\ &\geq \left| (a - d, b - (a_2 - a)) - (a_1 + 1, \lfloor \frac{b}{2} \rfloor) \right| \end{aligned}$$

Now, as $b \geq 2d - 1$, we know that $b - (a_2 - a) \geq \lfloor \frac{b}{2} \rfloor$. Moreover $a_1 + 1 \geq a - d$, then

$$\begin{aligned} \left| (a - d, b - (a_2 - a)) - (a_1 + 1, \lfloor \frac{b}{2} \rfloor) \right| &= b - \lfloor \frac{b}{2} \rfloor + d + 1 - (a_2 - a_1) \\ &\geq b \end{aligned}$$

Now for C^2_{int} , we now that $a \geq 5d$, then $|(4d, b - 1) - (x, y)| \geq |(4d, b - 1) - (a_1 + 1, \lfloor \frac{b}{2} \rfloor)| \geq |(a - d, b - (a_2 - a)) - (a_1 + 1, \lfloor \frac{b}{2} \rfloor)|$. The result follows as seen before.

For the nodes in $Z'_{\text{int}S}$, we will use the reflections of the cycles on the horizontal axis.

After solved these 2 cases, we conclude that all the nodes in Z' are covered by a cycle of width at least d . ■

Lemma 6.14 *Given an instance of the RWP where the graph G and the gateway g is placed at the corner in the coordinates $(0, 0)$. Let v the only node in G with a positive demand $b > 0$. If there exists a cycle going through g and v of length multiple of $d + 1$ and width at least d , then there exists a solution for the RWP in $\frac{d+1}{2}b$.*

Proof: As the length of the cycle is a multiple of $d + 1$, it can be covered with $d + 1$ rounds alternately. Moreover, the width of the cycle being at least d assures that there is no interference between edges of a same round. A cost of $\frac{1}{2}b$ will be given to each round. It assures that all the demand can be sent to the gateway. Thus, the cost of the solution is $\frac{d+1}{2}b$. ■

Theorem 6.3 *Given an instance of the RWP, where the graph G is a 2-dimensional grid and the gateway g is placed at the corner in $(0,0)$. Each node $v \neq g$ has a demand $b(v)$ and the size of the grid G satisfies one of the conditions of lemma 6.13. Then, there exists a solution for the instance of RWP with cost*

$$\frac{d+1}{2} \sum_{v \notin \{1,d-1\} \times \{1,d-1\}} b(v) + \sum_{v \in \{1,d-1\} \times \{1,d-1\}} d(v,g)b(v)$$

Proof: The idea is to route the demands of each node independently. It means that, for each node, we will use different rounds. For each node in $v \in \{1,d-1\} \times \{1,d-1\}$ we use a route corresponding to the one of the shortest paths between v and g . Each edge in the chosen shortest path, will be one different round with cost $b(v)$. Then, the cost for routing the demand of v is $d(v,g)b(v)$. For the remaining nodes $v \notin \{1,d-1\} \times \{1,d-1\}$, lemma 6.13 guarantees that there exists a cycle of length multiple of $d+1$ and width at least d between g and v . We can route their demands independently using lemma 6.14. Then, we obtain a cost of $\frac{d+1}{2}b(v)$ per node. ■

6.5.4.1 An upper bound for uniform demand

We start with the following remark.

Remark 6.1 *In order to attain the lower bound given in 6.5.3, 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.*

We present a solution with takes into account these nodes in order to attain the lower bound in 6.5.3.

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 6.5.3 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 6.5.3.

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 6.5.3. We will route the demand by different methods depending on the position in the grid. In figure 6.22, 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 6.1, 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_{odd} as the function with value 1 when d is odd and 0 when d is even. In the same way, we define 1_{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 by $d(v, g)$ 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 6.23. 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 deplacated 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 6.27(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 6.27(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 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 6.26. 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 6.29. 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 lemma 6.13, 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 6.5.3, we conclude the result.

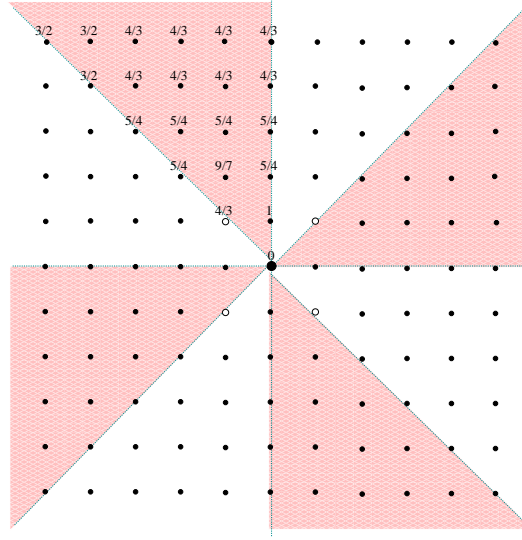


Figure 6.18: Cost to gather one unit of flow from each position to the center

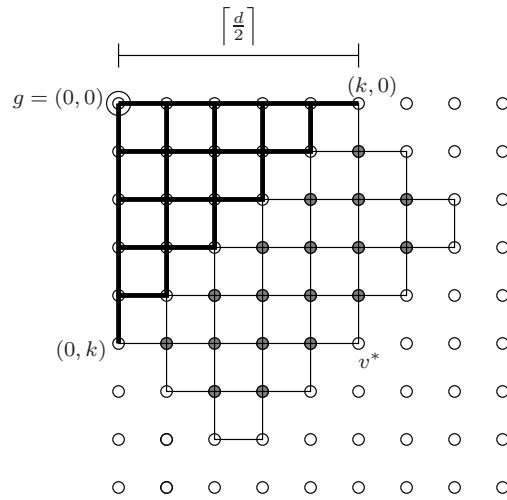


Figure 6.19: Call-clique K_{\max} for d odd with g at the corner. In this scheme, $d = 9$. The call-clique K_0 consists in all the wide edges.

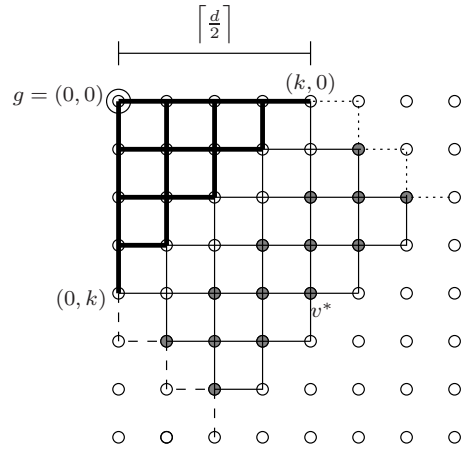


Figure 6.20: Two overlapped cliques for d even with g at the corner. In this scheme, $d = 8$.

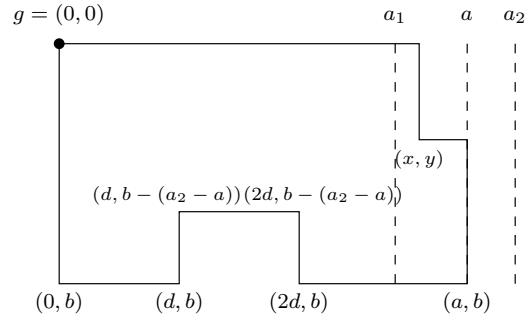


Figure 6.21: Example of the cycle C_{int}^1 .

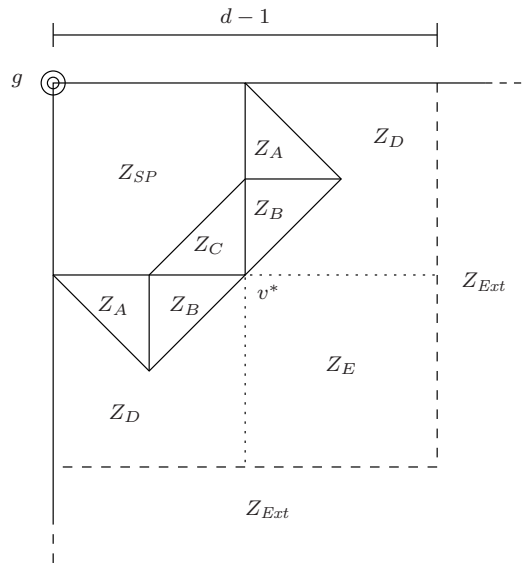


Figure 6.22: Scheme of the grid separated by method of routing.

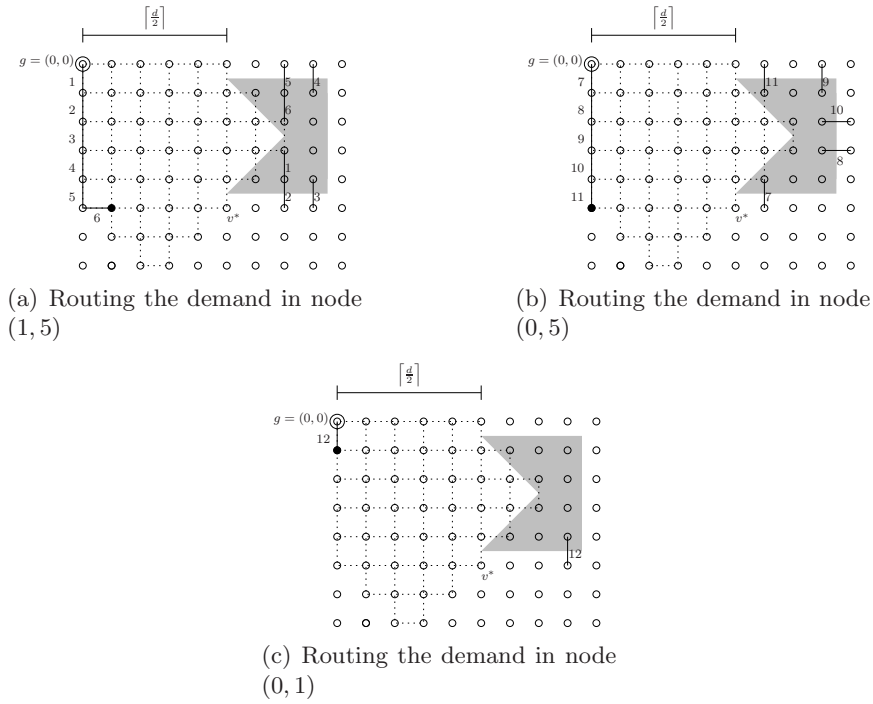


Figure 6.23: Example of moving the demand in Z_D using the routing of Z_B . In this example, $d = 9$.

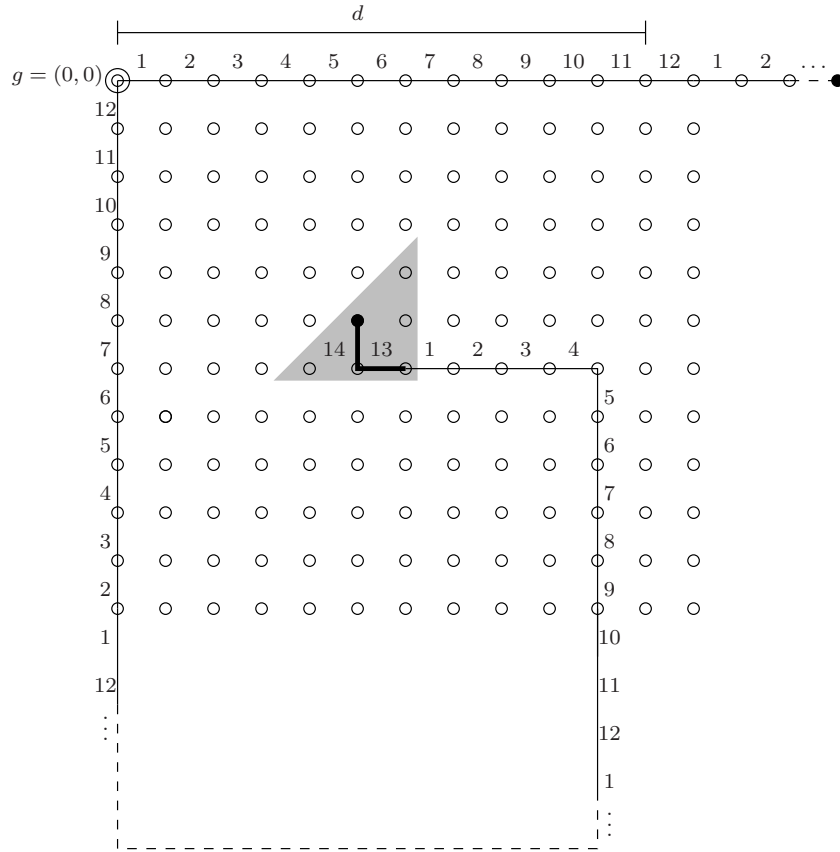


Figure 6.24: Example for Z_C with d odd. In this example, the demand.

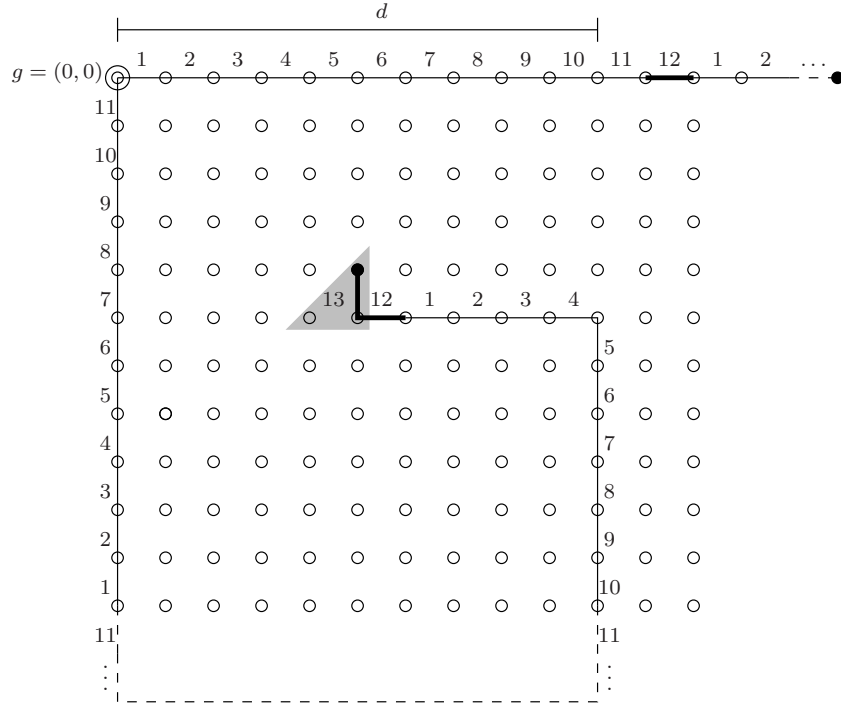


Figure 6.25: Example for Z_C with d even. In this example, the demand...

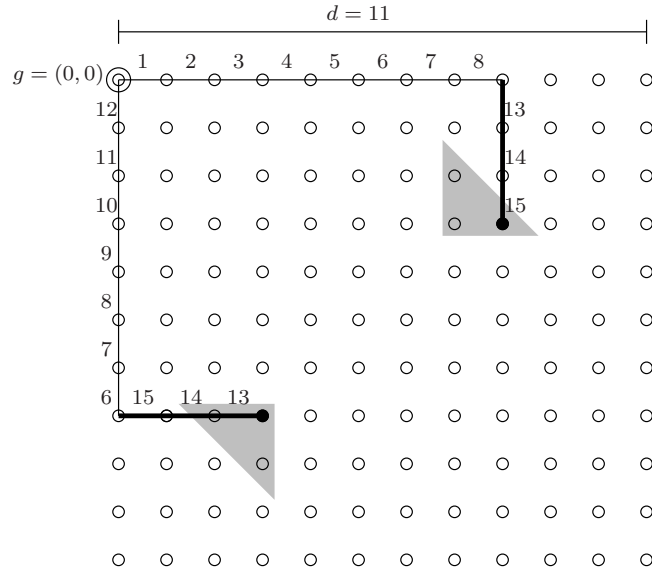


Figure 6.26: Example for Z_A with d odd. In this example, the demand. The even case is similar.

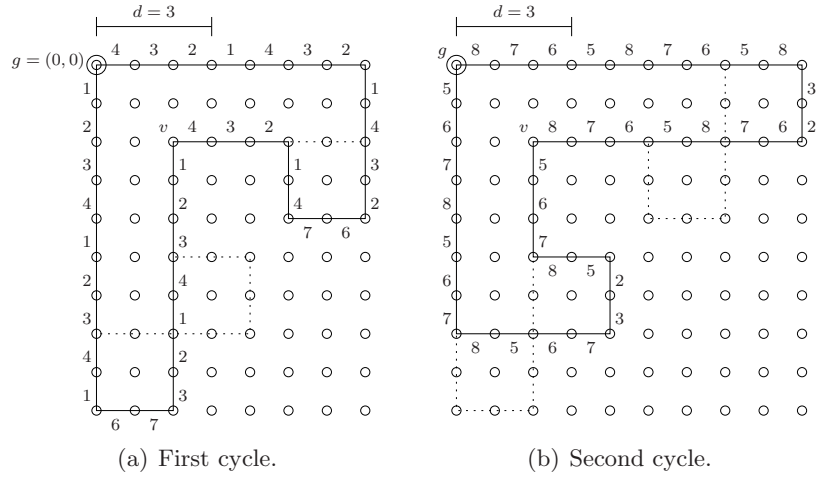


Figure 6.29: Example of routing with 2 cycles with rounds of weight $1/4$ for $v = (d-1, d-1)$. The weight needed to route the demand is $\frac{d+1}{2}b(v)$.

6.6 Conclusion and perspectives

In this work we have dealt with the ROUND WEIGHTING PROBLEM for gathering instances considering a symmetric variant of the binary interference model defined in [KMP08].

In the first part, we present methods to obtain lower bounds for general topologies using cliques of calls.

In the second part, we apply the lower bounds for the case of the grid. Moreover, we present solutions when the gateway is placed either at the center or at the corner. These solutions are optimal for uniform demands. For general demands, we have determined the zones of the grid where the demand is crucial for the cost of the solution.

We have shown that, in general, using a clique of calls around the gateway gives good lower bounds. However, in some cases these lower bounds do not attain the optimal solution. Indeed, for some cases of non-uniform demand, better lower bounds are obtained considering also cliques of calls which are not around the gateway.

Appendix A

Asymptotic Congestion in Wireless Ad-Hoc and Sensor Networks

A.1 Introduction

Wireless ad-hoc and sensor networks have gained much interest as inexpensive, energy-efficient, and miniaturized wireless devices are beginning to mature and take hold commercially. Wireless ad-hoc and sensor networks can be rapidly deployed as they do not require much existing infrastructure. Because of that, they are expected to find applications in many different settings, such as home appliance, disaster recovery, inventory tracking, battlefield surveillance, etc.

Congestion on this type of network is crucial as not only causes packet loss, and increases queueing delay, but also leads to unnecessary energy consumption, which causes lifetime reduction of the network. In wireless ad-hoc and sensor networks, extending the lifetime is important since all nodes contribute to collect the environment data and the early death of a node may lead to an incomplete monitoring. In a wireless ad-hoc and sensor network, two types of congestion can occur : node-level congestion, which is caused by buffer overflow in the node, or link-level congestion, when wireless channels are shared by several nodes and collisions occur when multiple active nodes try to seize the channel at the same time.

We will work on link-level congestion on randomly deployed static wireless ad-hoc and sensor networks. Wireless ad-hoc and sensor networks consist of nodes which share a common communication medium. On these networks, the signals intended for a receiver can cause interference at other receivers. The nodes on these networks cooperate in routing each other's data packets and communicate with each other over a wireless channel without any centralized control.

A wireless ad-hoc and sensor network can be seen as a graph G with a finite sets of nodes, and links connecting pairs of nodes (its ends). We consider the boolean model of connectivity, i.e., two nodes are connected if the distance between them is inferior to a certain threshold (called range of

connectivity), otherwise they are disconnected.

A path P in a graph G is a sequence $x_0, l_1, x_1, \dots, l_k, x_k$ where each x_i is a node, each l_i is an link, and the ends of link l_i are the nodes x_{i-1} and x_i . The length of the path P is k , i.e., the number of links on the path P . The network is connected if each node is connected by means of a path to every other node in the network.

On this setting, Gupta and Kumar [GK98] derived the critical power at which a node in the network needs to transmit in order to ensure that the network is connected with probability one as the number of nodes in the network goes to infinity.

The main theorem of that paper is the following:

Theorem A.1 (Gupta-Kumar, [GK98]) *If n nodes are randomly located, uniformly i.i.d., in a disc of unit area and each node transmits at a power level so as to cover an area of $\pi r^2(n) = \frac{\log n + \gamma_n}{n}$, then the resulting network is asymptotically connected with probability one as $n \rightarrow +\infty$ if and only if $\gamma_n \rightarrow +\infty$.*

On the rest of this chapter we will assume that the network is connected, i.e., the range of connectivity is greater than the threshold given by Gupta-Kumar's Theorem.

Our goal is to provide for different randomly deployed wireless ad-hoc and sensor network topologies, the congestion of the network. In this chapter we relate the notion of congestion to the number of paths of length k and the spectral radius of the generated graph. Using tools of random graph theory and random matrix theory we are able to determine the number of paths of length k there is on the network with k large enough and to relate this quantity to the congestion of the network.

A.2 Number of Paths of Length k and Congestion on the Network

In the following section, we analyze the relationship between the number of paths of length k in a wireless ad-hoc and sensor network and the link-level congestion over this network. In order to obtain this relationship we need to define some concepts of spectral graph theory (see [Chu97]) and analyze the asymptotic behavior of the number of paths of length k .

The ADJACENCY MATRIX of a graph G , denoted A , is the matrix with rows and columns labelled by graph vertices, defined as

$$A_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are connected by an link,} \\ 0 & \text{otherwise.} \end{cases}$$

The SPECTRAL RADIUS of a graph G , denoted λ_A , is the size of the largest eigenvalue of the

adjacency matrix of the graph that can be written as

$$\lambda_A = \sup_{x \in \mathbb{R}^d \setminus \{0\}} \frac{\langle x, Ax \rangle}{\langle x, x \rangle}.$$

Let A denote the adjacency matrix of a graph G . Then $(A^k)_{ij}$ is the number of paths of length k connecting the i -th and j -th vertices (proof by induction).

If we denote $\vec{1}$ the vector with all its components equal to 1s, then

$$\vec{1}^* A^k \vec{1} = \sum_{i,j} A_{ij}^k$$

is equal to the number of paths of length k on the graph G .

The adjacency matrix is symmetric, then by spectral decomposition we have $A = \sum \lambda_i v_i$ where v_i is the eigenvector of A associated with the eigenvalue λ_i . Remember that as A is a symmetric matrix, then the eigenvectors of A associated with distinct eigenvalues are orthogonal.

The next theorem shows the importance of the spectral radius of a graph G .

Theorem A.2 (Perron-Frobenius, [HJ90]) *Let A be an irreducible matrix with non-negative entries and spectral radius λ_A . Then*

1. $\lambda_A > 0$,
2. λ_A is an eigenvalue of A ,
3. There is a unique eigenvector v_A (up to a scale factor) with non-negative entries such that

$$Av_A = \lambda_A v_A,$$

4. λ_A is an algebraically simple eigenvalue of A .

From this theorem, the following result holds:

$$\begin{aligned} (\vec{1}^* A^k \vec{1})^{1/k} &= \left(\sum a_i v_i^* A^k \sum a_j v_j \right)^{1/k} \\ &= \left(\sum a_i v_i^* \sum a_j A^k v_j \right)^{1/k} \\ &= \left(\sum a_i v_i^* \sum a_j \lambda_j^k v_j \right)^{1/k} \\ &= \left(\sum |a_i|^2 \lambda_i^k \right)^{1/k}. \end{aligned}$$

This implies

$$\lim_{k \rightarrow +\infty} (\vec{1}^* A^k \vec{1})^{1/k} = \lambda_A.$$

From this result we obtain that the number of paths of length k in G is approximately λ_A^k , for k large enough.

Definition A.1 (Congestion Number) *Given a graph G we define the CONGESTION NUMBER as the inverse of the spectral radius of the graph λ_A^{-1} .*

The intuitive explanation to this definition is that while more paths of a fixed length we have in order to send information, we can split the information on these paths and coordinate it to arrive with the same number of hops at the receiver. This has the advantage of equalizing source-destination delays of packets that belong to the same class, which allows one to minimize the amount of packets that come out of sequence. This is desirable since in data transfers, out of order packets are misinterpreted to be lost which results not only in retransmissions but also in drop of systems throughput.

The following proposition give us another relationship between the spectral radius and on this case the degree of the nodes. The degree of a node in a graph is the number of links that connects to the node.

Proposition A.1 ([Lov07]) *Let d_{\min} denote the minimum degree of G , let \bar{d} be the average degree, and let d_{\max} be the maximum degree of G . For every graph G ,*

$$\max\{\bar{d}, \sqrt{d_{\max}}\} \leq \lambda_A \leq d_{\max}.$$

A.2.1 Discussion

A fundamental question about any network is whether or not it is κ -connected, i.e., for each pair of different nodes there exists at least κ link-disjoint paths in the graph connecting them of a fixed length that allow them to split their information and to send it through different paths. Additional requirements can be imposed, for instance the links can have small congestion.

We are conscious that the measure of congestion considered on this work has the limitation that the number of paths of fixed length are not necessarily link disjoint which would be an reasonable additional requirement. In that sense, a better measure would consider the possibility of splitting the information on independent paths without collision. However, for tractability reasons we consider this measure which is a good approximation and we can obtain explicit results.

A.3 Analysis Tools

In randomly deployed wireless ad-hoc and sensor networks the placement of the nodes and the links, which depend on the range of connectivity, are random. In order to derive the relation between congestion and spectral radius of a graph and to determine the spectral radius for different graphs,

we use tools from random graph theory and random matrix theory. In the asymptotic case, it enables us to have a tractable expression of the number of paths of a fixed length. Similar tools have been used on [GK98] and [CL06] to analyze wireless ad-hoc and sensor networks.

A.3.1 Random Graphs

In this section we introduce some basic notions of random graphs (see [Pen03]).

Given n nodes, x_1, \dots, x_n , in \mathbb{R}^d with $d = 2$ or 3 , we denote by $G(n, r(n))$ the graph with set of nodes $\{x_1, \dots, x_n\}$ and with links connecting all those nodes x_i, x_j , that satisfy $\|x_i - x_j\| < r(n)$ where $r(n)$ is the range of transmission and $\|\cdot\|$ is some norm in \mathbb{R}^d . We shall call $G(n, r(n))$ a GEOMETRIC GRAPH.

When the nodes are independent and identically distributed on \mathcal{D} with a specific probability density function, the geometric graph $G(n, r(n))$ is called a GEOMETRIC RANDOM GRAPH.

In the following, the domain on which nodes are deployed is the d -dimensional cube $\mathcal{D} = [-1/2, 1/2]^d$ where $d = 2$ or 3 .

On this domain each node is deployed with uniform distribution, i.e.,

$$f_U(x) := \begin{cases} 1 & \text{if } x \in [-1/2, 1/2]^d, \\ 0 & \text{otherwise.} \end{cases}$$

The most familiar random graph model, initiated by P. Erdős and A. Rényi [Erd59, ER59], consists of a graph with set of nodes $\{x_1, \dots, x_n\}$, obtained by including some of the links of the complete graph, each link being included independently with probability p . The graph derived by the latter scheme is called a Bernoulli random graph and is denoted $G(n, p)$.

Bernoulli random graphs (also called Erdős-Rényi random graphs) have been intensively studied and many of their properties are by now well understood; see Bollobás [Bol01] as a reference.

Bernoulli random graphs have the property of independence between the connectivity of different links, while for Geometric random graphs, if node x_i is close to node x_j , and node x_j is close to node x_k , then x_i will be fairly close to x_k . In wireless ad-hoc and sensor networks, this property is more realistic than the independence of links as in the Bernoulli random graphs.

Examples of Bernoulli random graphs $G(n, p)$ for different p 's and of Geometric random graphs $G(n, r)$ for different r 's can be found in figures A.1 and A.2, respectively.

From the figures A.3(a) and A.3(b) done by simulation, we see that the convergence of the k -th root of the numbers of paths of length k converges very fast to the spectral radius with respect to k on these two settings.

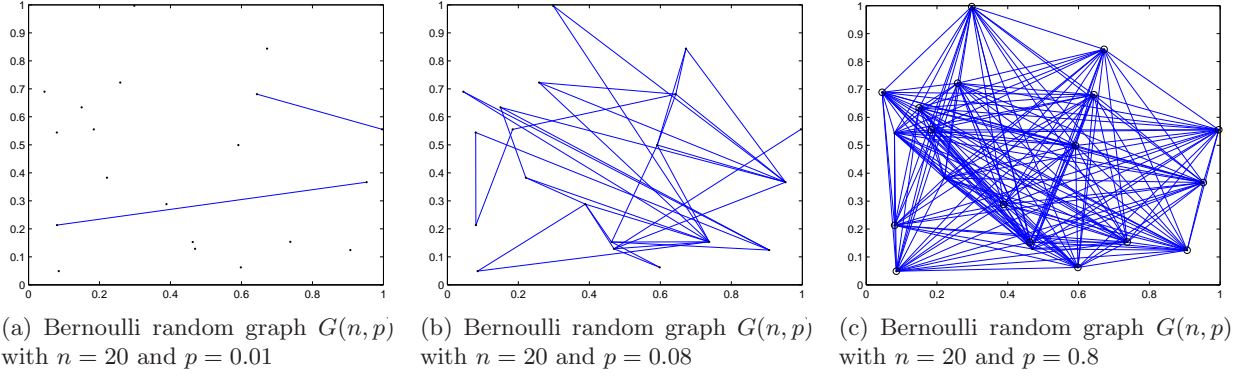


Figure A.1: Bernoulli random graphs

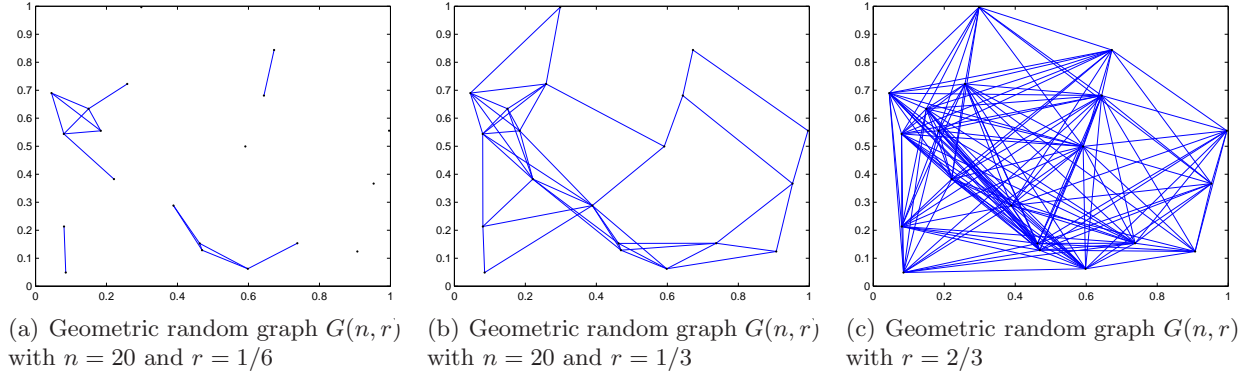


Figure A.2: Geometric random graphs

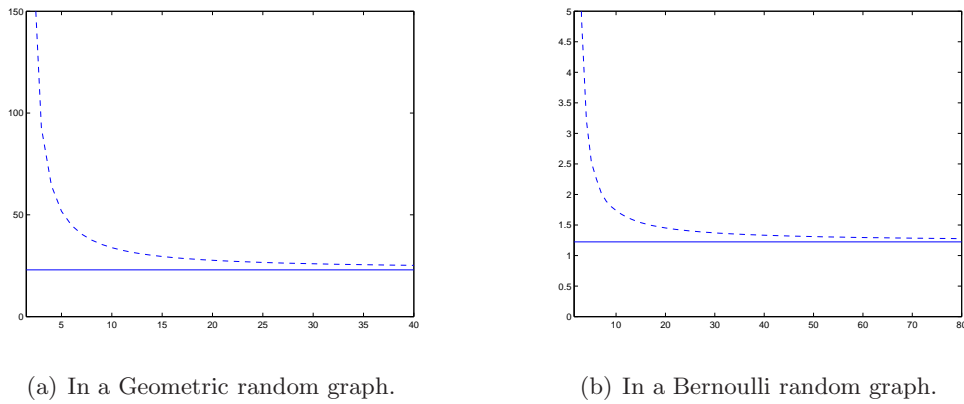


Figure A.3: Convergence of the k -th root of the number of paths of length k (depicted as a dashed curve) to the spectral radius of the graph λ_A (solid curve) with respect to k .

A.3.2 Random Matrix Theory

The main application of random matrix theory lies on the derivation of asymptotic results for large random matrices. In many practical cases, the eigenvalue distribution of large random hermitian matrices converges to a definite probability distribution, called empirical distribution or density of states. In particular, we can also find the value or bounds of the largest or smallest eigenvalues of large random hermitian matrices.

In this work we will use random matrix theory to derive the spectral radius of a Geometric random graph.

Definition A.2 *An EUCLIDEAN RANDOM MATRIX is an $n \times n$ matrix, A , whose entries are a function of the positions of n random points in a compact set \mathcal{D} of \mathbb{R}^d .*

More precisely, if n nodes, x_1, \dots, x_n , are located randomly, uniformly i.i.d., in a square of unit area \mathcal{D} and the matrix A is defined as

$$A := (F(x_i - x_j))_{1 \leq i \leq j \leq n}$$

where F is a measurable mapping from \mathbb{R}^d to \mathbb{C} . Then A is an Euclidean random matrix.

We consider the boolean model of connectivity, i.e., two nodes are connected if the distance between them is inferior to a certain threshold and otherwise they are disconnected. Therefore, if n nodes are located randomly, uniformly i.i.d., in a square of unit area and each node transmits at a power in order to cover an area of $r(n)$, then the adjacency matrix of this random graph is given by

$$A_{ij} = 1_{\{\|x_i - x_j\| \leq r(n)\}}$$

where

$$1_{\{\|x_i - x_j\| \leq r(n)\}} = \begin{cases} 1 & \text{if } \|x_i - x_j\| \leq r(n) \\ 0 & \text{otherwise.} \end{cases}$$

We would like to determine for this adjacency matrix the maximum eigenvalue or spectral radius and relate it to the congestion on the network.

In order to determine the congestion number, we explicit recent results of Bordenave on Geometric random graphs. Following the paper of Bordenave [Bor08], we assume that the discrete Fourier transform of F is defined for all $k \in \mathbb{Z}^d$ where

$$\hat{F}(k) = \int_{\mathcal{D}} F(x) e^{-2\pi i k \cdot x} dx$$

We assume that almost everywhere (a.e.) and at 0, the Fourier series of F exists and

$$F(x) = \sum_{k \in \mathbb{Z}^d} \hat{F}(k) e^{2\pi i k \cdot x}$$

A sufficient condition for the existence of the Fourier series of F (a.e.) is that

$$\sum_{k \in \mathbb{Z}^d} |\hat{F}(k)| < +\infty$$

and F to be continuous at zero.

Let's define $A_n = A/n$ and

$$\mu_n = \sum_{i=1}^n \delta_{\lambda_i(n)/n}$$

where $\{\lambda_i(n)\}_{1 \leq i \leq n}$ is the set of eigenvalues of A and δ is the Dirac function. Notice that $\{\lambda_i(n)/n\}_{1 \leq i \leq n}$ is the set of eigenvalues of A_n .

Let's define the measure

$$\mu = \sum_{k \in \mathbb{Z}} \delta_{\hat{F}(k)}$$

The following theorem gives us the convergence of the empirical distribution or density of states to a non-random distribution characterized by the Fourier transform of the function F .

Theorem A.3 (Bordenave, [Bor08]) *In the previous setting*

$$\lim_{n \rightarrow +\infty} \mu_n(K) = \mu(K) \quad a.e.$$

for all Borel sets K with $\mu(\partial K) = 0$ and $0 \notin \bar{K}$.

The following corollary gives us a formula to compute the spectral radius of a graph.

Corollary A.1 ([Bor08]) *The convergence of the spectral radius of A_n , almost surely, is given by*

$$\lim_{n \rightarrow +\infty} \max_{1 \leq i \leq n} \frac{|\lambda_i(n)|}{n} = \max_{k \in \mathbb{Z}^d} |\hat{F}(k)|.$$

Having this corollary in mind we can compute for different norms the spectral radius of a graph.

The following norms will be considered:

$$\|x\|_\infty := \max\{|x_1|, \dots, |x_n|\} \text{ (Infinity norm)}, \quad (\text{A.1})$$

$$\|x\|_p := \left(\sum_{i=1}^n |x_i|^p \right)^{1/p} \text{ (p-norm)} \quad \forall p > 1, \quad (\text{A.2})$$

$$\|x\|_1 := \sum_{i=1}^n |x_i| \text{ (Manhattan norm)}. \quad (\text{A.3})$$

Note that with the infinity norm case we obtain a closed form expression given by

$$F(x) = 1_{\{\max_{1 \leq i \leq d} |x_i| \leq r\}}(x),$$

for which its discrete Fourier transform writes as

$$\hat{F}(k) = r^d \prod_{i=1}^d \frac{\sin(2\pi k_i r)}{2\pi k_i r}$$

where $k = (k_1, \dots, k_d) \in \mathbb{Z}$.

Then for the infinity norm the spectral radius is given by r^d .

The figures A.5(a) and A.5(b) give us the asymptotic convergence of the spectral radius of the adjacency matrix to the maximum of the Fourier transform over the d -dimensional integer lattice with respect to the number of nodes on the network.

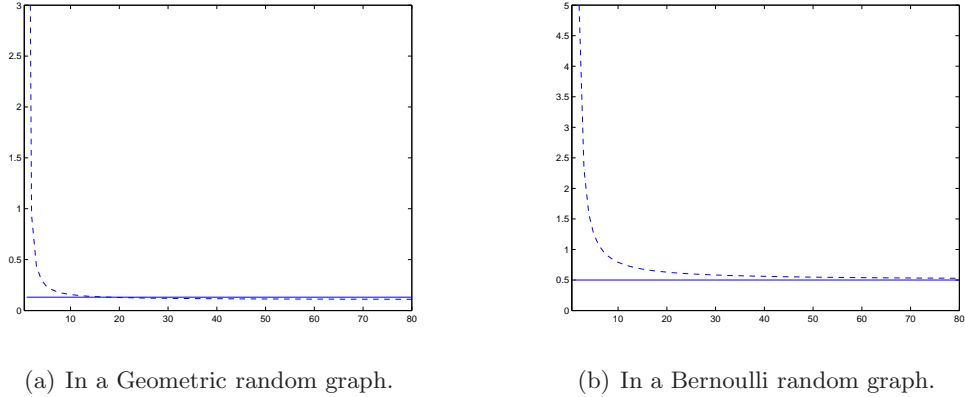


Figure A.4: Convergence of the k -th root of the number of paths of length k (depicted as a dashed curve) to the estimation of the spectral radius (solid curve) given by Bordenave's Theorem and Füredi-Komlós' Theorem, respectively, with respect to k .

There is a similar result on the spectral radius of a Bernoulli random graph A that we put for completeness.

Theorem A.4 (Füredi-Komlós, [FK81]) *Let $a_{ij}, i \geq j$, be independent (not necessarily identically distributed) random variables bounded with a common bound K . Assume that for $i > j$, the a_{ij} have a common expectation μ and variance σ^2 , further that $\mathbb{E}(a_{ii}) = \nu$. Define a_{ij} for $i < j$ by $a_{ij} = a_{ji}$ (the numbers K, μ, σ^2, ν will be kept fixed as n will tend to infinity).*

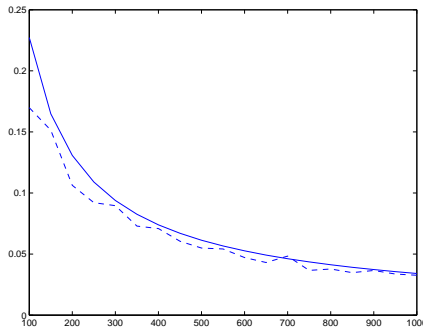
If $\mu > 0$ then the distribution of the largest eigenvalue of the random symmetric matrix $A = (a_{ij})$ can be approximated in order $1/\sqrt{n}$ by a normal distribution of expectation

$$(n-1)\mu + \nu + \sigma^2/\mu$$

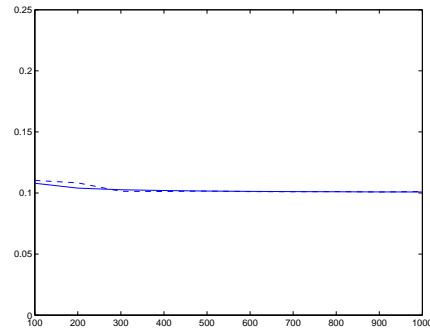
and variance $2\sigma^2$.

The result of this theorem stems from the analysis of the largest eigenvalue of non-zero mean random matrices with independent entries.

From this theorem in our case the constants are $K = 1, \mu = p, \nu = 0$ and $\sigma^2 = p(1-p)$ and then the expected spectral radius of a Bernoulli random graph is $(n-1)p + (1-p)$.



(a) In a Geometric random graph.



(b) In a Bernoulli random graph.

Figure A.5: Convergence of the largest eigenvalue (dashed curve) to the asymptotic approximation (solid curve) given by Bordenave's Theorem and Füredi-Komlós' Theorem, respectively.

A.4 Conclusions and Future Work

In this contribution, we have provided a model to deal with congestion of randomly deployed wireless nodes. For various cases of random graphs (Bernoulli random graphs and Geometric random graphs), we have provided, in the case of large networks, the congestion number which is linked to the number of connected paths of a given length. Quite remarkably, the mean congestion number can be explicitly derived using asymptotic results of random matrix theory and the results holds even for a not so large number of nodes. Further studies will focus on providing central limit theorems on the congestion number in order to have a better assessment of the quality of service

in the network. Other realistic models (beside the boolean model for connectivity) will also also studied in combination with other random distribution of the nodes.

Acknowledgments

The authors want to thank Dr. Charles Bordenave for helpful discussions.

Bibliography

- [BBK⁺04] H. Balakrishnan, C.L. Barret, V.S.A. Kumar, M.V. Marathe, and S. Thite. The distance-2 matching problem and its relationship to the MAC-layer capacity of ad-hoc wireless networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1069–1079, 2004.
- [BBS05] P Bertin, J.-F. Bresse, and B.L. Sage. Accès haut débit en zone rurale: une solution ad hoc. *France Telecom R&D*, 22:16–18, 2005.
- [BCY06] J.-C. Bermond, R. Corrêa, and M.-L. Yu. Gathering algorithms on paths under interference constraints. In *6th Conference on Algorithms and Complexity*, volume 3998 of *Lecture Notes in Computer Science*, pages 115–126, Roma, Italy, May 2006.
- [BCY09] J.-C. Bermond, R. Corrêa, and M.-L. Yu. Optimal gathering protocols on paths under interference constraints. *Discrete Mathematics*, 2009. To appear.
- [BGK⁺06a] 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)*, pages 85–88, Trégastel, France, May 2006.
- [BGK⁺06b] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Perennes. Hardness and approximation of gathering in static radio networks. *Parallel Processing Letters*, 16(2):165–183, June 2006.
- [BGK⁺06c] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Pérennes. Hardness and approximation of gathering in static radio networks. Research Report 5936, INRIA, June 2006.
- [BGK⁺06d] J.-C. Bermond, J. Galtier, R. Klasing, N. Morales, and S. Pérennes. Hardness and approximation of gathering in static radio networks. *Parallel Processing Letters*, 16(2):165–183, 2006.
- [BGN⁺09] Jean-Claude Bermond, Frédéric Guinand, Nicolas Nisse, Patricio Reyes, and Hervé Rivano. Minimum delay data gathering in closed half-duplex grid. On-going work and discussions, June 2009.

- [BGP98] J.-C. Bermond, L. Gargano, and S. Perennes. Optimal sequential gossiping by short messages. *Discrete Applied Mathematics*, 86:145–155, 1998.
- [BGR08] J.-C. Bermond, L. Gargano, and A.A. Rescigno. Gathering with minimum delay in tree sensor networks. In *SIROCCO 2008*, volume 5058 of *Lecture Notes in Computer Science*, pages 262–276, Villars-sur-Ollon, Switzerland, June 2008. Springer-Verlag.
- [BGRV98] J.-C. Bermond, L. Gargano, A.A. Rescigno, and U. Vaccaro. Fast gossiping by short messages. *SIAM Journal on Computing*, 27(4):917–941, 1998.
- [BHW00] C. Busch, M. Herlihy, and R. Wattenhofer. Hard-potato routing. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 278–285, Portland, Oregon, United States, 2000. ACM.
- [Bia00] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal of Selected Areas of Communication*, 18:535–547, 2000.
- [BKK⁺09a] V. Bonifaci, R. Klasing, P. Korteweg, L. Stougie, and A. Marchetti-Spaccamela. *Graphs and Algorithms in Communication Networks*, chapter Data Gathering in Wireless Networks. Springer Monograph Springer-Verlag. A. Koster and X. Munoz, editors, 2009.
- [BKK⁺09b] V. Bonifaci, R. Klasing, P. Korteweg, L. Stougie, and A. Marchetti-Spaccamela. *Graphs and Algorithms in Communication Networks*, chapter Data Gathering in Wireless Networks. Springer Monograph Springer-Verlag, 2009.
- [BKMS08a] V. Bonifaci, P. Korteweg, A. Marchetti-Spaccamela, and L. Stougie. An approximation algorithm for the wireless gathering problem. *Operations Research Letters*, 36(5):605 – 608, 2008.
- [BKMS08b] V. Bonifaci, P. Korteweg, A. Marchetti-Spaccamela, and L. Stougie. Minimizing average flow time in sensor data gathering. In *ALGOSENSORS*, volume 5389 of *Lecture Notes in Computer Science*, pages 18–29. Springer, 2008.
- [BNRR09a] J.-C. Bermond, N. Nisse, P. Reyes, and H. Rivano. Fast data gathering in radio grid networks. In *11èmes Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel’09)*, June 2009.
- [BNRR09b] J.-C. Bermond, N. Nisse, P. Reyes, and H. Rivano. Minimum delay data gathering in radio networks. In *Proceedings of the 8th international conference on Ad Hoc Networks and Wireless (AdHoc-Now)*, 2009. To appear.
- [Bol01] B. Bollobás. *Random Graphs*. Cambridge University Press, 2 edition, 2001.
- [Bor08] C. Bordenave. Eigenvalues of euclidean random matrices. *Random Struct. Algorithms*, 33(4):515–532, 2008.

- [BP05] 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)*, pages 103–106, Presqu'île de Giens, May 2005.
- [BP09] J.-C. Bermond and J. Peters. Optimal gathering in radio grids with interference. 2009. manuscript.
- [BY08] J.-C. Bermond and M.-L. Yu. Optimal gathering algorithms in multi-hop radio tree networks with interferences. In *ADHOC-NOW 2008*, volume 5198 of *Lecture Notes in Computer Science*, pages 204–217. Springer-Verlag, September 2008. Full version to appear in *AdHoc & Sensor Wireless Networks* 2009.
- [CGL02] M. Christersson, L. Gasieniec, and A. Lingas. Gossiping with bounded size messages in ad-hoc radio networks. In *Proceedings of 29th International Colloquium on Automata, Languages and Programming (ICALP'02)*, volume 2380 of *Lecture Notes in Computer Science*, pages 377–389. Springer-Verlag, 2002.
- [CGR02] M. Chrobak, L. Gasieniec, and W. Rytter. Fast broadcasting and gossiping in radio networks. *Journal of Algorithms*, 43(2):177–189, 2002.
- [Chu97] F. R. K Chung. *Spectral graph theory*, volume 92 of *CBMS Regional Conference Series in Mathematics*. Published for the Conference Board of the Mathematical Sciences, Washington, DC, 1997.
- [CL06] F. Chung and L. Lu. *Complex graphs and networks*, volume 107 of *CBMS Regional Conference Series in Mathematics*. Published for the Conference Board of the Mathematical Sciences, Washington, DC, 2006.
- [CLD05] L. Chen, S.H. Low, and J.C. Doyle. Joint congestion control and media access control design for wireless ad-hoc networks. In *IEEE INFOCOM*, 2005.
- [Col02] S. Coleri. PEDAMACS: power efficient and delay aware. Master's thesis, University of California, Berkeley, December 2002.
- [CW91] I. Chlamtac and O. Weinstein. The wave expansion approach to broadcasting in multihop radio networks. *IEEE Transaction on Communications*, 39(3):426–433, 1991.
- [DFTT04] O. Dousse, M. Franceschetti, D. Tse, and P. Thiran. Closing the gap in the capacity of random wireless networks. In *IEEE ISIT 2004*. IEEE Computer Society, 2004.
- [EK04] M.L. Elkin and G. Kortsarz. Logarithmic inapproximability of the radio broadcast problem. *Journal of Algorithms*, 52(1):8–25, 2004.
- [ER59] P Erdős and A Rényi. On random graphs, I. *Publicationes Mathematicae (Debrecen)*, 6:290, 297, 1959.

- [Erd59] P. Erdős. Graph theory and probability. *Canad. J. Math.*, 11:34–38, 1959.
- [FFM04a] C. Florens, M. Franceschetti, and R.J. McEliece. Lower bounds on data collection time in sensory networks. *IEEE Selected Areas in Communications*, 22(6):1110–1120, 2004.
- [FFM04b] C. Florens, M. Franceschetti, and R.J. McEliece. Lower bounds on data collection time in sensory networks. *Selected Areas in Communications, IEEE Journal on*, 22(6):1110–1120, 2004.
- [FK81] Z. Füredi and J. Komlós. The eigenvalues of random symmetric matrices. *Combinatorica*, 1(3):233–241, 1981.
- [Gal04] J. Galtier. Optimizing the IEEE 802.11b performance using slow congestion window decrease. In *Proc. 16th ITC Specialist Seminar on Performance Evaluation of Wireless and Mobile Systems*, pages 165–176, 2004.
- [Gar07] L. Gargano. Time optimal gathering in sensor networks. In Giuseppe Prencipe and Shmuel Zaks, editors, *Structural Information and Communication Complexity, 14th International Colloquium, SIROCCO 2007, Castiglione, Italy, June 5-8, 2007, Proceedings*, volume 4474 of *Lecture Notes in Computer Science*, pages 7–10. Springer, 2007.
- [GG93] A.G. Greenberg and J. Goodman. Sharp approximate models of deflection routing in mesh networks. *Communications, IEEE Transactions on*, 41(1):210–223, Jan 1993.
- [GH07] 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.
- [GK98] P. Gupta and P. R. Kumar. Critical power for asymptotic connectivity in wireless networks. *Stochastic Analysis, Control, Optimization and Applications: volume in honor of W.H. Fleming*, pages 547–566, 1998.
- [GK00] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, March 2000.
- [GM03] I. Gaber and Y. Mansour. Centralized broadcast in multihop radio networks. *Journal of Algorithms*, 46(1):1–20, 2003.
- [GP02] L. Gasieniec and I. Potapov. Gossiping with unit messages in known radio networks. In *Proceedings of the IFIP 17th World Computer Congress - TC1 Stream / 2nd IFIP International Conference on Theoretical Computer Science*, pages 193–205. Kluwer, B.V., 2002.

- [GPR08] C. Gomes, S. Pérennes, and H. Rivano. Bottleneck analysis for routing and call scheduling in multi-hop wireless networks. In *4th IEEE Workshop on Broadband Wireless Access (BWA)*, New-Orleans, US, 4th December 2008.
- [GPRR08] C. Gomes, S. Pérennes, P. Reyes, and H. Rivano. Bandwidth allocation in radio grid networks. In *10èmes Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel'08)*, May 2008.
- [GR06a] L. Gargano and A. A. Rescigno. Optimally fast data gathering in sensor networks. In Rastislav Kralovic and Pawel Urzyczyn, editors, *MFCS*, volume 4162 of *Lecture Notes in Computer Science*, pages 399–411. Springer, 2006.
- [GR06b] L. Gargano and A. A. Rescigno. Optimally fast data gathering in sensor networks. In Rastislav Kralovic and Pawel Urzyczyn, editors, *MFCS*, volume 4162 of *Lecture Notes in Computer Science*, pages 399–411. Springer, 2006.
- [GR09] L. Gargano and A. A. Rescigno. Collision-free path coloring with application to minimum-delay gathering in sensor networks. *Discrete Applied Mathematics*, 157(8):1858 – 1872, 2009.
- [HJ90] R. Horn and C. Johnson. *Matrix analysis*. Cambridge University Press, Cambridge, 1990. Corrected reprint of the 1985 original.
- [HKP⁺05] J. Hromkovic, R. Klasing, A. Pelc, P. Ruzicka, and W. Unger. *Dissemination of Information in Communication Networks: Part I. Broadcasting, Gossiping, Leader Election, and Fault-Tolerance*. Springer Monograph. Springer-Verlag, 2005.
- [JPPQ03] K. Jain, J. Padhye, V.N. Padmanabhan, and L. Qiu. Impact of interference on multi-hop wireless network performance. In *MobiCom'03*, pages 66–80, 2003.
- [KLNP05] R. Klasing, Z. Lotker, A. Navarra, and S. Pérennes. From balls and bins to points and vertices. In *Proceedings of the 16th Annual International Symposium on Algorithms and Computation (ISAAC 2005)*, volume 3827 of *Lecture Notes in Computer Science*, pages 757–766. Springer Verlag, December 2005.
- [KMP08] R. Klasing, N. Morales, and S. Pérennes. On the complexity of bandwidth allocation in radio networks. *Theoretical Computer Science*, 406(3):225–239, October 2008.
- [KMPS04] V. S. Anil Kumar, Madhav V. Marathe, Srinivasan Parthasarathy, and Aravind 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*, pages 1021–1030, Philadelphia, PA, USA, 2004. Society for Industrial and Applied Mathematics.

- [Kor08] P. Korteweg. *Online gathering algorithms for wireless networks*. PhD thesis, Technische Universiteit Eindhoven, 2008.
- [Lov07] L. Lovasz. Eigenvalues of graphs. Technical report, November 2007.
- [Mah00] M. Mahdian. The strong chromatic index of graphs. M.Sc. Thesis, Department of Computer Science, University of Toronto, ON, Canada, 2000.
- [MPR06] G. Méheut, S. Pérennes, and H. Rivano. Evaluation stochastique et simulation des réseaux radio. Research report 5989, INRIA, September 2006.
- [MPS95] Y. Mansour and B. Patt-Shamir. Many-to-one packet routing on grids. In *Proceedings of the twenty-seventh annual ACM symposium on Theory of computing*, pages 258–267, Las Vegas, Nevada, United States, 1995. ACM.
- [MSS06] R. Mazumar, G. Sharma, and N. Shroff. Maximum weighted matching with interference constraints. *FAWN, Pisa, Italy*, 2006.
- [Müh02] P. Mühlethaler. *802.11 et les réseaux sans fil*. Eyrolles, 2002.
- [Pen03] M. Penrose. *Random geometric graphs*, volume 5 of *Oxford Studies in Probability*. Oxford University Press, Oxford, 2003.
- [Rap96] T. Rappaport. *Wireless Communications: Principles and Practices*. Prentice Hall, 1996.
- [RS07] Y. Revah and M. Segal. Improved lower bounds for data-gathering time in sensor networks. In *Networking and Services, 2007. ICNS. Third International Conference on*, page 76, 2007.
- [RS08] Y. Revah and M. Segal. Improved bounds for data-gathering time in sensor networks. *Computer Communications*, 31(17):4026–4034, November 2008.
- [RTV09] H. Rivano, F. Theoleyre, and F. Valois. A framework for the capacity evaluation of multihop wireless networks. *Ad Hoc and Sensor Wireless networks (AHSWN)*, 2009. To appear.
- [SV82] L. Stockmeyer and V. Vazirani. NP-completeness of some generalizations of the maximum matching problem. *Information Processing Letters*, 15(1):14–19, 1982.
- [SW06] S. Schmid and R. Wattenhofer. Algorithmic models for sensor networks. In *14th International Workshop on Parallel and Distributed Real-Time Systems (WPDRTS), Island of Rhodes, Greece*, April 2006.
- [Wan09a] 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*, pages 85–94, New York, NY, USA, 2009. ACM.

- [Wan09b] P.-J. Wan. Multiflows in multihop wireless networks. In *ACM MobiHoc*, 2009.
- [WWLS08] 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.*, 19(12):1709–1726, 2008.

Collecte d'Information dans les Réseaux Radio

Résumé: Cette thèse concerne l'étude de l'algorithmique et de la complexité des communications dans les réseaux radio. En particulier, nous nous sommes intéressés au problème de rassembler les informations des sommets d'un réseau radio en un noeud central. Ce problème est motivé par une question de France Telecom (Orange Labs) "comment amener Internet dans les villages". Les sommets représentent les maisons des villages qui communiquent entre elles par radio, le but étant d'atteindre une passerelle connectée à Internet par une liaison satellite. Le même problème se rencontre dans les réseaux de senseurs où il s'agit de collecter les informations des senseurs dans une station de base. Une particularité des réseaux radio est que la distance de transmission est limitée et que les transmissions interfèrent entre elles (phénomènes d'interférences). Nous modélisons ces contraintes en disant que deux sommets (équipements radio) peuvent communiquer s'ils sont à distance au plus d_T et qu'un noeud interfère avec un autre si leur distance est au plus d_I . Les distances sont considérées dans un graphe représentant le réseau. Une étape de communication consistera donc en un ensemble de transmissions compatibles (n'interférant pas). Notre objectif est de trouver le nombre minimum d'étapes nécessaires pour réaliser un tel rassemblement et de concevoir des algorithmes réalisant ce minimum. Pour des topologies particulières comme le chemin et la grille, nous avons établi des résultats optimaux ou quasi optimaux. Nous avons aussi considéré le cas systolique (ou continu) où on veut maximiser le débit offert à chaque noeud.

Data Gathering in Radio Networks

Abstract: This thesis concerns the study of the algorithmic and the complexity of the communications in radio networks. In particular, we were interested in the problem of gathering information from the nodes of a radio network in a central node. This problem is motivated by a question of France Telecom (Orange Labs) "How to bring Internet in villages". Nodes represent the houses of the villages which communicate between them by radio, the goal being to reach a gateway connected to Internet by a satellite link. The same problem can be found in sensor networks where the question is to collect data from sensors to a base station. A peculiarity of radio networks is that the transmission distance is limited and that the transmissions interfere between them (interference phenomena). We model these constraints by saying that two nodes (radio devices) can communicate if they are at distance at most d_T and a node interferes with another one if their distance is at most d_I . The distances are considered in a graph representing the network. Thus, a communication step will consist in a compatible (non interfering) set of transmissions. Our goal is to find the minimum number of steps needed to achieve such a gathering and design algorithms achieving this minimum. For special topologies such as the path and the grid, we have proposed optimal or near optimal solutions. We also considered the systolic (or continuous) case where we want to maximize the throughput (bandwidth) offered to each node.