WARSAW UNIVERSITY OF TECHNOLOGY

DISCIPLINE OF SCIENCE INFORMATION AND COMMUNICATION TECHNOLOGY
FIELD OF SCIENCE ENGINEERING AND TECHNOLOGY

Ph.D. Thesis

Bartłomiej Ostrowski, M.Sc.

A Comprehensive Optimization Model for Multi-hop Wireless Networks with Multicast Traffic

Supervisor
Prof. Michał Pióro, Ph.D., D.Sc.

WARSAW 2022

Abstract

A Comprehensive Optimization Model for Multi-hop Wireless Networks with Multicast Traffic

In this thesis we study multi-hop wireless networks (MHWN) assuming time division multiple access (TDMA) to radio channel and multicast traffic. In the recent years, MHWN have become an important research object, mainly due to their great importance, especially in the Internet of Things context.

The goal of this thesis is to provide a comprehensive optimization model for MHWN that utilize TDMA and serve multicast traffic. The presented optimization model, understood here as the set of notions, notations, problem formulations, and solution algorithms, includes such important aspects as traffic throughput maximization (with routing, modulation and coding schemes assignment, and transmission power control optimization), energy consumption minimization, as well as packet delay minimization. The considerations related to each aspect are illustrated with numerical results.

The optimization model described in this thesis provides a consistent approach to MHWN optimization and allows for treating various problems related to MHWN in a rigid mathematical way. The main application of this model is to provide performance evaluation of MHWN under TDMA access to radio channel and multicast traffic assumptions.

Keywords: multi-hop wireless networks, TDMA, multicast traffic, optimization, integer programming, Internet of Things

Streszczenie

Całościowy model optymalizacji kratowych sieci radiowych z ruchem multicastowym

Przedmiotem niniejszej rozprawy są kratowe sieci radiowe wykorzystujące wielodostęp czasowy TDMA oraz obsługujące ruch multicastowy. W ostatnich latach sieci te stały się obiektem licznych badań naukowych, głównie z powodu ich rosnącego znaczenia, w szczególności w kontekście Internetu Rzeczy.

Celem niniejszej rozprawy jest przedstawienie całościowego modelu optymalizacji kratowych sieci radiowych z wielodostępem TDMA obsługujących ruch multicastowy. Przedstawiony model optymalizacji (rozumiany tutaj jako zbiór pojęć, notacji, sformułowań problemów oraz algorytmów służących do ich rozwiązywania) obejmuje następujące zagadnienia: maksymalizację przepustowości sieci (z uwzględnieniem optymalizacji routingu, przydziału modulacji i schematów kodowania oraz sterowania mocą), minimalizację zużycia energii oraz minimalizację opóźnienia pakietów. Rozważania dotyczącego każdego z zagadnień zilustrowane są wynikami numerycznymi.

Przedstawiony model optymalizacji pozwala rozwiązywać rozmaite problemy związane z rozważanymi sieciami w formalny, bazujący na modelowaniu matematycznym sposób. Głównym zastosowaniem opisanego modelu optymalizacji jest wiarygodne szacowanie wskaźników wydajnościowych rozważanych sieci.

Słowa kluczowe: kratowe sieci radiowe, TDMA, ruch multicastowy, optymalizacja, programowanie całkowitoliczbowe, Internet Rzeczy

Acknowledgement

I would like to express my gratitude to my supervisor Michał Pióro for his constant advice, patience, and support while writing this thesis and during my Ph.D. studies.

I am also grateful to Mariusz Mycek, Artur Tomaszewski, and Mateusz Żotkiewicz for their time and help they provided.

The presented work was supported by the National Science Centre, Poland, under grant no. 2017/25/B/ST7/02313: "Packet routing and transmission scheduling optimization in multi-hop wireless networks with multicast traffic".

Contents

Chapter 1

Introduction

This introductory chapter aims at providing a background needed to understand the main assumptions and ideas underlying this thesis. First, we make a brief overview of the area of the thesis applications. Then, the aim of the thesis is stated. Finally, we explain our motivation and provide a brief overview of the thesis layout. The content of this chapter is partially based on [1].

1.1 Application area

Multicast transmission has existed for many years now bringing substantial benefits for fixed as well as wireless networks. Initially envisioned for media streaming applications, over time it has been applied in multiple new scenarios including for example data dissemination in wireless sensor networks. Moreover, with the advent of 5G networking, multicast transmission is undergoing a resurgence as it is considered to constitute the core solution for 5G networks in a number of verticals sectors such as media and entertainment, automotive, Internet of Things (machine-type communications) as well as Public Warning & Safety (mission critical communications) [2].

In many applications fixed networks highly benefit from multicast transmission, and even higher gain can be obtained in wireless networks, which is due to the broadcast nature of wireless medium. In fixed networks multicast transmission requires that a copy of a given packet is sent to each recipient, while in wireless networks multiple recipients can receive the same packet via the same broadcast transmission, as long as they are within the range of the transmitter. This property may, and should be, further developed to take multicast transmission into account in determining routing in wireless networks since this can lead to significant gains in traffic throughput as well as in energy consumption. Basically, this idea constitutes the keynote of this thesis.

In the following we focus on the specific type of wireless networks, namely we consider multi-hop wireless networks (MHWN). Although the considerations of this thesis apply to MHWN in general, we can distinguish two network types that are particularly interesting in our considerations, i.e., Wireless Mesh Networks (WMN) and Wireless Sensor Networks (WSN). Both types have important applications and in both multicast transmission could be utilized to achieve a substantial performance gain, particularly in the context of Internet of Things (IoT) whose dynamic and rapid growth is observed in the recent years.

The idea behind IoT is to create a system of interconnected physical devices that collect and share data to provide ubiquitous connectivity among machines and physical devices. With such a system, such concepts as smart grid, smart home, smart water networks, and intelligent transportation can be realized, making our everyday lives more comfortable [3]. The importance of such a shift from a human type communication (HTC) to a machine type communication (MTC) has been proved by making massive machine type communication (mMTC) one of the most important use cases in the 5th generation (5G) of mobile networks [4]. Although IoT is not coupled with any specific networking technology, it is commonly believed that it is WSN that will be the main building blocks of modern IoT solutions [5], [6]. However, not only WSN are relevant when it comes to IoT, as some of the recent papers indicate the importance of using WMN in IoT [7], as well as interconnecting of WMN with WSN in IoT context [8]. This leads us to the conclusion that both WSN and WMN will play a significant role in modern networking.

Roughly speaking, wireless mesh networks are the networks composed of two types of nodes: mesh client nodes and mesh router nodes. Mesh routers are fixed and constitute the backbone of WMN while mesh clients can be either fixed or mobile. One of the most important features of such networks is that each node supports mesh networking which means the ability to forward packets to neighboring nodes. As far as the radio technologies are concerned, WMN can be realized by several current technologies such as Wi-Fi IEEE 802.11, Bluetooth IEEE 802.15, WiMAX 802.16, and IEEE 802.20. Besides multi-hop packet transfer, the other important features of WMN include the ability to self-organization and self-configuration which leads to important benefits such as robustness, reliability, as well as easy deployment, and maintenance. Because of these advantages WMN play an important role in numerous applications, out of which one of the most important is providing common and inexpensive broadband access to the Internet, especially in community and neighborhood networks. Finally, as already mentioned, WMN may play an important role when it comes to IoT, whose popularity is rapidly increasing. For more detailed surveys of WMN the reader is referred to [9], [10].

Wireless sensor networks are composed of a large number of sensor nodes distributed throughout an area that are capable of measuring environmental parameters. Such nodes communicate with each other by means of wireless links to exchange measured data. The main features of these networks are multi-hop and broadcast communication, as well as limited resources when it comes to computing capabilities, memory, and power. An example radio technology used in these networks is 802.15.4. Because of many important advantages including ease of deployment and flexibility, such networks have found applications in multiple domains such as the military, manufacturing, home automation, health care, and agriculture. As WMN, the most important potential field of WSN application is IoT. For comprehensive studies on WSN the reader is referred to [11], [12].

In this thesis we consider time-division multiple access (TDMA) to radio channel. This choice is justified by the following reasons. First, TDMA is supported by multiple radio technologies used in MHWN such as WiMax, WirelessHART [13], ISA100 [14], as well as low-rate personal area networks (LR-WPAN) based on the 802.15.4 specifications. Then, because of its properties (avoidance of collisions, idle listening, and overhearing), TDMA is predominantly used in industrial wireless networks (including industrial wireless sensor networks) [15] and is also believed to be one of the best choices for energy efficient multiple access protocol for WSN [16]. At the same time, TDMA is favored from the viewpoint of the applications that require high level of quality of service [17]. Finally, since TDMA outperforms other access control schemes (including random access based schemes) when it comes to network traffic throughput, the traffic performance measurement obtained for TDMA can be used as an upper bound on traffic performance of MHWN using other medium access control (MAC) schemes.

Finally, in this thesis we assume periodic packet traffic which means that data sources generate packets at regular intervals, and hence, transmissions in network follow a cyclic pattern. This type of traffic is common for the IoT applications, in particular for collecting sensors measurement and for actuators control. We should also note that such a traffic can be well handled with periodically repeated TDMA frames.

1.2 Goals of the thesis

The goal of this thesis is twofold. First, we provide a comprehensive optimization model for MHWN with TDMA access to radio channel that serve multicast periodic traffic. To achieve this goal we extend the optimization model from [18]. As a result, our model include the following important aspects related to MHWN:

- traffic throughput maximization (including optimization of multicast routing trees, modulations and coding schemes assignment, and transmission power control)
- energy consumption minimization
- packet delay minimization.

Let us first note that in this thesis by optimization model we mean notions, notations, problem formulations, and solution algorithms that all together provide a consistent approach to the general problem of MHWN optimization.

The second goal is to illustrate the considerations of the thesis with numerical results obtained by implementing the introduced optimization model.

1.3 Motivation

Despite the significant importance of MHWN and multicast transmission along with the vast amount of the existing literature on this subject, a comprehensive optimization model for

MHWN under the multicast traffic assumption, to the best of our knowledge, is not available (however, such a model exists, but for a substantially simpler unicast case [19]). By providing such an optimization model we can treat the considered problems in a rigid mathematical way and use it to provide performance evaluation of the networks in question. Further, it can also be used as a benchmark for the protocols intended for MHWN. Finally, because of the TDMA assumption such a model can be useful also for other multiple access control schemes where the results provided by the model can be used as an upper bound on network performance.

1.4 Layout of the thesis

This thesis is organized as follows. In Chapter 2 we study the existing literature on multicast transmission and MHWN. Then, Chapter 3 is devoted to presenting the optimization model introduced in [18] which is extended in the subsequent chapters. Chapter 4 and Chapter 5 describe extensions related to throughput maximization. Chapter 4 discusses dynamic and static modulation and coding schemes assignment, while Chapter 5 studies transmission power control; the latter contains also numerical study illustrating considerations of these two chapters. Then, energy consumption minimization is studied in Chapter 6, while Chapter 7 deals with packet delay minimization. Both these chapters contain numerical results illustrating the most important issues described in these chapters. In Chapter 8 we provide an alternative approach to routing optimization, and in Chapter 9 we deal with destination nodes selection optimization. Finally, in Chapter 10 we conclude the thesis by discussing the strengths and weaknesses of the introduced model, as well as potential directions for future work.

Chapter 2

Related work

The aim of this chapter is to provide the reader with an overview of the literature on the most important aspects of this thesis. We first focus on multicast transmission and TDMA in the context of MHWN, and then describe the work related to the performance metrics considered in our optimization model, i.e, traffic throughput, energy consumption, and packet delay. Note that some papers cover several aspects, and can therefore be classified into more than one category; in these cases our final classification was to some extent subjective.

2.1 Multicast transmission

Due to the significance of multicast transmission in MHWN, a vast amount of papers dealing with this subject can be found in the literature. The authors of [20] notice that WSN constitute a new field where multicast is advantageous. They discuss and evaluate by means of simulation the benefits of multicast in WSN. The general conclusion is that multicast reduces the number of transmitted packets leading to better bandwidth utilization and energy consumption reduction. However, implementation of multicast in WSN is not straightforward and potential problems to be resolved in order to realize efficient and reliable multicasting in WSN are described in [21]. Paper [22] discusses challenges and opportunities of multicasting in WMN with a conclusion that multicast transmission is advantageous when it comes to video dissemination, crisis management as well as home entertainment. Regarding recent works, multicast transmission in MHWN is discussed mainly in the context of MTC and

IoT. For example, a critical analysis of multicast in emerging 5G technologies including network coding, beamforming and device to device communication can be found in [23].

A vast amount of the literature is devoted to devising efficient routing algorithms and protocols. An energy optimal multicast routing protocol oCast for WSN is presented in [24]. The authors study two approaches: centralized and distributed, and show that these approaches yield optimal multicast trees as long as the number of destination nodes is small. In [25] a stateless High Power Node multicast protocol that is well suited for dynamic WSN is proposed. The authors of [26] propose a multipath-based multicast routing protocol that leads to improved energy utilization and network lifetime of hierarchical WSN. Similarly, numerous solutions were proposed when it comes to WMN. The authors of [27] make use of ant colony optimization to construct a multicast routing algorithm intended for WMN. In [28], a difference between unicast and multicast routing in the way data packets are transmitted at the link layer is pointed out and then used to appropriately adapt routing metrics for high-throughput multicast routing. The authors of [29] focus on a channel assignment strategy in order to decrease the level of interferences and hence maximize network traffic throughput. Particularly, to improve network performance the proposed algorithm considers not only orthogonal channels but also partially overlapping channels.

2.2 Time division multiple acces

As already mentioned, TDMA plays an important role in MHWN. The authors of [30] state that TDMA is perhaps the best option for WSN in terms of energy efficiency. In the same article the authors compare TDMA and CSMA/CA, pointing out that TDMA outperforms CSMA/CA when it comes to power consumption and bandwidth utilization. It is also observed that the higher the traffic level, the greater the benefits of using TDMA schemes. Such observation is also made in [31] where an analytical model is proposed to evaluate such network metrics as traffic throughput, packet delay as well as packet delivery ratio for a given transmission schedule. That paper is of particular interest since it is one of few papers using queueing theory in the context of MHWN. Note however that although such an approach allows for fast and strict performance evaluation of a given schedule, it is not able, contrary to optimization models, to provide a schedule that maximizes a given network metric.

Some optimization approaches for MHWN with TDMA can be found in [32] and [33]. In [32] a mixed integer-nonliner problem is solved to find optimal routing and link scheduling. For large network instances a heuristic approach is proposed. However, contrary to this thesis, only unicast transmission and the simplified protocol interference model is considered. In [33] integer linear programming is used to find energy efficient scheduling in WSN. Similarly to [32], the paper deals with only unicast transmission and assumes that concurrent transmissions occurs in several radio channels without interferences. Finally, some heuristic approaches are presented in [34], [35], and [36]. Paper [34] proposes a TDMA scheduling algorithm based on genetic algorithm and aims at minimizing the delay of communication in a data collection scenario. In [35] a heuristic algorithm that is intended for traffic throughput maximization and fair rate allocation under a given network lifetime requirement in WSN that utilize TDMA is described, while [36] considers a distributed scheduling problem for channel access in TDMA-based WMN. To solve this problem, the authors of [36] provide greedy, fully distributed algorithms for two different system models.

2.3 Traffic throughput maximization

The main performance metric considered in our optimization model is traffic throughput. Obviously, this issue was also well studied in the literature. In paper [37] joint link scheduling and power control in MHWN with the objective of throughput maximization is considered. The authors noticed that as long as traffic throughput is the only optimization criterion, an imbalance between bandwidth allocation among links may occur. Thus, to achieve a reasonable tradeoff between fairness and throughput, a parameter DSF (demand satisfaction factor) was introduced. Along with mixed-integer programming (MIP) formulations, the authors propose a polynomially solvable heuristic algorithm. Further, the problem of joint scheduling, power, and rate control in MHWN is studied in [38] whose authors aim at maximizing end-to-end data rates. To solve the problem the authors propose a MIP formulation and a computationally efficient heuristic algorithm. The authors of [39] describe the problem of finding resource (such as transmitting power or time slot fractions) allocation that results in fair end-to-end communication rates as a nonlinear mathematical program. Particularly, a solution method based on a nonlinear column generation technique was developed and proved to converge to the optimal solution. Also, several rate and power adaptation schemes are considered in this paper.

Further, paper [19] provides an optimization model for joint routing, scheduling, rate and power control, as well as channel assignment optimization in WMN. The aim is to optimize efficiency of the resource allocation by means of minimization of the number of time slots. The problem formulation from [19] is non-compact and makes use of the notion of the so called compatible sets, i.e., the sets of links that transmit simultaneously in the same time slot of a TDMA frame. In fact the notion of compatible set was considered as early as in 1992 in [40], and its multicast version is used in this thesis. The notion of compatible set is also used in [41] which deals with fair traffic allocation in WMN by assuming a max-min fairness traffic objective. Besides non-compact MIP formulations that make use of compatible sets the paper provides also formulations that explicitly assign transmissions to time slots as well as a heuristic solution algorithm. The problem of a proper assignment of modulations and coding schemes to broadcasting nodes is also studied in that paper: both static and dynamic cases. As far as a recent work is concerned, some interesting results can be found in [42] where the problem of traffic throughput maximization in WSN is studied. The paper considers a scenario of data collection from stationary sensor nodes to a mobile sink under one-hop communication assumption. Together with a MIP formulation two efficient heuristic algorithms are provided.

It should be highlighted that although all the previously mentioned papers deal with the problem of traffic throughput maximization in MHWN in the same way as we are, i.e., by means of the mathematical programming, they all consider only unicast transmission. In fact, the papers attempting at mathematical modeling are in minority with respect to the papers devoted to devising protocols that are based on heuristic algorithms. For example, paper [43] proposes a routing algorithm that uses a clustering mechanism aiming at increasing the traffic throughput of MHWN in IoT environment. Particularly, a routing metric taking into account traffic load as well as intra-flow and inter-flow interference is proposed in this paper. In [44] a protocol that manages channel assignment and minimizes the mutual interferences is proposed. The goal of the protocol is to increase the network throughput by providing a proper balance of traffic load. Finally, [45] maximizes the network throughput of solar-powered WSN by applying reinforcement learning. An adaptive power control mechanism is proposed in the paper to assure that the harvested energy is well balanced among the network nodes so that the effective traffic throughput can be maximized.

2.4 Energy consumption minimization

The problem of energy consumption minimization is important for MHWN in general, and especially for WSN. Because of their nature, it is important to construct solutions that not only increase network performance in terms of traffic throughput or packet delay but also take limited resources into consideration. It is not uncommon for such networks that their nodes are battery-powered and placed in hardly accessible locations; and hence it is crucial to reduce the total energy consumption in order to extend the network lifetime.

A numerous papers dealing with this problem exist in the literature. Paper [46] points out the importance of energy consumption minimization in a new trend called Green Networking that takes into account not only the network performance but also their carbon footprint. This paper presents MIP formulations for energy management in WMN. The aim is to minimize the energy consumption during the whole day. The day is divided into 8 intervals with demand volumes depending on a given interval. Further, paper [47] describes an optimization framework that is based on linear programming (LP) for the problem of joint energy consumption and network capacity optimization in WMN. To solve the problem, the column generation technique is used. A tradeoff between network capacity and energy consumption is also studied there.

When it comes to WSN, an interesting approach can be found in [48] in which network coding is used to minimize energy consumption in WSN with additional lifetime constraints. The authors present a LP formulation of the problem under multi-path routing assumption. Although the paper considers broadcast transmissions, such transmissions are used only for the coded traffic and all demands are defined as source-destination pairs. Paper [49] describes a cross-layer routing algorithm that minimizes transmission energy taking into account an additional constraint imposed on the maximum delay. Both optimal, brute force algorithm and suboptimal algorithm are presented. However, a significant limitation is that the system model assumes only one source node and one sink node. Another analytical approach can be found in [50] where the authors deal with the problem of energy minimization using queueing theory. Further, a heuristic approach to energy consumption minimization in WSN can be found in [51] where different variants of genetic algorithm were applied to find the minimum value of the energy consumed during communication. In [52] energy optimization is performed by means of ant colony optimization technique. Paper [53] proposes a routing protocol for energy optimization in WSN. The protocol is based on LEACH (Low-energy adaptive clustering hierarchy), which is a MAC protocol with routing and clustering aiming at decreasing energy consumption in TDMA-based WSN. The protocol proposed by the authors makes use of genetic algorithm to appropriately choose a cluster head and leads to a decrease of the energy consumption. Finally, [54] applies a genetic algorithm that is based on chicken swarm optimization to provide energy efficient clustering in WSN.

2.5 Packet delay minimization

Packet delay minimization is of significant importance for WMN and WSN. The applications that require strict delay requirement include IPTV and VoIP in WMN and information dissemination and code update in WSN.

Papers [55] and [56] focus on delivering routing methods aiming at delay minimization. In [55] a new routing metric that minimizes end-to-end delay in multi-radio WMN is proposed, while [56] proposes DIAR (delay-and-interference aware routing), a method that minimizes the delay taking into account the available bandwidth, probability of the transmission failure and interference level. To solve the optimization problem the authors use genetic algorithm. The authors of [57] use queueing theory to provide a method for obtaining lower bounds on the delay when a scheduling policy is given. The paper assumes multiple source-destination demand pairs and 2-hop interference model. Other papers dealing with delay analysis in MHWN by means of queueing theory are [58] and [59].

Papers [60, 61, 62], on the other hand, deal with packet delay minimization in a specific

WSN setting, i.e., WSN that work according to duty cycling which means that each node is capable of receiving only in its active time slots. Paper [60] deals with the scenario when one node has to broadcast a message to all other nodes in the network and constructs a collision-free schedule. In [61], the authors propose a routing protocol that is based on a Markov decision process which leads to energy savings, and at the same time ensures that the end-to-end delay is below a given bound. Paper [62] deals with a multicast scenario when one source sends data to K destination nodes. The aim of the authors is to provide a transmission schedule that keeps the delay below a given threshold. Several centralized algorithms were proposed in the paper: an optimal algorithm for $K = 2$, a method based on dynamic programming for K equals to the number of network nodes, and two approximation algorithms with approximation ratio $K - 1$ for $K > 2$. A distributed algorithm with low computational complexity was also proposed.

MIP is applied to solve a delay minimization problem in [63, 64, 65]. Paper [63] focuses on providing delay-bounded routing and scheduling for multi-hop WSN. A MIP formulation of the problem is presented in the paper, and since the problem is $\mathcal{NP}$-complete, the authors propose a solution method that is based on the column generation technique for providing suboptimal solutions. In [64], MIP formulations for packet delay minimization in MHWN under physical interference are presented. Two cases are considered in the paper. In the first case packet delay is optimized for a given TDMA frame that maximizes network capacity, while in the second case, the delay is optimized without any assumption of the transmissions pattern (in this case the constraint on frame periodicity is also removed). For large problem instances a greedy heuristic algorithm is proposed. A problem of delay minimization in MHWN with the physical interference model is studied also in [65], where the authors present the sets of MIP formulations some of which include forward interference cancellation and cooperative forwarding. To solve more computationally demanding cases the authors propose two heuristic algorithms. However, none of these three papers deals with multicast traffic.

2.6 Conclusion

Apparently, a lot of work has been done in the area studied in this thesis. The papers we described focus on both devising routing protocols for MHWN together with underlying algorithms and a mathematical modeling - either by means of queueing theory or by means of mathematical programming. The second group is especially interesting from this thesis viewpoint, however, although the papers in this group provide many interesting results we were not able to find a paper dealing with a particular problem setting considered in this thesis (i.e., multicast traffic, TDMA, and physical interferences). The papers most closely related to our work are [19] and [64], but even there only unicast traffic is considered. Therefore, to the best of our knowledge, this thesis is an original work that provide a comprehensive optimization model for MHWN serving multicast traffic and utilizing TDMA.

Chapter 3

Basic optimization model

As previously mentioned, the original optimization model which is the starting point for the considerations of this thesis was published in [18]. This chapter is devoted to presenting this model, and in particular to introducing notations, basic notions, and problems formulations used in this thesis. All these elements form a base for describing the extensions that constitute the main contribution of this thesis. In the following we refer to this model as basic optimization model (BOM). The description used in this chapter is taken from [1]; also the figures are reproduced from [1].

3.1 Network graph

In BOM and its extensions, MHWN is modelled by means of a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{A} \subseteq \mathcal{V} \times \mathcal{V} \backslash \{(v,v) : v \in \mathcal{V}\}$ is the set of directed links (arcs). The beginning of arc a is denoted by $b(a)$, while its end by $e(a)$. It is assumed that an ordered pair (w,v) is an arc ($(w,v) \in \mathcal{A}$) if, and only if, node v is in the transmission range of node w, i.e., the signal-to-noise ratio (SNR) condition is satisfied at node v when node w is transmitting and the remaining nodes are not: $\frac{p(w,v)}{\eta} \geqslant \gamma$, where $p(w,v)$ is the transmission power received at node v from node w, η is the noise power, and $\gamma > 1$ is the assumed power ratio threshold (SNR threshold here). The set of all nodes connected to node $v \in \mathcal{V}$ with the arcs originating at v is denoted by $\delta^+(v)$, and the set of all nodes connected to $v \in \mathcal{V}$ with the arcs terminating at v is denoted by $\delta^-(v)$; hence, $\delta^+(v) := \{w \in \mathcal{V} : (v,w) \in \mathcal{A}\}$ and

$\delta^-(v) := \{w \in \mathcal{V} : (w, v) \in \mathcal{A}\}$.

3.2 Packet streams and multicast trees

The network is supposed to carry a given set of packet streams denoted by $\mathcal{S}$. Each stream $s \in \mathcal{S}$ generates successive packets $s(1), s(2), \ldots$ (referred to as s-packets) to be delivered from its source node $o(s)$ to each node in a given set $\mathcal{D}(s)$ of destination nodes. The packet size is fixed and is the same for all streams.

In the packet transfer process, each s-packet is transmitted along paths of a given multicast (routing) tree $\mathcal{T}(s) = (\mathcal{V}(s), \mathcal{A}(s))$ rooted at $o(s)$, with the set of nodes $\mathcal{V}(s)$ and the set of arcs $\mathcal{A}(s)$ (where $\mathcal{V}(s) \subseteq \mathcal{V}$ and $\mathcal{A}(s) \subseteq \mathcal{A}$).

Since such described multicast tree may (and sometimes must) contain nodes other than $\{o(s)\} \cup \mathcal{D}(s)$, the undirected graph underlying $\mathcal{T}(s)$ is in general a Steiner tree in the undirected graph underlying the network graph; because of that the nodes in set $\{o(s)\} \cup \mathcal{D}(s)$ are called terminals, and the nodes in set $\mathcal{V}(s) \backslash (\{o(s)\} \cup \mathcal{D}(s))$ are called Steiner nodes. In the multicast tree $\mathcal{T}(s)$ each node has exactly one incoming arc except the root $o(s)$ which has no incoming arcs. A node $v \in \mathcal{V}(s)$ with no outgoing arcs is called a leaf. Note that each leaf is a terminal, but in general the nodes in $\mathcal{D}(s)$ do not have to be leaves and they may transit the packets other than s-packets. In the following, the set of leaves in $\mathcal{T}(s)$ is denoted by $\mathcal{V}''(s)$, and the set $\mathcal{V}(s) \backslash \mathcal{V}''(s)$ of the non-leaf nodes by $\mathcal{V}'(s)$.

Observe that since at each (non-leaf) node w in $\mathcal{V}'(s)$ the s-packet has to be transferred along all arcs in $\mathcal{A}(s)$ outgoing from w, i.e., to the set of nodes $\mathcal{V}(s, w) := \{v \in \mathcal{V} : (w, v) \in \mathcal{A}(s)\}$, for multicast traffic it is advantageous to make use of the broadcast nature of radio transmissions.

An example network with one packet stream is depicted in Figure 3.1. Node w_1 is the source node, while nodes w_4, w_5, and w_6 are the destination nodes (and, at the same time leaves).

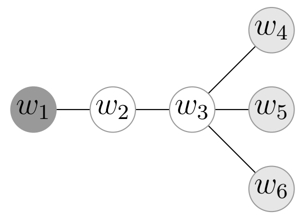

Figure 3.1: An example network with one packet stream.

3.3 Transmission patterns

The network is assumed to utilize time division multiple access to radio channel. Packet transmissions are executed in consecutive time slots (slots in short), each of duration of τ seconds, grouped into consecutive (time) frames of F slots each. Let b (expressed in Mbps) and B (expressed in kb) denote the data rate (common for all arcs) and the packet size (common for all streams), respectively. Then, assuming $\tau = \frac{1000 \times B}{b}$ we ensure that transmission of one packet over any arc takes exactly one time slot.

Slot transmission patterns applied in the consecutive frames are identical. This means that the f-th slot ($f = 1, 2, ..., F$) in all frames has the same set of broadcasting nodes and that each node in this set is associated with the same (node-dependent) set of receiving nodes. Thus, to describe the frame we need to specify such a slot transmission pattern, called compatible set (or c-set in short), for each slot $f = 1, 2, ..., F$.

A c-set c is described by the set $\mathcal{W}(c)$ (where $\mathcal{W}(c) \subseteq \mathcal{V}$) of simultaneously broadcasting nodes and, for each $w \in \mathcal{W}(c)$, the set $\mathcal{U}(c, w)$ (where $\mathcal{U}(c, w) \subseteq \mathcal{V} \backslash \mathcal{W}(c)$) of nodes receiving the signal from w. Certainly, such an arrangement is valid only when each of the nodes in $\mathcal{U}(c, w)$ can decode the signal from w, i.e., the nodes receiving from w are not interfered by the signals that are simultaneously broadcast from the nodes in $\mathcal{W}(c) \backslash \{w\}$. Actually, this requirement is satisfied if, and only if, the signal-to-interference-to-noise ratio (SINR) condition

$$\frac{p(w, u)}{\eta + \sum_{v \in \mathcal{W}(c) \backslash \{w, u\}} p(v, u)} \geqslant \gamma \tag{3.1}$$

is satisfied for all $w \in \mathcal{W}(c)$ and $u \in \mathcal{U}(c, w)$. Note that, since $\gamma > 1$ (in this context γ is referred to as SINR threshold), (3.1) implies pairwise disjointness of the family of receiving sets $\mathcal{U}(c, w)$, $w \in \mathcal{W}(c)$.

Formally, the c-set can be defined by means of a system of linear inequalities imposed on binary coefficients $X(w)$ ($w \in \mathcal{V}$) and $Y(w, u)$ ($w \in \mathcal{V}, u \in \delta^+(w)$), where $\mathcal{W}(c) := \{w \in \mathcal{V} : X(w) = 1\}$ and $\mathcal{U}(c, w) := \{u \in \delta^+(w) : Y(w, u) = 1\}$. The so described sets define a valid c-set if, and only if, the following system of inequalities is satisfied:

$$X(w) \geqslant Y(w, u), \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{3.2a}$$

$$X(w) \leqslant \textstyle\sum_{u \in \delta^+(w)} Y(w, u), \quad w \in \mathcal{V} \tag{3.2b}$$

$$X(w) + \textstyle\sum_{u \in \delta^-(w)} Y(u, w) \leqslant 1, \quad w \in \mathcal{V} \tag{3.2c}$$

$$p(w, u) + M(w, u)(1 - Y(w, u)) \geqslant \gamma\big(\eta + \textstyle\sum_{v \in \mathcal{V} \setminus \{w,u\}} p(v, u)X(v)\big), \quad (w, u) \in \mathcal{A}. \tag{3.2d}$$

Inequalities (3.2a) and (3.2b) force the node to transmit ($X(w) = 1$) if its transmission is to be received by at least one of its neighbors ($Y(w, u) = 1$); otherwise $X(w)$ is forced to be 0. Next, inequality (3.2c) ensures that if a node is transmitting then it cannot receive and if a node is not transmitting then it can be receiving but from at most only one other node. Finally, inequality (3.2d) expresses the SINR condition (3.1). The second term on the left-hand side of (3.2d) is added to cancel the constraint whenever $Y(w, u) = 0$ using a "big M" constant that can for example be defined as

$$M(w, u) := \gamma(\eta + \textstyle\sum_{v \in \mathcal{V} \setminus \{w,u\}} p(v, u)), \tag{3.3}$$

i.e., the upper bound on the right-hand side in (3.2d).

Below, the family of all c-sets is denoted by $\widehat{\mathcal{C}}$ and we use $\mathcal{C}$ to denote a given family of c-sets ($\mathcal{C} \subseteq \widehat{\mathcal{C}}$). Observe that in general the number of c-sets in $\widehat{\mathcal{C}}$ grows exponentially with the size of the network graph.

An example family of c-sets that can be applied to transmit packets from the previously described stream is depicted in Figure 3.2 (broadcasting nodes of the c-sets are depicted in dark gray, while receiving nodes in light gray).

With the notion of the compatible set we can now describe the frame by introducing the so called frame transmission pattern which is the sequence $\widehat{\mathcal{F}} := (c(f))_{f \in \mathcal{F}}$, where $c(f)$ denotes the c-set active in slot f. In other words, frame transmission pattern is the sequence that specifies the c-set active in each time slot of the frame.

Two example ways of realizing transmissions of packets from the stream within TDMA frames with two different subsets of c-sets are illustrated in Figure 3.3. Symbol $\times$ denotes that a node is not scheduled to transmit in a given time slots, while symbol $\circ$ denotes that it has, at that moment, nothing to transmit. Note that in Figure 3.3 (b) the packets are not delivered within single TDMA frame, however the chosen frame transmissions pattern, if periodically repeated, guarantees that the packets will eventually reach all their destination with a finite delay.

Observe finally, that when it comes to traffic throughput it is not necessary to have knowledge of the applied frame transmission pattern. In fact, the only thing we need to know is the number of times each c-set is applied in the frame (as is shown in the following part of this chapter). This is defined by frame composition $[\mathcal{C}, (F(c),\ c \in \mathcal{C})]$ - the pair of a c-sets family $\mathcal{C}$ together with the number $F(c), c \in \mathcal{C}$, indicating how many times each c-set from family $\mathcal{C}$ is used in the frame.

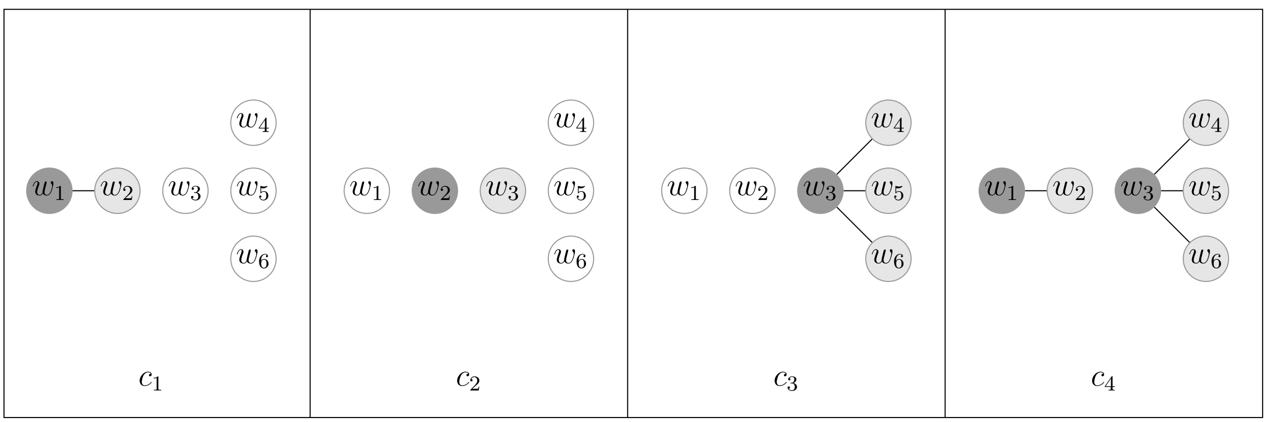

Figure 3.2: An example family of c-sets.

c	c_1	c_2	c_3
w_3	$\times$	$\times$	$s(1)$
w_2	$\times$	$s(1)$	$\times$
w_1	$s(1)$	$\times$	$\times$
f	1	2	3
$\mathcal{F}$	1		

(a)

c	c_4	c_2	c_4	c_2
w_3	$\circ$	$\times$	$s(1)$	$\times$
w_2	$\times$	$s(1)$	$\times$	$s(2)$
w_1	$s(1)$	$\times$	$s(2)$	$\times$
f	1	2	1	2
$\mathcal{F}$	1		2	

(b)

Figure 3.3: Transmissions within TDMA frames.

3.4 Simple frame minimization problem (SFMP)

In this section we present the first problem introduced in [18], namely the simple frame minimization problem SFMP. Note that this problem appeared in [18] under the acronym MFCP, and in the following we present its slightly modified equivalent from [1]. Together with the problem formulation we also describe a solution algorithm of SFMP since it forms a base for the solution algorithms of the extensions of the model.

3.4.1 Formulation of SFMP

The simple frame minimization problem can be verbally posed as follows.

- SFMP: assuming given sets of destination nodes $\mathcal{D}(s)$, $s \in \mathcal{S}$, and routing trees $\mathcal{T}(s)$, $s \in \mathcal{S}$, find the minimum frame length F, together with a frame composition $[\mathcal{C}, (F(c), c \in \mathcal{C})]$ such that for each stream $s \in \mathcal{S}$ the network is able to carry all packets $s(1), s(2), \ldots$ in case when the consecutive packets are generated at the beginning of the consecutive frames (i.e., for each $s \in \mathcal{S}$, $s(1)$ is generated at the beginning of frame 1, $s(2)$ is generated at the beginning of frame 2, and so on).

Note that the so posed problem is equivalent to traffic throughput maximization where traffic throughput is defined as the intensity of packet arrivals, which is equal to one per stream per frame. As we already stated, for solving SFMP it is not important in which particular slots

Table 3.1: Notation 1 (SFMP).

Notation	Description
$f = 1, 2, \dots, F$	(time) slots of the frame
$\mathcal{C}$	list of allowable c-sets used in frame optimization
$\mathcal{C}(a)$	subset of those c-sets in $\mathcal{C}$ that broadcast over arc $a = (w, v)$ ($\mathcal{C}(a) := \{c \in \mathcal{C} : w \in \mathcal{W}(c), v \in \mathcal{U}(c, w)\}$)
F_c	number of times the c-set c is used in the frame ($c \in \mathcal{C}$) – primal integer variable
h_{scw}	equals to 1 if, and only if, node w of c-set c is used to broadcast s-packets ($s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \cap \mathcal{V}'(s)$) – primal binary variable
$\mathcal{S}(c, w)$	set of streams for which $w \in \mathcal{V}'(s)$, i.e., s-packets can be broadcast from w (defined for $w \in \mathcal{W}(c)$)
$\mathcal{V}(s, w)$	set of nodes appearing after node $w \in \mathcal{V}'(s)$ in tree $\mathcal{T}(s)$ ($\mathcal{V}(s, w) := \delta^+(w) \cap \mathcal{V}(s)$)
$\mathcal{A}(s, c, w)$	set of arcs in $\mathcal{A}(s)$ which originate at node $w \in \mathcal{W}(c)$ and terminate in $\mathcal{U}(c, w)$ ($\mathcal{A}(s, c, w) := \{(w, v) : v \in \mathcal{U}(c, w)\} \cap \mathcal{A}(s)$)
$\mathcal{S}(w)$	set of streams that contain w as a broadcasting node ($\mathcal{S}(w) := \{s \in \mathcal{S} : w \in \mathcal{V}'(s)\}$)
$\mathcal{A}(s, w)$	set of arcs originating at node w in the multicast tree of stream s ($\mathcal{A}(s, w) := \{(w, v) \in \mathcal{A} : (w, v) \in \mathcal{A}(s)\}$)

of the frame a given packet is broadcast from a given node – what is important is only that within the frame each s-packet is successfully broadcast from each non-leaf node $w \in \mathcal{V}'(s)$ of tree $\mathcal{T}(s)$ to all destinations in the set $\mathcal{V}(s,w)$, where, for each $w \in \mathcal{V}'(s)$, $\mathcal{V}(s,w)$ denotes the set of all nodes that appear after node w in $\mathcal{T}(s)$. This will guarantee that all consecutive packets of stream s are eventually delivered (without losses) to all destinations (with the same finite delay, dependent on the destination node $v \in \mathcal{D}(s)$).

Thus, when solving SFMP, it suffices to know only how many times each c-set in the considered family $\mathcal{C}$ is used since each occurrence of a particular c-set can be placed anywhere in the frame and therefore it is the frame composition and not the frame transmission pattern that is needed to solve SFMP.

Now let us consider a given family of c-sets $\mathcal{C} \subseteq \widehat{\mathcal{C}}$. For the assumed c-set family, SFMP can be expressed by the following MIP formulation.

$$\text{SFMP}(\mathcal{C}): \quad \max \quad F = \textstyle\sum_{c \in \mathcal{C}} F_c \tag{3.4a}$$

$$[\lambda_{sa} \geqslant 0] \quad \textstyle\sum_{c \in \mathcal{C}(a)} h_{scb(a)} \geqslant 1, \quad s \in \mathcal{S}, a \in \mathcal{A}(s) \tag{3.4b}$$

$$[\pi_{cw} \geqslant 0] \quad \textstyle\sum_{s \in \mathcal{S}(c,w)} h_{scw} \leqslant F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{3.4c}$$

$$h_{scw} \in \mathbb{B}, \; s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \cap \mathcal{V}'(s) \tag{3.4d}$$

$$F_c \in \mathbb{R}, \; c \in \mathcal{C}. \tag{3.4e}$$

(The quantities λ_{sa} and π_{cw} on the left side of constraints (3.4b)-(3.4c) are dual variables to be used in the linear relaxation of (3.4) considered in the next section. Note that such a notation is used in the whole thesis, i.e., quantities in the square brackets on the left side of a given constraint always denote the dual variables corresponding to this constraint.)

The above formulation makes use of two sets of variables: non-negative integer variables h_{scw}, $s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \cap \mathcal{V}'(s)$, and continuous variables F_c, $c \in \mathcal{C}$. Note that although variables F_c are real valued and unconstrained in sign, in any optimal solution of SFMP they will be non-negative integers anyway.

Variable h_{scw} is equal to 1 if c-set c is used to broadcast an s-packet from node w (and to 0, otherwise). Feasibility of the values assumed by h_{scw} is ensured by constraint (3.4b): for each arc a in the multicast tree of stream s there must be at least one broadcast of an s-packet over arc a. $\mathcal{C}(a)$ on the left-hand side of (3.4b) denotes the subset of c-sets in $\mathcal{C}$ which broadcast along arc a, i.e., $c \in \mathcal{C}(a)$ if, and only if, $c \in \mathcal{C}$, $b(a) \in \mathcal{W}(c)$, and $e(a) \in \mathcal{U}(c, w)$.

Variable F_c, in turn, expresses the number of time slots that use c-set c. In order to ensure that c is applied sufficiently many times, constraint (3.4c) is introduced, one for each node broadcasting in c, i.e., for each $w \in \mathcal{W}(c)$; these constraints imply that the value of F_c is sufficient to cover all transmissions assigned to c by variables h_{scw}. Note that the summation on the left-hand side of constraint (3.4c) corresponding to $w \in \mathcal{W}(c)$ is taken over all streams s that can (potentially) use the broadcast from w. This means that w must be a non-leaf node of the multicast tree $\mathcal{T}(s)$ and hence the summation in question is taken over the set $\mathcal{S}(c, w) := \{s \in \mathcal{S} : w \in \mathcal{V}'(s) \wedge w \in \mathcal{W}(c)\}$.

Thus, objective (3.4a) minimizes the total frame length, and any set of optimal values F_c^*, $c \in \mathcal{C}$, specifies an optimal frame composition $\mathcal{C}' := \{c \in \mathcal{C} : F_c^* > 0\}$ and $F(c) := F_c^*$, $c \in \mathcal{C}'$.

Observe that formulations of (3.4) with a limited list of c-sets, i.e., SFMP($\mathcal{C}$) with $\mathcal{C} \subsetneq \widehat{\mathcal{C}}$, will in general not provide a globally optimal solution since, as can be easily seen, $\mathcal{C}$ may not contain the c-sets necessary to obtain the minimum frame length; this is guaranteed only when the full list $\widehat{\mathcal{C}}$ of c-sets is considered in SFMP. The resulting formulation SFMP($\widehat{\mathcal{C}}$) is non-compact since, as already mentioned, in general the size of $\widehat{\mathcal{C}}$ grows exponentially with the size of the network graph, and this results in an exponential number of variables and constraints in (3.4).

Finally we note that the frame minimization problem assuming the full family of c-sets (represented by formulation SFMP($\widehat{\mathcal{C}}$)) is $\mathcal{NP}$-hard, since, as shown in [66], its simpler unicast version is already $\mathcal{NP}$-hard.

3.4.2 SFMP: pricing problem for c-set generation

For the networks of practical size it is virtually impossible not only to use all possible c-sets in SFMP, but also to pre-generate a list $\mathcal{C}$ of a reasonable length that guarantees an optimal solution of SFMP. This issue can be addressed by considering the linear relaxation of SFMP($\widehat{\mathcal{C}}$), i.e., the non-compact linear program obtained from (3.4) by assuming $h_{scw} \in \mathbb{R}_+$ instead of $h_{scw} \in \mathbb{B}$ ($s \in \mathcal{S}$, $c \in \widehat{\mathcal{C}}, w \in \mathcal{W}(c) \cap \mathcal{V}'(s)$). Note that it is not necessary to assume that $h_{scw} \leqslant 1$ since such a constraint will be satisfied in any optimal solution of the linear relaxation anyway.

The linear relaxation in question can be solved by c-set generation in the following way. Let MP($\mathcal{C}$) (the so called master problem) denote the linear relaxation of SFMP($\mathcal{C}$), i.e., SFMP assuming a limited list of c-sets, and let $F^*(\mathcal{C})$ denote the minimum frame length achievable when only c-sets from $\mathcal{C}$ are allowed for transmission patterns. Now we pose the question: does there exist a c-set outside the current c-set list $\mathcal{C}$ that, when added to the list will make it possible to decrease the frame length? More precisely, we ask whether there exists $c \in \widehat{\mathcal{C}} \backslash \mathcal{C}$ such that $F^*(\mathcal{C} \cup \{c\}) < F^*(\mathcal{C})$, and, if it does, how to find such an "improving" c-set which locally maximizes the improvement $F^*(\mathcal{C}) - F^*(\mathcal{C} \cup \{c\})$.

This question can be answered through solving the so called pricing problem (PP) derived from the problem dual to MP($\mathcal{C}$), called the dual master problem (DMP). This problem is formulated using dual variables corresponding to constraints (3.4b)-(3.4c) in the linear relaxation of (3.4), i.e., in MP($\mathcal{C}$) as follows:

$$\text{DMP}(\mathcal{C}): \ \max \ \ W = \textstyle\sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}(s)} \lambda_{sa} \tag{3.5a}$$

$$\textstyle\sum_{w \in \mathcal{W}(c)} \pi_{cw} = 1, \quad c \in \mathcal{C} \tag{3.5b}$$

$$\textstyle\sum_{a \in \mathcal{A}(s,c,w)} \lambda_{sa} \leqslant \pi_{cw}, \quad c \in \mathcal{C}, w \in \mathcal{W}(c), s \in \mathcal{S}(c,w) \tag{3.5c}$$

$$\lambda_{sa} \in \mathbb{R}_+, \ s \in \mathcal{S}, a \in \mathcal{A}(s); \ \pi_{cw} \in \mathbb{R}_+, \ c \in \mathcal{C}, w \in \mathcal{W}(c). \tag{3.5d}$$

Above, $\mathcal{A}(s,c,w)$ denotes the set of all those arcs in $\mathcal{A}(s)$ that originate at node w and terminate at one of the nodes in $\mathcal{U}(c,w)$ (in other words, these are those arcs in $\mathcal{T}(s)$ that originate at w and are active in c-set c). Let $\lambda := (\lambda_{sa})_{s \in \mathcal{S}, a \in \mathcal{A}(s)}$ and let λ^* be the vector of optimal values of the dual variables λ.

PP consists in finding a c-set c in $\widehat{\mathcal{C}}$ that maximizes the expression:

$$P(c; \lambda^*) := \min_{\pi \geqslant 0, \sum_{w \in \mathcal{W}(c)} \pi_w = 1} Q(\pi; c), \tag{3.6}$$

where

$$Q(\pi; c) := \textstyle\sum_{w \in \mathcal{W}(c)} \sum_{s \in \mathcal{S}(c,w)} (\max\{0, \sum_{a \in \mathcal{A}(s,c,w)} \lambda^*_{sa} - \pi_w\}) \tag{3.7}$$

is the sum of violations of the dual constraints (3.5c) for given (feasible) values of dual variables $\pi := (\pi_w)_{w \in \mathcal{W}(c)}$ appearing in the dual formulation when c is added to the c-set list $\mathcal{C}$.

Hence, for a given vector λ^*, PP can be formulated as follows:

$$\text{maximize } P(c; \lambda^*) \text{ over } c \in \hat{\mathcal{C}}. \tag{3.8}$$

Now suppose that a c-set c is specified by means of binary coefficients $X(w), Y(w,u),\ w, u \in \mathcal{V}$: $\mathcal{W}(c) := \{w \in \mathcal{V} :\ X(w) = 1\}$, $\mathcal{U}(c, w) := \{u \in \delta^+(w) :\ Y(w,u) = 1\}$. Then, the quantity $P(c; \lambda^*)$ specified in (3.6) can be calculated by means of the following LP formulation in variables k_{sw} and π_w:

$$P(c; \lambda^*) = \min\ \sum_{w\in\mathcal{V}} \sum_{s\in\mathcal{S}(w)} k_{sw} \tag{3.9a}$$

$$[r] \quad \sum_{w\in\mathcal{V}} \pi_w X(w) = 1 \tag{3.9b}$$

$$[g_{sw} \geqslant 0] \quad k_{sw} \geqslant \sum_{u\in\mathcal{V}(s,w)} \lambda^*_{s(w,u)} Y(w,u) - \pi_w, \quad w \in \mathcal{V}, s \in \mathcal{S}(w) \tag{3.9c}$$

$$\pi_w \in \mathbb{R}_+,\ w \in \mathcal{V};\ k_{sw} \in \mathbb{R}_+, w \in \mathcal{V}, s \in \mathcal{S}(w), \tag{3.9d}$$

where $S(w) := \{s \in \mathcal{S} :\ w \in \mathcal{V}'(s)\}$.

Now, noting that constraint (3.9b) can be simplified to $\sum_{w\in\mathcal{V}} \pi_w = 1$ (because in optimal solutions of (3.9) for which $P(c; \lambda^*) > 0$, $\pi_w = 0$ when $X(w) = 0$, see [18]), and using the dual variables r (unconstrained in sign) and g_{sw} specified on the left-hand side of constraints (3.9b)-(3.9c), we can formulate the following dual to LP formulation (3.9).

$$\max\ D = -r + \sum_{w\in\mathcal{V}} \sum_{s\in\mathcal{S}(w)} \Big(\sum_{u\in\mathcal{V}(s,w)} \lambda^*_{s(w,u)} Y(w,u)\Big) g_{sw} \tag{3.10a}$$

$$\sum_{s\in\mathcal{S}(w)} g_{sw} \leqslant r, \quad w \in \mathcal{V} \tag{3.10b}$$

$$g_{sw} \leqslant 1, \quad s \in \mathcal{S}(w) \tag{3.10c}$$

$$r \in \mathbb{R};\ \ g_{sw} \in \mathbb{R}_+,\ w \in \mathcal{V}, s \in \mathcal{S}(w). \tag{3.10d}$$

The reason for considering the dual is that its optimization objective (i.e., maximization) is consistent with optimization objective in (3.8)—thanks to that formulation (3.10) can be embedded in the final formulation of PP:

$$\text{PP}(\lambda^*):\ \ \max\ \ P = -r + \sum_{w\in\mathcal{V}} \sum_{s\in\mathcal{S}(w)} \sum_{u\in\mathcal{V}(s,w)} \lambda^*_{s(w,u)} U_{swu} \tag{3.11a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{3.11b}$$

$$X_w \leqslant \sum_{u\in\delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{3.11c}$$

$$X_w + \sum_{u\in\delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{3.11d}$$

$$p(w,u) + M(w,u)(1 - Y_{wu}) \geqslant \gamma(\eta + \textstyle\sum_{v \in \mathcal{V}\setminus\{w,u\}} p(v,u) X_v), \quad (w,u) \in \mathcal{A} \quad (3.11e)$$

$$U_{swu} \leqslant g_{sw},\, U_{swu} \leqslant Y_{wu}, \quad w \in \mathcal{V}, s \in \mathcal{S}(w), u \in \mathcal{V}(s,w) \quad (3.11f)$$

$$\textstyle\sum_{s \in \mathcal{S}(w)} g_{sw} \leqslant r, \quad w \in \mathcal{V}; \;\; g_{sw} \leqslant 1, \quad w \in \mathcal{V}, s \in \mathcal{S}(w) \quad (3.11g)$$

$$Y_{wu} \in \mathbb{B},\, w \in \mathcal{V}, u \in \delta^+(w);\;\; X_w \in \mathbb{B},\, w \in \mathcal{V} \quad (3.11h)$$

$$r \in \mathbb{R};\;\; g_{sw} \in \mathbb{R}_+,\, w \in \mathcal{V}, s \in \mathcal{S}(w) \quad (3.11i)$$

$$U_{swu} \in \mathbb{R}_+,\, w \in \mathcal{V}, s \in \mathcal{S}(w), u \in \mathcal{V}(s,w). \quad (3.11j)$$

In the above PP formulation, variables X_w and Y_{wu} specify the constructed c-set and correspond to parameters $X(w)$ and $Y(w,u)$ that were used in (3.9) and (3.10) for describing the c-set that was fixed. These variables must obey constraints (3.11b)-(3.11e) that specify the necessary and sufficient conditions (corresponding to conditions (3.2a)-(3.2d) given in Section 3.3) for c-sets c in $\widehat{\mathcal{C}}$, where $\mathcal{W}(c) := \{w \in \mathcal{V} : X_w = 1\}$, $\mathcal{U}(c,w) := \{u \in \delta^+(w) : Y_{wu} = 1\}$, $w \in \mathcal{W}(c)$. Note that in (3.11e), a “big M” constants $M(w,u)$ are used to cancel the condition when $Y_{wu} = 0$. Next, variables U_{swu} (formally non-negative continuous), together with constraints (3.11f), are used to eliminate the bi-linearities $Y(w,u) \cdot g_{sw}$ implied by (3.10a). Finally, variables r and g_{sw}, and constraints (3.11g) are taken from the dual (3.10), and so is the objective in (3.11a).

In the following, c^* denotes a c-set defined by an optimal solution of (3.11): $\mathcal{W}(c^*) := \{w \in \mathcal{V} : X^*_w = 1\}$, $\mathcal{U}(c^*,w) := \{u \in \delta^+(w) : Y^*_{wu} = 1\}$, $w \in \mathcal{W}(c)$.

In fact, PP(λ^*) is a max-min problem that finds a c-set $c^* \in \widehat{\mathcal{C}} \setminus \mathcal{C}$ (if any, i.e., if $P(c^*;\lambda^*) > 0$) for which the constraints defining the dual polytope (considered for the family $\mathcal{C} \cup \{c^*\}$) projected onto the λ-space (i.e., onto $\mathbb{R}^{\sum_{s \in \mathcal{S}} |\mathcal{A}(s)|}$) are most violated by λ^*.

3.4.3 SFMP: solution algorithm

Our solution procedure of SFMP (specified in Algorithm 1) is based on price-and-branch (P&B) [67]. With P&B we first solve the linear relaxation of SFMP($\widehat{\mathcal{C}}$), and then apply the branch-and-bound (B&B) algorithm (using a MIP solver) to the MIP problem SFMP($\mathcal{C}^*$) where $\mathcal{C}^*$ denotes the final c-set list $\mathcal{C}$ obtained from Algorithm 1.

Having specified PP as a MIP problem, we can solve SFMP($\widehat{\mathcal{C}}$) using the following iterative algorithm.

Algorithm 1:

Step-1: Define an initial list $\mathcal{C}$ of feasible c-sets.

Step-2: Solve the dual master problem DMP($\mathcal{C}$) using a LP solver to obtain optimal dual variables λ^*.

Step-3: Solve the pricing problem PP(λ^*) using a MIP solver. If the maximum $P(c^*;\lambda^*)$ of the objective function is strictly greater than 0, then add the resulting c-set c^* to the c-set list $\mathcal{C}$ ($\mathcal{C} := \mathcal{C} \cup \{c^*\}$) and go to Step-2.

Step-4: Solve SFMP($\mathcal{C}$) for the resulting c-set list $\mathcal{C}$ and stop. □

Observe that the simplest feasible initial c-set list (i.e., a list $\mathcal{C}$ for which MP($\mathcal{C}$) is feasible) to be pre-computed in Step-1 is the family of all c-sets with only one node broadcasting to all its neighbors. Certainly, the initial list can be extended by adding all c-sets with two broadcasting nodes, etc. Another remark is that in the main iteration loop composed of Step-2 and Step-3, solving the MIP problem PP(λ^*) (which is $\mathcal{NP}$-hard, as demonstrated in [68], its unicast subcase is already $\mathcal{NP}$-hard) is much more time consuming than solving the LP problem DMP($\mathcal{C}$). However, as shown in the numerical section of [18], solving PP(λ^*) directly using a MIP solver is reasonably effective.

The presented procedure is heuristic in the sense that the c-set list $\mathcal{C}$ used in Step-4, although sufficient for solving the linear relaxation of SFMP($\widehat{\mathcal{C}}$) to optimality, may in general be not sufficient for solving SFMP($\widehat{\mathcal{C}}$) itself. Nevertheless, the so obtained c-set lists allow to find near-optimal solutions of SFMP. This is because (as demonstrated in [18]) the value of the objective function of SFMP($\mathcal{C}^*$) is in fact very close to the value of the objective function of MP($\mathcal{C}^*$), which is the lower bound on the value of the objective function of SFMP($\widehat{\mathcal{C}}$).

It should be highlighted that the presented SFMP solution algorithm, as illustrated in [18], is efficient in terms of computation time. For example, optimizing a 100 node network with almost 100 one-to-all multicast streams using CPLEX solver takes approximately one hour on a computing server with 20 logical processors and up to 80GB of RAM.

We finally note that the solution algorithm described in this section, unless stated otherwise, applies also to the extensions of BOM presented in the remaining part of the thesis.

3.5 Frame minimization problem with routing optimization (RFMP)

The second optimization problem introduced in [18] is the frame minimization problem with routing optimization RFMP (in [18] the problem appears under the acronym FROP). Contrary to SFMP, this time the routing trees are not fixed and are subject to optimization. In this chapter we present a MIP formulation of RFMP, together with PP used in the solution algorithm.

3.5.1 Formulation of RFMP

In order to embed routing trees optimization in SFMP a new set of binary variables is introduced, namely $y_{sa}, s \in \mathcal{S}, a \in \mathcal{A}$. Each variable y_{sa} equals 1 if, and only if, arc a belongs to the routing tree of s, in other words, variables y_{sa} for a given s define routing tree $\mathcal{T}(s)$ with $\mathcal{A}(s) = \{a \in \mathcal{A} : y_{sa} = 1\}$. Accordingly, to take into account that now packets transmissions are not realized over fixed routing trees, constraint (3.4b) has to be modified in the following way:

$$\textstyle\sum_{c \in \mathcal{C}(a)} h_{scb(a)} \geqslant y_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A}. \tag{3.12a}$$

Additionally, to construct proper routing trees, flow variables $z_{swa}, s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A}$, have to be introduced. The flow variables corresponding to a given stream s specify the paths from s to its destinations in $\mathcal{D}(s)$: the set of arcs forming the selected path from s to $w \in \mathcal{D}(s)$ is determined as $\{a \in \mathcal{A} : z_{zwa} = 1\}$; certainly, such variables depend on variables y_{sa}. Together with the flow variables additional constraints have to be introduced to ensure the correctness of the constructed tree:

$$\textstyle\sum_{u \in \delta^-(v)} z_{sw(u,v)} + I(s,w,v) = \sum_{u \in \delta^+(v)} z_{sw(v,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{3.13a}$$

$$z_{swa} \leqslant y_{sa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A}, \tag{3.13b}$$

where parameter $I(s,w,v)$ used in (3.13a) equals 1 for $v = o(s)$, -1 for $v = w$, and 0 otherwise.

Considering the above, the final formulation of RFMP is as follows:

$$\text{RFMP}(\mathcal{C}): \ \min \ F = \textstyle\sum_{c \in \mathcal{C}} F_c \tag{3.14a}$$

$$[\lambda_{sa} \geqslant 0] \quad \textstyle\sum_{c \in \mathcal{C}(a)} h_{scb(a)} \geqslant y_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{3.14b}$$

$$[\pi_{cw} \geqslant 0] \quad \textstyle\sum_{s \in \mathcal{S}} h_{scw} \leqslant F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{3.14c}$$

$$[\varphi_{swv}] \quad \textstyle\sum_{u \in \delta^-(v)} z_{sw(u,v)} + I(s, w, v) = \textstyle\sum_{u \in \delta^+(v)} z_{sw(v,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{3.14d}$$

$$[\sigma_{swa} \geqslant 0] \quad z_{swa} \leqslant y_{sa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{3.14e}$$

$$y_{sa} \in \mathbb{B}, \ s \in \mathcal{S}, a \in \mathcal{A} \tag{3.14f}$$

$$z_{swa} \in \mathbb{R}_+, \ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{3.14g}$$

$$h_{scw} \in \mathbb{B}, \ s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c); \quad F_c \in \mathbb{R}, \ c \in \mathcal{C}. \tag{3.14h}$$

Note that routing trees optimization forces some minor changes in the definition of the variables h_{scw} and in constraint (3.14c) (compared to their counterparts in (3.4)). Since it is impossible to predefine the set $\mathcal{V}'(s)$, variables h_{scw} are now defined for all $w \in \mathcal{W}(c)$ (and not for $w \in \mathcal{W}(c) \cap \mathcal{V}'(s)$). For the same reason the summation in (3.14c) is taken over all streams $s \in \mathcal{S}$ (instead of $s \in \mathcal{S}(c, w)$).

Besides the formulation presented above (which is referred to as formulation involving flows in [18]), two other formulations were presented in [18]: formulation involving aggregated flows and formulation involving node potentials. However, these formulations won't be used in the main part of this thesis, thus they are not considered here.

3.5.2 RFMP: pricing problem for c-set generation

The pricing problem for RFMP can be derived in a similar way as for SFMP. Now, the linear relaxation of the problem is obtained by relaxing the integrality constraints for variables h_{scw} and y_{sa}. As previously, it is not necessary to introduce the constraints $h_{scw} \leqslant 1$ and $y_{sa} \leqslant 1$ since such conditions will be fulfilled in optimal solutions anyway. The problem dual to such a relaxation is as follows:

$$\text{DMP}(\mathcal{C}): \ \max \ W = \textstyle\sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{D}(s)} (\varphi_{swo(s)} - \varphi_{sww}) \tag{3.15a}$$

$$\sum_{w \in D(s)} \sigma_{swa} \leqslant \lambda_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{3.15b}$$

$$\varphi_{swb(a)} - \varphi_{swe(a)} \leqslant \sigma_{swa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{3.15c}$$

$$\sum_{w \in \mathcal{W}(c)} \pi_{cw} = 1, \quad c \in \mathcal{C} \tag{3.15d}$$

$$\sum_{v \in \mathcal{U}(c,w)} \lambda_{s(w,v)} \leqslant \pi_{cw}, \quad s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{3.15e}$$

$$\varphi_{swv} \in \mathbb{R}, \; s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{3.15f}$$

$$\sigma_{swa} \in \mathbb{R}_+, \; s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{3.15g}$$

$$\lambda_{sa} \in \mathbb{R}_+, \; s \in \mathcal{S}, a \in \mathcal{A}; \;\; \pi_{cw} \in \mathbb{R}_+, \; c \in \mathcal{C}, w \in \mathcal{W}(c), \tag{3.15h}$$

where the dual variables $\lambda_{sa}, \pi_{cw}, \varphi_{swv}, \sigma_{swa}$ correspond, respectively, to the primal constraints (3.14a)–(3.14e).

We are looking for a c-set that maximize the following expression (again $\lambda = (\lambda^*_{sa})_{s \in \mathcal{S}, a \in \mathcal{A}}$ is a vector of optimal values of dual variables λ_{sa}):

$$P(c, \lambda^*) := \min_{\pi \geqslant 0, \sum_{w \in \mathcal{W}(c)} \pi_w = 1} Q(\pi; c), \tag{3.16}$$

where $\pi = (\pi_w)_{w \in \mathcal{W}(c)}$ and

$$Q(\pi; c) := \sum_{w \in \mathcal{W}(c)} \sum_{s \in \mathcal{S}} \max \big\{0, \sum_{u \in \mathcal{U}(c,w)} \lambda^*_{s(w,u)} - \pi_w \big\}. \tag{3.17}$$

The value of $P(c, \lambda^*)$ expressed in the form of a LP problem is as follows:

$$P(c, \lambda^*) = \; \min \; \sum_{w \in \mathcal{V}} \sum_{s \in \mathcal{S}} k_{sw} \tag{3.18a}$$

$$[r] \quad \sum_{w \in \mathcal{V}} \pi_w = 1 \tag{3.18b}$$

$$[g_{sw} \geqslant 0] \quad k_{sw} \geqslant \sum_{u \in \delta^+(w)} \lambda^*_{s(w,u)} Y(w,u) - \pi_w, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{3.18c}$$

$$\pi_w \in \mathbb{R}_+, \; w \in \mathcal{V}; \; k_{sw} \in \mathbb{R}_+, s \in \mathcal{S}, w \in \mathcal{V}. \tag{3.18d}$$

The dual to (3.18) takes the following form:

$$\max D = -r + \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} \big(\sum_{u \in \delta^+(w)} \lambda^*_{s(w,u)} Y(w,u)\big) g_{sw} \tag{3.19a}$$

$$\sum_{s \in \mathcal{S}} g_{sw} \leqslant r, \quad w \in \mathcal{V} \tag{3.19b}$$

$$g_{sw} \leqslant 1, \quad s \in \mathcal{S} \tag{3.19c}$$

$$r \in \mathbb{R}; \;\; g_{sw} \in \mathbb{R}_+, \quad w \in \mathcal{V}, s \in \mathcal{S}. \tag{3.19d}$$

Finally, PP for RFMP is as follows:

$$\text{PP}(\lambda^*): \quad \max P = -r + \sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{V}} \sum_{u\in\delta^+(w)} \lambda^*_{s(w,u)} U_{swu} \tag{3.20a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{3.20b}$$

$$X_w \leqslant \sum_{u\in\delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{3.20c}$$

$$X_w + \sum_{u\in\delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{3.20d}$$

$$p(w,u) + M(w,u)(1 - Y_{wu}) \geqslant \gamma(\eta + \sum_{v\in\mathcal{V}\setminus\{w,u\}} p(v,u)X_v), \quad (w,u) \in \mathcal{A} \tag{3.20e}$$

$$U_{swu} \leqslant g_{sw}, U_{swu} \leqslant Y_{wu}, \quad s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(v) \tag{3.20f}$$

$$\sum_{s\in\mathcal{S}} g_{sw} \leqslant r, \quad w \in \mathcal{V}; \quad g_{sw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{3.20g}$$

$$Y_{wu} \in \mathbb{B},\ w \in \mathcal{V}, u \in \delta^+(w);\ X_w \in \mathbb{B},\ w \in \mathcal{V} \tag{3.20h}$$

$$r \in \mathbb{R};\ g_{sw} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V};\ U_{swu} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w). \tag{3.20i}$$

Notice that the only difference between (3.20) and (3.11) is the range of variables g_{sw}, U_{swu} and the constraints imposed on these variables.

Chapter 4

Modulation and coding schemes assignment

So far we have assumed that the transmission data rate is the same for all transmitting nodes and equals one packet per time slot. However, such an assumption is not always met, since multiple modulation and coding schemes (MCS) can be used in various MHWN (e.g., see [69] for a list of MCS available for IEEE 802.11a). Further, an adaptive MCS assignment can be implemented to increase the overall capacity of the system by assigning an appropriate MCS for a given channel condition. Obviously, the higher the data rate of a given MCS, the higher the sensitivity to interferences, and thus a trade-off between data rate and spatial reuse (the number of parallel transmissions) is observed. Because of that, we cannot expect that choosing the MCS with the highest data rates will always provide the best possible solution. Actually, some interesting conclusions on that can be found in [69] where the authors pointed out that using more robust MCS instead of these with the highest data rates can lead to the improvement of the overall network throughput. This means that an appropriate assignment of MCS is not straightforward to find, and, as we show below, is in fact very complex and difficult to compute.

The main goal of this chapter is to extend the optimization model to include MCS assignments, both dynamic and static. In the first case the transmitting nodes are capable of changing MCS used by the broadcasting nodes from a time slot to a time slot of the frame while in the second case different nodes can use different MCS, but they are not allowed to

Table 4.1: Notation 2 (modulation and coding schemes).

notation	description
$\mathcal{M}$	set of MCS ($\mathcal{M} := \{1, 2, \ldots, M\}$)
$\gamma(m)$	SINR threshold for MCS m
$B(m)$	bitrate in [Mbps] for MCS m
$m(c, w)$	MCS assigned to node w in c-set c
$n(s)$	data volume in [Mb] generated at s

change MCS between time slots. However, since the problem of static MCS assignment is hardly tractable for the networks of realistic size, we present the numerical results only for the first case. The extensions provided in this chapter are applicable for RFMP. The content of this chapter is partially based on [70], however some new formulations for static MCS assignment were introduced here.

The numerical results illustrating this chapter are presented together with the results for transmission power control, at the end of the next chapter.

4.1 Additional notations

To include MCS assignment in BOM some new notations are needed.

Now, each node broadcasts a transmission signal using a MCS m selected from a given set of such schemes $\mathcal{M} = \{1, 2, \ldots, M\}$. Each scheme $m \in \mathcal{M}$ is characterized by the SINR threshold $\gamma(m) > 1$ (note that previously we have implicitly considered only one MCS, and thus only one value of the SINR threshold equal to γ), and the transmission rate $B(m)$ expressed in Mbps. It follows that when a given node $v \in \mathcal{V}$ is broadcasting using MCS m, and no other nodes are broadcasting, then its transmitted signal can be successfully decoded by all nodes $w \in \mathcal{V} \setminus \{v\}$ for which $\frac{p(v,w)}{\eta} \geqslant \gamma(m)$. That means that the definition of the set of arcs $\mathcal{A}$ in the network graph $\mathcal{G}$ (and defined with its use sets $\delta^+(w)$ and $\delta^-(w)$) should be modified, since now such a set depends on MCS used in the network nodes. To this end, we introduce a parameter $\delta^+(v, m)$ (called m-range of node v), $v \in \mathcal{W}$, $m \in \mathcal{M}$, defined as follows:

$$\delta^+(v,m) := \{w \in \mathcal{V}\backslash\{v\} : \tfrac{p(v,w)}{\eta} \geqslant \gamma(m)\}. \tag{4.1}$$

Thus, $\delta^+(v)$ (called here the node range) is defined as the m-range for the MCS the minimum $\gamma(m)$. Using the notion of the node range we can now define the set of arcs $\mathcal{A}$: $(v,w) \in \mathcal{A}$ if, and only if, $v \in \mathcal{W}$ and $w \in \delta^+(v)$. With such a definition of arcs, $\delta^+(v)$ denotes, as before, the set of nodes connected with node v by arcs outgoing from v and the symmetric set of all nodes connected with v by arcs incoming to v is defined as $\delta^-(v) := \{(w,v) \in \mathcal{A} : w \in \mathcal{V}\}$.

Further, we drop the assumption that the intensity of each network stream $s \in \mathcal{S}$ equals 1 packet per frame, and instead, we assume now that each stream s generates, at the beginning of each consecutive frame, data of volume $n(s)$ Mb.

Finally, also the c-set definition has to be modified since now a c-set is characterized not only by the set $\mathcal{W}(c)$ of (simultaneously) broadcasting nodes and the set $\mathcal{U}(c,w)$ of nodes receiving the signal from each node $w \in \mathcal{W}(c)$, but also by the MCS assignment $m(c,w), w \in \mathcal{W}(c)$ (formally $m(c,w)$ is the MCS assigned to node w in c-set c). Therefore the SINR condition (3.1) that has to be fulfilled for all $w \in \mathcal{W}(c)$ and $u \in \mathcal{U}(c,w)$ is now defined as follows:

$$\frac{p(w,u)}{\eta + \sum_{v \in \mathcal{W}(c)\backslash\{w,u\}} p(v,u)} \geqslant \gamma(m(c,w)). \tag{4.2}$$

All the above lead to the following formal c-set definition:

$$X(w) \geqslant Y(w,u), \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{4.3a}$$

$$X(w) \leqslant \textstyle\sum_{u \in \delta^+(w)} Y(w,u), \quad w \in \mathcal{V} \tag{4.3b}$$

$$X(w) + \textstyle\sum_{u \in \delta^-(w)} Y(u,w) \leqslant 1, \quad w \in \mathcal{V} \tag{4.3c}$$

$$\textstyle\sum_{m \in \mathcal{M}} z(m,w) = X(w), \quad w \in \mathcal{V} \tag{4.3d}$$

$$p(w,u) + M(w,u,m)(1 - Y(w,u))$$

$$\geqslant \gamma(m)(\eta + \textstyle\sum_{v \in \mathcal{V}\backslash\{w,u\}} p(v,u)X(v))z(m,w), \quad w \in \mathcal{V}, u \in \delta^+(w), m \in \mathcal{M}. \tag{4.3e}$$

Compared to (3.2), a new set of binary coefficients $z(m,w), m \in \mathcal{M}, w \in \mathcal{W}$, is introduced. Each binary coefficient $z(m,w)$ is equal to 1 if, and only if, node w is assigned to transmit using MCS m. Further, a new constraint (4.3d) is introduced to assure that only one MCS is assigned to each transmitting node. Finally, the SINR constraint (4.3e) is appropriately modified to take into account the use of different MCS.

4.2 Dynamic modulation and coding schemes assignment

With the new notations we can now move on to the problem formulations. We start our considerations with dynamic modulation and coding schemes assignment. As we already mentioned we extend the formulation of FRMP to include MCS assignment. At this point we should highlight that there is no point in including MCS assignment in SFMP since by predefining routing trees we can easily exclude some MCS from feasible solutions. For example, let us consider shortest-path multicast trees generated with the Dijkstra's algorithm with the hop count metric. In this case long links are more likely to be used than short ones. Clearly, this may lead to the situation where some MCS cannot be used in some nodes since their transmission range is not sufficient to send data along some links which begin at these nodes (note that the higher data rate of a given MCS m, the higher the value of $\gamma(m)$ and the shorter its transmission range, see [71]).

4.2.1 Problem formulation

To include dynamic MCS assignment the following modification has to be made in (3.14).

First, constraint (3.14b) is modified to include the fact that the intensity of each stream is not equal to 1 packet per frame anymore, but equals $n(s)$ Mb per frame:

$$\textstyle\sum_{c \in \mathcal{C}(a)} h_{scb(a)} \geqslant n(s) y_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A}. \tag{4.4a}$$

Then, we modify constraint (3.14c):

$$\textstyle\sum_{s \in \mathcal{S}} h_{scw} \leqslant B(m(c,w)) F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c). \tag{4.5a}$$

Note that by multiplying F_c by $B(m(c,w))$ we take into account that the nodes transmit with different data rates that depend on the used MCS. Notice that the same is possible also by multiplying $h_{scb(a)}$ by $B(m(c,w))$ in constraint (4.4a), however such an approach makes it impossible to transmit data from different streams in the same time slot. The rest of the constraints remain unchanged, and thus the complete formulation of RFMP with dynamic MCS assignment (referred to as DMRFMP) is as follows:

$$\text{DMRFMP}(\mathcal{C}):\quad \min\ F = \textstyle\sum_{c\in\mathcal{C}} F_c \tag{4.6a}$$

$$[\lambda_{sa} \geqslant 0]\quad \textstyle\sum_{c\in\mathcal{C}(a)} h_{scb(a)} \geqslant n(s) y_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{4.6b}$$

$$[\pi_{cw} \geqslant 0]\quad \textstyle\sum_{s\in\mathcal{S}} h_{scw} \leqslant B(m(c,w)) F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{4.6c}$$

$$[\varphi_{swv}]\quad \textstyle\sum_{u\in\delta^-(v)} z_{sw(u,v)} + I(s,w,v) = \textstyle\sum_{u\in\delta^+(v)} z_{sw(v,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{4.6d}$$

$$[\sigma_{swa} \geqslant 0]\quad z_{swa} \leqslant y_{sa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{4.6e}$$

$$y_{sa} \in \mathbb{B},\ s \in \mathcal{S}, a \in \mathcal{A} \tag{4.6f}$$

$$z_{swa} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{4.6g}$$

$$h_{scw} \in \mathbb{R}_+,\ s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c);\ \ F_c \in \mathbb{Z},\ c \in \mathcal{C}. \tag{4.6h}$$

We finally note that, as compared to (3.14), variables h_{scw} are now defined as non-negative and continuous; on the contrary, now the integrality constraints have to be imposed on variables F_c.

The dual to the linear relaxation of the above formulation (i.e., DMP used in the PP derivation) is as follows:

$$\text{DMP}(\mathcal{C}):\quad \max\ W = \textstyle\sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{D}(s)} (\varphi_{swo(s)} - \varphi_{sww}) \tag{4.7a}$$

$$\textstyle\sum_{w\in D(s)} \sigma_{swa} \leqslant n(s)\lambda_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{4.7b}$$

$$\varphi_{swb(a)} - \varphi_{swe(a)} \leqslant \sigma_{swa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{4.7c}$$

$$\textstyle\sum_{w\in\mathcal{W}(c)} B(m(c,w))\pi_{cw} = 1, \quad c \in \mathcal{C} \tag{4.7d}$$

$$\textstyle\sum_{u\in\mathcal{U}(c,w)} \lambda_{s(w,u)} \leqslant \pi_{cw}, \quad s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{4.7e}$$

$$\varphi_{swv} \in \mathbb{R},\ s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{4.7f}$$

$$\sigma_{swa} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{4.7g}$$

$$\lambda_{sa} \in \mathbb{R}_+,\ s \in \mathcal{S}, a \in \mathcal{A};\ \ \pi_{cw} \in \mathbb{R}_+,\ c \in \mathcal{C}, w \in \mathcal{W}(c). \tag{4.7h}$$

4.2.2 Pricing problem for c-set generation

In this section we present a PP relevant for c-set generation for DMRFMP. The derivation follows the steps given below, analogous to those described in Section 3.4.2.

Now, because of $B(m(c,w))$ in (4.7d), $P(c,\lambda^*)$ is expressed in a different way:

$$P(c,\lambda^*) := \min_{\pi \geqslant 0, \sum_{w\in\mathcal{W}(c)} B(m(c,w))\pi_w = 1} Q(\pi;c), \tag{4.8}$$

where $\pi = (\pi_w)_{w\in\mathcal{W}(c)}$ and

$$Q(\pi;c) := \sum_{w\in\mathcal{W}(c)} \sum_{s\in\mathcal{S}} \max\left\{0, \sum_{u\in\mathcal{U}(c,w)} \lambda^*_{s(w,u)} - \pi_w\right\}. \tag{4.9}$$

The LP formulation that is used to calculate $P(c,\lambda^*)$ is as follows:

$$P(c,\lambda^*) = \quad \min \sum_{w\in\mathcal{V}} \sum_{s\in\mathcal{S}} k_{sw} \tag{4.10a}$$

$$[r] \quad \sum_{w\in\mathcal{V}} \left(\sum_{m\in\mathcal{M}} B(m) z(m,w)\right)\pi_w = 1 \tag{4.10b}$$

$$[g_{sw} \geqslant 0] \quad k_{sw} \geqslant \sum_{u\in\delta^+(w)} \lambda^*_{s(w,u)} Y(w,u) - \pi_w, \quad s\in\mathcal{S}, w\in\mathcal{V} \tag{4.10c}$$

$$\pi_w \in \mathbb{R}_+,\ w\in\mathcal{V};\ k_{sw}\in\mathbb{R}_+, s\in\mathcal{S}, w\in\mathcal{V}. \tag{4.10d}$$

The dual problem to (4.10):

$$\max D = -r + \sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{V}} \left(\sum_{u\in\delta^+(w)} \lambda^*_{s(w,u)} Y(w,u)\right) g_{sw} \tag{4.11a}$$

$$\sum_{s\in\mathcal{S}} g_{sw} \leqslant \left(\sum_{m\in\mathcal{M}} B(m) z(m,w)\right) r, \quad w\in\mathcal{V} \tag{4.11b}$$

$$g_{sw} \leqslant 1, \quad s\in\mathcal{S} \tag{4.11c}$$

$$r\in\mathbb{R};\ \ g_{sw}\in\mathbb{R}_+,\ w\in\mathcal{V}, s\in\mathcal{S}. \tag{4.11d}$$

Finally, PP for DMRFMP is as follows:

$$\mathrm{PP}(\lambda^*): \ \max\ P = -r + \sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{V}} \sum_{u\in\delta^+(w)} \lambda^*_{s(w,u)} U_{swu} \tag{4.12a}$$

$$X_w \geqslant Y_{wu}, \quad w\in\mathcal{V}, u\in\delta^+(w) \tag{4.12b}$$

$$X_w \leqslant \sum_{u\in\delta^+(w)} Y_{wu}, \quad w\in\mathcal{V} \tag{4.12c}$$

$$X_w + \sum_{u\in\delta^-(w)} Y_{uw} \leqslant 1, \quad w\in\mathcal{V} \tag{4.12d}$$

$$\sum_{m\in\mathcal{M}} z^m_w = X_w, \quad w\in\mathcal{V} \tag{4.12e}$$

$$p(w,u) + M(w,u,m)(1-Y_{wu})$$

$$\geqslant \gamma(m) z_w^m \eta + \gamma(m) \sum_{v \in \mathcal{V} \setminus \{w,u\}} p(v,u) Z_{wv}^m, \quad w \in \mathcal{V}, u \in \delta^+(w), m \in \mathcal{M} \tag{4.12f}$$

$$Z_{wv}^m \leqslant z_w^m, \;\; Z_{wv}^m \leqslant X_v, \;\; Z_{wv}^m \geqslant z_w^m + X_v - 1, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in \mathcal{V} \setminus \{w\} \tag{4.12g}$$

$$U_{swu} \leqslant g_{sw}, U_{swu} \leqslant Y_{wu}, \quad w \in \mathcal{V}, s \in \mathcal{S}, u \in \delta^+(w) \tag{4.12h}$$

$$\sum_{s \in \mathcal{S}} g_{sw} \leqslant \sum_{m \in \mathcal{M}} B(m) R_w^m, \quad w \in \mathcal{V}; \; g_{sw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{4.12i}$$

$$R_w^m \leqslant |\mathcal{S}| z_w^m, \;\; R_w^m \leqslant r, \;\; R_w^m \geqslant r - |\mathcal{S}|(1 - z_w^m), \quad m \in \mathcal{M}, w \in \mathcal{V}. \tag{4.12j}$$

$$Y_{wu} \in \mathbb{B}, \; w \in \mathcal{V}, u \in \delta^+(w); \; X_w \in \mathbb{B}, \; w \in \mathcal{V} \tag{4.12k}$$

$$z_w^m \in \mathbb{B}, \; w \in \mathcal{V}, m \in \mathcal{M}; \; Z_{wv}^m \in \mathbb{B}, \; w \in \mathcal{V}, v \in \mathcal{V} \setminus \{w\}, m \in \mathcal{M} \tag{4.12l}$$

$$r \in \mathbb{R}; \; R_w^m \in \mathbb{R}, \; m \in \mathcal{M}, w \in \mathcal{V}; \; g_{sw} \in \mathbb{R}_+, \; s \in \mathcal{S}, w \in \mathcal{V} \tag{4.12m}$$

$$U_{swu} \in \mathbb{R}_+, \; s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w). \tag{4.12n}$$

In the above formulation, binary variables $z_w^m, m \in \mathcal{M}, w \in \mathcal{W}$ correspond to the binary coefficients $z(m, w)$ introduced in (4.3), constraint (4.12e) is the counterpart of constraint (4.3d), and (4.12f) is the counterpart of the modified SINR constraint (4.3e). The non-negative continuous variables Z_{wv}^m, and R_w^m, together with constraints (4.12g) and (4.12j), are introduced to eliminate bi-linearities that otherwise would appear because of the variables multiplication: $z_w^m \cdot X_v$ and $z_w^m \cdot r$.

4.3 Static modulation and coding schemes assignment

The second case worth considering is static modulation and coding schemes assignment (referred to as SMRFMP) where different nodes can use different MCS, but they are not allow to change MCS between time slots. Although this may appear to be a simpler problem than dynamic MCS assignment, it turns out that the formulation of static MCS assignment as well as the PP for c-set generation are in fact more complicated than their dynamic counterparts described in Section 4.2.

4.3.1 Problem formulation

To consider static MCS assignment we modify (4.6) by introducing a new set of binary variables $o_w^m, m \in \mathcal{M}, w \in \mathcal{W}$, together with additional constraints. A variable o_w^m is equal to

1 if, and only if, node w is using MCS m in the frame. Since each node is allowed to use only one MCS in the frame, the following constraint is used:

$$\sum_{m \in \mathcal{M}} o_w^m = 1, \quad w \in \mathcal{V}. \tag{4.13a}$$

Further, for each node, we have to determine which MCS it actually uses. Clearly, it is not possible to do this by checking MCS assignments $m(c,w)$ for each c-set containing a given node, since usually only a subset of all generated c-sets $\mathcal{C}$ is used within the frame (and this is determined by variables h_{scw}). Hence, we introduce the following constraint:

$$\sum_{s \in \mathcal{S}} \sum_{c \in \mathcal{C}(w,m)} h_{scw} \leqslant M o_w^m, \quad m \in \mathcal{M}, w \in \mathcal{V}, \tag{4.14a}$$

where $\mathcal{C}(w,m)$ denotes the subset of c-sets in which node w transmits with MCS m ($\mathcal{C}(w,m) := \{c \in \mathcal{C} : w \in \mathcal{W}(c) \wedge m(c,w) = m\}$) and M is a "big M" constant (equal for example to $\sum_{s \in \mathcal{S}} n(s)$) that ensure that the constraint can be satisfied when the left-hand side is greater than 0 and variable o_w^m is equal to 1. Note that it is not necessary to add a constraint that force o_w^m to be equal to 0 when MCS m is not used by node w since such a behaviour will be guaranteed by constraint (4.13a).

Considering the above we can now present the complete formulation of SMRFMP:

$$\text{SMRFMP}(\mathcal{C}): \quad \min\ F = \sum_{c \in \mathcal{C}} F_c \tag{4.15a}$$

$$[\lambda_{sa} \geqslant 0] \quad \sum_{c \in \mathcal{C}(a)} h_{scb(a)} \geqslant n(s) y_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{4.15b}$$

$$[\pi_{cw} \geqslant 0] \quad \sum_{s \in \mathcal{S}} h_{scw} \leqslant B(m(c,w)) F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{4.15c}$$

$$[\varphi_{swv}] \quad \sum_{u \in \delta^-(v)} z_{sw(u,v)} + I(s,w,v) = \sum_{u \in \delta^+(v)} z_{sw(v,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{4.15d}$$

$$[\sigma_{swa} \geqslant 0] \quad z_{swa} \leqslant y_{sa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{4.15e}$$

$$[\alpha_w] \quad \sum_{m \in \mathcal{M}} o_w^m = 1, \quad w \in \mathcal{V} \tag{4.15f}$$

$$[\beta_w^m \geqslant 0] \quad \sum_{s \in \mathcal{S}} \sum_{c \in \mathcal{C}(w,m)} h_{scw} \leqslant M o_w^m, \quad m \in \mathcal{M}, w \in \mathcal{V} \tag{4.15g}$$

$$y_{sa} \in \mathbb{B},\ s \in \mathcal{S}, a \in \mathcal{A} \tag{4.15h}$$

$$z_{swa} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{4.15i}$$

$$h_{scw} \in \mathbb{R}_+,\ s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c); \quad F_c \in \mathbb{Z},\ c \in \mathcal{C}. \tag{4.15j}$$

The dual to the linear relaxation of SMRFMP is as follows:

$$\text{DMP}(\mathcal{C}): \ \max\ W = \textstyle\sum_{s\in\mathcal{S}}\sum_{w\in\mathcal{D}(s)}(\varphi_{swo(s)} - \varphi_{sww}) \tag{4.16a}$$

$$\textstyle\sum_{w\in D(s)} \sigma_{swa} \leqslant n(s)\lambda_{sa}, \quad s\in\mathcal{S}, a\in\mathcal{A} \tag{4.16b}$$

$$\varphi_{swb(a)} - \varphi_{swe(a)} \leqslant \sigma_{swa}, \quad s\in\mathcal{S}, w\in\mathcal{D}(s), a\in\mathcal{A} \tag{4.16c}$$

$$\textstyle\sum_{w\in\mathcal{W}(c)} B(m(c,w))\pi_{cw} = 1, \quad c\in\mathcal{C} \tag{4.16d}$$

$$\textstyle\sum_{u\in\mathcal{U}(c,w)} \lambda_{s(w,u)} \leqslant \pi_{cw} + \beta_w^{m(c,w)}, \quad s\in\mathcal{S}, c\in\mathcal{C}, w\in\mathcal{W}(c) \tag{4.16e}$$

$$\alpha_w \geqslant M\beta_w^m, \quad m\in\mathcal{M}, w\in\mathcal{W}(c) \tag{4.16f}$$

$$\varphi_{swv}\in\mathbb{R},\ s\in\mathcal{S}, w\in\mathcal{D}(s), v\in\mathcal{V} \tag{4.16g}$$

$$\sigma_{swa}\in\mathbb{R}_+,\ s\in\mathcal{S}, w\in\mathcal{D}(s), a\in\mathcal{A} \tag{4.16h}$$

$$\lambda_{sa}\in\mathbb{R}_+,\ s\in\mathcal{S}, a\in\mathcal{A};\ \ \pi_{cw}\in\mathbb{R}_+,\ c\in\mathcal{C}, w\in\mathcal{W}(c) \tag{4.16i}$$

$$\alpha_w\in\mathbb{R}, \quad w\in\mathcal{W};\ \ \beta_w^m\in\mathbb{R}_+, \quad m\in\mathcal{M}, w\in\mathcal{W}. \tag{4.16j}$$

4.3.2 Pricing problem for c-set generation

The changes in the dual problem (as compared to (4.7)) require some modifications in the process of derivation of the pricing problem. First, because of the $\beta_w^{m(c,w)}$ in (4.16e), the value of $P(c,\lambda^*)$ that we use to express the minimum violations of the constraints defining dual polytope for given (feasible) values of dual variables $\pi := (\pi_w)_{w\in\mathcal{W}(c)}$ now depends also on β^*, i.e., the vector of optimal values of the dual variables β:

$$P(c,\lambda^*,\beta^*) := \min_{\pi\geqslant 0,\sum_{w\in\mathcal{W}(c)} B(m(c,w))\pi_w = 1} Q(\pi; c), \tag{4.17}$$

where $\pi = (\pi_w)_{w\in\mathcal{W}(c)}$ and

$$Q(\pi;c) := \textstyle\sum_{w\in\mathcal{W}(c)}\sum_{s\in\mathcal{S}} \max\left\{0, \sum_{u\in\mathcal{U}(c,w)} \lambda^*_{s(w,u)} - \pi_w - \beta^*_{m(c,w)w}\right\}. \tag{4.18}$$

Thus, the LP formulation that can be used to calculate $P(c,\lambda^*,\beta^*)$ is as follows:

$$P(c,\lambda^*,\beta^*) = \ \min\ \textstyle\sum_{w\in\mathcal{V}}\sum_{s\in\mathcal{S}} k_{sw} \tag{4.19a}$$

$$[r] \quad \textstyle\sum_{w\in\mathcal{V}}(\sum_{m\in\mathcal{M}} B(m)z(m,w))\pi_w = 1 \tag{4.19b}$$

$$[g_{sw}\geqslant 0] \quad k_{sw} \geqslant \textstyle\sum_{u\in\delta^+(w)} \lambda^*_{s(w,u)} Y(w,u) - \pi_w$$

$$-\sum_{m\in\mathcal{M}}\beta^*_{mw}z(m,w),\quad s\in\mathcal{S},w\in\mathcal{V} \tag{4.19c}$$

$$\pi_w\in\mathbb{R}_+,\ w\in\mathcal{V};\ k_{sw}\in\mathbb{R}_+,s\in\mathcal{S},w\in\mathcal{V}. \tag{4.19d}$$

This leads to the following problem dual to (4.19):

$$\max D=-r+\sum_{s\in\mathcal{S}}\sum_{w\in\mathcal{V}}(\sum_{u\in\delta^+(w)}\lambda^*_{s(w,u)}Y(w,u))g_{sw}$$

$$+\sum_{s\in\mathcal{S}}\sum_{w\in\mathcal{V}}(\sum_{m\in\mathcal{M}}\beta^*_{mw}z(m,w))g_{sw} \tag{4.20a}$$

$$\sum_{s\in\mathcal{S}}g_{sw}\leqslant(\sum_{m\in\mathcal{M}}B(m)z(m,w))r,\quad w\in\mathcal{V} \tag{4.20b}$$

$$g_{sw}\leqslant 1,\quad s\in\mathcal{S} \tag{4.20c}$$

$$r\in\mathbb{R};\ \ g_{sw}\in\mathbb{R}_+,\ w\in\mathcal{V},s\in\mathcal{S}. \tag{4.20d}$$

Finally, below we formulate PP for c-set generation for SMRFMP:

$$\text{PP}(\lambda^*):\ \max\ P=-r+\sum_{s\in\mathcal{S}}\sum_{w\in\mathcal{V}}\sum_{u\in\delta^+(w)}\lambda^*_{s(w,u)}U_{swu}$$

$$+\sum_{s\in\mathcal{S}}\sum_{w\in\mathcal{V}}\sum_{m\in\mathcal{M}}\beta^*_{mw}G^m_{sw} \tag{4.21a}$$

$$X_w\geqslant Y_{wu},\quad w\in\mathcal{V},u\in\delta^+(w) \tag{4.21b}$$

$$X_w\leqslant\sum_{u\in\delta^+(w)}Y_{wu},\quad w\in\mathcal{V} \tag{4.21c}$$

$$X_w+\sum_{u\in\delta^-(w)}Y_{uw}\leqslant 1,\quad w\in\mathcal{V} \tag{4.21d}$$

$$\sum_{m\in\mathcal{M}}z^m_w=X_w,\quad w\in\mathcal{V} \tag{4.21e}$$

$$p(w,u)+M(w,u,m)(1-Y_{wu})$$

$$\geqslant\gamma(m)z^m_w\eta+\gamma(m)\sum_{v\in\mathcal{V}\setminus\{w,u\}}p(v,u)Z^m_{wv},\quad w\in\mathcal{V},u\in\delta^+(w),m\in\mathcal{M} \tag{4.21f}$$

$$Z^m_{wv}\leqslant z^m_w,\ \ Z^m_{wv}\leqslant X_v,\ \ Z^m_{wv}\geqslant z^m_w+X_v-1,\quad m\in\mathcal{M},w\in\mathcal{V},v\in\mathcal{V}\setminus\{w\} \tag{4.21g}$$

$$G^m_{sw}\leqslant g_{sw},\,G^m_{sw}\leqslant z^m_w,\ \ G^m_{sw}\geqslant g_{sw}+z^m_w-1,\quad m\in\mathcal{M},s\in\mathcal{S},w\in\mathcal{V} \tag{4.21h}$$

$$U_{swu}\leqslant g_{sw},\,U_{swu}\leqslant Y_{wu},\ \ U_{swu}\geqslant g_{sw}+Y_{wu}-1,\quad w\in\mathcal{V},s\in\mathcal{S},u\in\delta^+(w) \tag{4.21i}$$

$$\sum_{s\in\mathcal{S}}g_{sw}\leqslant\sum_{m\in\mathcal{M}}B(m)R^m_w,\quad w\in\mathcal{V};\ g_{sw}\leqslant 1,\quad s\in\mathcal{S},w\in\mathcal{V} \tag{4.21j}$$

$$R^m_w\leqslant|\mathcal{S}|z^m_w,\ \ R^m_w\leqslant r,\ \ R^m_w\geqslant r-|\mathcal{S}|(1-z^m_w),\quad m\in\mathcal{M},w\in\mathcal{V}. \tag{4.21k}$$

$$Y_{wu}\in\mathbb{B},\,w\in\mathcal{V},u\in\delta^+(w);\ X_w\in\mathbb{B},\,w\in\mathcal{V} \tag{4.21l}$$

$$z^m_w\in\mathbb{B},\,w\in\mathcal{V},m\in\mathcal{M};\ Z^m_{wv}\in\mathbb{B},\,w\in\mathcal{V},v\in\mathcal{V}\setminus\{w\},m\in\mathcal{M} \tag{4.21m}$$

$$r\in\mathbb{R};\ R^m_w\in\mathbb{R},\,m\in\mathcal{M},w\in\mathcal{V};\ g_{sw}\in\mathbb{R}_+,\ s\in\mathcal{S},w\in\mathcal{V} \tag{4.21n}$$

$$U_{swu} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w);\ G^m_{sw} \in \mathbb{R}_+,\ m \in \mathcal{M}, s \in \mathcal{S}, w \in \mathcal{V}, \qquad (4.21o)$$

where variable G^m_{sw} together with constraints (4.21h) were introduced to get rid of bilinearities that otherwise would occur because of the multiplication $z^m_w \cdot g_{sw}$.

Chapter 5

Transmission power control

At this point we know how to maximize network traffic throughput in MHWN taking into account routing trees and MCS assignment optimization. However, in all previous cases we assumed that the transmission power is fixed and the same for all nodes. In this chapter we drop this assumption and include the final aspect in traffic throughput optimization, i.e, transmission power control (TPC). We take the MIP formulations for dynamic and static MCS assignment and we extend them to both the discrete and continuous TPC cases. In the former case the transceivers can use only a finite, usually small, set of power levels (this is common when it comes to low-cost sensor nodes). In the latter case the transmissions power applied by the nodes can change continuously.

It should be highlighted that when it comes to TPC the primal problems remains unchanged and the differences appear only in the pricing problems. In the following we present pricing problems for four cases:

- discrete TPC with dynamic MCS assignment
- discrete TPC with static MCS assignment
- continuous TPC with dynamic MCS assignment
- continuous TPC with static MCS assignment.

Finally, at the end of this chapter, we present a numerical study illustrating the considerations of this chapter and Chapter 4. Because of the complexity of the static MCS assignment, we consider only dynamic MCS assignment case. However, note that the results for the dynamic case can be used as a performance upper bound of the static case.

This chapter is based on paper [72] which is extended by formulations for both discrete and continuous TPC with static MCS assignment.

5.1 Discrete transmission power control

In the discrete TPC case, we assume that in each time slot each network node can transmit with the transmission power level from a discrete set of predefined power levels that is denoted by $\mathcal{P}$. Compared to the formulations without TPC, in the following we use a new set of binary variables, namely h_w^p, $p \in \mathcal{P}, w \in \mathcal{V}$, where h_w^p equals 1 if, and only if, node w transmits at power level p, and 0 otherwise. In consequence, new constraints ((5.1f) and (5.2f), respectively for dynamic and static MCS assignment case) are introduced to ensure that each transmitting node is assigned exactly one transmission power level from $\mathcal{P}$. Finally, the SINR constraint is modified in the following way: $p(w,v)$ is replaced by multiplying power $\mathcal{P}(p)$ (corresponding to the power level p assigned to node w that is identified by variable h_w^p) by the path gain denoted by $G(w,v)$. Finally, binary variables $Z_{wv}^{mp} \in \mathbb{B}$, $m \in \mathcal{M}, p \in \mathcal{P}, w \in \mathcal{V}, v \in V\backslash\{w\}$, together with additional constraints, are introduced to eliminate bi-linearities that otherwise occur because of the multiplication $h_v^p \cdot z_w^m$.

5.1.1 Dynamic modulation and coding schemes assignment

The pricing problem for discrete TPC with dynamic MCS assignment is as follows:

$$\max \;\; P = -r + \textstyle\sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{V}} \sum_{u\in\delta^+(w)} \lambda^*_{s(w,u)} U_{swu} \tag{5.1a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{5.1b}$$

$$X_w \leqslant \textstyle\sum_{u\in\delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{5.1c}$$

$$X_w + \textstyle\sum_{u\in\delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{5.1d}$$

$$\textstyle\sum_{m\in\mathcal{M}} z_w^m = X_w, \quad w \in \mathcal{V} \tag{5.1e}$$

$$\textstyle\sum_{p\in\mathcal{P}} h_w^p = X_w, \quad w \in \mathcal{V} \tag{5.1f}$$

$$\textstyle\sum_{p\in\mathcal{P}} h_w^p \mathcal{P}(p) G(w,u) + M(w,u,m)(1 - Y_{wu})$$

$$\geqslant \gamma(m) z_w^m \eta + \gamma(m) \sum_{v \in \mathcal{V} \setminus \{w,u\}} \sum_{p \in \mathcal{P}} \mathcal{P}(p) G(v,u) Z_{wv}^{mp}, \quad (w,u) \in \mathcal{A}, m \in \mathcal{M} \tag{5.1g}$$

$$Z_{wv}^{mp} \leqslant z_w^m,\ Z_{wv}^{mp} \leqslant h_v^p,\ Z_{wv}^{mp} \geqslant h_v^p + z_w^m - 1, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V \setminus \{w\}, p \in \mathcal{P} \tag{5.1h}$$

$$U_{swu} \leqslant g_{sw},\ U_{swu} \leqslant Y_{wu}, \quad s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(v) \tag{5.1i}$$

$$\sum_{s \in \mathcal{S}} g_{sw} \leqslant \sum_{m \in \mathcal{M}} B(m) R_w^m, \quad w \in \mathcal{V};\ \ g_{sw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{5.1j}$$

$$R_w^m \leqslant |\mathcal{S}| z_w^m,\ R_w^m \leqslant r, R_w^m \geqslant r - |\mathcal{S}|(1 - z_w^m), \quad m \in \mathcal{M}, w \in \mathcal{V} \tag{5.1k}$$

$$Y_{wu} \in \mathbb{B},\ w \in \mathcal{V}, u \in \delta^+(w);\ X_w \in \mathbb{B},\ w \in \mathcal{V} \tag{5.1l}$$

$$z_w^m \in \mathbb{B},\ m \in \mathcal{M}, w \in \mathcal{V};\ h_v^p \in \mathbb{B},\ p \in \mathcal{P}, v \in \mathcal{V} \tag{5.1m}$$

$$r \in \mathbb{R};\ g_{sw} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V};\ R_w^m \in \mathbb{R},\ m \in \mathcal{M}, w \in \mathcal{V} \tag{5.1n}$$

$$Z_{wv}^{mp} \in \mathbb{B},\ m \in \mathcal{M}, p \in \mathcal{P}, w \in \mathcal{V}, v \in V \setminus \{w\};\ U_{swu} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w). \tag{5.1o}$$

5.1.2 Static modulation and coding schemes assignment

The pricing problem for discrete TPC with static MCS assignment is as follows:

$$\max\ P = -r + \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} \sum_{u \in \delta^+(w)} \lambda_{s(w,u)}^* U_{swu} + \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} \sum_{m \in \mathcal{M}} \beta_{mw}^* G_{sw}^m \tag{5.2a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{5.2b}$$

$$X_w \leqslant \sum_{u \in \delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{5.2c}$$

$$X_w + \sum_{u \in \delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{5.2d}$$

$$\sum_{m \in \mathcal{M}} z_w^m = X_w, \quad w \in \mathcal{V} \tag{5.2e}$$

$$\sum_{p \in \mathcal{P}} h_w^p = X_w, \quad w \in \mathcal{V} \tag{5.2f}$$

$$\sum_{p \in \mathcal{P}} h_w^p \mathcal{P}(p) G(w,u) + M(w,u,m)(1 - Y_{wu})$$

$$\geqslant \gamma(m) z_w^m \eta + \gamma(m) \sum_{v \in \mathcal{V} \setminus \{w,u\}} \sum_{p \in \mathcal{P}} \mathcal{P}(p) G(v,u) Z_{wv}^{mp}, \quad (w,u) \in \mathcal{A}, m \in \mathcal{M} \tag{5.2g}$$

$$Z_{wv}^{mp} \leqslant z_w^m,\ Z_{wv}^{mp} \leqslant h_v^p,\ Z_{wv}^{mp} \geqslant h_v^p + z_w^m - 1, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V \setminus \{w\}, p \in \mathcal{P} \tag{5.2h}$$

$$G_{sw}^m \leqslant g_{sw},\ G_{sw}^m \leqslant z_w^m,\ \ G_{sw}^m \geqslant g_{sw} + z_w^m - 1, \quad m \in \mathcal{M}, s \in \mathcal{S}, w \in \mathcal{V} \tag{5.2i}$$

$$U_{swu} \leqslant g_{sw},\ U_{swu} \leqslant Y_{wu},\ \ U_{swu} \geqslant g_{sw} + Y_{wu} - 1, \quad w \in \mathcal{V}, s \in \mathcal{S}, u \in \delta^+(w) \tag{5.2j}$$

$$\sum_{s \in \mathcal{S}} g_{sw} \leqslant \sum_{m \in \mathcal{M}} B(m) R_w^m, \quad w \in \mathcal{V};\ g_{sw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{5.2k}$$

$$R_w^m \leqslant |\mathcal{S}| z_w^m,\ \ R_w^m \leqslant r,\ \ R_w^m \geqslant r - |\mathcal{S}|(1 - z_w^m), \quad m \in \mathcal{M}, w \in \mathcal{V}. \tag{5.2l}$$

$$Y_{wu} \in \mathbb{B},\ w \in \mathcal{V}, u \in \delta^+(w);\ X_w \in \mathbb{B},\ w \in \mathcal{V} \tag{5.2m}$$

$$z_w^m \in \mathbb{B},\ m \in \mathcal{M}, w \in \mathcal{V};\ h_v^p \in \mathbb{B},\ p \in \mathcal{P}, v \in \mathcal{V} \tag{5.2n}$$

$$r \in \mathbb{R};\ g_{sw} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V};\ R_w^m \in \mathbb{R},\ m \in \mathcal{M}, w \in \mathcal{V} \tag{5.2o}$$

$$Z_{wv}^{mp} \in \mathbb{B},\ m \in \mathcal{M}, p \in \mathcal{P}, w \in \mathcal{V}, v \in V \backslash \{w\};\ U_{swu} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w) \tag{5.2p}$$

$$G_{sw}^m \in \mathbb{R}_+,\ m \in \mathcal{M}, s \in \mathcal{S}, w \in \mathcal{V}. \tag{5.2q}$$

5.2 Continuous transmission power control

In the continuous power control the transmission power that is applied by the nodes can change continuously. Hence, instead of binary variables h_w^p used previously, the continuous variables $p_w, w \in \mathcal{V}$, are now introduced to express the actual power that is applied at node w. These variables are used in the SINR constraint modified analogously to the discrete case. It is assumed that the value of p_w cannot be smaller than the minimum power $\check{P}$ (a parameter), and it cannot exceed the maximum power $\widehat{P}$, which is assured by new constraints ((5.3g) and (5.4g), respectively for dynamic and static MCS assignment case). Variables P_{wv}^m, together with auxiliary constraints are used to get rid of bi-linearities that otherwise appear in the SINR constraint because of the multiplication $z_w^m \cdot p_v$.

5.2.1 Dynamic modulation and coding schemes assignment

The pricing problem for continuous TPC with dynamic MCS assignment is as follows:

$$\max\ P = -r + \textstyle\sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} \sum_{u \in \delta^+(w)} \lambda^*_{s(w,u)} U_{swu} \tag{5.3a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{5.3b}$$

$$X_w \leqslant \textstyle\sum_{u \in \delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{5.3c}$$

$$X_w + \textstyle\sum_{u \in \delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{5.3d}$$

$$\textstyle\sum_{m \in \mathcal{M}} z_w^m = X_w, \quad w \in \mathcal{V} \tag{5.3e}$$

$$p_w G(w,u) + M(w,u,m)(1 - Y_{wu})$$

$$\geqslant \gamma(m) z_w^m \eta + \gamma(m) \textstyle\sum_{v \in \mathcal{V} \backslash \{w,u\}} G(v,u) P_{wv}^m, \quad (w,u) \in \mathcal{A}, m \in \mathcal{M} \tag{5.3f}$$

$$\check{P} X_w \leqslant p_w \leqslant \widehat{P} X_w, \quad w \in \mathcal{V} \tag{5.3g}$$

$$P_{wv}^m \leqslant \widehat{P} z_w^m, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V \backslash \{w\} \tag{5.3h}$$

$$P^m_{wv} \leqslant p_v, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V\backslash\{w\} \tag{5.3i}$$

$$P^m_{wv} \geqslant p_v - (1 - z^m_w)\widehat{P}, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V\backslash\{w\} \tag{5.3j}$$

$$U_{swu} \leqslant g_{sw}, U_{swu} \leqslant Y_{wu}, \quad s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(v) \tag{5.3k}$$

$$\textstyle\sum_{s\in\mathcal{S}} g_{sw} \leqslant \sum_{m\in\mathcal{M}} B(m)R^m_w, \quad w \in \mathcal{V}; \;\; g_{sw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{5.3l}$$

$$R^m_w \leqslant |\mathcal{S}|z^m_w, \; R^m_w \leqslant r, R^m_w \geqslant r - |\mathcal{S}|(1 - z^m_w), \quad m \in \mathcal{M}, w \in \mathcal{V} \tag{5.3m}$$

$$Y_{wv} \in \mathbb{B}, \; w \in \mathcal{V}, v \in \delta^+(w); \;\; X_w \in \mathbb{B}, \; w \in \mathcal{V} \tag{5.3n}$$

$$z^m_w \in \mathbb{B}, \; m \in \mathcal{M}, w \in \mathcal{V}; \;\; p_w \in \mathbb{R}_+, \; w \in \mathcal{V}; \;\; P^m_{wv} \in \mathbb{R}_+, \; m \in \mathcal{M}, w \in \mathcal{V}, v \in V\backslash\{w\} \tag{5.3o}$$

$$r \in \mathbb{R}; \;\; g_{sw} \in \mathbb{R}_+, \; s \in \mathcal{S}, w \in \mathcal{V}; \;\; R^m_w \in \mathbb{R}, \; m \in \mathcal{M}, w \in \mathcal{V} \tag{5.3p}$$

$$U_{swu} \in \mathbb{R}_+, \; s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w). \tag{5.3q}$$

5.2.2 Static modulation and coding schemes assignment

The pricing problem for continuous TPC with static MCS assignment is as follows:

$$\max \;\; P = -r + \textstyle\sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{V}} \sum_{u\in\delta^+(w)} \lambda^*_{s(w,u)} U_{swu} + \sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{V}} \sum_{m\in\mathcal{M}} \beta^*_{mw} G^m_{sw} \tag{5.4a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{5.4b}$$

$$X_w \leqslant \textstyle\sum_{u\in\delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{5.4c}$$

$$X_w + \textstyle\sum_{u\in\delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{5.4d}$$

$$\textstyle\sum_{m\in\mathcal{M}} z^m_w = X_w, \quad w \in \mathcal{V} \tag{5.4e}$$

$$p_w G(w,u) + M(w,u,m)(1 - Y_{wu})$$

$$\geqslant \gamma(m) z^m_w \eta + \gamma(m) \textstyle\sum_{v\in\mathcal{V}\backslash\{w,u\}} G(v,u) P^m_{wv}, \quad (w,u) \in \mathcal{A}, m \in \mathcal{M} \tag{5.4f}$$

$$\check{P} X_w \leqslant p_w \leqslant \widehat{P} X_w, \quad w \in \mathcal{V} \tag{5.4g}$$

$$P^m_{wv} \leqslant \widehat{P} z^m_w, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V\backslash\{w\} \tag{5.4h}$$

$$P^m_{wv} \leqslant p_v, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V\backslash\{w\} \tag{5.4i}$$

$$P^m_{wv} \geqslant p_v - (1 - z^m_w)\widehat{P}, \quad m \in \mathcal{M}, w \in \mathcal{V}, v \in V\backslash\{w\} \tag{5.4j}$$

$$G^m_{sw} \leqslant g_{sw}, \; G^m_{sw} \leqslant z^m_w, \;\; G^m_{sw} \geqslant g_{sw} + z^m_w - 1, \quad m \in \mathcal{M}, s \in \mathcal{S}, w \in \mathcal{V} \tag{5.4k}$$

$$U_{swu} \leqslant g_{sw}, U_{swu} \leqslant Y_{wu}, \;\; U_{swu} \geqslant g_{sw} + Y_{wu} - 1, \quad w \in \mathcal{V}, s \in \mathcal{S}, u \in \delta^+(w) \tag{5.4l}$$

$$\textstyle\sum_{s\in\mathcal{S}} g_{sw} \leqslant \sum_{m\in\mathcal{M}} B(m)R^m_w, \quad w \in \mathcal{V}; \;\; g_{sw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{5.4m}$$

$$R_w^m \leqslant |\mathcal{S}|z_w^m, \quad R_w^m \leqslant r, \quad R_w^m \geqslant r - |\mathcal{S}|(1 - z_w^m), \qquad m \in \mathcal{M}, w \in \mathcal{V}. \tag{5.4n}$$

$$Y_{wu} \in \mathbb{B},\ w \in \mathcal{V}, u \in \delta^+(w);\ X_w \in \mathbb{B},\ w \in \mathcal{V} \tag{5.4o}$$

$$z_w^m \in \mathbb{B},\ m \in \mathcal{M}, w \in \mathcal{V};\ p_w \in \mathbb{R}_+,\ w \in \mathcal{V};\ P_{wv}^m \in \mathbb{R}_+,\ m \in \mathcal{M}, w \in \mathcal{V}, v \in V\backslash\{w\} \tag{5.4p}$$

$$r \in \mathbb{R};\ g_{sw} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V};\ R_w^m \in \mathbb{R},\ m \in \mathcal{M}, w \in \mathcal{V} \tag{5.4q}$$

$$U_{swu} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w);\ G_{sw}^m \in \mathbb{R}_+,\ m \in \mathcal{M}, s \in \mathcal{S}, w \in \mathcal{V}. \tag{5.4r}$$

5.3 Numerical Study

Below we present numerical results obtained by means of the optimization approach described in this and the previous chapter. Although, as already mentioned, we focus on the dynamic MCS assignment case combined with both discrete and continuous TPC, these results are useful for the static case, since they provide upper bounds on the static case. The aim of this study is to answer the question to what extent the network traffic throughput can be increased through joint MCS assignment and TPC optimization. Besides, we discuss the computation time efficiency of the presented optimization approach. The results presented here are taken from [72].

All of the optimization problems considered in this study were implemented and solved by means of C# and CPLEX 12.9.0, respectively. All computations were executed on an Intel Core i7-8550U CPU (four cores, each up to 4 GHz) with 16 GB RAM.

5.3.1 Network Setting

In the study we consider irregular networks generated by [73] of three different sizes: small networks with 18 nodes, medium networks with 24 nodes, and large networks with 30 nodes. For each network, the nodes were placed in a given square area according to the uniform distribution. The size of the network area is equal to 220 m × 220 m, 250 m × 250 m, and 280 m × 280 m, respectively for small, medium, and large networks. Each small network has 2 source nodes and 10 destinations, each medium network has 4 source nodes and 16 destinations, while each large network has 6 source nodes and 22 destinations; all of the remaining nodes in each network serve as purely transit nodes. Figures 5.1–5.3 show the

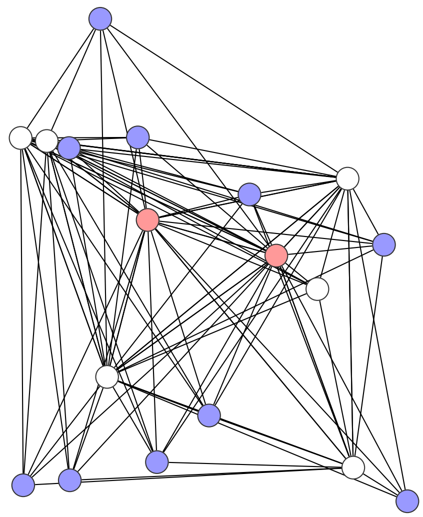

Figure 5.1: A small network example.

example networks for each size. The source nodes are depicted in red, destination nods in blue, and transit nodes in white.

In the study, we assume that the noise power η is equal to -101 dBm and that the network nodes are operating in 5 GHz band. The power received at the nodes is calculated by means of the simplified path-loss model from [74]:

$$p(w,u) = P \cdot G(w,u) = P\Big(\frac{\lambda}{4\pi d_0}\Big)^2 \Big(\frac{d_0}{d}\Big)^\alpha, \tag{5.5}$$

where $p(w,u)$ (expressed in mW) is the power received by node u when node w is broadcasting, P is the transmission power applied at node w, d_0 (in m) is a reference distance, d (in m) is the distance between w and u, λ (in m) is the signal wavelength, and α is the path-loss exponent. In the study, we assume $d_0 = 10$ m and $\alpha = 4$. As mentioned above, the transceivers are operating in 5 GHz band and, thus, $\lambda = 0.06$ m.

Clearly, the maximum transmission range depends on the transmission power and the MCS used by the broadcasting node. In the following, we assume that the set $\mathcal{M}$ is composed of three MCS: BPSK 3/4, 16-QAM 1/2, and 16-QAM 3/4. The power ratio thresholds $\gamma(m)$ and transmission rates $B(m)$ of these MCS are, respectively, equal to 6.5 dB and 12 Mbps, 12.8 dB and 18 Mbps, and 16.2 dB and 24 Mbps [71]. For the assumed MCS and the maximum transmission power considered in this study equal to 130 mW the maximum transmission range equals 170 m.

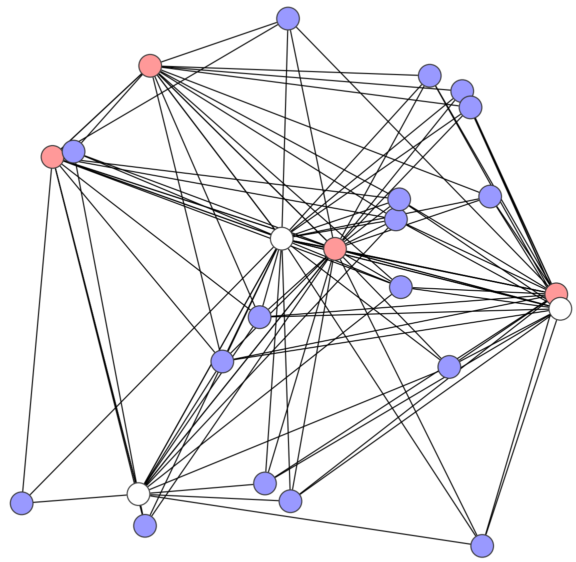

Figure 5.2: A medium network example.

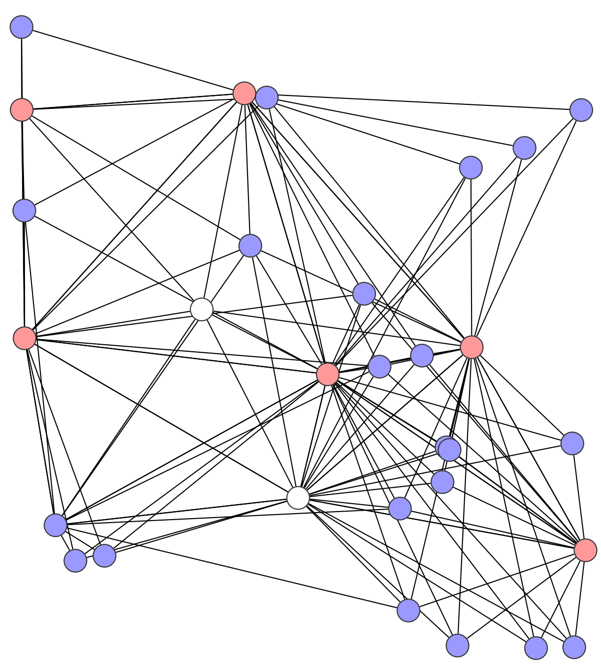

Figure 5.3: A large network example.

5.3.2 Joint Modulation and Coding Schemes Assignment and Transmission Power Control

In this section we answer the question to what extent traffic throughput can be increased by joint MCS assignment and TPC optimization. In the study we focus on the following four cases differing in the number of MCS available for each node and the type of TPC applied:

- A: one selected MCS from $\mathcal{M}$, namely BPSK 3/4, and no TPC; the transmission power at each node equals 90 mW,
- B: all three MCS from $\mathcal{M}$ and no TPC; the transmission power at each node equals 90 mW,
- C: all three MCS from $\mathcal{M}$ and discrete TPC; the transmission power at each node can take a value from the set {50 mW, 90 mW, 130 mW},
- D: all three MCS from $\mathcal{M}$ and continuous TPC; the transmission power at each node can take the values from the interval [50 mW, 130 mW].

For additional results, more comprehensively presenting the impact of the MCS assignment, the reader is referred to [70].

The frame lengths obtained for the considered cases are presented in Tables 5.1–5.3, respectively for small, medium, and large networks. Each table contains the results for 10 randomly generated networks of a given size, as well as the results that are averaged over all 10 network instances. All of the frame lengths reported below are expressed in the number of time slots. The averaged results are illustrated in Figure 5.4.

The results show that a significant gain (with respect to the case where only one MCS is applied, i.e., case A) can be achieved where all three available MCS are applied (case B). Regarding the averaged results, the value of this gain is equal to 32.5%, 33.5%, and 19.9% for small, medium, and large networks, respectively.

The frame length can be further decreased by applying TPC; however, it is not crucial whether it is discrete or continuous TPC. For small networks, the average frame length after applying discrete TPC is further decreased by 22.2% and by 23% after applying continuous

Table 5.1: Optimized frame lengths for small networks.

	A	**B**	**C**	**D**
Network 1	54	35	31	31
Network 2	41	19	14	14
Network 3	41	31	32	32
Network 4	68	50	33	33
Network 5	68	43	31	31
Network 6	95	70	41	41
Network 7	41	25	23	20
Network 8	70	43	36	36
Network 9	28	21	14	14
Network 10	41	32	32	32
Avg	54.7	36.9	28.7	28.4

Table 5.2: Optimized frame lengths for medium networks.

	A	**B**	**C**	**D**
Network 1	110	84	79	78
Network 2	97	84	63	63
Network 3	109	64	52	51
Network 4	176	132	98	97
Network 5	121	64	51	50
Network 6	95	49	44	44
Network 7	110	62	52	50
Network 8	148	100	63	61
Network 9	122	91	95	88
Network 10	135	83	60	58
Avg	122.3	81.3	65.7	64

Table 5.3: Optimized frame lengths for large networks.

	A	B	C	D
Network 1	162	133	118	116
Network 2	242	194	122	121
Network 3	149	107	108	107
Network 4	188	141	124	119
Network 5	215	194	125	121
Network 6	202	126	114	111
Network 7	190	137	125	121
Network 8	254	218	144	144
Network 9	175	239	105	101
Network 10	281	258	177	173
Avg	205.8	164.7	126.2	123.4

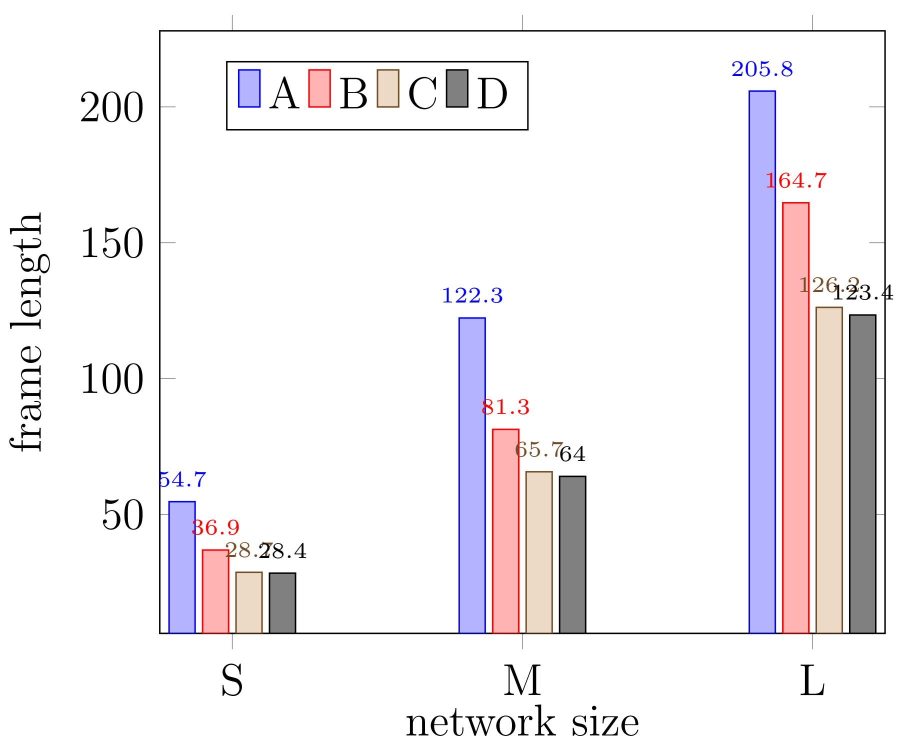

Figure 5.4: Frame length as a function of the network size for TPC cases A,B,C,D.

TPC (with respect to case B). Notice also that in nine out of ten small networks, it does not matter which type of TPC we choose—a difference exists only for Network 7.

The difference is also not significant for medium networks. When compared to case B (three MCS, no TPC), the averaged frame length can be decreased by 19.2% by applying discrete TPC and by 21,3% applying continuous TPC. However, contrary to the previous case, there are small differences between discrete and continuous TPC for most networks.

A similar, again not significant, difference between discrete and continuous TPC is observed for large networks. In this case, the averaged frame length is decreased by 23.1% by applying discrete TPC and by 25.1% when continuous TPC is applied. It should also be noticed that the influence of TPC strongly depends on the network topology (compare, for example, the results obtained for large Network 8 and large Network 9).

Finally, we should highlight a substantial difference between case A (no power control with single MCS) and case D (continuous power with multiple MCS), which is equal to 48,1%, 47.7%, and 40.1% for small, medium, and large networks, respectively, when it comes to the averaged results.

5.3.3 Time Efficiency of the Optimization Model

In this section we analyze computation times of the solution algorithm based on the CPLEX solver applied in our optimization model. Table 5.4 summarizes the notations used for this purpose.

Tables 5.5–5.7 present the results averaged over all networks of a given size that illustrate time efficiency of the optimization model for small, medium, and large networks, and all TPC cases (A – D). As before, the frame length is expressed in the number of time slots (note that in the case of the linear relaxation of the problem, the number of time slots is not necessarily integer).

We first note that all problem variants were solved in a reasonable time (also for large networks). Yet, the time needed to solve a particular problem increases significantly with the network size. Next, including MCS assignment and TPC leads to a substantial increase of the number of generated c-sets. This suggests that it is the increased number of simultaneous transmissions possible that lead to the gain in traffic throughput discussed in the previous

Table 5.4: Numerical results—explanation.

notation	description
F^{LR}	optimal frame length obtained from the linear relaxation of the problem after the c-sets generation process
F^{MIP}	optimal frame length obtained from frame length minimization
$\lvert\mathcal{C}\rvert$	number of generated c-sets
t^{MP}	total computation time of solving master problem during the c-sets generation process
t^{PP}	total computation time of solving pricing problem during the c-sets generation process
t^{MIP}	computation time of solving the final MIP version of the problem

Table 5.5: Optimization model efficiency-small networks.

	A	**B**	**C**	**D**
F^{LR}	91.09	65.89	50.50	49.86
F^{MIP}	122.3	81.3	65.7	64
$\lvert\mathcal{C}\rvert$	28.4	62.2	61.6	71.3
t^{MP}	22.4s	47.7s	563s	1m6s
t^{PP}	30.2s	6m56s	16m57s	14m44s
t^{MIP}	4m21s	3m37s	1m46s	4m44s

Table 5.6: Optimization model efficiency-medium networks.

	A	**B**	**C**	**D**
F^{LR}	91.09	65.89	50.50	49.86
F^{MIP}	122.3	81.3	65.7	64
$\lvert\mathcal{C}\rvert$	28.4	62.2	61.6	71.3
t^{MP}	22.4s	47.7s	563s	1m6s
t^{PP}	30.2s	6m56s	16m57s	14m44s
t^{MIP}	4m21s	3m37s	1m46s	4m44s

Table 5.7: Optimization model efficiency-large networks.

	A	**B**	**C**	**D**
F^{LR}	167.92	140.82	101.43	100.32
F^{MIP}	205.8	164.7	126.2	123.4
$\lvert\mathcal{C}\rvert$	41.5	79.8	86	103.7
t^{MP}	1m7s	2m17s	2m52s	3m35s
t^{PP}	41s	12m21s	23m51s	25m15s
t^{MIP}	12m42s	4m44s	12m47s	13m47s

section. Finally, the quality of linear relaxation is not high, contrary to the basic optimization model from [18]. This issue should be addressed in the future work, since it affects the effectiveness of the B&B algorithm underlying the CPLEX solver.

Chapter 6

Energy consumption minimization

The considerations of the previous chapters deal with different variants of traffic throughput maximization in MHWN. As already mentioned, it is not only traffic throughput that is important in MHWN, but also such aspects as packet delay or the total network energy consumption. In this chapter we consider the energy consumption issue. We present our assumption on the energy consumption model applied in this chapter, and then introduce an optimization problem aiming at energy consumption minimization as well as the frame minimization problem with constrained energy. Finally, we illustrate considerations of this chapter with a numerical study. For simplicity, in this chapter we do not cover MCS assignment nor TPC. However, these aspects could be easily included using the methods developed in the previous chapters. It should be noted, that, as in SFMP, we again assume that the intensity of each stream $s \in \mathcal{S}$ is equal to 1 packet per frame.

6.1 Energy consumption model

Below we make use of the node energy model introduced in [75] applied to energy consumed in transceivers operation. According to [75] such an energy consumption can be modeled as the sum of the energy consumed in a given state of a transceiver and the energy consumed during state transitions. In the following we focus on the first component, i.e., we consider only the energy consumed in a given state of a transceiver. Six transceiver states are assumed in that model: transmitting, receiving, off, idle, sleep, and CCA/ED (Clear Channel Assessment/Energy Detection). Notice that since we assume TDMA, the last state can be omitted. Also, we can assume that when a node neither transmits nor receives it is in sleep mode. Since the energy consumed in sleep mode is negligible (compared to the energy consumed in transmitting and receiving states), in the following we compute the total energy consumption as the total energy consumed by all nodes when transmitting the packets and the energy consumed by all nodes when receiving the packets. Note that the total value of the energy consumption (i.e., the value including the energy consumed also in the sleeping mode) can be easily obtained in postprocessing. In the following we assume that the energy needed to transmit a packet along a link is the same for each node and denoted by E_T. Similarly, the energy needed to receive a packet is the same for each node and denoted by E_R.

6.2 Energy consumption minimization problem

In this section we introduce the energy consumption minimization problem ECMP that aims at minimizing the total energy consumption. Note that since we express the total energy consumption as the sum of the energy needed to transmit packets and the energy needed to receive packets there is no point in considering fixed and predefined routing trees (the number of packets transmissions and the number of packets receptions is fixed in this case and there will be nothing to optimize). Because of that, the ECMP problem considered in this section includes routing trees optimization.

ECMP can be formally formulated as a MIP problem as follows (the meaning of all variables is the same as in the previous formulations):

$$\text{ECMP}(\mathcal{C}): \ \min \ E = \textstyle\sum_{c\in\mathcal{C}} \sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{W}(c)} E_T h_{scw} + \sum_{s\in\mathcal{S}} \sum_{a\in\mathcal{A}} E_R y_{sa} \tag{6.1a}$$

$$\textstyle\sum_{c\in\mathcal{C}(a)} h_{scb(a)} \geqslant y_{sa}, \quad s\in\mathcal{S}, a\in\mathcal{A} \tag{6.1b}$$

$$\textstyle\sum_{u\in\delta^-(v)} z_{sw(u,v)} + I(s,w,v) = \sum_{u\in\delta^+(v)} z_{sw(v,u)}, \quad s\in\mathcal{S}, w\in\mathcal{D}(s), v\in\mathcal{V} \tag{6.1c}$$

$$z_{swa} \leqslant y_{sa}, \quad s\in\mathcal{S}, w\in\mathcal{D}(s), a\in\mathcal{A} \tag{6.1d}$$

$$y_{sa}\in\mathbb{B}, \ s\in\mathcal{S}, a\in\mathcal{A} \tag{6.1e}$$

$$z_{swa}\in\mathbb{R}_+, \ s\in\mathcal{S}, w\in\mathcal{D}(s), a\in\mathcal{A} \tag{6.1f}$$

$$h_{scw} \in \mathbb{B}, \; s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c). \tag{6.1g}$$

Note that constraint (6.1b) is identical to constraint (3.14b) from SFMP, and since it assures that all necessary transmissions are scheduled, it is included in ECMP. Further, since in ECMP we do not minimize the frame length, constraint (3.14c) and variables $F_c, c \in \mathcal{C}$, from SFMP may now be skipped. Notice that the frame composition can still be obtained in postprocessing using variables h_{scw}. Constraints (6.1c)-(6.1d) are the flow constraints defining the multicast routing trees. Finally, objective function (6.1a) minimizes the total energy consumption expressed as the sum of the energy needed to transmit the packets and the energy needed to receive the packets. Variables h_{scw} in the first component of the sum are used to calculate how many times a given node broadcasts in the frame. To calculate the second component we need to determine the total number of time slots during which nodes are receiving; we do this by using variables y_{sa}.

An important difference between the minimum frame composition problem SFMP and energy consumption minimization problem is that now we do not need to perform c-set generation. Although formulation is still non-compact, in this case any initial family of c-sets providing a feasible solution is also sufficient to provide an optimal solution. That is because it is the number of packets transmissions and packets receptions that matters now and that means that the simultaneous transmissions do not influence the optimal solutions.

Finally, let us notice that although ECMP allows for achieving the minimum possible energy consumption, ECMP does not control the frame length, and in consequence energy consumption minimization can increase the frame length. This important issue is addressed in the next section.

6.3 Frame minimization problem with constrained energy

In this section we introduce the frame minimization problem with constrained energy RFMPE so that we could minimize the frame length while keeping the energy consumption on a given satisfactory level (for example by using the value obtained by solving ECMP).

6.3.1 Problem formulation

To keep energy consumption under an assumed fixed level while minimizing the frame length we need to add to RFMP (frame minimization problem with routing optimization) constraint (6.2d). Parameter $\widehat{E}$ is the upper bound on the total energy consumption while the left-hand side of this constraint is the expression we minimize in (6.1), i.e., the total energy consumption. The complete formulation of RFMPE is as follows:

$$\text{RFMPE}(\mathcal{C}): \quad \min \ F = \textstyle\sum_{c \in \mathcal{C}} F_c \tag{6.2a}$$

$$[\lambda_{sa} \geqslant 0] \quad \textstyle\sum_{c \in \mathcal{C}(a)} h_{scb(a)} \geqslant y_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{6.2b}$$

$$[\pi_{cw} \geqslant 0] \quad \textstyle\sum_{s \in \mathcal{S}} h_{scw} \leqslant F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{6.2c}$$

$$[\alpha \geqslant 0] \quad \textstyle\sum_{c \in \mathcal{C}} \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{W}(c)} E_T h_{scw} + \sum_{s \in \mathcal{S}} \sum_{a \in \mathcal{A}} E_R y_{sa} \leqslant \widehat{E} \tag{6.2d}$$

$$[\varphi_{swv}] \quad \textstyle\sum_{u \in \delta^-(v)} z_{sw(u,v)} + I(s,w,v)$$

$$= \textstyle\sum_{u \in \delta^+(v)} z_{sw(v,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{6.2e}$$

$$[\sigma_{swa} \geqslant 0] \quad z_{swa} \leqslant y_{sa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{6.2f}$$

$$y_{sa} \in \mathbb{B}, \ s \in \mathcal{S}, a \in \mathcal{A} \tag{6.2g}$$

$$z_{swa} \in \mathbb{R}_+, \ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{6.2h}$$

$$h_{scw} \in \mathbb{B}, \ s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c); \quad F_c \in \mathbb{R}, \ c \in \mathcal{C}. \tag{6.2i}$$

The dual to the linear relaxation of (6.2) is thus as follows:

$$\text{DMP}(\mathcal{C}): \quad \max \ W = \textstyle\sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{D}(s)} (\varphi_{swo(s)} - \varphi_{sww}) - \widehat{E}\alpha \tag{6.3a}$$

$$\textstyle\sum_{w \in D(s)} \sigma_{swa} \leqslant \lambda_{sa} + E_R, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{6.3b}$$

$$\varphi_{swb(a)} - \varphi_{swe(a)} \leqslant \sigma_{swa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{6.3c}$$

$$\textstyle\sum_{w \in \mathcal{W}(c)} \pi_{cw} = 1, \quad c \in \mathcal{C} \tag{6.3d}$$

$$\textstyle\sum_{u \in \mathcal{U}(c,w)} \lambda_{s(w,u)} \leqslant \pi_{cw} + E_T\alpha, \quad s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{6.3e}$$

$$\varphi_{swv} \in \mathbb{R}, \ s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{6.3f}$$

$$\sigma_{swa} \in \mathbb{R}_+, \ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{6.3g}$$

$$\lambda_{sa} \in \mathbb{R}_+, \ s \in \mathcal{S}, a \in \mathcal{A}; \quad \pi_{cw} \in \mathbb{R}_+, \ c \in \mathcal{C}, w \in \mathcal{W}(c), \tag{6.3h}$$

6.3.2 Pricing problem for c-set generation

Since we now minimize the frame length, we now (contrary to ECMP) need a way to generate c-sets that could potentially improve on the objective function. To achieve this we now derive the pricing problem for c-set generation following the steps we used for with SFMP. The current PP consist in finding a c-set that maximizes the value $P(c, \lambda^*, \alpha^*)$ where α^* is the vector of optimal values of the dual variables α:

$$P(c, \lambda^*, \alpha^*) := \min_{\pi \geqslant 0, \sum_{w \in \mathcal{W}(c)} \pi_w = 1} Q(\pi; c), \tag{6.4}$$

where $\pi = (\pi_w)_{w \in \mathcal{W}(c)}$ and

$$Q(\pi; c) := \sum_{w \in \mathcal{W}(c)} \sum_{s \in \mathcal{S}(c,w)} (\max\{0, \sum_{a \in \mathcal{A}(s,c,w)} \lambda^*_{sa} - E_T \alpha^* - \pi_w\}. \tag{6.5}$$

The LP formulation for calculating $P(c, \lambda^*, \alpha^*)$ is as follows:

$$P(c, \lambda^*, \alpha^*) = \min \; \sum_{w \in \mathcal{V}} \sum_{s \in \mathcal{S}} k_{sw} \tag{6.6a}$$

$$[r] \quad \sum_{w \in \mathcal{V}} \pi_w = 1 \tag{6.6b}$$

$$[g_{sw} \geqslant 0] \quad k_{sw} \geqslant \sum_{u \in \delta^+(w)} \lambda^*_{s(w,u)} Y(w,u) - \pi_w - E_T \alpha^*, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{6.6c}$$

$$\pi_w \in \mathbb{R}_+, \; w \in \mathcal{V}; \; k_{sw} \in \mathbb{R}_+, s \in \mathcal{S}, w \in \mathcal{V}. \tag{6.6d}$$

The dual problem to (6.6) is formulated as follows:

$$\max D = -r + \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} (\sum_{u \in \delta^+(w)} \lambda^*_{s(w,u)} Y(w,u)) g_{sw} - \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} E_T g_{sw} \alpha^* \tag{6.7a}$$

$$\sum_{s \in \mathcal{S}} g_{sw} \leqslant r, \quad w \in \mathcal{V} \tag{6.7b}$$

$$g_{sw} \leqslant 1, \quad s \in \mathcal{S} \tag{6.7c}$$

$$r \in \mathbb{R}; \; g_{sw} \in \mathbb{R}_+, \; w \in \mathcal{V}, s \in \mathcal{S}. \tag{6.7d}$$

Finally, the pricing problem for c-set generation in RFMPE is as follows:

$$\mathrm{PP}(\lambda^*): \; \max \; P = -r + \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} \sum_{u \in \delta^+(w)} \lambda^*_{s(w,u)} U_{swu} - \sum_{s \in \mathcal{S}} \sum_{w \in \mathcal{V}} \alpha^* E_T g_{sw} \tag{6.8a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{6.8b}$$

$$X_w \leqslant \sum_{u \in \delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{6.8c}$$

$$X_w + \sum_{u \in \delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{6.8d}$$

$$p(w,u) + M(w,u)(1 - Y_{wu}) \geqslant \gamma(\eta + \textstyle\sum_{v\in\mathcal{V}\setminus\{w,u\}} p(v,u)X_v), \quad (w,u) \in \mathcal{A} \tag{6.8e}$$

$$U_{swu} \leqslant g_{sw},\, U_{swu} \leqslant Y_{wu}, \quad s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(v) \tag{6.8f}$$

$$\textstyle\sum_{s\in\mathcal{S}} g_{sw} \leqslant r, \quad w \in \mathcal{V}; \;\; g_{sw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{6.8g}$$

$$Y_{wu} \in \mathbb{B},\, w \in \mathcal{V}, u \in \delta^+(w);\; X_w \in \mathbb{B},\, w \in \mathcal{V} \tag{6.8h}$$

$$r \in \mathbb{R};\; g_{sw} \in \mathbb{R}_+,\, s \in \mathcal{S}, w \in \mathcal{V};\; U_{swu} \in \mathbb{R}_+,\, s \in \mathcal{S}, w \in \mathcal{V}, u \in \delta^+(w). \tag{6.8i}$$

6.4 Numerical study

In the following study we illustrate to what extent energy consumption can be minimized when using our optimization model as well as the tradeoff between energy consumption and the frame length.

As in Chapter 5, all optimization problems were implemented and solved by means of C# and CPLEX 12.9.0, respectively. All computations were executed on an Intel Core i7-8550U CPU (four cores, each up to 4 GHz) with 16 GB RAM.

6.4.1 Network setting

In the main part of this study we consider 10 randomly generated networks with 30 nodes and network area equal to 199.5 m × 199.5 m (Network M1-Network M10). Additionally, to illustrate the tradeoff between energy consumption and the frame length we consider two other networks, one with 24 nodes and network area equal to 180 m × 180 m (Network S1) and the other with 36 nodes and network area equal to 220 m × 220 m (Network L1). The nodes in all networks were placed according uniform distribution. All networks were generated with [73]. Networks S1, M1, and L1 are depicted in Figures 6.1-6.3, respectively.

The links depicted in the figures are bi-directed and represent two oppositely directed arcs between its end nodes. Such a link is provided between two given nodes if, and only if, the distance between these nodes does not exceed the maximal transmission range, which is calculated as follows. First, to obtain the power received at a node we use the propagation model described in [18]: the power (expressed in mW) received at node w when node v is broadcasting is defined as the power of the transmitter (which is the same for each node and

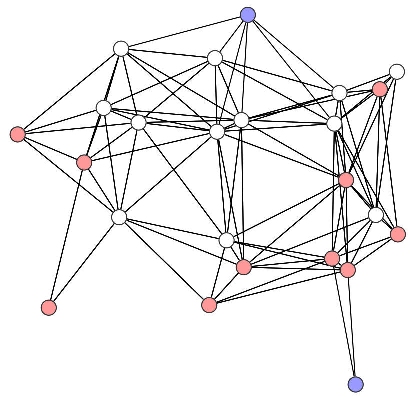

Figure 6.1: Network S1.

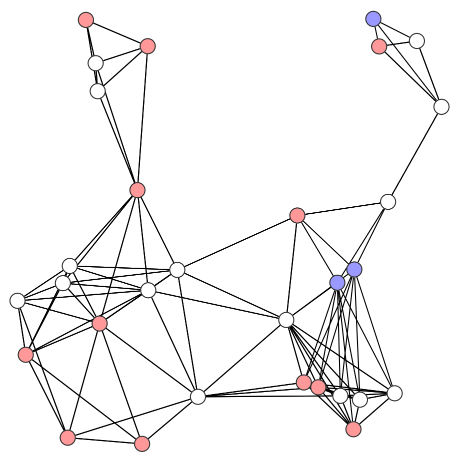

Figure 6.2: Network M1.

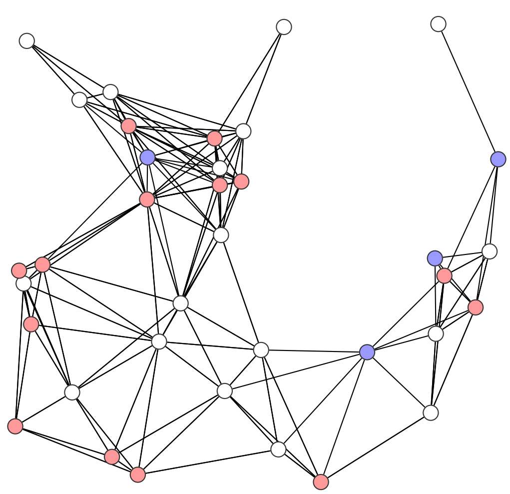

Figure 6.3: Network L1.

equals 100 mW) multiplied by $d(v,w)^{-4}$, where $d(v,w)$ (in m) is the distance between the nodes. Then, assuming noise power equal to -101 dBm, and setting the threshold SNR to 8 dB, we calculate the maximal transmission range, which is equal to 66.8 m.

6.4.2 Results

The notation used in this study is summarized in Table 6.1. The initial energy consumption is the energy consumed when the shortest-path multicast trees are used and the initial frame length is obtained for the initial family of c-sets (one-node c-sets in this case). As it is commonly done in the literature, in this study we treat energy consumption as electrical current consumption and express it in A. Because of that, to obtain the actual energy consumption in J we need to multiply the obtained values by the value of voltage supply (note that we assume fixed voltage supply and the effects related to batteries discharging are out of the scope of this chapter) and the frame length in s. E_T and E_R are the values of the electrical current consumption presented in [76] for Semtech SX1276 transceiver and the transmission power 100 mW, and are equal to 120 mA and 12 mA, respectively. We obtain the optimized energy consumption by solving ECMP, the optimized frame length by solving RFMP, and the frame length for the optimized energy consumption by solving RFMPE. However, it was more complicated to obtain energy consumption for the optimized frame length. To do this we first solve RFMPE for the optimized energy consumption and then we gradually increase the maximum energy consumption (by the minimal value of the change in energy consumption which is equal to 12 mA) until we obtain the minimum frame length.

The results obtained for Network M1-Network M10 are presented in Table 6.2. The frame lengths are reported in the number of time slots and the values of energy consumption are the values of the total energy consumption during one TDMA frame. The most important conclusions we can draw from these results are as follows. First, a significant gain in the total energy consumption can be obtained by using energy consumption minimization, namely for Network M9 we can save around 13% and around 10% for Network M1. Note that such savings are especially important in WSN since they can increase the network lifetime. Second, it should be noticed that in one case (Network M8) energy consumption for the shortest-path multicast trees is equal to optimal energy consumption. Finally, in almost

Table 6.1: Numerical results—explanation.

notation	description
E	initial energy consumption
E^*	optimized energy consumption
F	initial frame length
F^*	optimized frame length
F^E	frame length for the optimized energy consumption
E^F	energy consumption for the optimized frame length

all cases there is the tradeoff between the minimum frame length and energy consumption. Note that sometimes the energy consumption needed to obtain the minimum frame length is substantially greater than its minimum value obtained by solving ECMP. This is the case for example in Network M1 where the energy consumption for the minimum frame length is almost 19% greater than the energy consumption obtained by solving ECMP. The tradeoff is even more visible in Figures 6.4-6.6 that present the minimum frame length obtained for different values of energy consumption upper bound, respectively for Networks S1, M1, and L1.

Table 6.2: Optimized frame lengths for small networks.

	E	E^*	F	F^*	F^E	E^F
Network M1	1.992A	1.776A	62	37	45	2.112A
Network M2	0.984A	0.960A	28	22	24	1.056A
Network M3	1.752A	1.632A	56	29	33	1.704A
Network M4	1.680A	1.560A	53	27	33	1.680A
Network M5	1.008A	0.984A	29	23	24	1.008A
Network M6	2.064A	2.016A	72	32	33	2.064A
Network M7	2.160A	2.040A	73	30	32	2.088A
Network M8	1.464A	1.464A	49	31	31	1.464A
Network M9	1.392A	1.320A	43	31	31	1.320A
Network M10	1.824A	1.584A	54	31	53	1.632A
Avg	1.632A	1.534A	51.9	29.3	33.9	1.613A

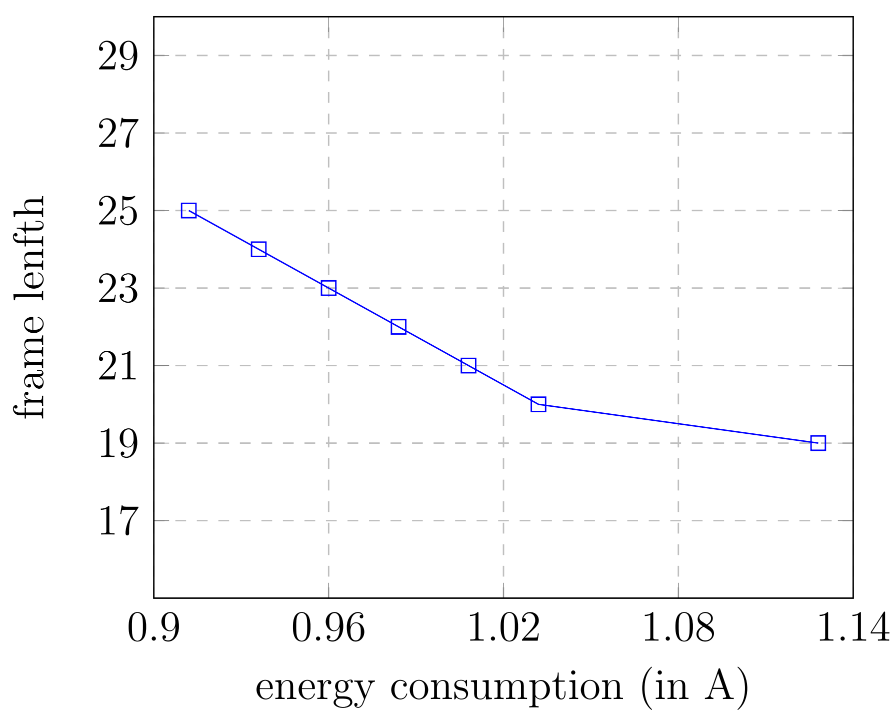

Figure 6.4: Frame vs. energy consumption - Network S1.

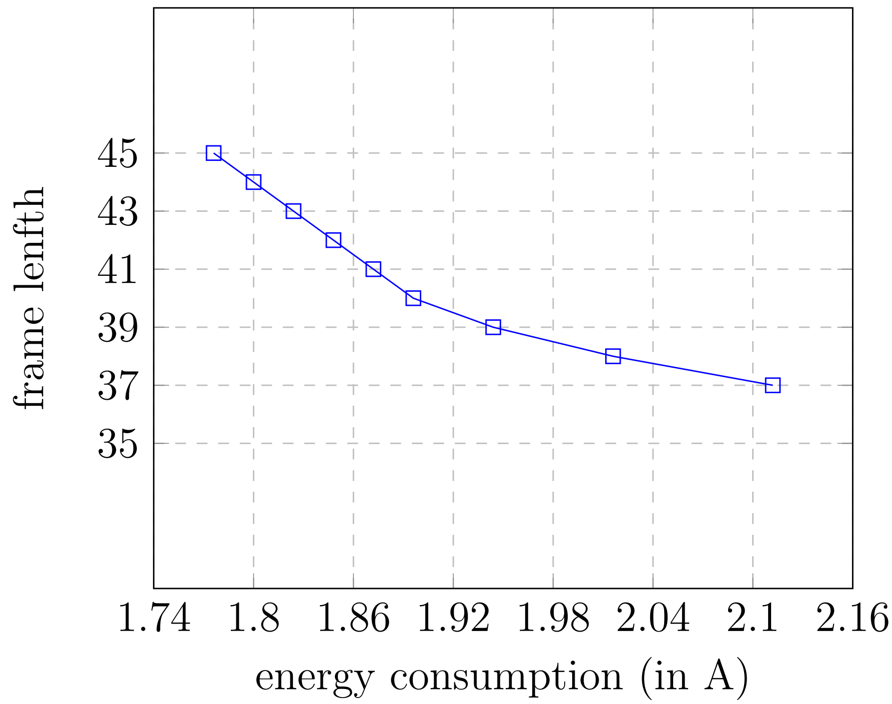

Figure 6.5: Frame vs. energy consumption - Network M1.

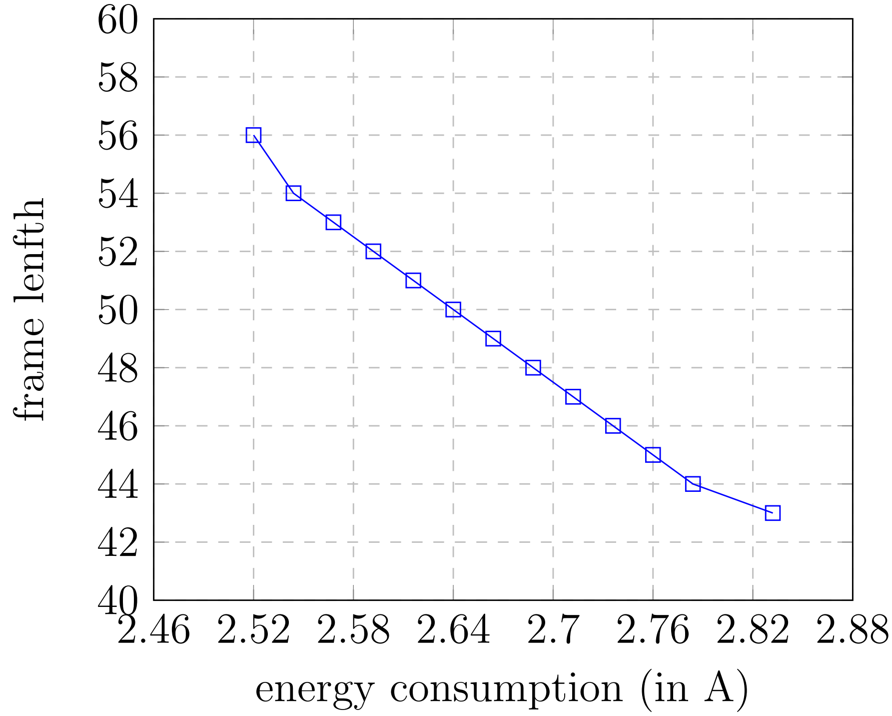

Figure 6.6: Frame vs. energy consumption - Network L1.

Chapter 7

Packet delay minimization

After discussing different variants of the problems aiming at throughput maximization as well as at energy consumption minimization, in this chapter we proceed to another important issue, namely to packet delay minimization. Now we assume that the composition of the TDMA frame is given and sufficient to carry the packets of all packet streams with a finite delay, and aim at finding a packet broadcast scheduling pattern that minimizes the maximum delay over all streams. To achieve this, we present an exact MIP formulation for the related optimization problem. Since the problem formulation can be effectively solved only for the networks of a limited size, we propose a heuristic method, and demonstrate, through a numerical study, its capability of finding near optimal solutions in a reasonable time.

The content of this chapter is based on [77], and is relevant to the simplest version of the frame minimization problem with predefined routing trees and without MCS assignment and power control, i.e., the problem setting considered in SFMP. Possible modifications of the considered problem are described in Section 7.1.2.

7.1 Additional notions

To deal with packet delay minimization in MHWN some new notions are needed.

First, recall that we use the term frame composition to refer to the pair $[\mathcal{C}, (F(c),\, c \in \mathcal{C})]$, and the term frame transmission pattern refers to the sequence $\widehat{\mathcal{F}} := (c(f))_{f\in\mathcal{F}}$. Then, we stated that the traffic carrying capability of a network depends only on the composition, and not on a particular frame transmission pattern $\widehat{\mathcal{F}}$ corresponding to this composition. For example, a reader may remember that in SFMP the frame composition that guarantees the shortest frame was determined by the values of variables h_{scw} obtained by solving (3.4). Let us treat the values of variables h_{scw} as the sequence of binary coefficients $h = (h(s,c,w))_{s\in\mathcal{S},c\in\mathcal{C},w\in\mathcal{W}(c)\cap\mathcal{V}'(s)}$, and notice that any sequence h where an $h(s,c,w)$ is equal to 1 if, and only if, node w broadcasts an s-packet in one of the slots that applies c-set c, determines a frame composition and is referred to as packet assignment requirement. Further, if the following constraints:

$$\textstyle\sum_{c\in\mathcal{C}(a)} h(s,c,b(a)) \geqslant 1, \quad s\in\mathcal{S}, a\in\mathcal{A}(s) \tag{7.1a}$$

$$\textstyle\sum_{s\in\mathcal{S}(c,w)} h(s,c,w) \leqslant F(c), \quad c\in\mathcal{C}, w\in\mathcal{W}(c) \tag{7.1b}$$

are satisfied by packet assignment requirements, the frame composition $[\mathcal{C},(F(c),\, c\in\mathcal{C})]$ is capable of carrying the packets of all packet streams with a finite delay. We should also note that in general for a given frame composition there are multiple solutions h that fulfil conditions (7.1b). Finally, since packet assignment requirement is not sufficient to determine packet delay (it determines only a frame composition) we introduce the notion of packet broadcast schedule. Packet broadcast schedule (defined for a given frame transmission pattern) is an assignment of the particular packet transmissions to time slots. In other words, packet broadcast schedule assigns for each time slot $f\in\mathcal{F}$ and each transmitting node $w\in\mathcal{W}(c(f))$, a particular packet stream s, $s\in\mathcal{S}$ whose packet is transmitted from node w in time slot f. Note that such an assignment has to be constructed according to packet assignment requirement.

7.1.1 PDMP formulation

Given the above definitions it is now possible to describe the particular problem we deal with in this chapter. Consider a network where (i) fixed multicast trees $\mathcal{T}(s), s\in\mathcal{S}$, (ii) a feasible composition $[\mathcal{C},\ (F(c), c\in\mathcal{C})]$, and (iii) a corresponding packet assignment requirement h are given. Now, in order to fully describe the network operation we need to make the following selections:

- a frame transmission pattern: choose one of $\frac{F!}{\prod_{c\in\mathcal{C},F(c)>0} F(c)!}$ possible sequences $\widehat{\mathcal{F}} := (c(f))_{f\in\mathcal{F}}$ of c-sets that realized the assumed frame composition (each such pattern is a permutation with repetitions of the elements c of $\mathcal{C}$, where element c is repeated $F(c)$ times)

- a packet broadcast schedule: for each $f \in \mathcal{F}$ and each $w \in \mathcal{W}(c(f))$ choose an s-packet such that $h(s, c(f), w) = 1$ and reserve the broadcasts from node $w \in \mathcal{W}(c(f))$ for the consecutive s-packets.

Observe that a given packet broadcast schedule (specified for a transmission pattern) determines packet delays. In fact (see [18]), for each $s \in \mathcal{S}$ all consecutive s-packets (denoted by $s(1), s(2), \dots$) will appear at each destination node $d \in \mathcal{D}(s)$ with the same delay $L(s, d)$ measured in the number of time slots elapsed between the epoch in which such a packet is generated at s (note that packet $s(k)$ is generated at the beginning of frame number k) and the time epoch it is delivered to d. Clearly, these delays will in general be different for different choices of the packet broadcast schedule. Thus, defining the maximum delay for stream s (called packet stream delay) as $L(s) := \max_{d\in\mathcal{D}(s)} L(s, d)$, $s \in \mathcal{S}$, we can finally state the packet delay minimization problem.

- PDMP: given a frame composition and a packet assignment requirement, find a packet broadcast schedule that minimizes the packet streams delay over all packet streams.

The so described problem can be illustrated by the following example studied in [18]. Consider the network shown in Figure 7.1 that is supposed to carry only one multicast packet stream with the source node (marked in dark grey) and the set of destination nodes (marked in light gray). Assume that the frame is composed of five time slots and each time slot uses one c-set from the family of c-sets depicted in Figure 7.2. Clearly, each of the considered c-sets has to be used in the frame, i.e., $F(c_1) = F(c_2) = F(c_3) = F(c_4) = F(c_5) = 1$. It is easy to notice that the order in which the considered c-sets are situated in the frame greatly influences the packet delay. Indeed, the sequence (c_1, c_2, c_3, c_4, c_5) results in the packet delay equal to 5 time slots, while the reversed sequence increases the delay to as many as 21 time slots, which shows the importance of appropriate scheduling as far as of the packet delay is concerned.

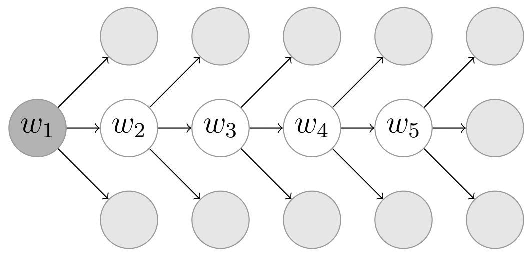

Figure 7.1: A network with a multicast packet stream.

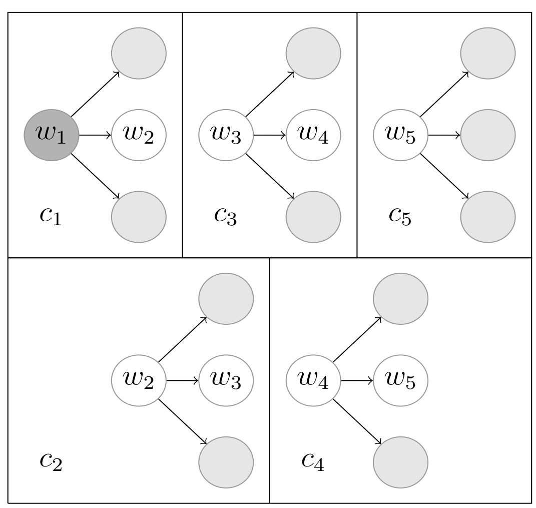

Figure 7.2: A family of compatible sets.

A MIP problem corresponding to the above verbal statement is given in formulation (7.2) (presented already in [18]). The parameters in that formulation are the frame composition $[\mathcal{C}, (F(c), c \in \mathcal{C})]$, the packet assignment requirement $h = (h(s,w,c),\ s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \cap \mathcal{V}'(s))$, and the upper bound M on the maximum number of frames necessary to deliver a packet to its destination; this means that the maximum delay expressed in the number of slots is not greater than $\overline{F} := M \cdot F$, and hence $\overline{\mathcal{F}} := \{1, 2, \ldots, \overline{F}\}$ is the set all considered time slots. Additional notations used in (7.2) are as follows. The (unique) arc preceding arc a in $\mathcal{A}(s), a \in \mathcal{A}(s) \backslash \{\delta^+(o(s))\}$ is denoted by $p(s,a)$, $q(s,d)$ denotes the (unique) arc in $\mathcal{A}(s)$ incoming to node $d \in \mathcal{D}(s)$, $\mathcal{V}'(s,w)$ denotes the set of nodes appearing after node $w \in \mathcal{V}'(s)$ in tree $\mathcal{T}(s)$ $(\mathcal{V}'(s,w) := \delta^+(w) \cup \mathcal{A}(s))$, while $C(s,w)$ denotes the set of c-sets applicable for s at node $w \in \mathcal{V}'(s)$; $C(s,w) := \{c \in \mathcal{C} : w \in \mathcal{W}(c), \mathcal{U}(c,w) \cap \mathcal{V}'(s,w) \neq \varnothing\}$.

The formulation makes use of the following optimization variables:

- l (continuous) – delay
- $x_c^f = 1$ if c-set $c \in \mathcal{C}$ is used in time slot $f \in \mathcal{F}$ of the frame; otherwise $x_c^f = 0$ (binary)
- $Y_{sw}^f = 1$ if node $w \in \mathcal{V}'(s)$ is reserved for broadcasting s-packets ($s \in \mathcal{S}$) in time slot $f \in \mathcal{F}$ of the frame; otherwise $Y_{sw}^f = 0$ (binary)
- $z_{sa}^f = 1$ if packet $s(1)$ from stream $s \in \mathcal{S}$ is sent over arc $a \in \mathcal{A}(s)$ in time slot $f \in \overline{\mathcal{F}}$; otherwise $z_{sa}^f = 0$ (binary)
- $Z_{sa}^f = 1$ if packet $s(1)$ from stream $s \in \mathcal{S}$ is sent over $a \in \mathcal{A}(s)$ in one of the slots between 1 and $f \in \overline{\mathcal{F}}$; otherwise $Z_{sa}^f = 0$ (binary),

and is as follows:

$$\text{PDMP:}\ \min\ l \tag{7.2a}$$

$$l \geqslant \textstyle\sum_{f \in \overline{\mathcal{F}}} f \cdot z_{sq(s,d)}^f, \quad s \in \mathcal{S}, d \in \mathcal{D}(s) \tag{7.2b}$$

$$\textstyle\sum_{c \in \mathcal{C}} x_c^f = 1, \quad f \in \mathcal{F} \tag{7.2c}$$

$$\textstyle\sum_{f \in \mathcal{F}} x_c^f = F_c, \quad c \in \mathcal{C} \tag{7.2d}$$

$$x_c^{f+(k-1)F} = x_c^f, \quad c \in \mathcal{C}, f \in \mathcal{F}, 2 \leqslant k \leqslant K \tag{7.2e}$$

$$\textstyle\sum_{s\in\mathcal{S}(w)} Y^f_{sw} \leqslant 1, \quad w\in\mathcal{V}, f\in\mathcal{F} \tag{7.2f}$$

$$Y^f_{sw} \leqslant \textstyle\sum_{c\in\mathcal{C}(s,w)} h(s,c,w)x^f_c, \quad s\in\mathcal{S}, w\in\mathcal{V}'(s), f\in\mathcal{F} \tag{7.2g}$$

$$\textstyle\sum_{f\in\mathcal{F}} Y^f_{sw} = \sum_{c\in\mathcal{C}(s,w)} h(s,c,w), \quad s\in\mathcal{S}, w\in\mathcal{V}'(s) \tag{7.2h}$$

$$Y^{f+(k-1)F}_{sw} = Y^f_{sw}, \quad s\in\mathcal{S}, w\in\mathcal{V}'(s), f\in\mathcal{F}, 2\leqslant k\leqslant K \tag{7.2i}$$

$$\textstyle\sum_{f\in\overline{\mathcal{F}}} z^f_{sa} = 1, \quad s\in\mathcal{S}, a\in\mathcal{A}(s) \tag{7.2j}$$

$$Z^f_{sa} = \textstyle\sum_{\tau=1}^{f} z^\tau_{sa}, \quad s\in\mathcal{S}, a\in\mathcal{A}(s), f\in\overline{\mathcal{F}} \tag{7.2k}$$

$$z^f_{sa} \leqslant Z^f_{sp(s,a)}, \quad s\in\mathcal{S}, a\in\mathcal{A}(s)\backslash\{(o(s),v)\in\mathcal{A} : v\in\delta^+(o(s))\}, f\in\overline{\mathcal{F}} \tag{7.2l}$$

$$z^f_{sa} \leqslant Y^f_{sb(a)}, \quad s\in\mathcal{S}, a\in\mathcal{A}(s), f\in\overline{\mathcal{F}} \tag{7.2m}$$

$$z^f_{sa} \leqslant \textstyle\sum_{c\in\mathcal{C}(s,a)} x^f_c, \quad s\in\mathcal{S}, a\in\mathcal{A}(s), f\in\overline{\mathcal{F}} \tag{7.2n}$$

$$l\in\mathbb{R}; \quad x^f_c\in\mathbb{B},\ c\in\mathcal{C}, f\in\overline{\mathcal{F}} \tag{7.2o}$$

$$Y^f_{sw}\in\mathbb{B},\ s\in\mathcal{S}, w\in\mathcal{V}'(s), f\in\overline{\mathcal{F}} \tag{7.2p}$$

$$z^f_{sa}, Z^f_{sa}\in\mathbb{B},\ s\in\mathcal{S}, a\in\mathcal{A}(s), f\in\overline{\mathcal{F}}. \tag{7.2q}$$

Due to constraint (7.2b), objective (7.2a) minimizes, over all packet streams s, the maximum number of slots required to deliver packet $s(1)$ (and hence all subsequent packets $s(2), s(3), \dots$) to all destinations nodes d in $\mathcal{D}(s)$. This particular objective is referred to as min-max delay.

The three subsequent constraints specify a frame transmission pattern $\widehat{\mathcal{C}}$ in which $c(f) = c$, if, and only if, $x^f_c = 1$. Constraint (7.2c) assigns exactly one c-set from $\mathcal{C}$ to each slot of the frame, while constraint (7.2d) ensures that pattern $\widehat{\mathcal{C}}$ is consistent with the assumed frame composition. Constraint (7.2e), in turn, ensures that $\widehat{\mathcal{C}}$ is repeated in the consecutive frames.

Next, constraints (7.2f)-(7.2i) define the packet broadcast schedule consistent with the specified frame transmission pattern and the assumed packet assignment requirement h.

Then, by means of variables z^f_{sa}, and Z^f_{sa}, constraints (7.2j)-(7.2n) specify, for each arc a in the tree of stream s, which time slot in $1, 2, \dots, \overline{F}$ is used to transmit packet $s(1)$ over this arc. Note that constraint (7.2l) prevents transmitting packet $s(1)$ over a if it has not been transmitted over a yet.

The final three constraints specify the range of variables, where $\mathbb{R}$ denotes the set of real numbers and $\mathbb{B} := \{0, 1\}$.

PDMP, represented by MIP formulation (7.2), is an $\mathcal{NP}$-hard problem, as demonstrated in [64] for its special case assuming unicast rather than multicast packet traffic assumed in this paper (the unicast case is substantially simpler to treat than the multicast one). Certainly, PDMP can be directly solved by MIP solvers. However, as illustrated in Section 7.3, such exact solving of (7.2) becomes inefficient for large networks, and therefore in Section 7.2 we introduce a heuristic algorithm for PDMP.

7.1.2 Comments

The first simple comment is that the min-max delay objective used in formulation (7.2) is not the only relevant for delay optimization. For example, the total packet delay $l = \sum_{s\in\mathcal{S}}\sum_{d\in\mathcal{D}(s)} L(s,d)$ (where $L(s,d) = \sum_{f\in\overline{\mathcal{F}}} f \cdot z^f_{sq(s,d)}$) could be used, possibly with additional upper-bound constraints on $d(s)$. Another possibility is min-max fair optimization of the delay $(L(s), s \in \mathcal{S})$ [78].

The min-max delay optimization problem stated in Section 7.1.1 assumes that frame composition $[\mathcal{C},\ (F(c), c \in \mathcal{C})]$ and packet assignment requirement h are fixed, and hence have to be preprocessed in a reasonable way. This can be done by finding a frame composition and packet assignment requirement that result in minimum frame length F, i.e., by solving SFMP described in (3.4) (note that quantities $F(c)$ and $h(s,c,w)$ are optimization variables in (3.4) and that is why are denoted by F_c and h_{scw}). It is important to note that formulation (3.4) assumes that the multicast trees $\mathcal{T}(s)$, $s \in \mathcal{S}$, are fixed (for example, for each s the shortest-path tree rooted at $o(s)$ can be used). In general combining frame length minimization with tree optimization can lead to a significant decrease of the minimum frame length. This is shown in papers [79, 18], where such combined optimization is considered. In any case, the trees used/optimized during frame length minimization are used in formulation (7.2).

Clearly, the optimized c-set family used in the frame composition resulting from (3.4) will contain only those c-sets c for which $F_c > 0$. Then, the packet assignment requirement assumed for PDMP will involve only values of those h_{scw} for which c belongs to the obtained composition. The so obtained frame compositions is used in the numerical study presented in Section 7.3.

Another issue related to PDMP is that the minimum delay l resulting from (7.2) depends in general on the particular packet assignment requirement h (out of many possible for a given frame composition) used as an element of the input data. This issue can be treated by assuming that only the frame composition is fixed and given, making $h(s, c, w)$ optimization variables h_{scw}, and adding constraints (3.4b) and (3.4c) to the formulation.

Further, when packet assignment requirement optimization is added to (7.2), we can add frame composition optimization as well. This is achieved by converting the c-set occurrence parameters $F(c)$ to non-negative continuous variables F_c, and adding a new constraint

$$\sum_{c \in \mathcal{C}} F_c \leqslant F \tag{7.3}$$

to formulation (7.2).

Although this extension will in general allow only for a marginal improvement of the minimum packet delay when the family $\mathcal{C}$ resulting from the frame length minimization (3.4) is used as input to the so modified PDMP, it may decrease the minimum packet delay to a more substantial degree when a larger family $\mathcal{C}$ is assumed for (7.2). This is because there is a tradeoff between minimization of packet delays and minimization of the frame length. Thus, from the delay minimization viewpoint it could be beneficial to increase the value of F above its value resulting from (3.4) and possibly extend the family $\mathcal{C}$ by adding other c-sets generated during the frame length minimization solution process but not used in the final frame composition. Note that in general this procedure will decrease the delay at the expense of decreasing the network throughput since the increasing the frame length implies smaller packet intensity (one per frame) of the packet streams.

Observe also that in order to consider all possible c-sets we could still apply P&B to the just described version of the delay minimization problem, using its linear relaxation for the master problem. However, the effectiveness of such an approach would be unacceptable. Also, we could explicitly embed the problem of finding the c-sets for all the slots in $\mathcal{F}$ into (7.2). Such a full version of the problem would require adding, for each slot $f \in \mathcal{F}$, a set of constraints (and associated binary variables) forcing that the broadcasts used in the slots represent c-sets (these constraints are given by conditions (2) in [18]). However, this would make the problem virtually intractable for realistic networks.

Finally, let us notice that tree optimization could be added to formulation (7.2) as well, in the way we did in RFMP for frame length minimization. Then, the trees $\mathcal{T}(s)$ would be characterized by additional binary variables, constrained by additional conditions added to the formulation. However, this again would lead to virtually intractable problem formulation.

7.2 Solution Algorithm

Since for realistic networks the number of binary variables in formulation (7.2) becomes tremendous and thus the likelihood of solving it in reasonable time by a MIP solver is low, a heuristic algorithm is in place. For that we make use of the simulated annealing (SA) meta-heuristic [80] given below in the form of Algorithm 2.

Roughly speaking, the SA algorithm represents a random walk through the points (called solutions) of a feasible set $\mathcal{P}$ (called the solution space) aiming at minimizing the objective function $O : \mathcal{P} \to \mathbb{R}$. Each solution $P \in \mathcal{P}$ has a specified neighborhood $\mathcal{N}(P)$ (where $\mathcal{N}(P) \subseteq \mathcal{P}$), and the moves of the random walk are allowed only to the neighboring nodes. Starting from an initial solution, in each step (the steps are executed in the **while** loop composed of lines 7-19 in Algorithm 2) the algorithm moves from the current solution P to its randomly selected neighbor $Q \in \mathcal{N}(P)$ if such a move does not increase the objective function ($\Delta O \leqslant 0$) or it passes the so called Metropolis test applied in line 15. Note that the test allows for the "uphill" moves that increase the objective function. The idea behind the test is that the chance of passing the test depends on the value of ΔO (the higher the value the lower the chance) and on the current value of the temperature parameter Θ (the lower the value the lower the chance) and when $\Delta O > 0$ or the test is not passed, the constructed random walk trajectory stays at the current point P.

Note that for any fixed temperature value Θ, K steps of the random walk are executed, and since in the main **while** loop (comprising lines 5-21) the temperature is reduced, in the consecutive sequences of K steps the chance of accepting an uphill move is decreasing. When the stopping criterion is fulfilled, the best solution visited by the generated random walk trajectory is stored in P^{best} and O^{best}.

In order to apply the SA procedure to PDMP (with a given frame composition $[\mathcal{C}, (F(c),\ c \in \mathcal{C})]$ and the corresponding packet assignment requirement h) we need to characterize the so-

Algorithm 2 Simulated annealing (SA)

```
procedure SA
    P := initial_solution;
    P^best := P; O^best := O(P^best);
    Θ := Θ^0;
    while (stopping_criterion = false) do
        k := 0
        while k < K do
            Q := neighbor(P)
            ΔO := O(Q) − O(P)
            if ΔO ⩽ 0 then
                P := Q
                if O(P) < O^best then
                    O^best := O(P); P^best := P;
                end if
            else if random < e^(−ΔO/Θ) then
                P := Q
            end if
            k := k + 1
        end while
        reduce_temperature(Θ)
    end while
end procedure
```

lution space and specify the functions required in Algorithm 2. This is done as follows.

- Solution space $\mathcal{P}$. Clearly, the solution space $\mathcal{P}$ of PDMP is composed of all possible packet broadcast schedules. However, in the proposed solution algorithm we perform a random walk through a particular subspace of $\mathcal{P}$ that we denote by $\mathcal{P}'$. To define the subspace $\mathcal{P}'$ let us introduce some new notions. First, recall that there are many frame transmission patterns corresponding to a given packet assignment requirement and frame composition. Also, numerous packet broadcast schedules can be constructed for a given frame transmission pattern. Now, we say that the variant of a packet broadcast schedule is the packet broadcast schedule obtained from the original packet broadcast schedule by the time slots permutation (preserving the c-sets and the reservations assigned to the broadcasting nodes used in those time slots). Let us illustrate this with a simple example.

  Consider two time slots (f' and f''), two packet streams (s' and s''), and a c-set c with two transmitting nodes (w' and w''). Assume that as a result of the frame length minimization we obtain the following values: $F(c) = 2$, $h(s', c, w') = 1$, $h(s', c, w'') = 1$, $h(s'', c, w') = 1$, $h(s'', c, w'') = 1$. Now consider the packet broadcast schedule with both nodes of c used in slot f' reserved for s'-packets, and both nodes of c used in slot f'' reserved for s''-packets. This packet broadcast schedule has only one other variant, namely the variant with both nodes of c used in slot f' reserved for s''-packets, and both nodes of c used in slot f'' reserved for s'-packets. However, the set of all possible packet broadcast schedules contains also the schedule with nodes w' and w'' in time slot f' reserved for s'-packets and s''-packets, respectively, and nodes w' and w'' in time slot f'' reserved for s''-packets and s'-packets, respectively, as well as its swapped version. Notice that it is not possible to obtain these packet broadcast schedules by the time slots permutations of the initial schedule.

  In fact, the time slots permutation defines equivalence relation between two packet broadcast schedules in the set of all packet broadcast schedules. That means that all variants of a given packet broadcast schedule constitute the equivalence class of that packet broadcast schedule. Using these notions we define the subspace $\mathcal{P}'$ as the equivalence class of a given packet broadcast schedule.

- Function *initial_solution*. To construct an initial solution we first randomly select a transmission pattern $\widehat{\mathcal{F}} := (c(f))_{f \in \mathcal{F}}$ realizing the assumed frame composition. Then for each $c \in \mathcal{C}$ and for each $w \in \mathcal{W}(c)$ we create the set $\mathcal{S}(c, w) = \{s \in \mathcal{S} : h(s, c, w) = 1\}$. Next, for each time slot f and for each node broadcasting in this slot $w \in \mathcal{W}(c(f))$ we randomly assign a packet stream s from the set $\mathcal{S}(c(f), w)$ whose packet will be transmitted from node w in time slot f. After the assignment the packet stream s is removed from the set $\mathcal{S}(c(f), w)$. Clearly, the so obtained packet broadcast schedule is consistent with the assumed packet assignment requirement h because inequalities (3.4c) are fulfilled. Note also that in this way it is possible to generate any possible packet broadcast schedule.

- Objective function O. In order to calculate the value of $O(P)$ notice, that solution P constructed by function *initial_solution* (and every solution Q generated during execution of Algorithm 2) defines a solution of PDMP, i.e., a feasible solution of formulation (7.2). More precisely, solution P specifies feasible values $x(f, c)$ corresponding to variables x_c^f, and $Y(f, s, w)$ corresponding to variables Y_{sw}^f in (7.2). Using these values, the values of $z(f, s, a)$ and $Z(f, s, a)$ (corresponding to variables z_{sa}^f and Z_{sa}^f, respectively) can be computed from conditions (7.2j)-(7.2n), leading to the value $O(P)$ of the SA objective function. This value is simply the smallest l fulfilling all inequalities in (7.2b).

- Function *stopping_criterion*. Returns **true** when Θ becomes less than or equal to an assumed final value $\overline{\Theta}$.

- Function *neighbor*(P). A solution obtained by swapping the c-sets (preserving the reservations assigned to broadcasting nodes in each of the swapped c-sets) in P between two randomly selected time slots in $\mathcal{F}$. Note that this new solution belongs to the equivalence class of P.

- Function *random*. Returns a random number from interval $[0, 1]$.

- Function *reduce_temperature*(Θ): $\Theta := \alpha \times \Theta$ for a given parameter α $(0 < \alpha < 1)$.

Note that frame composition $[\mathcal{C}, (F(c),\ c \in \mathcal{C})]$, packet assignment requirement $h = (h(s, c, w),$ $s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \cap \mathcal{V}'(s))$, initial temperature Θ^0, final temperature $\overline{\Theta}$, temperature reduction factor α, and number of steps in a single iteration K are input parameters for the SA algorithm.

Finally, let us note that in our numerical study presented in Section 7.3 the fact that we perform a random walk through a subspace $\mathcal{P}'$ of the PDMP solution space $\mathcal{P}$ is taken into account by starting multiple SA algorithm instances concurrently, each from a different (randomly generated) initial solution.

7.3 Numerical study

Below we present a numerical study illustrating effectiveness of solving PDMP by means of the simulated annealing algorithm described in Section 7.2, as well as in an exact way by solving formulation (7.2) using a MIP solver. The SA algorithm as well as problem formulations (7.2) and (3.4) together with related solution algorithms were implemented in C# and executed on a dedicated Windows Server 2016 Datacenter x64 virtual machine, configured for 20 logical processors and up to 80 GB of RAM. CPLEX 12.9.0 was used for solving the MIP formulations. The results presented in this section were taken from [77]. All delay values and frame lengths reported in this study are expressed in the number of time slots.

7.3.1 Network setting

We consider irregular network topologies of five different sizes: extra small networks with $|\mathcal{V}| = 20$ nodes, small networks with $|\mathcal{V}| = 30$ nodes, medium networks with $|\mathcal{V}| = 40$ nodes, large networks with $|\mathcal{V}| = 50$ nodes, and extra large networks with $|\mathcal{V}| = 60$ nodes. The networks were generated using the tool available at [73] that places nodes randomly in an assumed square network area according to the uniform distribution. To achieve constant network density we increase the network area proportionally to $|\mathcal{V}|$ and assume the network area equal to 163 m $\times$ 163 m for extra small networks, 199.5 m $\times$ 199.5 m for small networks, 230 m $\times$ 230 m for medium networks, 257.5 m $\times$ 257.5 m for large networks, and 282 m $\times$

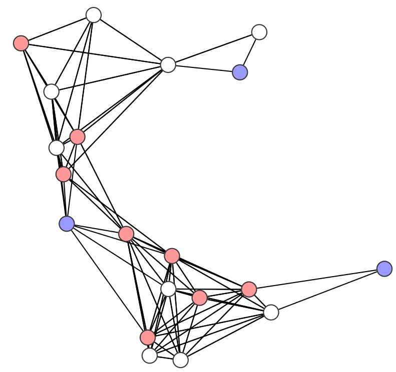

Figure 7.3: Network N1.

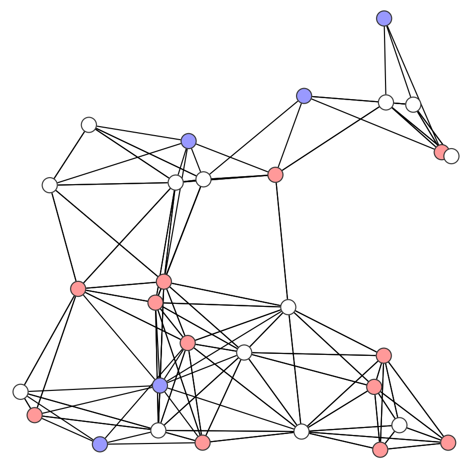

Figure 7.4: Network N2.

282 m for extra large networks. In each network, 40% of nodes are the traffic originating nodes, i.e., the nodes that constitute sources of packet streams (note that such nodes are also capable of transiting packets), 15% of nodes are the destination nodes (for small and large networks the number of the destination nodes was rounded up), and the remaining nodes are the (purely) transit nodes. Examples of such networks (one for each of the considered sizes) are depicted in Figures 7.3-7.7. The numbers of nodes and links in networks N1-N5 are equal to (20, 97), (30, 174), (40, 263), (50, 406), and (60, 451), respectively. In the figures, the traffic originating nodes, the destination nodes, and the transit nodes are shown in red, blue, and white, respectively.

In this study we used the same propagation model as in Chapter 6.

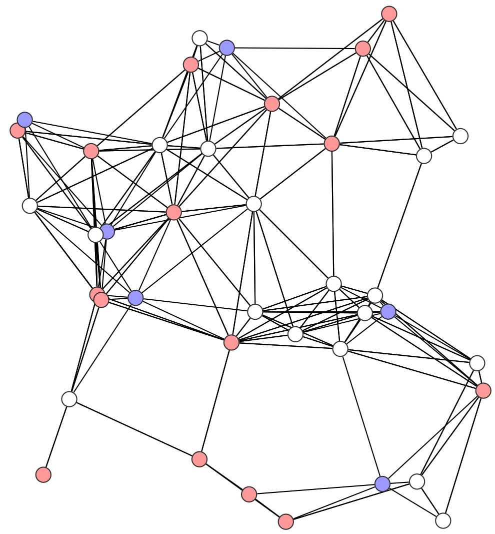

Figure 7.5: Network N3.

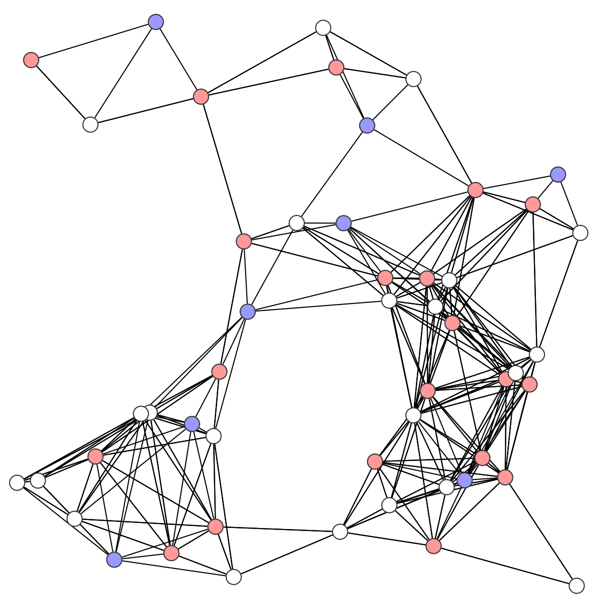

Figure 7.6: Network N4.

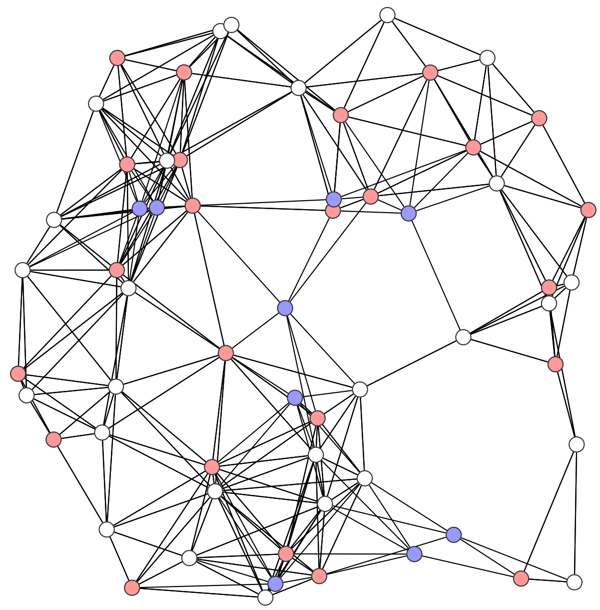

Figure 7.7: Network N5.

7.3.2 SA parameters setting

The parameter values used in the SA algorithm described in Section 7.2 are as follows. In the case of extra small, small, medium, and large networks the initial and the final temperatures are set to $\Theta^0 = 5$ and $\overline{\Theta} = 0.1$; for extra large networks we assume $\Theta^0 = 3$ and $\overline{\Theta} = 0.06$. The temperature is reduced with factor α set to 0.9. Note that since reaching the final temperature is used as the stopping criterion, in both cases the number of iterations is equal to 38. In each iteration (i.e., for each temperature value), $K = 20\ 000$ steps of the algorithm are performed, and hence the SA random walk through the solution space is 760 000 steps long. As explained in Section 7.2, for better exploration of the solution space when optimizing each network instance, we have used several different random initial solution. In the computations we have used 20 initial solutions for each network and we have run the corresponding SA algorithm instances in parallel, each in a separate thread of a logical server.

7.3.3 Results of the delay optimization

Below we present the numerical results for delay optimization. In the considered cases each traffic originating node is the source of one packet stream that sends the packets to all destinations nodes in a network (i.e., the set $\mathcal{D}(s)$ for each stream $s \in \mathcal{S}$ is composed of all

Table 7.1: Numerical results—explanation.

Notation	Description
F^*	optimal frame length obtained from frame length minimization
l^*_{MIP}	minimum delay obtained by solving the MIP formulation of PDMP
l^0_{avg}	average initial delay for initial solutions used in concurrent SA algorithm instances
l^*_{avg}	average delay for final solutions of concurrent SA algorithm instances
l^*_{SA}	the best minimum delay found by means of the concurrent SA algorithm
t^f	computation time required for solving frame length minimization
t^l_{MIP}	computation time required for solving the MIP formulation of PDMP (timeout for the computation time equal to 3h was set in this case, and (*) denotes that this timeout was reached)
t^l_{SA}	computation time required for solving all concurrent SA instances

Table 7.2: Delay optimization results.

	F^*	l^*_{MIP}	l^0_{avg}	l^*_{avg}	l^*_{SA}	t^f	t^l_{MIP}	t^l_{SA}
Network N1	23	23	60.55	24.85	23	1.5s	31s	3m28s
Network N2	47	48	147.55	52.9	52	10s	55m47s	9m19s
Network N3	52	102	176.7	64.35	61	1m5s	3h (*)	16m35s
Network N4	72	-	286.3	100.55	95	1m15s	3h (*)	35m50s
Network N5	79	-	339.2	114.95	110	1m50s	3h (*)	45m33s

destinations nodes in a network). For solving PDMP, the frame composition, the packet assignment requirement and the multicast routing trees for the traffic originating nodes (which are the shortest-path trees, pre-computed by means of Dijkstra's algorithm) obtained from the frame minimization problem (3.4) are assumed. Notations used in this section are described in Table 7.1.

The results for the networks depicted in Figures 7.3-7.7 are presented in Table 7.2. The main observations are as follows. First, the optimization process leads to a significant delay decrease. The optimized delay is up to 3.08 times smaller than the average initial delay (this is the case for N5; for N1-N4 these values are equal to 2.63, 2.84, 2.89, and 3.01, respectively).

Next, the SA algorithm is able to find solutions that are optimal or close to the lower bound, i.e., the minimum frame length F^*. (Note that the minimum frame length is indeed the lower bound for l since if the minimum l were strictly less than F^*, then all packets would reach their destinations before the frame end, and hence the slots after the slot no. l would not have to be used.) For N1 the optimal solution was found, while for N2-N5 the optimized delay is 10.64%, 17.30%, 31.94%, and 39.24% greater than the lower bound, respectively, and this means that all packets are delivered well before the second frame ends.

Let us observe that the MIP formulation (7.2) of PDMP was successfully solved (before the 3 hour timeout was reached) only for the two smallest networks, and provided the exact minimum delay $l^* = 23$ (equal to the minimum frame length in this case) for N1 and $l^* = 48$ for N2. Notice that for the case of N1 SA algorithm found the optimal solution as well, while for $N2$ the difference between SA solution and optimal solution is equal to only 8.33%. Note also that within the timeout, only feasible PDMP solutions for N3 were found, while even this was not possible for N4 and N5. Yet, using the SA algorithm we were able to obtain the near optimal solutions even in those computationally demanding cases in a reasonable time. Note that the SA solution ($l^*_{SA} = 61$) for N3 found in 16 minutes and 35 seconds is substantially better than the best feasible solution ($l^*_{MIP} = 102$) achieved by the MIP solver in 3 hours.

The results obtained with the SA algorithm averaged over 10 randomly generated networks for each of the considered sizes are presented in Table 7.3 and depicted in Figure 7.8 (where the letters XS, S, M, L, and XL denote extra small, small, medium, large, and extra large networks, respectively). The results show that the SA approach is effective. For all extra small networks the optimal solutions were found. For small, medium, large, and extra large networks, the optimized delay is on the average 5.94%, 19.04%, 29.21%, and 45.26% greater than the lower bound, respectively.

7.3.4 Convergence of the SA delay minimization process

Convergence of a selected SA algorithm run for N3 (the one that found the best solution) is illustrated in Figure 7.9. In this case the gap between the initial solution ($l^0 = 161$) and the final solution ($l^*_{SA} = 61$) was equal to 100. This gap was reduced by 48% already in the first

Table 7.3: Averaged delay optimization results.

	F^*	l^0_{avg}	l^*_{avg}	l^*	t^f	t^l_{SA}
XS	20.5	48.1	20.84	20.5	2.54s	2m38s
S	38.7	141.11	43.42	41	11.31s	8m19s
M	56.7	215.66	72.17	67.5	42.5s	18m26s
L	75.3	302.97	102.05	97.3	1m39s	38m15s
XL	91.9	423.5	141.27	133.5	2m48s	58m12s

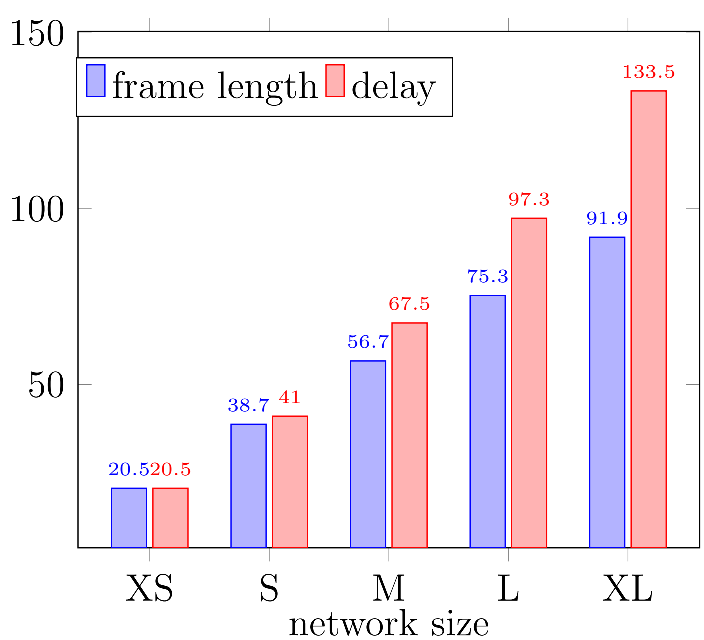

Figure 7.8: Frame length and minimized delay as a function of the network size.

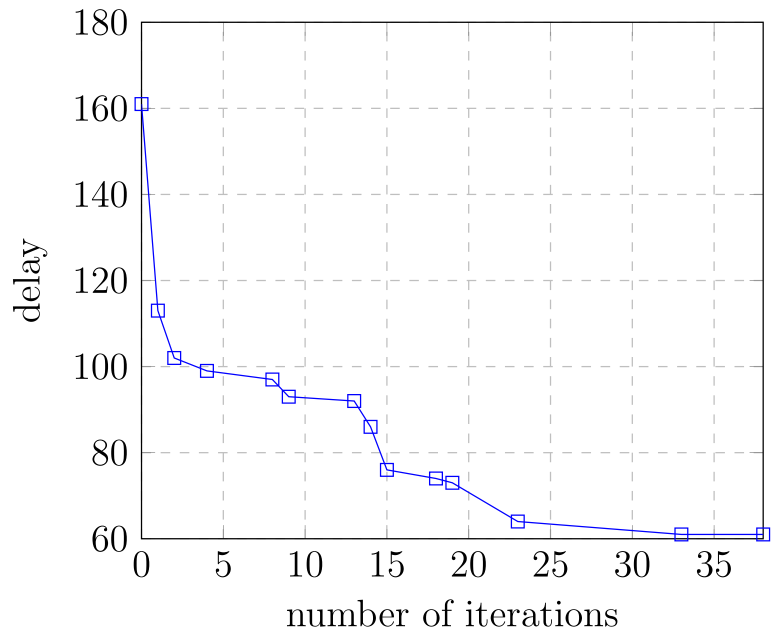

Figure 7.9: Convergence of the delay minimization process for Network N3.

iteration, and became equal to 15% of the initial gap after 15 iterations. Then, in the next 8 iterations, the gap was further decreased to 3% (this took place in iteration number 23). Finally, it took 10 extra iterations (when the algorithm terminated) to reduce the gap to 0. The observed behaviour suggests that if the computation time is important and slightly worse solutions can be accepted, the SA iterative process could be terminated substantially earlier than implied by the stopping criterion.

Chapter 8

Routing optimization - an alternative formulation

In this chapter we introduce an alternative formulation for frame minimization problem with routing optimization (RFMP) called TFMP. This formulation in some cases outperforms the formulation presented in Section 3.5 and because of its nature could be interesting from the engineering viewpoint. Besides, TFMP can be easily extended to include destination nodes optimization, an issue discussed in Chapter 9. The material presented below is based on the work described in [1]. However, while in [1] only the strong version of TFMP is given, here we present a detailed way how this version was obtained.

8.1 Formulation of TFMP

Generally speaking, the main idea behind TFMP is to consider not all possible routing trees like it is implicitly done in formulation (3.14), but those routing trees that are needed for achieving an optimal solution of the linear relaxation of the problem – just like we did for c-sets. Because of that, we now alternately generate c-sets and routing trees. Since the formulation of TFMP is substantially more difficult than its counterpart from Chapter 3, we start by introducing its weak version, and then we proceed to its strong form. The notations used in this section are described in Table 8.1.

Table 8.1: Notation 3 (TFMP).

Notation	Description
$\widehat{\mathscr{T}}(s)$	list of all multicast trees for stream $s \in \mathcal{S}$
$\widehat{\mathscr{T}}$	list of all trees ($\widehat{\mathscr{T}} := \bigcup_{s \in \mathcal{S}} \widehat{\mathscr{T}}(s)$)
$\mathscr{T}(s)$	a list of trees for stream $s \in \mathcal{S}$ ($\mathscr{T}(s) \subseteq \widehat{\mathscr{T}}(s)$)
$\mathscr{T}$	list of trees composed of $\mathscr{T}(s)$, $s \in \mathcal{S}$ ($\mathscr{T} := \bigcup_{s \in \mathcal{S}} \mathscr{T}(s)$)
$\mathcal{T}(s,t)$	tree t in $\widehat{\mathscr{T}}(s)$
$\mathcal{A}(s,t)$	arcs of tree $\mathcal{T}(s,t)$
$\mathcal{V}'(s,t)$	broadcasting nodes in tree $\mathcal{T}(s,t)$
u_s^t	equals to 1 if, and only if, tree t is selected from the list $\mathscr{T}(s)$ ($u_s^t \in \mathbb{B}$, $s \in \mathcal{S}, t \in \mathscr{T}(s)$) – primal binary variable
g_{scw}^t	equals to 1 if, and only if, node w of c-set c is used to broadcast s-packets in tree t in $\mathscr{T}(s)$ ($s \in \mathcal{S}$, $t \in \mathscr{T}(s)$, $c \in \mathcal{C}$, $w \in \mathcal{W}(c)$) – primal non-negative continuous variable

8.1.1 TFMP: a weak formulation

The weak version of TFMP is as follows:

$$\text{TFMP}(\mathcal{C}, \mathscr{T}): \quad \min \quad F = \textstyle\sum_{c \in \mathcal{C}} F_c \tag{8.1a}$$

$$\textstyle\sum_{t \in \mathscr{T}(s)} u_s^t = 1, \quad s \in \mathcal{S} \tag{8.1b}$$

$$\textstyle\sum_{c \in \mathcal{C}(a)} h_{scb(a)} \geqslant u_s^t, \quad s \in \mathcal{S}, t \in \mathscr{T}(s), a \in \mathcal{A}(s,t) \tag{8.1c}$$

$$\textstyle\sum_{s \in \mathcal{S}} h_{scw} \leqslant F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{8.1d}$$

$$F_c \in \mathbb{R},\ c \in \mathcal{C};\ u_s^t \in \mathbb{B},\ s \in \mathcal{S}, t \in \mathscr{T}(s) \tag{8.1e}$$

$$h_{scw} \in \mathbb{B},\ s \in \mathcal{S},\ c \in \mathcal{C},\ w \in \mathcal{W}(c). \tag{8.1f}$$

Now, besides the list of c-sets $\mathcal{C}$, the formulation assumes a given list of multicast trees $\mathscr{T}$. The main difference with respect to formulation (3.14) is the use of variable $u_s^t, s \in \mathcal{S}, t \in \mathscr{T}(s)$. Each variable u_s^t is equals 1 if, and only if, tree t is selected from the list $\mathscr{T}(s)$. Constraint (8.1b) assures that only one tree is selected for each stream, while constraint (8.1c), a counterpart of constraint (3.14b), ensures that the packets of each stream have to be transmitted along all arcs of its routing tree. Finally, it should be noticed that TFMP is non-compact also in terms of routing trees.

The above formulation, although intuitive and straightforward, turns out to be inefficient since its linear relaxation is weak. This is because in the linear relaxation variables u_s^t for a given s tend to take non-zero values for multiple routing trees, and as result, variables h_{scw} take small fractional values (close to zero) which leads to much shorter frame than in the MIP version of the problem. Let us consider a part of a MHWN illustrated in Figure 8.1 with one packet stream s, three routing trees t_1 (depicted in green), t_2 (depicted in violet), t_3 (depicted in blue), and c-set c with one transmitting node w_1, and tree receiving nodes w_2, w_3, w_4. Node w_1 is the source of stream s, and the packets of stream s have to be sent along one of arcs (w_1, w_2), (w_1, w_3), or (w_1, w_4) which belong to routing trees t_1, t_2, and t_3, respectively. In the linear relaxation this transmission will be realized with three routing trees, i.e., variables $u_s^{t_1}$ will be equal to $\frac{1}{3}$, and thus variable h_{scw_1} will also be equal to $\frac{1}{3}$ (see constraints (8.1c)), and the same is true for variable F_c. However, in the MIP version of the problem F_c will be equal to 1, and thus the optimality gap equals 66.67%.

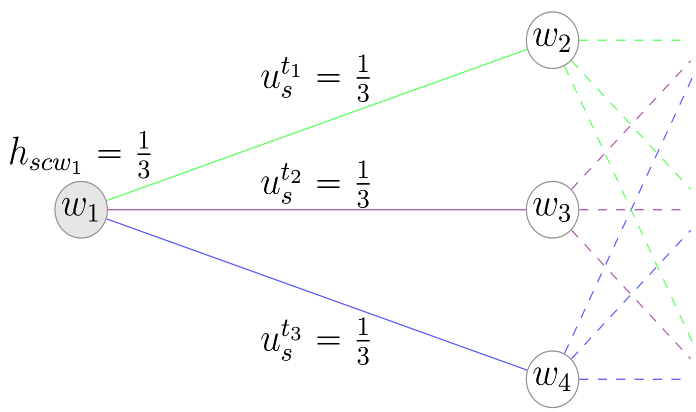

Figure 8.1: Routing trees selection example.

8.1.2 TFMP: a strong formulation

Fortunately, the aforementioned issue can be alleviated by aggregating the values of variables h_{scw} over all tress of a given stream. Such a modification can be introduced in the following way. First, we introduce variables g_{scw}^t, $s \in \mathcal{S}, t \in \mathscr{T}(s), c \in \mathcal{C}, w \in \mathcal{W}(c)$. Each variable g_{scw}^t equals 1 if, and only if, node w of c-set c is used to broadcast s-packets in tree t in $\mathscr{T}(s)$. Next, a new constraint that expresses aggregation of the values of variables h_{scw} over all trees of a given stream is introduced:

$$h_{scw} \geqslant \sum_{t\in\mathscr{T}(s)} g^t_{scw}, \quad s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c). \tag{8.2a}$$

Finally, constraint (8.1c) is modified in the following way:

$$\sum_{c\in\mathcal{C}(a)} g^t_{scb(a)} \geqslant u^t_s, \quad s \in \mathcal{S}, t \in \mathscr{T}(s), a \in \mathcal{A}(t). \tag{8.3a}$$

The above modifications applied to the example from the previous sections are as follows. Clearly, the value of each variable u^t_s remains the same and equals $\frac{1}{3}$. The same value is taken by the newly introduced variables g^t_{scw} but because of constraint (8.2a) the value of variable h_{scw_1} is now equal to 1, and thus F_c is also equal to 1 which means that there is no more optimality gap in this simple example.

The strong formulation of TFMP is thus as follows :

$$
\begin{aligned}
\text{TFMP}(\mathcal{C}, \mathscr{T})\text{:} \quad & \min \ \ F = \sum_{c\in\mathcal{C}} F_c && \text{(8.4a)}\\
[\alpha_s] \quad & \sum_{t\in\mathscr{T}(s)} u^t_s = 1, \quad s \in \mathcal{S} && \text{(8.4b)}\\
[\beta^t_{sa} \geqslant 0] \quad & \sum_{c\in\mathcal{C}(a)} g^t_{scb(a)} \geqslant u^t_s, \quad s \in \mathcal{S}, t \in \mathscr{T}(s), a \in \mathcal{A}(s,t) && \text{(8.4c)}\\
[\gamma_{scw} \geqslant 0] \quad & h_{scw} \geqslant \sum_{t\in\mathscr{T}(s)} g^t_{scw}, \quad s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) && \text{(8.4d)}\\
[\pi_{cw} \geqslant 0] \quad & \sum_{s\in\mathcal{S}} h_{scw} \leqslant F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) && \text{(8.4e)}\\
& F_c \in \mathbb{R},\ c \in \mathcal{C};\ u^t_s \in \mathbb{B},\ s \in \mathcal{S}, t \in \mathscr{T}(s) && \text{(8.4f)}\\
& h_{scw} \in \mathbb{B},\ s \in \mathcal{S},\ c \in \mathcal{C},\ w \in \mathcal{W}(c) && \text{(8.4g)}\\
& g^t_{scw} \in \mathbb{R}_+,\ s \in \mathcal{S},\ t \in \mathscr{T}(s),\ c \in \mathcal{C},\ w \in \mathcal{W}(c). && \text{(8.4h)}
\end{aligned}
$$

(Recall that the quantities in the square brackets on the left side of a given constraint denote the dual variables corresponding to this constraint.)

The dual to the linear relaxation of (8.4) is as follows:

$$
\begin{aligned}
\text{DMP}(\mathcal{C}, \mathscr{T})\text{:} \quad & \max \ \ W = \sum_{s\in\mathcal{S}} \alpha_s && \text{(8.5a)}\\
& \sum_{w\in\mathcal{W}(c)} \pi_{cw} = 1, \quad c \in \mathcal{C} && \text{(8.5b)}\\
& \sum_{a\in\mathcal{A}(s,t)} \beta^t_{sa} \geqslant \alpha_s, \quad s \in \mathcal{S}, t \in \mathscr{T}(s) && \text{(8.5c)}\\
& \pi_{cw} \geqslant \gamma_{scw}, \quad s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) && \text{(8.5d)}
\end{aligned}
$$

$$\gamma_{scw} \geqslant \sum_{u \in \mathcal{U}(c,w) \cap \mathcal{A}(s,t)} \beta^t_{s(w,u)}, \quad s \in \mathcal{S},\, t \in \mathscr{T}(s),\, c \in \mathcal{C},\, w \in \mathcal{W}(c) \tag{8.5e}$$

$$\alpha_s \in \mathbb{R},\, s \in \mathcal{S};\ \pi_{cw} \in \mathbb{R}_+,\, c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{8.5f}$$

$$\beta^t_{sa} \in \mathbb{R}_+,\, s \in \mathcal{S},\, t \in \mathscr{T}(s), a \in \mathcal{A}(s,t) \tag{8.5g}$$

$$\gamma_{scw} \in \mathbb{R}_+,\, s \in \mathcal{S},\, c \in \mathcal{C},\, w \in \mathcal{W}(c). \tag{8.5h}$$

This dual is used for deriving the pricing problem PP($\mathcal{C}$) for c-set generation (analogously as for SFMP and RFMP), and the pricing problem PP($\mathscr{T}$) for tree generation. The first of them is derived in Section 8.3, and the second in Section 8.2.

8.1.3 TFMP: formulations comparison

Before we proceed to the pricing problems derivation we show, by means of a numerical example, that the strong version of TFMP is indeed superior to its weak version. The results presented in Table 8.3 were obtained for grid network with 36 nodes and 128 links presented in Figure 8.2. We consider a simple scenario when the measurement data generated by the source nodes (light gray) are to be delivered to all destinations nodes (dark gray). The propagation model used in this example is the same as in Chapters 6 and 7, i.e., we assume that the power received by each node is inversely proportional to the fourth power of the distance from the transmitting node. The routing trees were obtained by performing the routing trees generation process, i.e., by solving an appropriate pricing problem. All frame lengths reported in this section are expressed in the number of time slots. The formulations were implemented in C# and executed on an Intel Core i7-8550U CPU (4 cores, each up to 4 GHz) with 16 GB RAM and solved with CPLEX 12.8.0.

Notice, that even for only two routing trees per stream the difference between the value of the linear relaxation of the weak formulation and its MIP version, i.e., the optimality gap, equals 8.5%. The optimality gap increases significantly with the number of routing trees per stream and is equal to as much as 34.6% for 20 routing trees per stream. In contrary, for the strong version the optimality gap is negligible even for 20 routing trees per stream. Actually, in this case the gap is practically not observed since we can obtain the optimal value of the MIP version of the problem only by rounding up the value of the liner relaxation which is the lower bound on the frame length (as the value of the optimal frame length is not fractional).

Table 8.2: Numerical results—explanation.

Notation	Description
$\lvert\mathscr{T}(s)\rvert$	number of trees per stream (the same for each stream)
F_w^{LR}	frame length obtained from the linear relaxation of the weak formulation of TFMP
F_s^{LR}	frame length obtained from the linear relaxation of the strong formulation of TFMP
F^{MIP}	frame length obtained from the MIP version of TFMP
Δ_w	optimality gap for the weak formulation of TFMP
Δ_s	optimality gap for the strong formulation of TFMP
$time_w$	MIP computation time of the weak formulation of TFMP
$time_s$	MIP computation time of the final formulation of TFMP

Moreover, note that the quality of the linear relaxation significantly affects computational efficiency of the formulations - it takes less than 40 seconds to solve the MIP version of the strong formulation for 20 trees per stream while for its weak version this time increases to as much as 1 hour and 25 minutes.

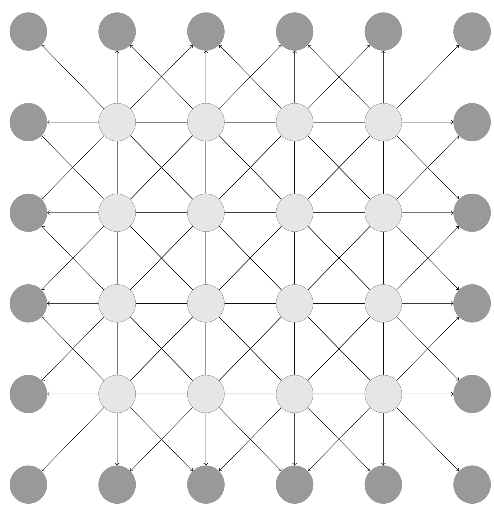

Figure 8.2: Grid network 6 x 6.

Table 8.3: TFMP - formulations comparison.

$\lvert\mathscr{T}(s)\rvert$	F_w^{LR}	F_s^{LR}	F^{MIP}	Δ_w	Δ_s	$time_w$	$time_s$
1	120	120	120	0	0	0.12s	0.24s
2	88.8	96.2	97	8.5%	0.8%	1.11s	0.66s
3	75.82	87.73	88	13.8%	0.3%	2.83s	1.21s
4	69.37	82.93	83	16.4%	0.08%	4.65s	1.10s
5	65.33	79.81	80	18.3%	0.2%	9.77s	2.13s
10	53.39	74.09	75	28.8%	1.2%	1m3s	5.41s
15	49.45	72.18	73	32.2%	1.1%	18m3s	13.71s
20	47.04	71.26	72	34.6%	1%	1h25m	36.57s

8.2 TFMP: pricing problem for tree generation

Consider a given tree $t \in \widehat{\mathscr{T}}(s)$ (denoted by $\mathcal{T}(s,t)$) and suppose that the values α^* and $\gamma^* := (\gamma^*_{scw})_{s\in\mathcal{S},c\in\mathcal{C},w\in\mathcal{W}(c)}$ are a part of an optimal solution for DMP$(\mathcal{C},\mathscr{T})$. The violation of the dual constraints by such a tree is expressed as follows:

$$P(s,t;\alpha^*,\gamma^*) := \min_{\beta\geqslant 0} Q(\beta;s,t), \tag{8.6}$$

where

$$Q(\beta;s,t) := \max\{0,\alpha^*_s - \textstyle\sum_{a\in\mathcal{A}(s,t)}\beta_a\} + \sum_{c\in\mathcal{C}}\sum_{w\in\mathcal{W}(c)}\max\{0,\sum_{v\in\mathcal{U}(c,w)\cap\mathcal{A}(s,t)}\beta_{(w,v)} - \gamma^*_{scw}\}.$$

It follows that for a fixed stream $s \in \mathcal{S}$ and given α^* and γ^*, the PP in question can be stated as follows:

$$\text{maximize } P(s,t;\alpha^*,\gamma^*) \text{ over } t \in \widehat{\mathscr{T}}(s). \tag{8.7}$$

Next we notice that the quantity $P(s,t;\alpha^*,\gamma^*)$ defined in (8.6) can be found using the following LP formulation in variables z, H_{cw} and β_a:

$$P(s,t;\alpha^*,\gamma^*) = \min\ z + \textstyle\sum_{c\in\mathcal{C}}\sum_{w\in\mathcal{W}(c)} H_{cw} \tag{8.8a}$$

$$[r \geqslant 0] \quad z \geqslant \alpha^*_s - \textstyle\sum_{a\in\mathcal{A}}\beta_a x(a) \tag{8.8b}$$

$$[m_{cw} \geqslant 0] \quad H_{cw} \geqslant \textstyle\sum_{u\in\mathcal{U}(c,w)}\beta_{(w,u)}x((w,u)) - \gamma^*_{scw}, \quad c\in\mathcal{C}, w\in\mathcal{W}(c) \tag{8.8c}$$

$$z \in \mathbb{R}_+;\ H_{cw} \in \mathbb{R}_+,\ c \in \mathcal{C}, w \in \mathcal{W}(c);\ \beta_a \in \mathbb{R}_+,\ a \in \mathcal{A}. \tag{8.8d}$$

Above, binary coefficients $x(a)$, $a \in \mathcal{A}$, specify the given tree (s,t): $x(a) = 1$ if, and only if, $a \in \mathcal{A}(s,t)$ (i.e., when arc a belongs to $\mathcal{T}(s,t)$).

Now, using the dual variables r (unconstrained in sign) and m_{cw} specified on the left-hand side of constraints (8.8b)-(8.8c), we formulate the dual to (8.8):

$$\max\ D = r\alpha_s^* - \textstyle\sum_{c\in\mathcal{C}} \sum_{w\in\mathcal{W}(c)} \gamma_{scw}^* m_{cw} \tag{8.9a}$$

$$\textstyle\sum_{c\in\mathcal{C}(a)} m_{cb(a)} x(a) \geqslant e \cdot x(a), \quad a \in \mathcal{A} \tag{8.9b}$$

$$r \leqslant 1;\ \ m_{cw} \leqslant 1, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{8.9c}$$

$$r \in \mathbb{R}_+;\ m_{cw} \in \mathbb{R}_+,\ c \in \mathcal{C}, w \in \mathcal{W}(c). \tag{8.9d}$$

Finally, transforming the tree-defining binary coefficients $x(a)$, $a \in \mathcal{A}$, to binary variables x_a, $a \in \mathcal{A}$, we formulate the TFMP pricing problem for tree generation (for a given stream $S \in \mathcal{S}$):

$$\mathrm{PP}(s;\alpha^*,\gamma^*):\ \max\ P = \alpha_s^* r - \textstyle\sum_{c\in\mathcal{C}} \sum_{w\in\mathcal{W}(c)} \gamma_{scw}^* m_{cw} \tag{8.10a}$$

$$\textstyle\sum_{c\in\mathcal{C}(a)} U_{ca} \geqslant p_a, \quad a \in \mathcal{A} \tag{8.10b}$$

$$r \leqslant 1; \quad m_{cw} \leqslant 1, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{8.10c}$$

$$p_a \leqslant r,\ p_a \leqslant x_a,\ p_a \geqslant r + x_a - 1, \quad a \in \mathcal{A} \tag{8.10d}$$

$$U_{ca} \leqslant m_{cb(a)},\ U_{ca} \leqslant x_a,\ U_{ca} \geqslant m_{cb(a)} + x_a - 1, \quad c \in \mathcal{C}(a), a \in \mathcal{A} \tag{8.10e}$$

$$\textstyle\sum_{u\in\delta^-(v)} z_{w(v,u)} + I(s,w,v) = \sum_{u\in\delta^+(v)} z_{w(v,u)}, \quad w \in \mathcal{D}(s), v \in \mathcal{V} \tag{8.10f}$$

$$z_{wa} \leqslant x_a, \quad w \in \mathcal{D}(s), a \in \mathcal{A} \tag{8.10g}$$

$$\textstyle\sum_{u\in\delta^-(o(s))} x_{(u,o(s))} = 0;\ \ \sum_{u\in\delta^-(v)} x_{(u,v)} \leqslant 1, \quad v \in \mathcal{V}\backslash\{o(s)\} \tag{8.10h}$$

$$r \in \mathbb{R}_+;\ m_{cw} \in \mathbb{R}_+,\ c \in \mathcal{C}, w \in \mathcal{W}(c);\ U_{ca} \in \mathbb{R}_+,\ c \in \mathcal{C}(a), a \in \mathcal{A} \tag{8.10i}$$

$$z_{wa} \in \mathbb{R}_+,\ w \in \mathcal{V}, a \in \mathcal{A};\ x_a \in \mathbb{B},\ a \in \mathcal{A};\ p_a \in \mathbb{R}_+,\ a \in \mathcal{A}. \tag{8.10j}$$

Above, variables p_a and U_{ca} represent respective products rx_a and $m_{cb(a)}x_a$. Arc-flow variables z_{wa}, in turn, specify the flow from $o(s)$ to $w \in \mathcal{D}(s)$ needed to define the set of arcs $\mathcal{A}(s,t) := \{a \in \mathcal{A} :\ x_a = 1\}$ of the constructed tree $\mathcal{T}(s,t)$ in $\widehat{\mathscr{T}}(s)$. This is done through constraints (8.10f)-(8.10h), where $I(s,w,v)$ is equal to 1 when $w = v$, and 0 otherwise.

8.3 TFMP: pricing problem for c-set generation

The pricing problem, used for c-set generation in TFMP, referred to as PP(β^*) and based on the dual DMP($\mathcal{C}, \mathscr{T}$) formulated in (8.5), is derived essentially in the same way as its SFMP counterpart PP(λ^*) formulated in (3.11). The derivation follows the steps given below, analogous to those described in Section 3.4.2.

$$P(c;\beta^*) := \min_{\pi,\gamma\geqslant 0:\, \sum_{w\in\mathcal{W}(c)} \pi_w = 1 \wedge \pi_w \geqslant \gamma_{sw}} Q(\pi,\gamma;c). \tag{8.11}$$

$$Q(\pi,\gamma;c) := \sum_{s\in\mathcal{S}} \sum_{t\in\mathscr{T}(s)} \sum_{w\in\mathcal{V}} (\max\{0, \sum_{u\in\delta^+(w,c)} \beta^*_{st(w,u)} - \gamma_{scw}\}).$$

$$P(c;\beta^*) = \min \sum_{w\in\mathcal{V}} \sum_{s\in\mathcal{S}} k_{sw} \tag{8.12a}$$

$$[r] \quad \sum_{w\in\mathcal{V}} \pi_w = 1 \tag{8.12b}$$

$$[\varphi \geqslant 0] \quad k_{sw} \geqslant \sum_{u\in\delta^+(w)} \beta^*_{st(w,u)} Y(w,u) - \gamma_{sw}, \quad s \in \mathcal{S}, t \in \mathscr{T}(s), w \in \mathcal{V} \tag{8.12c}$$

$$[g_{sw} \geqslant 0] \quad \pi_w \geqslant \gamma_{sw}, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{8.12d}$$

$$k_{sw} \in \mathbb{R}_+, s \in \mathcal{S}, w \in \mathcal{V};\ \pi_w \in \mathbb{R}_+,\ w \in \mathcal{V};\ \gamma_{sw} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V}. \tag{8.12e}$$

$$\max\ D = -r + \sum_{s\in\mathcal{S}} \sum_{w\in\mathcal{V}} \sum_{t\in\mathscr{T}(s)} \sum_{u\in\delta^+(w)} \beta^*_{st(w,u)} Y(w,u) \varphi_{stw} \tag{8.13a}$$

$$\sum_{t\in\mathscr{T}(s)} \varphi_{stw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{8.13b}$$

$$\sum_{t\in\mathscr{T}(s)} \varphi_{stw} \leqslant g_{sw}, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{8.13c}$$

$$\sum_{s\in S} g_{sw} \leqslant r, \quad w \in \mathcal{V} \tag{8.13d}$$

$$r \in \mathbb{R};\ \ g_{sw} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{V};\ \varphi_{stw} \in \mathbb{R}_+, s \in \mathcal{S}, t \in \mathscr{T}(s), w \in \mathcal{V}. \tag{8.13e}$$

$$\text{PP}(\beta^*):\ \max\ P = -r + \sum_{s\in\mathcal{S}} \sum_{t\in\mathscr{T}(s)} \sum_{w\in\mathcal{V}} \sum_{u\in\delta^+(w)} \beta^*_{st(w,u)} U_{stwu} \tag{8.14a}$$

$$X_w \geqslant Y_{wu}, \quad w \in \mathcal{V}, u \in \delta^+(w) \tag{8.14b}$$

$$X_w \leqslant \sum_{u\in\delta^+(w)} Y_{wu}, \quad w \in \mathcal{V} \tag{8.14c}$$

$$X_w + \sum_{u\in\delta^-(w)} Y_{uw} \leqslant 1, \quad w \in \mathcal{V} \tag{8.14d}$$

$$p(w,u) + M(w,u)(1 - Y_{wu}) \geqslant \gamma(\eta + \sum_{v\in\mathcal{V}\setminus\{w,u\}} p(v,u)X_v), \quad (w,u) \in \mathcal{A} \tag{8.14e}$$

$$U_{stwu} \leqslant \varphi_{stw},\, U_{stwu} \leqslant Y_{wu}, \quad s \in \mathcal{S}, t \in \mathscr{T}(s), w \in \mathcal{V}, u \in \delta^+(w) \tag{8.14f}$$

$$\sum_{t\in\mathscr{T}(s)} \varphi_{stw} \leqslant 1, \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{8.14g}$$

$$g_{sw} \geqslant \sum_{t\in\mathscr{T}(s)} \varphi_{stw} \quad s \in \mathcal{S}, w \in \mathcal{V} \tag{8.14h}$$

$$r \geqslant \sum_{s\in\mathcal{S}} g_{sw}, \quad w \in \mathcal{V} \tag{8.14i}$$

$$Y_{wu} \in \mathbb{B},\, w \in \mathcal{V}, u \in \delta^+(w);\; X_w \in \mathbb{B},\, w \in \mathcal{V} \tag{8.14j}$$

$$r \in \mathbb{R};\; g_{sw} \in \mathbb{R}_+,\, w \in \mathcal{V}, s \in \mathcal{S};\; \varphi_{stw} \in \mathbb{R}_+,\, s \in \mathcal{S}, t \in \mathcal{T}(s), w \in \mathcal{W} \tag{8.14k}$$

$$U_{stwu} \in \mathbb{R}_+,\, s \in \mathcal{S}, t \in \mathscr{T}(s), w \in \mathcal{V}, u \in \delta^+(w). \tag{8.14l}$$

8.3.1 TFMP: solution algorithm

As for SFMP, we apply the P&B approach for TFMP. We first solve the linear relaxation of TFMP($\widehat{\mathcal{C}}, \widehat{\mathscr{T}}$), and then apply the B&B algorithm to the MIP problem TFMP($\mathcal{C}^*, \mathscr{T}^*$) where $\mathcal{C}^*$ and $\mathscr{T}^*$ denote the final c-set list $\mathcal{C}$ and the final tree list $\mathscr{T}$ obtained from solving the relaxation. The algorithm is as follows.

Algorithm 3:

Step-1: Define an initial list $\mathcal{C}$ of feasible c-sets and an initial list $\mathscr{T}$ of feasible trees.

Step-2: Fix $\mathscr{T}$ and solve the dual master problem DMP($\widehat{\mathcal{C}}, \mathscr{T}$) (8.5), starting from $\mathcal{C}$, by c-set generation using pricing problem (8.14). Let $\mathcal{C}^*$ be the c-set list resulting from the solution (including the initial list $\mathcal{C}$ plus all c-sets that have been generated during the solution process); put $\mathcal{C} := \mathcal{C}^*$.

Step-3: Fix $\mathcal{C}$ and solve the dual master problem DMP($\mathcal{C}, \widehat{\mathscr{T}}$) (8.5), starting from $\mathscr{T}$, by tree generation using pricing problem (8.10). Let $\mathscr{T}^*$ be the tree list resulting from the solution (including the initial list $\mathscr{T}$ plus all trees that have been generated during the solution process); put $\mathscr{T} := \mathscr{T}^*$.

Step-4: If at least one tree was generated in Step-3 go to Step-2.

Step-5: Solve FFMP($\mathcal{C}^*, \mathscr{T}^*$) and stop. □

Note that the c-set list $\mathcal{C}^*$ and the tree list $\mathscr{T}^*$ used in Step-5 are sufficient to reach the global optimum of the linear relaxation of TFMP (when all c-sets and trees are considered) but they may not be sufficient to reach the optimum of TFMP. As in the case of SFMP, the presented algorithm is heuristic but produces very good suboptimal solutions.

The c-set generation procedure applied in Step-2 of the above algorithm is essentially the same as described in Step-2 and Step-3 of Algorithm 1. As far as the tree-generation is concerned, the procedure is similar to the c-set generation procedure. The only difference is that when it comes to generating trees for α^* and γ^* obtained from solving the current DMP$(\mathcal{C}, \mathscr{T})$, the pricing problem PP$(s; \alpha^*, \gamma^*)$ is solved for each $s \in \mathcal{S}$ so in general at each iteration a new tree can be added to the current tree list $\mathscr{T}$ for many streams. A reasonable initial tree list can be obtained through finding, for each $s \in \mathcal{S}$, the shortest path tree from $o(s)$ to all $w \in \mathcal{D}(s)$ using Dijkstra's shortest path algorithm.

8.4 Discussion

In many cases TFMP outperforms RFMP in terms of computational efficiency. Moreover, it poses several important features making it particularly interesting from the engineering viewpoint.

First, as it is illustrated in Figure 8.3, significant improvements of the TFMP objective function value are observed in a relatively small number of the initial iterations of the tree generation process, these improvements are then followed by a long tail of small changes. (In fact, such a behaviour is typical for the column generation technique and was observed also for the c-set generation in [18].) Because of that it could be reasonable to introduce some "soft" stopping criterion. For example we could terminate the tree generation process when the improvement of the objective function becomes small. In this way we can save a lot of time and still obtain a good suboptimal solution.

Further, since TFMP considers a set of routing trees per stream, we can use the routing trees that we know that for a given problem instance can lead to a good solution. (For example we can try to obtain such trees by applying some heuristic or approximation algorithms.) In this case we could skip the tree generation process and obtain a solution almost immediately

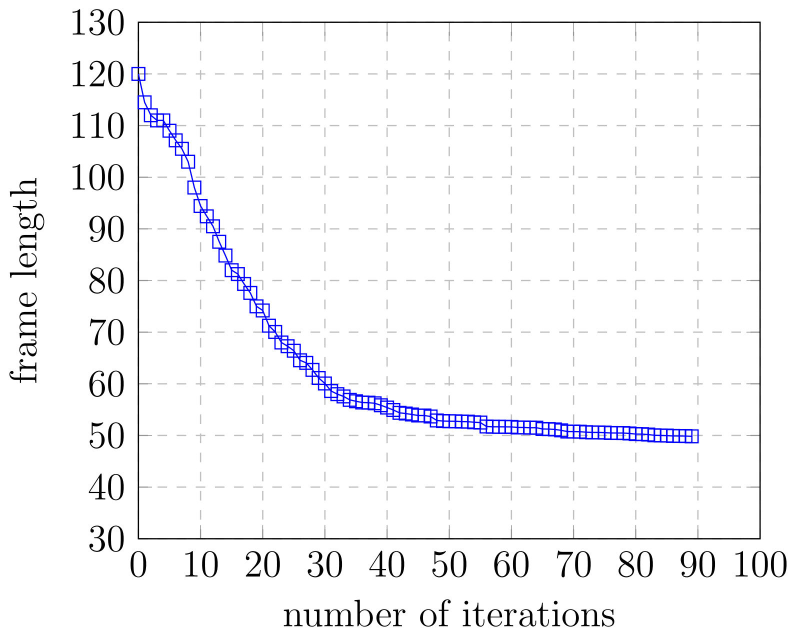

Figure 8.3: Convergence of the tree generation process for the 6 x 6 grid network.

(note that the time needed to solve the strong formulation of TFMP does not increase rapidly with the number of routing trees per stream, see Table 8.3).

Finally, by using TFMP we can easily include destination nodes selection in the optimization process, while in RFMP such an extension requires the derivation of the c-set pricing problem from the beginning. The extension of the model that introduce destination nodes optimization is shown in the next chapter.

Chapter 9

Destination nodes selection

Thus far we have assumed that the set of destination nodes $\mathcal{D}(s)$ for each stream $s \in \mathcal{S}$ is fixed and given. However, this assumption in not always proper. For example, when several gateways (bridges between WSN and external networks) with the same functionality are placed in a WSN and there is no necessity to send data from sensors to all of them, optimization of destination nodes selection becomes reasonable. When the sets of destination nodes are subject to optimization, we can treat sets $\mathcal{D}(s)$ as the potential locations of destinations nodes to which s-packets are delivered, and we require that the s-packets must be delivered to one of the $N(s)$-elements subset of $\mathcal{D}(s)$ (where $N(s)$ is a given and fixed parameter; $1 \leqslant N(s) \leqslant |\mathcal{D}(s)|$). In this chapter we extend RFMP and TFMP to destination nodes selection optimization. We end this chapter with a short numerical study. It should be noted that some aspects related to destination nodes selection are discussed in [81] and [1] (from which the formulation described in Section 9.2 was taken).

9.1 Destination nodes selection in RFMP

Destination nodes selection in RFMP can be achieved by replacing constraint (3.14d) with the following set of constraints:

$$\sum_{u \in \delta^-(o(s))} z_{sw(u,o(s))} + x_{sw} = \sum_{u \in \delta^+(o(s))} z_{sw(o(s),u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s) \tag{9.1a}$$

$$\sum_{u \in \delta^-(w)} z_{sw(u,w)} - x_{sw} = \sum_{u \in \delta^+(w)} z_{sw(w,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s) \tag{9.1b}$$

$$\sum_{u \in \delta^-(v)} z_{sw(u,v)} = \sum_{u \in \delta^+(v)} z_{sw(v,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \backslash \{o(s), w\} \tag{9.1c}$$

$$\sum_{w \in \mathcal{D}(s)} x_{sw} = N(s), \quad s \in \mathcal{S}, \tag{9.1d}$$

where $x_{sw}, s \in \mathcal{S}, w \in \mathcal{D}(s)$ are binary variables equal to 1, if and only if, node w from set $\mathcal{D}(s)$ has been chosen to be a destination node for stream s. Thus, the complete formulation of RFMP with destinations nodes selection (referred to as DRFMP) is as follows:

$$\text{DRFMP}(\mathcal{C}): \quad \min \ F = \sum_{c \in \mathcal{C}} F_c \tag{9.2a}$$

$$[\lambda_{sa} \geqslant 0] \quad \sum_{c \in \mathcal{C}(a)} h_{sb(a)c} \geqslant y_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{9.2b}$$

$$[\pi_{cw} \geqslant 0] \quad \sum_{s \in \mathcal{S}} h_{scw} \leqslant F_c, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{9.2c}$$

$$[\sigma_{swa} \geqslant 0] \quad \sum_{u \in \delta^-(o(s))} z_{sw(u,o(s))} + x_{sw} = \sum_{u \in \delta^+(o(s))} z_{sw(o(s),u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s) \tag{9.2d}$$

$$[\alpha_{sw}] \quad \sum_{u \in \delta^-(w)} z_{sw(u,w)} - x_{sw} = \sum_{u \in \delta^+(w)} z_{sw(w,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s) \tag{9.2e}$$

$$[\beta_{sw}] \quad \sum_{u \in \delta^-(v)} z_{sw(u,v)} = \sum_{u \in \delta^+(v)} z_{sw(v,u)}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \backslash \{o(s), w\} \tag{9.2f}$$

$$[\chi_s] \quad \sum_{w \in \mathcal{D}(s)} x_{sw} = N(s), \quad s \in \mathcal{S} \tag{9.2g}$$

$$[\varphi_{swv}] \quad z_{swa} \leqslant y_{sa}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{9.2h}$$

$$y_{sa} \in \mathbb{B}, \ s \in \mathcal{S}, a \in \mathcal{A}; \ \ x_{sw} \in \mathbb{B}, \ s \in \mathcal{S}, w \in \mathcal{D}(s) \tag{9.2i}$$

$$z_{swa} \in \mathbb{R}_+, \ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{9.2j}$$

$$h_{scw} \in \mathbb{B}, \ s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c); \ \ F_c \in \mathbb{R}, \ c \in \mathcal{C}. \tag{9.2k}$$

The dual to the linear relaxation of (9.2):

$$\text{DMP}(\mathcal{C}): \quad \max \ W = \sum_{s \in \mathcal{S}} \chi_s N(s) \tag{9.3a}$$

$$\sum_{w \in D(s)} \sigma_{swa} \leqslant \lambda_{sa}, \quad s \in \mathcal{S}, a \in \mathcal{A} \tag{9.3b}$$

$$\sum_{w \in \mathcal{W}(c)} \pi_{cw} = 1, \quad c \in \mathcal{C} \tag{9.3c}$$

$$\sum_{v \in \mathcal{U}(c,w)} \lambda_{s(w,v)} \leqslant \pi_{cw}, \quad s \in \mathcal{S}, c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{9.3d}$$

$$-\alpha_{sw} + \beta_{sw} - \chi_s \geqslant 0, \quad s \in \mathcal{S}, w \in \mathcal{D}(s) \tag{9.3e}$$

$$\begin{aligned} &\sigma_{swa} + \alpha_{sw} I^+(a, o(s)) - \alpha_{sw} I^-(a, o(s)) \\ &+\beta_{sw} I^+(a, w) - \beta_{sw} I^-(a, w) + \sum_{v \in \mathcal{V} \backslash \{o(s), w\}} \varphi_{swv} I^+(a, v) \\ &- \sum_{v \in \mathcal{V} \backslash \{o(s), w\}} \varphi_{swv} I^-(a, v) \geqslant 0, \quad s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \end{aligned} \tag{9.3f}$$

$$\varphi_{swv} \in \mathbb{R}, \ s \in \mathcal{S}, w \in \mathcal{D}(s), v \in \mathcal{V} \tag{9.3g}$$

$$\alpha_{sw} \in \mathbb{R},\ s \in \mathcal{S}, w \in \mathcal{D}(s);\ \ \beta_{sw} \in \mathbb{R}, \quad s \in \mathcal{S}, w \in \mathcal{D}(s) \tag{9.3h}$$

$$\sigma_{swa} \in \mathbb{R}_+,\ s \in \mathcal{S}, w \in \mathcal{D}(s), a \in \mathcal{A} \tag{9.3i}$$

$$\lambda_{sa} \in \mathbb{R}_+,\ s \in \mathcal{S}, a \in \mathcal{A};\ \ \pi_{cw} \in \mathbb{R}_+,\ c \in \mathcal{C}, w \in \mathcal{W}(c), \tag{9.3j}$$

where $I^+(a,w),\ a \in \mathcal{A}, w \in \mathcal{W}$ is a parameter equals to 1 if, and only if, $b(a) = w \wedge e(a) \in \delta^-(w)$, and $I^+(a,w),\ a \in \mathcal{A}, w \in \mathcal{W}$, is a parameter equals to 1 if, and only if, $e(a) = w \wedge b(a) \in \delta^+(w)$. Note that since constraints (9.3c) and (9.3d) in the above dual formulation remains the same as in its counterpart for RFMP ((3.15d) and (3.15e)), the pricing problem for c-set generation in DRFMP also remains the same and is expressed with formulation (3.20).

9.2 Destination nodes selection in TFMP

Destination nodes selection can be included also in TFMP. In fact, this extension (referred to as DTFMP) of the problem setting does not change the TFMP formulation (8.4), so that DTFMP$(\mathcal{C}, \mathscr{T})$ is identical to TFMP$(\mathcal{C}, \mathscr{T})$. The only difference is the pricing problem for tree generation, where optimization of the destination nodes selection is included. A version of PP adequate for DTFMP, a modification of its TFMP counterpart (8.10), is given below. Additional (binary) variables used in formulation (9.4), $y_w, w \in \mathcal{D}(s)$, determine the subset $\mathcal{D}'(s)$ of nodes in set $\mathcal{D}(s)$ of potential destinations for stream $s \in \mathcal{S}$ that are actually selected for the destinations $\mathcal{D}'(s) := \{w \in \mathcal{D}(s) :\ y_w = 1\}$.

$$\text{PP}(s; \alpha^*, \gamma^*):\ \ \max\ \ P(s) = r\alpha^*_s - \textstyle\sum_{c \in \mathcal{C}} \sum_{w \in \mathcal{W}(c)} \gamma^*_{scw} m_{cw} \tag{9.4a}$$

$$\textstyle\sum_{c \in \mathcal{C}(a)} U_{ca} \geqslant p_a, \quad a \in \mathcal{A} \tag{9.4b}$$

$$e \leqslant 1;\ \ m_{cw} \leqslant 1, \quad c \in \mathcal{C}, w \in \mathcal{W}(c) \tag{9.4c}$$

$$p_a \leqslant r,\ p_a \leqslant x_a,\ p_a \geqslant r + x_a - 1, \quad a \in \mathcal{A} \tag{9.4d}$$

$$U_{ca} \leqslant m_{cb(a)},\ U_{ca} \leqslant x_a,\ U_{ca} \geqslant m_{cb(a)} + x_a - 1, \quad c \in \mathcal{C}(a), a \in \mathcal{A} \tag{9.4e}$$

$$\textstyle\sum_{u \in \delta^-(o(s))} z_{w(u,o(s))} + y_w = \sum_{u \in \delta^+(o(s))} z_{w(o(s),u)}, \quad w \in \mathcal{D}(s) \tag{9.4f}$$

$$\textstyle\sum_{u \in \delta^-(w)} z_{w(u,w)} - y_w = \sum_{u \in \delta^+(w)} z_{w(w,u)}, \quad w \in \mathcal{D}(s) \tag{9.4g}$$

$$\textstyle\sum_{u \in \delta^-(v)} z_{w(u,v)} = \sum_{u \in \delta^+(v)} z_{w(v,u)}, \quad w \in \mathcal{D}(s), v \in \mathcal{V} \backslash \{o(s), w\} \tag{9.4h}$$

$$\sum_{w \in \mathcal{D}(s)} y_w = N(s) \quad (9.4i)$$

$$z_{wa} \leqslant x_a, \quad w \in \mathcal{D}(s), a \in \mathcal{A} \quad (9.4j)$$

$$\sum_{w \in \delta^-(o(s))} x_{(w,o(s))} = 0, \ \sum_{w \in \delta^-(v)} x_{(w,v)} \leqslant 1, \quad v \in \mathcal{V} \backslash \{o(s)\}, \quad (9.4k)$$

$$r \in \mathbb{R}_+; \ m_{cw} \in \mathbb{R}_+, \ c \in \mathcal{C}, w \in \mathcal{W}(c); \ U_{ca} \in \mathbb{R}_+, \ c \in \mathcal{C}(a), a \in \mathcal{A} \quad (9.4l)$$

$$y_w \in \mathbb{B}, \ w \in \mathcal{D}(s); \ z_{wa} \in \mathbb{R}_+, \ w \in \mathcal{V}, a \in \mathcal{A}; \ x_a \in \mathbb{B}, \ p_a \in \mathbb{R}_+, \ a \in \mathcal{A}. \quad (9.4m)$$

9.3 Numerical study

Below we present a short numerical study that illustrates the influence of destination nodes selection optimization on the frame length. The optimization problems were implemented in C# and executed on an Intel Core i7-8550U CPU (4 cores, each up to 4GHz) and 16 GB RAM. CPLEX 12.8 solver was used to solve LP and MIP problems. The results presented in this section were taken from [1].

9.3.1 Network setting

In this study we consider three irregular networks depicted in Figures 9.1-9.3, generated by [73]. For each network, the nodes are placed randomly in a given square area according to the uniform distribution. The areas are as follows: 230 m × 230 m for Network 1, 235 m × 235 m for Network 2, and 240 m × 240 m for Network 3. Each network has 36 nodes and the number of links is equal to 231, 200, and 169, respectively for Network 1, Network 2, and Network 3. In each network there are 12 source nodes (red nodes in the figures), 6 destination nodes (blue nodes), and 18 transit nodes (white nodes). We assume that the source nodes are capable of transiting packets (like the transit nodes), but the destination nodes can only terminate packet traffic.

In this study we use the same propagation model (and the same values of the radio parameters) as in Chapters 6, 7, and 8.

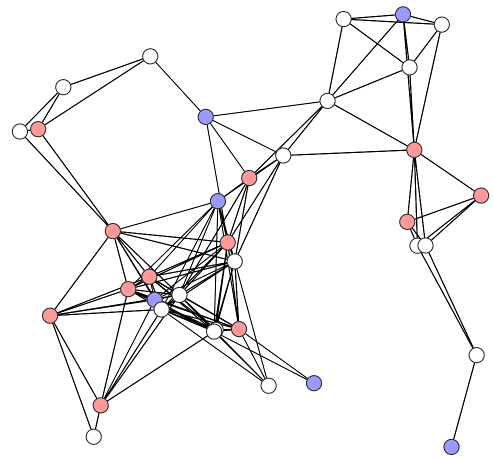

Figure 9.1: Network 1.

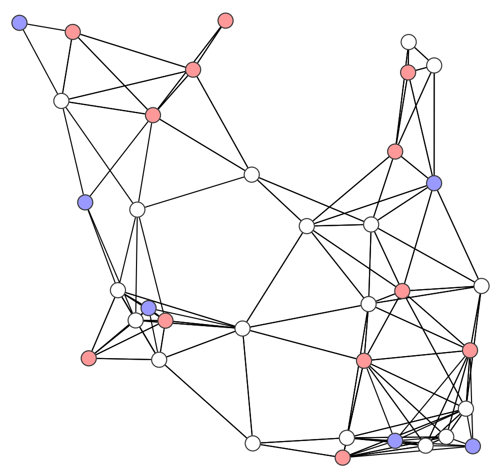

Figure 9.2: Network 2.

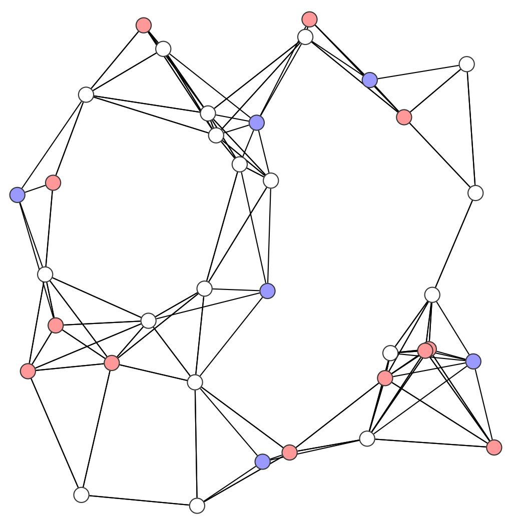

Figure 9.3: Network 3.

9.3.2 Influence of the destination nodes selection optimization

In this section we study the influence of the destination nodes selection optimization on the frame length examined by means of three optimization problems SFMP, TFMP, and DTFMP. We assume that the sets of the potential locations of destinations nodes ($\mathcal{D}(s), s \in \mathcal{S}$) are the same for all streams and contain all destination nodes in a network. In each considered case, the number of the actual destination nodes is the same for all streams and is denoted by N, while the minimum frame length obtained for a given N is denoted by $F(N)$. In SFMP and TFMP, for each N and for each stream, the destination nodes are obtained by finding N nearest destination nodes (among all destination nodes in a network) through Dijkstra's algorithm, which at the same time produces the shortest path trees used as the multicast routing trees in SFMP. The same trees are used in TFMP and DTFMP as the initial sets of multicast trees. In all problems, the initial family of c-sets consists of all one-node c-sets. The optimization frame lengths (expressed in the number of time slots) for the considered problems and the different number of the destination nodes are presented in Tables 9.1-9.3 and illustrated in Figures 9.4-9.6.

Table 9.1: The optimized frame lengths for Network 1.

	$F(1)$	$F(2)$	$F(3)$	$F(4)$	$F(5)$	$F(6)$
SFMP	8	12	20	28	34	46
TFMP	8	12	15	19	26	34
DTFMP	7	9	14	18	26	34

Table 9.2: The optimized frame lengths for Network 2.

	$F(1)$	$F(2)$	$F(3)$	$F(4)$	$F(5)$	$F(6)$
SFMP	5	10	16	26	33	38
TFMP	5	10	15	21	26	28
DTFMP	5	8	11	19	23	28

First, the results show that for $N \geqslant 3$, routing optimization leads to a significant decrease in the frame length in each considered network. The value of this decrease is in some cases

Table 9.3: The optimized frame lengths for Network 3.

	$F(1)$	$F(2)$	$F(3)$	$F(4)$	$F(5)$	$F(6)$
SFMP	6	10	18	25	33	38
TFMP	6	10	16	20	25	30
DTFMP	6	9	13	18	24	30

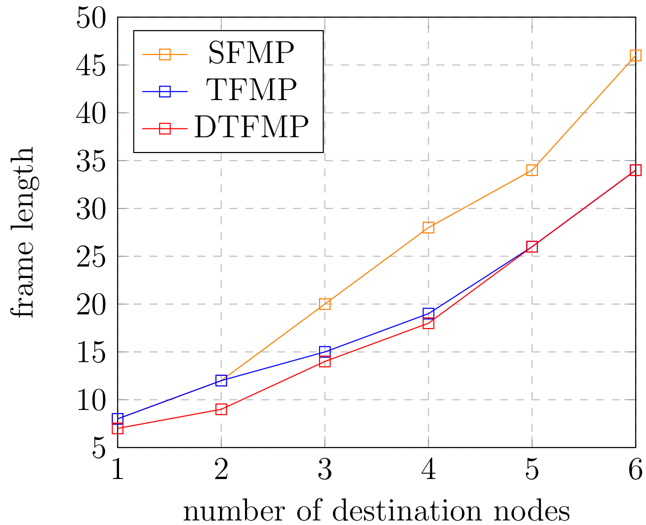

Figure 9.4: Frame length vs. number of destination nodes - Network 1.

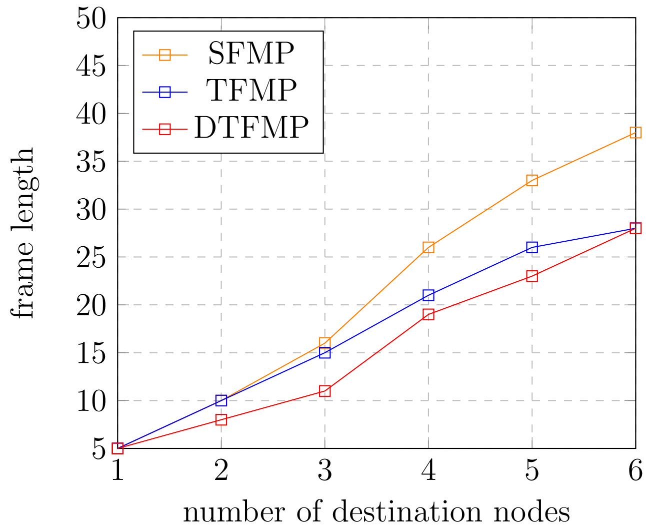

Figure 9.5: Frame length vs. number of destination nodes - Network 2.

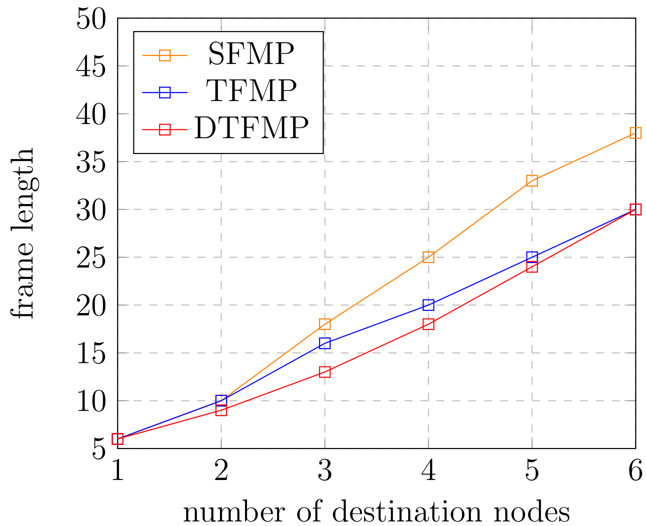

Figure 9.6: Frame length vs. number of destination nodes - Network 3.

really significant, and it varies from 6.25% (Network 2, $N = 3$) to 27.32% (Network 1, $N = 4$). (Recall that the shorter the frame, the greater traffic throughput.) For more results illustrating the influence of routing optimization on the frame length the reader is referred to [18].

Next, the frame can be further shortened by introducing destination nodes selection optimization. For the considered networks the largest decreases in the frame length are observed for $N = 2$ and $N = 3$, and their value are equal to 18.75% (Network 3, $N = 3$), 20% (Network 2, $N = 2$), 25% (Network 1, $N = 2$), and 26.67% (Network 2, $N = 3$).

Finally, it should be noted that the results strongly depend on a particular network topology. Therefore, network topology may play an important role in choosing the most appropriate strategy for destination nodes selection.

9.3.3 Efficiency of the optimization model

In this section we analyze computational efficiency of the considered optimization problems. For this purpose we consider Network 2, as this case turned out to be the most computationally demanding. The notations used in this chapter are presented in Table 9.4. The results are presented in Table 9.5 and lead to the following observations.

First, SFMP is solved efficiently (within seconds) for each N, while its extended versions

Table 9.4: Numerical results—explanation.

Notation	Description
F^0	frame length obtained from the initial linear relaxation of the problem
F^{LR}	frame length obtained from the final linear relaxation of the problem
F^{MIP}	frame length obtained from the final MIP version of the problem
|TPP|	number of tree pricing problem executions (for all s in all steps)
|CPP|	number of c-set pricing problem executions (in all steps)
$\|T^g\|$	number of generated trees (for all s in all steps)
$\|\mathcal{C}^g\|$	number of generated c-sets (in all steps)
t/MP	average computation time required for solving the master problem
t/CPP	average computation time required for solving the c-set pricing problem
t/TPP	average computation time required for solving the tree pricing problem
t/MIP	computation time of solving the final MIP version of the problem
total	total computation time

(TFMP and DTFMP) turn out to be much more computationally demanding (e.g., for $N = 6$ the total computation time exceeds 5 hours). Note that for $N \geqslant 3$ relative differences in the total computation time of FTMP and DTFMP are quite small. Thus, in these cases, it is tree optimization that makes the problem difficult.

Second, it is important to note that the linear relaxations of the problems are of high quality—in most cases F^{MIP} is equal to the ceiling of F^{LR}. This advantageous feature helps to improve the performance of the B&B algorithm underlying the MIP solver. Note that even for $N = 6$ the time needed to solve the MIP versions of the problems does not exceed several minutes.

Finally, as the time needed to solve the MIP versions of TFMP and DTFMP is in fact negligible, the long computation time in these cases stems from the large number of generated trees and c-sets. As discussed in [18] and in Section 8.4 this is mainly because the c-sets/trees generated in the late stages of the c-set/tree generation process improve the value of the linear relaxation only to a very limited extent, and most of the master problem objective function improvements are achieved already in the initial stages. Similar behaviour was noticed here.

Thus, a soft stopping criterion could result in substantial time savings, at the expense of a reasonable increase in the achieved frame length.

Table 9.5: Efficiency of the optimization model.

N		F^0	F^{LR}	F^{MIP}	|TPP|	$\|T^g\|$	|CPP|	$\|\mathcal{C}^g\|$	t/MP	t/TPP	t/CPP	t/MIP	total
1	SFMP	14	5	5	-	-	13	12	0.09s	-	0.33s	0.06s	5.73s
	TFMP	14	5	5	12	0	13	12	0.06s	0.05s	0.26s	0.04s	5.01s
	DTFMP	14	5	5	25	4	17	14	0.05s	0.18s	0.27s	0.06s	8.41s
2	SFMP	27	10	10	-	-	25	24	0.06s	-	0.29s	0.05s	9.20s
	TFMP	27	10	10	53	8	29	26	0.06s	0.11s	0.26s	0.06s	15.58s
	DTFMP	27	7.47	8	143	41	90	84	0.05s	1.17s	3.72s	1.24s	9m11s
3	SFMP	42	15.75	16	-	-	36	35	0.07s	-	0.33s	0.11s	14.85s
	TFMP	42	14.63	15	190	52	114	109	0.27s	0.75s	1.72s	1s	8m26s
	DTFMP	42	10.52	11	276	115	171	166	0.61s	3.75s	9.52s	5.54s	47m32s
4	SFMP	64	26	26	-	-	28	27	0.06s	-	0.35s	0.03s	12.01s
	TFMP	64	20.71	21	324	125	208	201	0.92s	13.41s	13.24s	25.95s	2h4m
	DTFMP	64	17.86	19	300	146	190	186	1.01s	16.05s	19.96s	53.24s	2h24m
5	SFMP	79	33	33	-	-	27	26	0.05s	-	0.39s	0.03s	12.39s
	TFMP	79	24.94	26	384	138	202	193	1.17s	11.15s	39.75s	1m2s	3h33m
	DTFMP	79	21.88	23	348	144	204	199	1.19s	27.75s	19.51s	30.79s	3h55m
6	SFMP	99	37	38	-	-	34	33	0.06s	-	0.41s	0.04s	16.64s
	TFMP	99	27.34	28	396	164	209	203	1.56s	31.40s	40.11s	2m41s	5h59m
	DTFMP	99	27.32	28	492	196	198	192	1.63s	27.49s	23.01s	2m51s	5h15m

Chapter 10

Final comments

This chapter gives the final comments on the optimization model presented in this thesis. First, we briefly summarize the thesis. Next, we assess our contribution by pointing out strengths and weaknesses of the introduced optimization model. Finally, we discuss potential directions of future work.

10.1 Conclusions

The goal formulated in the introduction was achieved. By extending the work described in [18] we provide a comprehensive optimization model for MHWN with TDMA access to radio channel that serve multicast periodic traffic. In the model we include the most important aspects related to MHWN. In Chapters 4 and 5 we deal with traffic throughput maximization by means of modulation and coding schemes assignment (both dynamic and static) and transmission power control (both continuous and discrete). Chapter 6 deals with the energy consumption minimization problem and studies the tradeoff between the frame length and energy consumption. In Chapter 7 we focus on the packet delay minimization of the optimized frame length. An efficient heuristic algorithm was also proposed for solving this problem. Finally, Chapter 8 describes an alternative (to Chapter 3) formulation for routing optimization, while Chapter 9 extends the routing optimization by including destination nodes selection optimization. The considerations of this thesis were illustrated by numerous numerical examples.

MHWN have been important research objects for a long time, and because of the current rapid growth of IoT their importance will even increase in the coming years. The optimization model introduced in this thesis fills the gap observed in the literature, and allows us to treat a number of problems related to MHWN in a rigid mathematical way. The main application of this model is to yield performance evaluation of the networks in question. Note that in this context the fact that we assume TDMA access to radio channel is especially valuable since the results provided by the model can be treated as upper bounds for other MAC schemes. Moreover, with the advent of SDN (for work on SDN in MHWN the reader is referred to [82, 83, 84, 85]) and its logically centralized network control, the scheduling obtained from the model becomes potentially applicable. It seems that such an attempt could be reasonable especially for WMN which are characterized by stable topologies.

10.2 Contribution

The main strength of our optimization model for MHWN is its comprehensiveness. This means that we not only deal with the most important performance metrics such as traffic throughput, energy consumption, and packet delay, but we also handle the aspects that are not common in the literature, i.e., physical interferences and multicast traffic. The physical interference model is the most accurate (and the most computationally demanding) interference model, which is often simplified in the literature by using the protocol interference model (relation between the physical interference model and the protocol interference model can be found in [86]). We also consider a general traffic model, i.e., multicast traffic which is hardly considered in the literature dealing with mathematical modelling.

Obviously, our optimization model has also some weaknesses out of which the main issue is its scalability. Although due to $\mathcal{NP}$-hardness of the considered problems we cannot expect that any problem instance can be solved within seconds or even minutes, there are some things we can try to do to make this problem less severe. First, we can try to improve the strength of the linear relaxation of the formulations including routing optimization (used in Chapters 4-5). Notice that this linear relaxation is quite weak which certainly influences the computation time needed to solve the MIP version of the problem, as well as the time needed to perform the c-set generation process (note that we generate c-sets useful from the linear relaxation viewpoint). Then, we can try to modify the MIP formulation from Chapter 7. It seems that the idea behind the formulation given in [64] can also be applied in this case to decrease the number of variables in the formulation. However, even with such a modification the formulation would remain complicated, and thus only slight improvement in its computational efficiency can be expected. Finally, let us note that scalability would not be an issue if we wanted to consider multi-path routing (such routing is widely studied in the MHWN context, see for example [87] and [88]). Although introducing multi-path routing in BOM seems to be unrealistic (since we deal with single packets), such routing could be applied in the formulations from Chapters 4-5 where multicast packet streams are characterized by the amount of Mb per frame. This can be done by removing integrality constraints imposed on variables y_{sa} which leads to the LP formulation that can be effectively solved to optimality by means of the column generation technique.

10.3 Future work

Certainly, issues related to MHWN are numerous (for a comprehensive list of design problems the reader is referred to [89]); therefore, it is clear that our model can be further modified and extended. First, the modifications mentioned in the previous section can be applied. Then, some aspects that are out of the scope of this thesis can be considered. From the vast amount of problems to be solved in MHWN, three problems seem to be of a particular importance. The first problem is traffic scheduling in duty-cycled WSN. Such networks are especially important when energy efficient solutions are needed. Hence, in the recent years they have become an important research object. Then, much remains to be done in the field of network resilience (this aspect was to some extent addressed in [81] and [1]). Finally, the problem of determining appropriate nodes location can also be included in the model. Besides these major aspects, recent radio technologies like MIMO or beamforming (see for example [90] and [91]) are worth considering.

Bibliography

[1] B. Ostrowski, M. Pióro, A. Tomaszewski, E. Fitzgerald, Resilience through multicast – an optimization model for multi-hop wireless sensor networks, Ad Hoc Networks 107 (2020) 102239. doi:https://doi.org/10.1016/j.adhoc.2020.102239.
URL https://www.sciencedirect.com/science/article/pii/S1570870520301116

[2] T. Tran, D. Navrátil, P. Sanders, J. Hart, R. Odarchenko, C. Barjau, B. Altman, C. Burdinat, D. Gomez-Barquero, Enabling multicast and broadcast in the 5g core for converged fixed and mobile networks, IEEE Transactions on Broadcasting 66 (2) (2020) 428–439. doi:10.1109/TBC.2020.2991548.

[3] Internet of things: Wireless sensor networks, Tech. rep., International Electrotechnical Commission (2014).

[4] 5g use cases and requirements, Tech. rep., Nokia.

[5] X. Zhang, L. Dong, H. Peng, H. Chen, S. Zhao, C. Li, Collusion-aware privacy-preserving range query in tiered wireless sensor networks, Sensors 14 (12) (2014) 23905–23932. doi:10.3390/s141223905.
URL https://www.mdpi.com/1424-8220/14/12/23905

[6] M. Mamdouh, M. A. I. Elrukhsi, A. Khattab, Securing the internet of things and wireless sensor networks via machine learning: A survey, in: 2018 International Conference on Computer and Applications (ICCA), 2018, pp. 215–218. doi:10.1109/COMAPP.2018.8460440.

[7] Y. Liu, K.-F. Tong, X. Qiu, Y. Liu, X. Ding, Wireless mesh networks in iot networks, in: 2017 International Workshop on Electromagnetics: Applications and Student Innovation Competition, 2017, pp. 183–185. doi:10.1109/iWEM.2017.7968828.

[8] S. Bouckaert, E. De Poorter, P. De Mil, I. Moerman, P. Demeester, Interconnecting wireless sensor and wireless mesh networks: Challenges and strategies, in: GLOBECOM 2009 - 2009 IEEE Global Telecommunications Conference, 2009, pp. 1–7. doi:10.1109/GLOCOM.2009.5425861.

[9] I. F. Akyildiz, X. Wang, W. Wang, Wireless mesh networks: a survey, Computer Networks 47 (4) (2005) 445–487. doi:https://doi.org/10.1016/j.comnet.2004.12.001. URL https://www.sciencedirect.com/science/article/pii/S1389128604003457

[10] D. Benyamina, A. Hafid, M. Gendreau, Wireless mesh networks design — a survey, IEEE Communications Surveys Tutorials 14 (2) (2012) 299–310. doi:10.1109/SURV.2011.042711.00007.

[11] V. Potdar, A. Sharif, E. Chang, Wireless sensor networks: A survey, in: 2009 International Conference on Advanced Information Networking and Applications Workshops, 2009, pp. 636–641. doi:10.1109/WAINA.2009.192.

[12] P. Rawat, K. Singh, H. Chaouchi, J. Bonnin, Wireless Sensor Networks: a Survey on Recent Developments and Potential Synergies, J Supercomput 68 (2014) 1–48.

[13] O. Khader, A. Willig, A. Wolisz, WirelessHART TDMA Protocol Performance Evaluation Using Response Surface Methodology, in: IEEE BWCCA, 2011, pp. 197–206, published online, DOI: 10.1109/BWCCA.2011.32.

[14] M. Nixon, A Comparison of WirelessHART and ISA100.11, White Paper: HCF_SPEC-xxx, 2012, revision 1.0, Preliminary A.

[15] E. Shayo, P. Mafole, A. Mwambela, A survey on time division multiple access scheduling algorithms for industrial networks, SN Applied Sciences 2 (12) (2020) 2140. doi:10.1007/s42452-020-03923-4.
URL https://doi.org/10.1007/s42452-020-03923-4

[16] V. Cionca, T. Newe, V. Dadârlat, Tdma protocol requirements for wireless sensor networks, in: 2008 Second International Conference on Sensor Technologies and Applications (sensorcomm 2008), 2008, pp. 30–35.

[17] O. Yevsyeyeva, A.-A. E. Mohammed, Mathematical model for resource allocation in tdma-based wireless meshnetworks, Eastern-European Journal of Enterprise Technologies 3 (9) (2014) 4–9. doi:10.15587/1729-4061.2014.24617.

[18] M. Pióro, A. Tomaszewski, A. Capone, Maximization of multicast periodic traffic throughput in multi-hop wireless networks with broadcast transmissions, Ad Hoc Networks 77 (2018) 119–142.

[19] A. Capone, G. Carello, I. Filippini, S. Gualandi, F. Malucelli, Routing, scheduling and channel assignment in wireless mesh networks: Optimization models and algorithms, Ad Hoc Networks 8 (6) (2010) 545–563. doi:https://doi.org/10.1016/j.adhoc.2009.11.003.
URL https://www.sciencedirect.com/science/article/pii/S1570870509001152

[20] J. S. Silva, T. Camilo, A. Rodrigues, M. Silva, F. Gaudencio, F. Boavida, Multicast in wireless sensor networks the next step, in: 2007 2nd International Symposium on Wireless Pervasive Computing, 2007. doi:10.1109/ISWPC.2007.342598.

[21] G. Wagenknecht, M. Anwander, M. Brogle, T. Braun, Reliable multicast in wireless sensor networks, 2008, pp. 69–72.

[22] K. Ajish Kumar, S. Hegde, Multicasting in wireless mesh networks: Challenges and opportunities, in: 2009 International Conference on Information Management and Engineering, 2009, pp. 514–518. doi:10.1109/ICIME.2009.92.

[23] G. Araniti, M. Condoluci, P. Scopelliti, A. Molinaro, A. Iera, Multicasting over emerging 5g networks: Challenges and perspectives, IEEE Network 31 (2) (2017) 80–89. doi:10.1109/MNET.2017.1600067NM.

[24] L. Su, B. Ding, Y. Yang, T. F. Abdelzaher, G. Cao, J. C. Hou, ocast: Optimal multicast routing protocol for wireless sensor networks, in: 2009 17th IEEE International Conference on Network Protocols, 2009, pp. 151–160. doi:10.1109/ICNP.2009.5339689.

[25] R. S. Krishnan, E. Golden Julie, Y. Harold Robinson, L. H. Son, R. Kumar, M. Abdel-Basset, P. H. Thong, A new algorithm for high power node multicasting in wireless sensor networks, IEEE Access 7 (2019) 38584–38592. doi:10.1109/ACCESS.2019.2905135.

[26] S. Mohapatra, B. K. Ratha, K. Tanaya Dhal, Implementation of multipath-based multicast routing protocol in hierarchical wireless sensor network, in: S. Borah, V. Emilia Balas, Z. Polkowski (Eds.), Advances in Data Science and Management, Springer Singapore, Singapore, 2020, pp. 345–354.

[27] D.-R. Pan, Y. Xue, L.-J. Zhan, A multicast wireless mesh network (wmn) network routing algorithm with ant colony optimization, in: 2008 International Conference on Wavelet Analysis and Pattern Recognition, Vol. 2, 2008, pp. 744–748. doi:10.1109/ICWAPR.2008.4635876.

[28] S. Roy, D. Koutsonikolas, S. Das, Y. C. Hu, High-throughput multicast routing metrics in wireless mesh networks, Ad Hoc Networks 6 (6) (2008) 878–899. doi:https://doi.org/10.1016/j.adhoc.2007.07.005.
URL https://www.sciencedirect.com/science/article/pii/S1570870507001321

[29] W. Shi, S. Wang, Z. Wang, E. Wang, An efficient channel assignment algorithm for multicast wireless mesh networks, AEU - International Journal of Electronics and Communications 89 (2018) 62–69. doi:https://doi.org/10.1016/j.aeue.2018.03.023.
URL https://www.sciencedirect.com/science/article/pii/S1434841117315443

[30] V. Cionca, T. Newe, V. Dadârlat, Tdma protocol requirements for wireless sensor networks, in: 2008 Second International Conference on Sensor Technologies and Applications (sensorcomm 2008), 2008, pp. 30–35. doi:10.1109/SENSORCOMM.2008.69.

[31] F. Kauer, V. Turau, An analytical model for wireless mesh networks with collision-free tdma and finite queues, EURASIP Journal on Wireless Communications and Networking

2018 (1) (2018) 149. doi:10.1186/s13638-018-1146-x.
URL https://doi.org/10.1186/s13638-018-1146-x

[32] P. Cappanera, L. Lenzini, A. Lori, G. Stea, G. Vaglini, Optimal joint routing and link scheduling for real-time traffic in tdma wireless mesh networks, Computer Networks 57 (11) (2013) 2301–2312. doi:https://doi.org/10.1016/j.comnet.2012.11.021.
URL https://www.sciencedirect.com/science/article/pii/S138912861300008X

[33] S. Kumar, H. Kim, Energy efficient scheduling in wireless sensor networks for periodic data gathering, IEEE Access 7 (2019) 11410–11426. doi:10.1109/ACCESS.2019.2891944.

[34] W. Osamy, A. A. El-Sawy, A. M. Khedr, Effective tdma scheduling for tree-based data collection using genetic algorithm in wireless sensor networks, Peer-to-Peer Networking and Applications 13 (3) (2020) 796–815. doi:10.1007/s12083-019-00818-z.
URL https://doi.org/10.1007/s12083-019-00818-z

[35] M. Yao, C. Lin, P. Zhang, Y. Tian, S. Xu, Tdma scheduling with maximum throughput and fair rate allocation in wireless sensor networks, in: 2013 IEEE International Conference on Communications (ICC), 2013, pp. 1576–1581. doi:10.1109/ICC.2013.6654739.

[36] M. Cheng, Q. Ye, L. Cai, Cross-Layer Schemes for Reducing Delay in Multihop Wireless Networks, IEEE Transactions on Wireless Communications 12 (2) (2013) 928–937.

[37] R. Cruz, A. Santhanam, Optimal routing, link scheduling and power control in multihop wireless networks, in: IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428), Vol. 1, 2003, pp. 702–711 vol.1. doi:10.1109/INFCOM.2003.1208720.

[38] G. Kulkarni, V. Raghunathan, M. Srivastava, Joint end-to-end scheduling, power control and rate control in multi-hop wireless networks, in: IEEE Global Telecommunications Conference, 2004. GLOBECOM '04., Vol. 5, 2004, pp. 3357–3362 Vol.5. doi:10.1109/GLOCOM.2004.1378971.

[39] M. Johansson, L. Xiao, Cross-layer optimization of wireless networks using nonlinear column generation, IEEE Transactions on Wireless Communications 5 (2) (2006) 435–445. doi:10.1109/TWC.2006.1611067.

[40] L. Tassiulas, A. Ephremides, Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks, IEEE Transactions on Automatic Control 37 (12) (1992) 1936–1948. doi:10.1109/9.182479.

[41] M. Pióro, M. Żotkiewicz, B. Staehle, D. Staehle, D. Yuan, On max–min fair flow optimization in wireless mesh networks, Ad Hoc Networks 13 (2014) 134–152. doi:https://doi.org/10.1016/j.adhoc.2011.05.003.
URL https://www.sciencedirect.com/science/article/pii/S1570870511000989

[42] A. Mehrabi, K. Kim, General framework for network throughput maximization in sink-based energy harvesting wireless sensor networks, IEEE Transactions on Mobile Computing 16 (7) (2017) 1881–1896. doi:10.1109/TMC.2016.2607716.

[43] J. Li, B. N. Silva, M. Diyan, Z. Cao, K. Han, A clustering based routing algorithm in iot aware wireless mesh networks, Sustainable Cities and Society 40 (2018) 657–666. doi:https://doi.org/10.1016/j.scs.2018.02.017.
URL https://www.sciencedirect.com/science/article/pii/S2210670717312775

[44] X. Deng, J. Luo, L. He, Q. Liu, X. Li, L. Cai, Cooperative channel allocation and scheduling in multi-interface wireless mesh networks, Peer-to-Peer Networking and Applications 12 (1) (2019) 1–12. doi:10.1007/s12083-017-0619-8.
URL https://doi.org/10.1007/s12083-017-0619-8

[45] Y. Ge, Y. Nan, X. Guo, Maximizing network throughput by cooperative reinforcement learning in clustered solar-powered wireless sensor networks, International Journal of Distributed Sensor Networks 17 (4) (2021) 15501477211007411. arXiv:https://doi.org/10.1177/15501477211007411, doi:10.1177/15501477211007411.
URL https://doi.org/10.1177/15501477211007411

[46] A. Capone, F. Malandra, B. Sansò, Energy savings in wireless mesh networks in a time-variable context, Mob. Netw. Appl. 17 (2) (2012) 298–311. doi:10.1007/s11036-011-0339-x.
URL https://doi.org/10.1007/s11036-011-0339-x

[47] A. Ouni, H. Rivano, F. Valois, C. Rosenberg, Energy and throughput optimization of wireless mesh networks with continuous power control, IEEE Transactions on Wireless Communications 14 (2) (2015) 1131–1142. doi:10.1109/TWC.2014.2364815.

[48] N. Gaddam, S. A. Gathala, D. Lastine, A. Somani, Energy minimization through network coding for lifetime constrained wireless networks, in: 2008 Proceedings of 17th International Conference on Computer Communications and Networks, 2008, pp. 1–6. doi:10.1109/ICCCN.2008.ECP.97.

[49] A. Awad, O. A. Nasr, M. M. Khairy, Energy-aware routing for delay-sensitive applications over wireless multihop mesh networks, in: 2011 7th International Wireless Communications and Mobile Computing Conference, 2011, pp. 1075–1080. doi:10.1109/IWCMC.2011.5982690.

[50] P. Jayarajan, R. Maheswar, G. R. Kanagachidambaresan, Modified energy minimization scheme using queue threshold based on priority queueing model, Cluster Computing 22 (5) (2019) 12111–12118.

[51] S. K. Jha, E. M. Eyong, An energy optimization in wireless sensor networks by using genetic algorithm, Telecommunication Systems 67 (1) (2018) 113–121. doi:10.1007/s11235-017-0324-1.
URL https://doi.org/10.1007/s11235-017-0324-1

[52] R. Arya, S. C. Sharma, Energy optimization of energy aware routing protocol and bandwidth assessment for wireless sensor network, International Journal of System Assurance Engineering and Management 9 (3) (2018) 612–619. doi:10.1007/s13198-014-0289-3.
URL https://doi.org/10.1007/s13198-014-0289-3

[53] M. Radhika, P. Sivakumar, Energy optimized micro genetic algorithm based leach protocol for wsn, Wireless Networks 27 (1) (2021) 27–40. doi:10.1007/s11276-020-02435-8.
URL https://doi.org/10.1007/s11276-020-02435-8

[54] N. Ajmi, A. Helali, P. Lorenz, R. Mghaieth, Mwcsga—multi weight chicken swarm based genetic algorithm for energy efficient clustered wireless sensor network, Sensors 21 (3) (2021). doi:10.3390/s21030791.
URL https://www.mdpi.com/1424-8220/21/3/791

[55] H. Li, Y. Cheng, C. Zhou, W. Zhuang, Minimizing end-to-end delay: A novel routing metric for multi-radio wireless mesh networks, in: IEEE INFOCOM 2009, 2009, pp. 46–54. doi:10.1109/INFCOM.2009.5061905.

[56] Y. Chai, X.-J. Zeng, Delay- and interference-aware routing for wireless mesh network, IEEE Systems Journal 14 (3) (2020) 4119–4130. doi:10.1109/JSYST.2020.2966795.

[57] G. R. Gupta, N. B. Shroff, Delay analysis and optimality of scheduling policies for multihop wireless networks, IEEE/ACM Transactions on Networking 19 (1) (2011) 129–141. doi:10.1109/TNET.2010.2095506.

[58] L. Bui, R. Srikant, A. Stolyar, Novel architectures and algorithms for delay reduction in back-pressure scheduling and routing, in: IEEE INFOCOM 2009, 2009, pp. 2936–2940. doi:10.1109/INFCOM.2009.5062262.

[59] B. Ji, C. Joo, N. B. Shroff, Delay-based back-pressure scheduling in multihop wireless networks, IEEE/ACM Transactions on Networking 21 (5) (2013) 1539–1552. doi:10.1109/TNET.2012.2227790.

[60] D. T. Le, T. Le Duc, V. V. Zalyubovskiy, D. S. Kim, H. Choo, Labs: Latency aware broadcast scheduling in uncoordinated duty-cycled wireless sensor networks, Journal of Parallel and Distributed Computing 74 (11) (2014) 3141–3152. doi:https://doi.org/10.1016/j.jpdc.2014.07.011.
URL https://www.sciencedirect.com/science/article/pii/S0743731514001336

[61] J. Hao, Z. Yao, K. Huang, B. Zhang, C. Li, An energy-efficient routing protocol with controllable expected delay in duty-cycled wireless sensor networks, in: 2013 IEEE International Conference on Communications (ICC), 2013, pp. 6215–6219. doi:10.1109/ICC.2013.6655601.

[62] Q. Chen, H. Gao, S. Cheng, X. Fang, Z. Cai, J. Li, Centralized and distributed delay-bounded scheduling algorithms for multicast in duty-cycled wireless sensor networks, IEEE/ACM Transactions on Networking 25 (6) (2017) 3573–3586. doi:10.1109/TNET.2017.2754405.

[63] Q. Wang, P. Fan, D. O. Wu, K. Ben Letaief, End-to-end delay constrained routing and scheduling for wireless sensor networks, in: 2011 IEEE International Conference on Communications (ICC), 2011, pp. 1–5. doi:10.1109/icc.2011.5962517.

[64] Y. Li, A. Capone, D. Yuan, On End-to-end Delay Minimization in Wireless Network under Physical Interference Model, in: IEEE INFOCOM, 2015, pp. 2020–2028, published online, DOI: 10.1109/INFOCOM.2015.7218586.

[65] A. Capone, Y. Li, M. Pióro, D. Yuan, Minimizing end-to-end delay in multi-hop wireless networks with optimized transmission scheduling, Ad Hoc Networks 89 (2019) 236 – 248.

[66] V. Angelakis, A. Ephremides, Q. He, D. Yuan, Minimum-Time Link Scheduling for Emptying Wireless Systems: Solution Characterization and Algorithmic Framework, IEEE Transactions on Information Theory 60 (2) (2014) 1083–1100.

[67] M. Pióro, Network optimization techniques, in: E. Serpedin, T. Chen, D. Rajan (Eds.), Mathematical Foundations for Signal Processing, Communications, and Networking, CRC Press, Boca Raton, USA, 2012, Ch. 18, pp. 627–684.

[68] O. Goussevskaia, Y. A. Oswald, R. Wattenhofer, Complexity in Geometric SINR, in: ACM Mobihoc, 2007, pp. 100–109.

[69] B. Staehle, D. Staehle, R. Pries, Effects of link rate assignment on the max-min fair throughput of wireless mesh networks, in: 2009 21st International Teletraffic Congress, 2009, pp. 1–8.

[70] B. Ostrowski, M. Pióro, A. Tomaszewski, Multicast traffic throughput maximization through dynamic modulation and coding scheme assignment in wireless sensor networks, in: 2020 International Conferences on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), 2020, pp. 135–141. doi:10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics50389.2020.00039.

[71] M. Pióro, M. Żotkiewicz, B. Staehle, D. Staehle, D. Yuan, On Max-min Fair Flow Optimization in Wireless Mesh Networks, Ad Hoc Networks 13 (2014) 134–152.

[72] B. Ostrowski, M. Pióro, A. Tomaszewski, Multicast traffic throughput maximization through joint dynamic modulation and coding schemes assignment, and transmission power control in wireless sensor networks, Sensors 21 (10) (2021). doi:10.3390/s21103411.
URL https://www.mdpi.com/1424-8220/21/10/3411

[73] E. Fitzgerald, Wireless network scenario generator, https://bitbucket.org/EIT_networking/network_generator, accessed 2020-01-10.

[74] A. Goldsmith, Cambridge University Press, 2005. [link].
URL https://app.knovel.com/hotlink/toc/id:kpWC000002/wireless-communications/wireless-communications

[75] H.-Y. Zhou, Modeling of node energy consumption for wireless sensor networks, Wireless Sensor Network 03 (2011) 18–23. doi:10.4236/wsn.2011.31003.

[76] L. Casals, B. Mir, R. Vidal, C. Gomez, Modeling the energy performance of lorawan, Sensors 17 (10) (2017). doi:10.3390/s17102364.
URL https://www.mdpi.com/1424-8220/17/10/2364

[77] B. Ostrowski, M. Pióro, A. Tomaszewski, Packet delay minimization in multi-hop wireless sensor networks with periodic traffic, in: 2020 IFIP Networking Conference (Networking), 2020, pp. 1–9.

[78] W. Ogryczak, M. Pióro, A. Tomaszewski, Telecommunications Network Design and Max-Min Optimization Problem, Journal of Telecommunications and Information Technology 3 (2005) 43–56.

[79] A. Tomaszewski, M. Pióro, Packet Routing and Frame Length Optimization in Wireless Mesh Networks with Multicast Communications, in: Proc. of the 17th International Network Strategy and Planning Symposium (NETWORKS 2016), 2016, pp. 1–8.

[80] M. Pióro, D. Medhi, Routing, Flow, and Capacity Design in Communication and Computer Networks, Morgan Kaufmann, 2004.

[81] B. Ostrowski, M. Pióro, A. Tomaszewski, M. Nowak, I. Kalesnikau, Throughput vs. resilience in multi-hop wireless sensor networks with periodic packet traffic, in: 2020 16th International Conference on the Design of Reliable Communication Networks DRCN 2020, 2020, pp. 1–7. doi:10.1109/DRCN48652.2020.1570610581.

[82] K. Bao, J. D. Matyjas, F. Hu, S. Kumar, Intelligent software-defined mesh networks with link-failure adaptive traffic balancing, IEEE Transactions on Cognitive Communications and Networking 4 (2) (2018) 266–276. doi:10.1109/TCCN.2018.2790974.

[83] S. S. A. Gilani, A. Qayyum, R. N. B. Rais, M. Bano, Sdnmesh: An sdn based routing architecture for wireless mesh networks, IEEE Access 8 (2020) 136769–136781. doi:10.1109/ACCESS.2020.3011651.

[84] F. Olivier, G. Carlos, N. Florent, Sdn based architecture for clustered wsn, in: 2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, 2015, pp. 342–347. doi:10.1109/IMIS.2015.52.

[85] Y. Wang, H. Chen, X. Wu, L. Shu, An energy-efficient sdn based sleep scheduling algorithm for wsns, Journal of Network and Computer Applications 59 (2016) 39–45. doi:https://doi.org/10.1016/j.jnca.2015.05.002.
URL https://www.sciencedirect.com/science/article/pii/S1084804515000910

[86] J. Wildman, S. Weber, On protocol and physical interference models in poisson wireless networks (2016). arXiv:1609.05314.

[87] W.-H. Tam, Y.-C. Tseng, Joint multi-channel link layer and multi-path routing design for wireless mesh networks, in: IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications, 2007.

[88] K. Jaiswal, V. Anand, Eomr: An energy-efficient optimal multi-path routing protocol to improve qos in wireless sensor network for iot applications, Wireless Personal Communications 111 (4) (2020) 2493–2515. doi:10.1007/s11277-019-07000-x.
URL https://doi.org/10.1007/s11277-019-07000-x

[89] P. H. Pathak, R. Dutta, A Survey of Network Design Problems and Joint Design Approaches in Wireless Mesh Networks, IEEE Communications Surveys Tutorials 13 (3) (2011) 396–428.

[90] W. Jaafar, W. Ajib, S. Tabbane, The capacity of mimo-based wireless mesh networks, in: 2007 15th IEEE International Conference on Networks, 2007, pp. 259–264. doi:10.1109/ICON.2007.4444096.

[91] M. Siraj, M. Shoaib, S. Alshebeili, Performance enhancement of cognitive radio wireless mesh networks by link scheduling and beamforming, in: 2013 Saudi International Electronics, Communications and Photonics Conference, 2013, pp. 1–6. doi:10.1109/SIECPC.2013.6551025.

Index

List of Abbreviations

5G	5th Generation
BOM	Basic Optimization Model
BPSK	Binary Phase Shift Keying
B&B	Branch-and-Bound
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
c-set	Compatible Set
DMP	Dual Master Problem
DMRFMP	Frame Minimization Problem with Routing and Dynamic Modulation and Coding Schemes Assignment Optimization
DRFMP	Frame Minimization Problem with Routing and Destination Nodes Selection Optimization
DTFMP	Frame Minimization Problem with Routing Optimization by means of Routing Tree Generation and Destination Nodes Selection Optimization
ECMP	Energy Consumption Minimization Problem
RFMPE	Frame Minimization Problem with Routing Optimization and Constrained Energy
IoT	Internet of Things
IPTV	Internet Protocol Television
LP	Linear Programming
LR-WPAN	Low-rate Personal Area Networks
MAC	Medium Access Control
MCS	Modulation and Coding Scheme/Schemes

MHWN	Multi-hop Wireless Network/Networks
MIMO	Multiple Input Multiple Output
MIP	Mixed-Integer Programming
mMTC	Massive Machine Type Communication
PP	Pricing Problem
P&B	Price-and-Branch
PDMP	Packet Delay Minimization Problem
QAM	Quadrature Amplitude Modulation
RFMP	Frame Minimization Problem with Routing Optimization
SA	Simulated Annealing
SFMP	Simple Frame Minimization Problem
SDN	Software-defined Networking
SINR	Signal-to-Interference-to-Noise Ratio
SMRFMP	Frame Minimization Problem with Routing and Static Modulation and Coding Schemes Assignment Optimization
SNR	Signal-to-Noise Ratio
TDMA	Time Division Multiple Access
TFMP	Frame Minimization Problem with Routing Optimization by means of Routing Tree Generation
WMN	Wireless Mesh Network/Networks
WSN	Wireless Sensor Network/Networks
VoIP	Voice over Internet Protocol

List of Figures

List of Tables