



HAL
open science

Maximum Bounded Rooted-Tree Problem : Algorithms and Polyhedra

Jinhua Zhao

► **To cite this version:**

Jinhua Zhao. Maximum Bounded Rooted-Tree Problem : Algorithms and Polyhedra. Combinatorics [math.CO]. Université Clermont Auvergne [2017-2020], 2017. English. NNT : 2017CLFAC044 . tel-01730182

HAL Id: tel-01730182

<https://theses.hal.science/tel-01730182>

Submitted on 13 Mar 2018

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

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

Numéro d'Ordre : D.U. 2816

EDSPIC : 799

Université Clermont Auvergne

École Doctorale
Sciences Pour l'Ingénieur de Clermont-Ferrand

THÈSE

Présentée par

Jinhua ZHAO

pour obtenir le grade de

Docteur d'Université

Spécialité : Informatique

**Le Problème de l'Arbre Enraciné Borné Maximum :
Algorithmes et Polyèdres**

Soutenue publiquement le 19 juin 2017 devant le jury

M. ou Mme Mohamed DIDI-BIHA Rapporteur et examinateur
Sourour ELLOUMI Rapporteuse et examinatrice
Ali Ridha MAHJOUB Rapporteur et examinateur
Fatiha BENDALI Examinatrice
Hervé KERIVIN Directeur de Thèse
Philippe MAHEY Directeur de Thèse

Acknowledgments

First and foremost I would like to thank my PhD advisors, Professors Philippe Mahey and Hervé Kerivin. Without Professor Philippe Mahey, I would never have had the chance to come to ISIMA or LIMOS here in France in the first place, and I am really grateful and honored to be accepted as a PhD student of his. My PhD could not be even started and needless to say finished without his support. I would also like to express my greatest appreciation and thanks to Professor Hervé Kerivin, who have taught me countless things over the course of my master and PhD, including but not limit to knowledge of all aspects of combinatorial optimization, scientific writing, methodology of research, means of tackle a complex problem, etc. We have had so many inspirational meetings during the last 5 years, and as a result of which, many ideas were inspected, implemented and eventually turned into new discoveries as a part of this dissertation. His patience, passion and step-by-step guidance was instrumental in helping me walk into the field of combinatorial optimization and go further in depth in order to crank out this dissertation.

I would like to thank my committee members Professors Fatiha Bendali, Mohamed Didi-Biha, Sourour Elloumi, Ali Ridha Mahjoub for their time, interest, insightful questions and comments. I would also like to express my appreciation for the French Ministry of Higher Education and Research as the funding source of my PhD, and LIMOS as the lab that hosted me.

My time at France was made enjoyable largely due to the many friends and colleagues I have met. Among them Maxime Chassaing and Benjamin Vincent are the first friends I have made after my arrival. They soon became my peer PhD students, and we shared a lot of common experiences in our academic and personal life. I would like to thank them for having my back all the time. I also thank Libo Ren as a forerunner in the PhD and researcher life in LIMOS, providing me valuable advises about research and how to live here in Clermont-Ferrand with his firsthand experience.

I am have also appreciated the lovely atmosphere in the office I have worked for the past 6 years, which was created by all the colleagues I have worked with, Samuel Deleplanque, Jocelyn De Goer, Vanel Siyou, Danh Cong Nguyen, Suan Tay, and most importantly Professor Philippe Vaslin. I would like also express my thanks to all other colleagues I

have met and worked with, to name a few, Professor Christophe Duhamel, Raksmei Phan, Bin Tian, Xiaofeng Zhao, Jean Connier. My special thanks also go to Professors Hong Sun, Jianjin Li and Kun Mean Hou, who were in charge of the ISIMA-WHU exchange program I was in.

I will forever be grateful to my former internship advisor and 6-year co-worker Professor Philippe Vaslin. During my time in France, he helped me in every possible way, even more than I could ever hope for. As soon as I arrived in France, I was signed to an internship supervised by him. Since then, we have worked together in the same office for more than 6 years. During this time, he has helped me in all aspects of both scientific work and daily life, ranging from research suggestions, writing suggestions, dealing with all sorts of things in everyday life to teaching me French history, European geography and swimming.

I could not have done anything without the support, love and encouragement of my friends and family members. I thank all my friends for providing support and friendship and having faith in me all the time. I thank my parents and parents in law for their unconditional love and supporting me in my pursuits abroad even if it means they have to suffer from the separation between them and their children and grandson.

Last but definitely not the least, I would like to thank my loving and supportive wife Jie Guan and my beloved son Yicen Zhao for always being there for me. Thank you.

Jinhua Zhao

September 2017

*So, we'll go no more a-roving
So late into the night*

– George Gordon Byron

Abstract

Given a simple undirected graph $G = (V, E)$ with a so-called root node $r \in V$, a rooted tree, or an r -tree, of G is either the empty graph (\emptyset, \emptyset) , or a tree containing r . If a node-capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$ is given, then a subgraph of G is said to be bounded if the degree of each node in the subgraph does not exceed its capacity. Let \mathbf{w} be an edge-weight vector in \mathbb{R}^E and \mathbf{p} a node-price vector in \mathbb{R}^V . The Maximum Bounded r -Tree (MBrT) problem consists of finding a bounded r -tree $T = (U, F)$ of G such that $\sum_{e \in F} w_e + \sum_{v \in U} p_v$ is maximized. If the capacity constraint from the MBrT problem is relaxed, we then obtain the Maximum r -Tree (MrT) problem. This dissertation contributes to the study of the MBrT problem and the MrT problem.

First we introduce the problems with their definitions and complexities. We define the associated polytopes along with a formulation for each of them. We present several polynomial-time combinatorial algorithms for both the MBrT problem (and thus the MrT problem) on trees, cycles and cactus graphs. Particularly, a dynamic-programming-based algorithm is used to solve the MBrT problem on trees, whereas on cycles we reduce it to some polynomially solvable problems in three different cases. For cactus graphs, we first show that the MBrT problem can be solved in polynomial time on a so-called cactus basis, then break down the problem on any cactus graph into a series of subproblems on trees and on cactus basis.

The second part of this work investigates the polyhedral structure of three polytopes associated with the MBrT problem and the MrT problem, namely $B_{xy}(G, r, \mathbf{c})$, $B_x(G, r, \mathbf{c})$ and $R_x(G, r)$. $B_{xy}(G, r, \mathbf{c})$ and $B_x(G, r, \mathbf{c})$ are polytopes associated with the MBrT problem, where $B_{xy}(G, r, \mathbf{c})$ considers both edge- and node-indexed variables and $B_x(G, r, \mathbf{c})$ considers only edge-indexed variables. $R_x(G, r)$ is the polytope associated with the MrT problem that only considers edge-indexed variables. For each of the three polytopes, we study their dimensions, facets as well as possible ways of decomposition. We introduce some newly discovered constraints for each polytope, and show that these new constraints allow us to characterize them on several graph classes. Specifically, we provide characterization for $B_{xy}(G, r, \mathbf{c})$ on cactus graphs with the help of a decomposition through 1-sum. On the other hand, a TDI-system that characterizes $B_x(G, r, \mathbf{c})$ is given in each case of

trees and cycles. The characterization of $R_x(G, r)$ on trees and cycles then follows as an immediate result.

Finally, we discuss the separation problems for all the inequalities we have found so far, and present algorithms or cut-generation heuristics accordingly. A couple of branch-and-cut frameworks are implemented to solve the MBrT problem together with a greedy-based matheuristic. We compare the performances of the enhanced formulations with the original formulations through intensive computational test, where the results demonstrate convincingly the strength of the enhanced formulations.

Keywords: combinatorial optimization, bounded rooted-tree, algorithm, polyhedral study, branch-and-cut

Résumé

Étant donné un graphe simple non orienté $G = (V, E)$ et un sommet particulier $r \in V$ appelé *racine*, un *arbre enraciné*, ou *r-arbre*, de G est soit le graphe nul (\emptyset, \emptyset) soit un arbre contenant r . Si un vecteur de capacités sur les sommets est donné, un sous-graphe de G est dit borné si le degré de chaque sommet dans le sous-graphe est inférieur ou égal à sa capacité. Soit \mathbf{w} un vecteur de poids sur les arêtes dans \mathbb{R}^E et \mathbf{p} un vecteur de profits sur les sommets dans \mathbb{R}^V . Le problème du *r-arbre borné maximum* (MBrT, de l'anglais Maximum Bounded *r*-Tree) consiste à trouver un *r-arbre borné* $T = (U, F)$ de G tel que $\sum_{e \in F} w_e + \sum_{v \in U} p_v$ soit maximisé. Si la contrainte de capacité du problème MBrT est relâchée, nous obtenons le problème du *r-arbre maximum* (MrT, de l'anglais Maximum *r*-Tree). Cette thèse contribue à l'étude des problèmes MBrT et MrT.

Tout d'abord, ces deux problèmes sont formellement définis et leur complexité est étudiée. Nous présentons ensuite des polytopes associés ainsi qu'une formulation pour chacun d'entre eux. Par la suite, nous proposons plusieurs algorithmes combinatoires pour résoudre le problème MBrT (et donc le problème MrT) en temps polynomial sur les arbres, les cycles et les cactus. En particulier, un algorithme de programmation dynamique est utilisé pour résoudre le problème MBrT sur les arbres. Pour les cycles, nous sommes amenés à considérer trois cas différents pour lesquels le problème MBrT se réduit à certains problèmes polynomiaux. Pour les cactus, nous montrons tout d'abord que le problème MBrT peut être résolu en temps polynomial sur un type de graphes appelé *cactus basis*. En utilisant une série de décompositions en sous-problèmes sur les arbres et les cactus basis, nous obtenons un algorithme pour les graphes de type cactus.

La deuxième partie de ce travail étudie la structure polyédrale de trois polytopes associés aux problèmes MBrT et MrT. Les deux premiers polytopes, $B_{xy}(G, r, \mathbf{c})$ et $B_x(G, r, \mathbf{c})$ sont associés au problème MBrT. Tous deux considèrent des variables sur les arêtes de G , mais seuls $B_{xy}(G, r, \mathbf{c})$ possède également des variables sur les sommets de G . Le troisième polytope, $R_x(G, r)$, est associé au problème MrT et repose uniquement sur les variables sur les arêtes. Pour chacun de ces trois polytopes, nous étudions sa dimension, caractérisons certaines inégalités définissant des facettes, et présentons les moyens possibles de décomposition. Nous introduisons également de nouvelles familles de

contraintes. L'ajout de ces contraintes nous permettent de caractériser ces trois polytopes dans plusieurs classes de graphes.

Pour finir, nous étudions les problèmes de séparation pour toutes les inégalités que nous avons trouvées jusqu'ici. Des algorithmes polynomiaux de séparation sont présentés, et lorsqu'un problème de séparation est \mathcal{NP} -difficile, nous donnons des heuristiques de séparation. Tous les résultats théoriques développés dans ce travail sont implémentés dans plusieurs algorithmes de coupes et branchements auxquels une matheuristique est également jointe pour générer rapidement des solutions réalisables. Des expérimentations intensives ont été menées via le logiciel CPLEX afin de comparer les formulations renforcées et originales. Les résultats obtenus montrent de manière convaincante la force des formulations renforcées.

Mots-clefs: optimisation combinatoire, arbre enraciné borné, algorithme, étude polyédrique, algorithme de coupe

Contents

Introduction	1
1 Preliminaries and notation	5
1.1 Algorithms and computational complexity	5
1.2 Polyhedra and linear optimization	7
1.3 Integer optimization and polyhedral approach	12
1.4 Graph theory	16
2 Maximum bounded rooted-tree problem	19
2.1 Introduction	19
2.2 Associated polyhedra	22
2.3 State of the art	28
2.3.1 Degree-constrained problems	28
2.3.2 Non-degree-constrained related problems	30
2.3.3 Applications in the telecommunication field	32
2.4 Conclusion	34
3 Combinatorial algorithms for the MB^rT problem	35
3.1 On trees	36
3.2 On cycles	39
3.3 On cactus graphs	46
3.4 General graph decomposition	50
3.5 Conclusion	51

4	Polyhedral study on Extended Bounded r-Tree Polytope	53
4.1	Dimension	54
4.2	Decomposition through 1-sum	59
4.3	Facets	62
4.4	Characterization	80
4.4.1	On trees	81
4.4.2	On cycles	83
4.4.3	On cactus graphs	87
4.5	Conclusion	87
5	Polyhedral study on r-Tree Polytope	89
5.1	Dimension	90
5.2	Facets	90
5.3	New valid inequalities	95
5.3.1	Matching-partition inequalities	95
5.3.2	Acyclicity-connectivity inequalities	98
5.4	Decomposition	101
5.5	Conclusion	102
6	Polyhedral study on Bounded r-Tree Polytope	105
6.1	Dimension	106
6.2	Decomposition	106
6.2.1	Decomposition at the root node	107
6.2.2	Decomposition with respect to bridges	108
6.3	Facets	109
6.3.1	General results	109
6.3.2	Box inequalities	111
6.3.3	Capacity inequalities	114
6.3.4	Connectivity inequalities	118
6.3.5	Subtour elimination inequalities	122
6.4	New valid inequalities	124

6.4.1	Matching-partition inequalities	124
6.4.2	Acyclicity-connectivity inequalities	127
6.4.3	Upload capacity inequalities	128
6.4.4	Capacity-2 inequalities	133
6.4.5	Capacity-i inequalities	134
6.4.6	i-articulation inequalities	136
6.4.7	Tightening inequalities	137
6.5	Projection from the Extended Polytope	138
6.5.1	Matching-partition inequalities	138
6.5.2	Upload capacity inequalities	139
6.5.3	Acyclicity-connectivity inequalities	141
6.5.4	Capacity-i inequalities	142
6.5.5	Tightening inequalities	144
6.6	Bounds on Chvátal-Gomory rank	145
6.6.1	Matching-partition inequalities	145
6.6.2	Upload capacity inequalities	151
6.7	TDI-ness	155
6.7.1	On trees	155
6.7.2	On cycles	163
6.8	Conclusion	181
7	Computational study	183
7.1	Methodology	183
7.2	Separation problems for valid inequalities	186
7.2.1	Connectivity inequalities	187
7.2.2	Upload capacity inequalities	189
7.2.3	Subtour elimination inequalities	190
7.2.4	Matching-partition inequalities	192
7.2.5	i-articulation inequalities and tightening inequalities	197
7.2.6	Acyclicity-connectivity inequalities	200

7.3	Primal matheuristic	200
7.4	Properties related to optimal solutions	202
7.5	Results	203
7.5.1	SteinLib instances	204
7.5.2	Random generated instances	208
7.6	Conclusion	214
	Conclusion	215
	References	221

List of Figures

1.1	A polyhedron and its integral hull	11
1.2	Tree order in a rooted-tree	18
3.1	A cycle G with v_o and its subpaths P_o, P'_o	44
3.2	A cactus basis	47
5.1	An example of matching-partition inequality cutting off a fractional extreme point	96
5.2	An example of an acyclicity-connectivity inequality cutting off a fractional extreme point	98
5.3	Counter example of decomposition involving 2 blocks	101
5.4	Another counter example of decomposition involving 3 blocks	102
6.1	Decomposition of graph with a bridge	108
6.2	An example of inequality (6.6) exclusively defining a facet	119
6.3	Example of a connectivity inequality with a triangle substructure	121
6.4	An example with a subtour elimination inequality that is not facet-defining	123
6.5	General example with an articulation set	124
6.6	An example of upload capacity inequality cutting off a fractional extreme point	129
6.7	A matching-partition inequality with Chvátal-Gomory rank 2	148
6.8	An upload capacity inequality with Chvátal-Gomory rank 2	154
6.9	Examples of alternating edge partition	172
7.1	Percentage of solved instances on SteinLib testsets	204

7.2	Lower bounds on SteinLib instances	205
7.3	Upper bounds on SteinLib instances	205
7.4	Percentage of solved instances on random dense graphs	208
7.5	Bounds on random dense graphs	209
7.6	Percentage of solved instances on random sparse graphs	210
7.7	Bounds on random sparse graphs	211
7.8	Percentage of solved instances on random sparse graphs with $O = \emptyset$. . .	212
7.9	Bounds on random sparse graphs with $O = \emptyset$	213

List of Tables

7.1	Options for the instances	184
7.2	Options for the branch-and-cut frameworks	186
7.3	Initial models and cuts of four branch-and-cut frameworks	187
7.4	Test results on instances of testset B	206
7.5	Test results on instances of testset C	206
7.6	Test results on instances of testsets I320, I640 and MSM	207
7.7	Test results on instances on dense graphs	210
7.8	Test results on instances on sparse graphs	211
7.9	Test results on instances on sparse graphs with $O = \emptyset$	213
7.10	Test results with $B RTP_{xy}$, 10 - 199 nodes	217

List of Algorithms

1.1	General cutting-plane algorithm	15
3.1	Algorithm to compute $g(r)$ on trees	38
3.2	Algorithm to obtain $\zeta_{Min}(P_r)$ on cycles	43
3.3	Algorithm for the MBrT problem on cactus	49
6.1	Algorithm on trees to obtain Δ^1	161
6.2	Algorithm to obtain an alternating edge partition on cycles	172
6.3	Algorithm to obtain a matching-partition and its associated dual variable based on ρ^k and \mathbf{w}^k	173
6.4	Algorithm to update an alternating edge partition ρ^k into ρ^{k+1} according to an edge-weight vector \mathbf{w}^{k+1}	174
6.5	Dual algorithm on cycles with $c_r \geq 2$ and $O = \emptyset$	175
6.6	Dual algorithm on cycles with $c_r = 1$ and $O = \emptyset$	179
7.1	Algorithm to separate connectivity inequalities	188
7.2	Cut generation algorithm for matching-partition inequalities	197
7.3	Primal matheuristic	201

Introduction

Consider a simple undirected graph $G = (V, E)$ with a *root node* $r \in V$ and a node-capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$. An *r-tree* of G is either the empty graph (\emptyset, \emptyset) , or a tree containing the root node r . We call a subgraph of G *bounded* if the degree of each node in the subgraph does not exceed its capacity. Given an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$ and a node-price vector $\mathbf{p} \in \mathbb{R}^V$, the Maximum Bounded *r*-Tree (MBrT) problem consists of finding a bounded *r*-tree $T = (U, F)$ of G such that $\sum_{e \in F} w_e + \sum_{v \in U} p_v$ is maximized.

This problem has been addressed recently as a new combinatorial optimization problem by Chakareski et al. [2009]. Its application arises in the content delivery networks, specifically the delivery of video streams in under-provisioned peer-to-peer (P2P) networks, where the resources (most of all, the upload capacities) of the peers are generally recognized as the bottleneck of the networks (Chakareski et al. [2009], Massoulié et al. [2007], Liu et al. [2010]). Such a P2P network can be represented by a simple undirected graph, where the source of the video stream naturally corresponds to the root node, and for each peer its upload-capacity limit can be converted into a degree limit. In Kerivin et al. [2011], the MBrT problem was proved to be \mathcal{NP} -hard by reducing the 3-SAT problem (Garey and Johnson [1979]) to it, and polynomial-time algorithms were given on certain classes of graphs such as trees and complete graphs.

If the node-capacity constraints are relaxed from the MBrT problem, we obtain the Maximum *r*-Tree (MrT) problem. The MrT problem was first introduced in Goemans [1994], where a formulation with both node- and edge-indexed variables was proposed. A study on the facets of the corresponding polytope was given and eventually the author proved that the formulation is ideal on series-parallel graphs.

Nonetheless, to the best of our knowledge, polytopes associated with the MBrT problem have not been previously studied in the literature, and neither has the polytope associated with the MrT problem considering only edge-indexed variables.

This dissertation contributes to three different aspects of the MBrT problem and the MrT problem: algorithmic issues, polyhedral structure, and computational test.

First of all, we define the aforementioned two problems and their associated polytopes

explicitly, provide their complexity results and formulations. Polynomial-time algorithms are proposed to solve both problems on several classes of graphs. We then study the problems' polytopes, and strengthen them by adding new families of valid equations and facet-defining inequalities. Through this theoretical study, complete descriptions of these polytopes are obtained on certain classes of graphs. Finally, we embed the different formulations into a branch-and-cut framework and compare their performance in the computational simulation.

This dissertation is organized as follows.

Chapter 1 provides the theoretical basis and the notation that will be used throughout this dissertation.

Chapter 2 firstly gives the definition and complexity results of the $MBrT$ problem and the MrT problem. The associated polyhedra are defined with respect to two sets of variables. The first one considers both node- and edge-indexed variables, while only the edge-indexed variables are present in the second.

After that, a literature survey of both the $MBrT$ problem and the MrT problem is presented. Notably, the degree constraints is one of the factors that contribute to the difficulty of the $MBrT$ problem. Hence we address the connection between the $MBrT$ problem and a few other well-studied degree-constrained problems, such as the b -matching related problems and the minimum bounded degree spanning tree (MBDST).

On the other hand, we also review a few results concerning some non-degree-constrained problems that are closely related to the $MBrT$ problem and the MrT problem. Particularly, as an r -tree is can be seen as a Steiner tree with one terminal, Goemans [1994] related the MrT problem to the Steiner tree problem and presented a polyhedral study with respect to the MrT problem's polytope considering both node- and edge-indexed variables. Besides, the connected subgraph problem is similar to the $MBrT$ problem and the MrT problem in the sense that they all aim at finding a connected subgraph that non-necessarily contains a node cover.

We also introduce the background of the application of the $MBrT$ problem in the telecommunication industry, and demonstrate the need of significant effort towards its application, and the lack thereof at present.

Chapter 3 presents a few polynomial-time combinatorial algorithms that can be used to solve the $MBrT$ problem on certain classes of graphs, particularly, on trees, cycles and cactus graphs. Since the MrT problem is a relaxation of the $MBrT$ problem, the proposed algorithms are also feasible for the MrT problem.

On trees, an algorithm based on dynamic programming is used to solve the $MBrT$ problem, where we break it down to a collection of subproblems corresponding to each node

in the graph. More precisely, for each node $v \in V$ we define a subproblem that aims at finding a maximum bounded r -tree in its up-closure while considering v as the root node. We can then solve the MBrT problem by solving the subproblems from the leaves to the root node r , where at each node a greedy algorithm is used since the subproblem reduces to an optimization problem over a uniform matroid.

On cycles, we consider the MBrT problem in three different cases according to the capacity setting of the graph. For each case, we show that the MBrT problem can be reduced to a certain number of polynomially solvable problems that aim at finding a maximum (or minimum) subpath of a given path.

For cactus graphs, as they can be considered as a combination of trees and cycles with a tree hierarchy, we use a dynamic programming scheme similar to the one on trees. We first show that the MBrT problem can be solved in polynomial time on a so-called *cactus basis*, which is a graph composed of a cycle and some *pending edges*. Then the MBrT problem on a cactus graph is reduced to a series of subproblems on trees and on cactus basis for each cycle-component of the graph.

We also show that if the MBrT problem is polynomially solvable on two graphs, it is also polynomially solvable on the 1-sum of them.

Chapter 4, 5 and 6 provide results concerning the polytopes associated with the MBrT problem and the MrT problem.

Chapter 4 focuses on the MBrT problem's polytope with considering both node- and edge-indexed variables. We introduce several sets of valid equations that relate to the non-root nodes that have capacity 1, and the blocks in the subgraph obtained by removing those nodes from the graph. Based on these results the dimension of the polytope is given. We then present a decomposition through 1-sum for the polyhedral study. Necessary and sufficient conditions to be facet-defining are examined for each known set of valid inequalities. We show that with all the introduced constraints, one is able to characterize the polytope on trees, cycles, and also, as a result of the decomposition through 1-sum, on cactus graphs.

Chapter 5 studies the MrT problem's polytope with considering only edge-indexed variables. We first prove that it is full-dimensional, which is followed by a facial study of the valid inequalities. Moreover, we propose two sets of newly discovered inequalities, namely the so-called *matching-partition inequalities* and the *acyclicity-connectivity inequalities*, where the former is an adaption of a similar constraint considered by Didi-Biha et al. [2015] for the connected subgraph problem. We then give a few counter examples which show that the polytope cannot be decomposed through 1-sum, contrary to the case for the polytope considering both node- and edge-indexed variables. Same conclusion regarding the decomposition can be drawn as well for the MBrT problem's polytope with

considering only edge-indexed variables, for which the same counter examples also exist. Chapter 6 presents results on the MBrT problem's polytope with considering only edge-indexed variables. We show that it is also full-dimensional. Two different approaches to decompose the polytope are proposed. The first one is a decomposition through 1-sum at the root node, while the second one decomposes a graph containing a bridge into two subgraphs that both contain that bridge. Similar to the previous chapters, we also give necessary and sufficient conditions for each set of valid inequalities to be facet-defining. We then introduce several sets of newly discovered inequalities, including those two sets of inequalities mentioned for the MrT problem. We show that they can all be obtained by projection from the constraints for the polytope considering both node- and edge-indexed variables. Bounds on their Chvátal-Gomory rank are also discussed. With the help of the newly discovered inequalities, we prove that one can obtain a linear system that is Totally Dual Integral (TDI), and completely characterize the polytope on trees and cycles.

In Chapter 7, we discuss the separation problems for all the inequalities considered in the previous chapters. A couple of branch-and-cut frameworks are presented to solve the MBrT problem together with a greedy-based matheuristic that generates feasible solutions to the problem. These branch-and-cut algorithms were implemented with CPLEX and intensive computational experiments are presented and analyzed.

Chapter 1

Preliminaries and notation

This chapter introduces some preliminary definitions, notation and some background theories that are used in this dissertation.

1.1 Algorithms and computational complexity

The complexity theory was born following the work of Edmonds [1962] and Cook [1971]. It offers a framework to classify problems according to their difficulty. More information about complexity theory can be found in Karp [1972], Garey and Johnson [1979], von Leeuwen [1990], and Papadimitriou [1994].

A *problem* is a general question to be answered and it usually possesses several formal parameters with no specific values. Such problem is characterized by a general description of all its parameters and a statement of properties that the answer, or the *solution*, should satisfy. An *instance* of a problem is when all the problem's parameters have specified values. A *decision problem* is a problem whose solution is either “yes” or “no”. For instance, “Given a graph $G = (V, E)$, does there exist a Hamiltonian path in G ?” is a decision problem. An *optimization problem*, on the other hand, generally aims at maximizing or minimizing a certain objective function. An optimization problem is not a decision problem, nonetheless it can often be transformed into a decision problem. For instance, “Given a graph $G = (V, E)$ with a set of terminals $S \subseteq V$, find a Steiner tree in G with minimum number of edges.” is an optimization problem. Its decision problem can be described as “Given a graph $G = (V, E)$ with a set of terminals $S \subseteq V$ and a value $B \in \mathbb{Z}_+$, does there exist a Steiner tree in G with at most B edges?”.

An *algorithm* is a general step-by-step procedure for solving problems. The efficiency of an algorithm, evaluated as the amount of computing resources it requires, decides whether an algorithm is good or not. However, the *time complexity*, or *computational*

complexity, is the most common way to determine the efficiency of an algorithm. The time complexity of an algorithm is expressed as a function of the *size* of the problem instance, where the instance size is usually represented by the length of the input data required to describe the instance. If an algorithm has time complexity bounded from above by a polynomial function in the size of the problem instance, then the algorithm is said to be a *polynomial-time algorithm*.

The computational complexity of a problem can be related to the algorithms that solve it. Moreover, any problem can be classified into some *complexity classes* according to its complexity. The most well-known complexity classes \mathcal{P} (*Polynomial*) and \mathcal{NP} (*Non-deterministic Polynomial*) are collections of decision problems. We hereafter provide more formal definitions of classes \mathcal{P} , \mathcal{NP} , \mathcal{NP} -complete and \mathcal{NP} -hard.

A problem is said to be *polynomially solvable* if there exists a polynomial-time algorithm to solve it. The collection of all polynomially solvable decision problems is denoted by \mathcal{P} . For example, the decision problem of the minimum spanning tree problem is in \mathcal{P} (Kruskal [1956], Prim [1957]).

The class \mathcal{NP} is defined as the collection of decision problems with the property that, for any instance for which the answer is “yes”, there is a polynomial-time checkable certificate of the “yes”. It can be seen that the class \mathcal{P} belongs to \mathcal{NP} .

In order to compare the computational complexity associated with two problems, the definition of *polynomial-time reduction* needs to be introduced. Let Q and R be two problems in \mathcal{NP} . If the input and output of any instance of Q can be *transformed* by a polynomial-time algorithm into input and output of an instance of R , then Q is *polynomially reducible* to R . It ensures that any polynomial-time algorithm for R can be converted into a corresponding polynomial-time algorithm for Q . Intuitively, this means that Q is not more difficult than R , or conversely that R is at least as hard as Q .

Now with the concept of polynomial-time reduction, the class of \mathcal{NP} -complete problems can be defined as follows. A problem R in \mathcal{NP} is *\mathcal{NP} -complete* if any other problem in \mathcal{NP} is polynomially reducible to R . Cook [1971] was the first one to prove that a problem, namely the boolean satisfiability problem (SAT), is \mathcal{NP} -complete. A list of \mathcal{NP} -complete problems can be found in Garey and Johnson [1979] (see also Ausiello et al. [1999]).

Finally, the class of *\mathcal{NP} -hard* problems is composed of problems for which any algorithm solving them can be converted into one solving any problem in \mathcal{NP} . In other words, \mathcal{NP} -hard problems are at least as hard as the most difficult problems in \mathcal{NP} . It is worth noting that an optimization problem is \mathcal{NP} -hard if its decision problem is \mathcal{NP} -complete.

In this dissertation, vectors are column vectors, and they are represented by boldface

letters. The transpose of a vector is represented by adding the superscript of T , e.g., \mathbf{a}^T is the transpose of vector \mathbf{a} . Matrices are represented by regular-face letters, and the transpose of a matrix A is denoted as A^T .

In Whitney [1935], the author introduced the concept and the properties of *matroid*. A pair (S, \mathcal{I}) is called a matroid if S is a finite set and \mathcal{I} is a nonempty collection of subsets of S that satisfies:

1. if $A \in \mathcal{I}$ and $B \subseteq A$, then $B \in \mathcal{I}$,
2. if $A, B \in \mathcal{I}$ and $|A| < |B|$, then $A \cup \{e\} \in \mathcal{I}$ for some $e \in B \setminus A$.

The sets in \mathcal{I} are called the *independent sets* and are said to be *independent*.

Given a weight vector $\mathbf{w} : S \rightarrow \mathbb{R}$, the *greedy algorithm* to find a set I in \mathcal{I} maximizing $\sum_{e \in I} w_e$ is as follows. Set $I = \emptyset$, and next repeatedly choosing $e' \in S \setminus I$ with $I \cup \{e'\} \in \mathcal{I}$ and with $w_{e'}$ as large as possible. The algorithm stops if no such e' exists. It has been proved that the greedy algorithm works and only works on matroids (see Rado [1957], Gale [1968], Edmonds [1971] and also Schrijver [2003]).

A *uniform matroid*, as a trivial class of matroid, is determined by a set S and a number k , where a subset I of S is independent if $|I| \leq k$.

1.2 Polyhedra and linear optimization

This section gives some fundamental results on linear and integer optimization. Further references are Schrijver [1986b], Nemhauser and Wolsey [1988], Wolsey [1998] and Schrijver [2003].

Given $n \in \mathbb{N}$, the symbol \mathbb{R}^n (\mathbb{Z}^n , respectively) represents the set of *vectors*, or *points*, having n real components (integer components, respectively). If all the components of a vector are zeros, it is denoted by $\mathbf{0}$.

Given $\lambda \in \mathbb{R}$, $\lfloor \lambda \rfloor$ ($\lceil \lambda \rceil$, respectively) represents the largest integer lower than or equal to (the smallest integer greater than or equal to, respectively) λ .

Given a subset A of U , the *complement* of A is denoted $U \setminus A$, or \bar{A} if U is clear from the context.

Additionally, given a set A , we also use \mathbb{R}^A (\mathbb{Z}^A , respectively) to represent the set of vectors having $|A|$ real components (integral components, respectively), each of which being indexed by one element in A .

Given any vector \mathbf{a} in \mathbb{R}^A and a subset $S \subseteq A$, let

$$a(S) = \sum_{e \in S} a_e.$$

A vector $\mathbf{x} \in \mathbb{R}^n$ is a *linear combination* of vectors $\mathbf{x}^1, \dots, \mathbf{x}^k$ in \mathbb{R}^n if there exist k real scalars $\lambda_1, \dots, \lambda_k$ such that

$$\mathbf{x} = \sum_{i=1}^k \lambda_i \mathbf{x}^i.$$

In addition, if $\lambda_1, \dots, \lambda_k$ satisfy

$$\sum_{i=1}^k \lambda_i = 1,$$

then \mathbf{x} is said to be an *affine combination* of $\lambda_1, \dots, \lambda_k$. Moreover, if \mathbf{x} is an affine combination of $\mathbf{x}^1, \dots, \mathbf{x}^k$ with non-negative λ -coefficients, that is, $\lambda_i \geq 0$ for all $i \in \{1, \dots, k\}$, then \mathbf{x} is a *convex combination* of vectors $\mathbf{x}^1, \dots, \mathbf{x}^k$.

A set of vectors is said to be *linearly independent* if none of the vectors can be written as a linear combination of the other vectors in the set. Similarly, a set of vectors is *affinely independent* if none of the vectors can be written as an affine combination of the other vectors in the set.

The *convex hull* of a set X of \mathbb{R}^n , denoted $\text{conv}(X)$, is the set of all vectors in \mathbb{R}^n which can be written as a convex combination of vectors in X .

A subset C of \mathbb{R}^n is called a *cone* if $\lambda \mathbf{x} \in C$ for any $\mathbf{x} \in C$ and $\lambda \in \mathbb{R}_+$.

A subset P of \mathbb{R}^n is called a *polyhedron* if there exist a matrix $A \in \mathbb{R}^{m \times n}$ and a vector $\mathbf{b} \in \mathbb{R}^m$ such that $P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$. Any inequality $\boldsymbol{\alpha}^T \mathbf{x} \leq \beta$ with $\boldsymbol{\alpha} \in \mathbb{R}^n, \beta \in \mathbb{R}$ is called *valid* for P if $\boldsymbol{\alpha}^T \mathbf{x} \leq \beta$ holds for any $\mathbf{x} \in P$. A polyhedron P is *bounded* if there exist $\mathbf{l}, \mathbf{u} \in \mathbb{R}^n$ such that $\mathbf{l} \leq \mathbf{x} \leq \mathbf{u}$ for any $\mathbf{x} \in P$. A subset P of \mathbb{R}^n is called a *polytope* if it is the convex hull of finitely many vectors in \mathbb{R}^n . One has that a set P is a polytope if and only if P is a bounded polyhedron (Minkowski [1896], Steinitz [1916], Weyl [1934]).

A polyhedron P of \mathbb{R}^n has *dimension* d if the maximum number of affinely independent vectors in P is $d + 1$. The dimension of P is denoted by $\dim P$. If $\dim P = n$, then the polyhedron P is said to be *full-dimensional*.

Given a polyhedron $P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$, a subset F of P is called a *face* if $F = \{\mathbf{x} \in P : A'\mathbf{x} = \mathbf{b}'\}$, where $A'\mathbf{x} \leq \mathbf{b}'$ is a sub-system of $A\mathbf{x} \leq \mathbf{b}$. The face F is *proper* if $F \neq P$. If $\boldsymbol{\alpha}^T \mathbf{x} \leq \beta$ is valid for P , then the inequality $\boldsymbol{\alpha}^T \mathbf{x} \leq \beta$ is said to *define* or *induce* the face $F = \{\mathbf{x} \in P : \boldsymbol{\alpha}^T \mathbf{x} = \beta\}$. Any face F of P satisfies $\dim F \leq \dim P$. The *recession cone* associated with the polyhedron P is $P^0 = \{\mathbf{r} \in \mathbb{R}^n : A\mathbf{r} \leq \mathbf{0}\}$.

Nonzero members of the recession cone are called *rays* of P . A ray \mathbf{r} of P is an *extreme ray* if there are $n - 1$ linearly independent constraints binding at \mathbf{r} .

A face of P is called a *facet* if it is nonempty, proper, and inclusion-wise maximal. In other words, a face F of P is a facet if it has dimension one less than the dimension of P , that is, $\dim F = \dim P - 1$. An inequality inducing a facet of P is called *facet-defining*.

A vector $\mathbf{x} \in \mathbb{R}^n$ is an *extreme point* of a pointed polyhedron P if it cannot be written as a convex combination of other vectors in P .

We call any vector $\mathbf{x} \in \mathbb{R}^n$ that satisfies $A\mathbf{x} \leq \mathbf{b}$ a *feasible solution* for the linear system $A\mathbf{x} \leq \mathbf{b}$. The feasibility of a system of linear inequalities, which indicates whether the system admits feasible solutions or not, is characterized by *Farkas' Lemma* (Farkas [1894]).

Theorem 1.2.1 (Farkas' Lemma). *The polyhedron $P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$ with $A \in \mathbb{R}^{m \times n}$ and $\mathbf{b} \in \mathbb{R}^m$ is not empty if and only if $\mathbf{y}^T \mathbf{b} \geq 0$ holds for any $\mathbf{y} \geq \mathbf{0}$ with $\mathbf{y}^T A = \mathbf{0}^T$. \square*

A *linear optimization problem* consists of maximizing (or minimizing) a linear function over a polyhedron. A linear optimization problem over the polyhedron P has the form

$$\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in P\}.$$

The linear function $\mathbf{c}^T \mathbf{x}$ is called the *objective function*.

If it is known that $P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$, the optimization problem also has an equivalent form as the following.

$$\max\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}. \quad (1.1)$$

If we take (1.1) as the *primal* problem, its corresponding *dual* problem is

$$\min\{\mathbf{y}^T \mathbf{b} : \mathbf{y}^T A \geq \mathbf{c}^T, \mathbf{y} \geq \mathbf{0}\}. \quad (1.2)$$

Duality is a fundamental concept in linear optimization that characterizes the relation between the primal problem and the dual problem. Further details on duality theory can be found in von Neumann [1947], Gale et al. [1951], and Dantzig [1963].

The primal-dual relation is described in the following duality theorems.

Theorem 1.2.2 (Weak Duality Theorem). *If $\mathbf{x} \in \mathbb{R}^n$ is feasible for (1.1) and $\mathbf{y} \in \mathbb{R}^m$ is feasible for (1.2), then*

$$\mathbf{c}^T \mathbf{x} \leq \mathbf{y}^T \mathbf{b}. \quad \square$$

It can be seen that for a maximization problem, every feasible solution $\bar{\mathbf{x}}$ to (1.1) provides a *primal bound* or *lower bound* of the optimal solution, that is, $\mathbf{c}^T \bar{\mathbf{x}} \leq z^*$, where z^* is the optimal value of (1.1). Conversely, every feasible solution $\bar{\mathbf{y}}$ to (1.2) provides a *dual bound* or *upper bound* of the optimal solution, that is, $\bar{\mathbf{y}}^T \mathbf{b} \geq z^*$.

Theorem 1.2.3 (Strong Duality Theorem).

$$\max\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\} = \min\{\mathbf{y}^T \mathbf{b} : \mathbf{y}^T A \geq \mathbf{c}^T, \mathbf{y} \geq \mathbf{0}\},$$

if at least one of these optima is finite. □

The strong duality theorem states that if one of the two problems has a finite optimal solution, then both of them have finite optimal solutions and equal optimal value.

Furthermore, the following *complementary slackness* theorem characterizes necessary and sufficient conditions for a pair of primal-dual feasible solutions to be optimal.

Theorem 1.2.4 (Complementary Slackness Theorem). *The vector $\mathbf{x} \in \mathbb{R}^n$ is an optimal solution to (1.1) and the vector $\mathbf{y} \in \mathbb{R}^m$ is an optimal solution to (1.2) if and only if \mathbf{x} and \mathbf{y} are feasible for (1.1) and (1.2) respectively, and*

$$(\mathbf{y}^T A - \mathbf{c}^T) \mathbf{x} = 0, \tag{1.3}$$

$$\mathbf{y}^T (A\mathbf{x} - \mathbf{b}) = 0. \quad \square \tag{1.4}$$

Conditions (1.3) and (1.4) are called the *complementary slackness conditions*. These conditions provide an effective way to develop a test of optimality for a putative solution to either (1.1) or (1.2).

Moreover, a dual bound can also be obtained by solving a *relaxation* of the primal problem. A relaxation may consist of maximizing the same objective function over a larger polyhedron, e.g., a polyhedron defined by a subsystem of the original linear system. The inequalities that are left out are said to be *relaxed*. For instance, let $P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$ and $P' = \{\mathbf{x} \in \mathbb{R}^n : A'\mathbf{x} \leq \mathbf{b}'\}$ be two polyhedra such that $A'\mathbf{x} \leq \mathbf{b}'$ is a subsystem of $A\mathbf{x} \leq \mathbf{b}$, then one has $P \subseteq P'$, and $\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in P'\}$ is a relaxation of the problem $\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in P\}$. As a result, $\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in P\} \leq \max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in P'\}$.

A vector $\mathbf{x} \in \mathbb{R}^n$ is called *integer* or *integral* if each of its components is an integer, that is, if $\mathbf{x} \in \mathbb{Z}^n$. The *integral hull* of a polyhedron $P \subseteq \mathbb{R}^n$ is the convex hull of integer vectors in P , that is, $\text{conv}(P \cap \mathbb{Z}^n)$. Figure 1.1 illustrates a polyhedron P and its integral hull P_I . A polyhedron is said to be *integral* if it is its own integral hull, or alternatively, each of its faces contains an integral vector.

In Edmonds and Giles [1977], a strong notion called *Totally Dual Integrality (TDI-ness)*

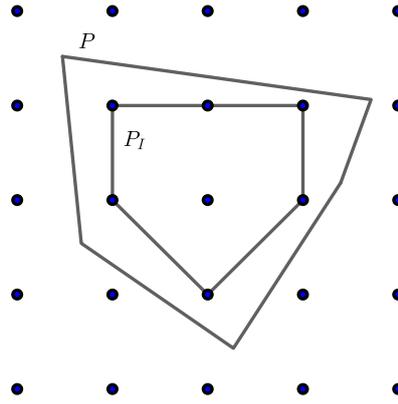


Figure 1.1: A polyhedron and its integral hull

is introduced.

Definition 1.2.5. A linear system of inequalities $A\mathbf{x} \leq \mathbf{b}$ is called Totally Dual Integral (TDI) if for each integral vector $\mathbf{c} \in \mathbb{Z}^n$, the dual problem of $\max\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}$ has an integer optimal solution if it is finite.

Edmonds and Giles [1977] showed that TDI-ness is a sufficient condition for integrality, as stated in the following theorem.

Theorem 1.2.6. Let $A\mathbf{x} \leq \mathbf{b}$ be a TDI system where A is rational and \mathbf{b} is integral. Then the polyhedron $\{\mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}$ is integral. \square

Further information on integrality of polyhedra and properties of TDI systems can be found in Hoffman [1974], Edmonds and Giles [1977], Schrijver [1986a,b, 2003], Korte and Vygen [2012].

Given a polyhedron P of \mathbb{R}^{n+m} in the variables $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$, the *projection* of P onto the x -space \mathbb{R}^n is defined as

$$\text{proj}_x(P) := \{\mathbf{x} \in \mathbb{R}^n : \exists \mathbf{y} \in \mathbb{R}^m \text{ with } (\mathbf{x}, \mathbf{y}) \in P\}.$$

If $P = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{n+m} : A\mathbf{x} + B\mathbf{y} \leq \mathbf{b}\}$, let the *projection cone* associated with $\text{proj}_x(P)$ be

$$W = \{\mathbf{v} : \mathbf{v}^T A = \mathbf{0}, \mathbf{v} \geq \mathbf{0}\}.$$

One has

$$\text{proj}_x(P) := \{\mathbf{x} \in \mathbb{R}^n : (\mathbf{v}^T B)\mathbf{x} \leq \mathbf{v}^T \mathbf{b}, \mathbf{v} \text{ is an extreme ray of } W\}.$$

Alternatively, $\text{proj}_x(P)$ can also be obtained either by Fourier-Motzkin elimination.

The following relation holds concerning the integrality of P and any of its projection $\text{proj}_x(P)$, see Balas [2005].

Theorem 1.2.7. *If P is an integral polyhedron of \mathbb{R}^{n+m} , then $\text{proj}_x(P)$ is an integral polyhedron of \mathbb{R}^m .*

1.3 Integer optimization and polyhedral approach

Similar to (1.1) for linear optimization, if we add the restriction that all variables must take integer values, we then have an *integer optimization* problem with the following form

$$\max\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}, \mathbf{x} \in \mathbb{Z}^n\}.$$

The corresponding linear optimization problem $\max\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}$ is called the *linear relaxation* of the integer optimization problem. A *formulation* of this integer optimization problem is a polyhedron of \mathbb{R}^n whose integral hull is $\text{conv}\{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$.

No polynomial-time algorithm is known to solve an integer optimization problem. In fact, the general integer optimization problem has been proved to be \mathcal{NP} -hard by Karp [1972] through a polynomial reduction of SAT to a variation of a 0-1 integer optimization problem.

The *branch-and-bound algorithm* provides an approach that solves a large-scale problem (or a hard problem) by considering smaller-scale ones (or easier ones), which is often used to tackle integer optimization problems.

Recall that the branch-and-bound algorithm breaks down an optimization problem over a set of feasible solutions into optimization problems over its subsets. This operation is referred as branching. Additionally, the branching process can be shortened by pruning the infeasible branches based on primal and dual bounds.

For instance, consider the integer optimization problem $\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in S\}$, $S \subseteq \{0, 1\}^n$. An *enumeration tree* can be constructed, where S is decomposed into $k \geq 2$ sets S_1, \dots, S_k which can also be decomposed into several smaller sets. The problem can be solved by carrying out the complete enumeration. However, for most problems with a large number of variables, complete enumeration is impossible. Furthermore, improvements on the primal and dual bounds, often made by other means such as heuristics, may allow one to prune some branches of the enumeration tree.

For \mathcal{NP} -hard integer optimization problems, where finding optimal solutions efficiently is generally not possible, various approaches can be used for them aside from branch-and-bound. *Approximation algorithms*, as one of them, are often used in such cases aiming at

finding near-optimal or *sub-optimal* solutions in polynomial time. Moreover, the use of approximation algorithms is also a rising trend for problems where exact polynomial-time algorithms are known but are too costly due to the input size.

An approximation algorithm usually exploits the algorithmically relevant combinatorial structure of a problem by using certain algorithmic techniques just as exact algorithms normally do. Nonetheless, it obtains a feasible solution within a fixed multiplicative factor of an optimal solution, instead of the optimal solution itself. An approximation algorithm is called a $f(n)$ -*approximation algorithm*, or a *factor $f(n)$ approximation algorithm*, for input size n if it can be proved that the solution the algorithm finds is at most a multiplicative factor of $f(n)$ times worse than the optimal solution. For instance, if the optimal value of a given maximization problem is denoted z^* , an $f(n)$ -approximation algorithm is an algorithm that always finds a feasible solution with value not less than $z^*/f(n)$.

Another approach that also aims at obtaining good feasible solutions is *heuristic*. Heuristics are often used when classic methods are too slow or fail to find any optimal solution. A heuristic generally finds a feasible solution, or primal bounds in other words, in polynomial time based on some experimental data or empirical knowledge of the problem. Oftentimes heuristics are much more quickly than solving the problem straightforwardly, and thus with the the primal bounds obtained from heuristics, it may shorten the overall time consumed on solving the problem. Unlike an approximation algorithm, a heuristic produces solutions with no guarantee, and its performance may vary largely for differently instances.

Given a *ground set* E , and \mathcal{S} as a set of vectors representing subsets of E , a *combinatorial optimization* problem \mathbf{P} which optimizes an objective function over \mathcal{S} can be defined as follows.

$$\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in \mathcal{S}\},$$

where $\mathbf{c} : E \rightarrow \mathbb{R}$ is a vector carrying the weights associated with components in E .

As one of the foremost technique to solve hard combinatorial optimization problems, the *polyhedral approach* has been widely used to solve combinatorial optimization problems after it was first introduced by Edmonds [1965].

Consider the convex hull $\text{conv}(\mathcal{S})$ of \mathcal{S} . The problem \mathbf{P} is equivalent to the linear optimization problem

$$\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in \text{conv}(\mathcal{S})\}. \tag{1.5}$$

If one succeeds to describe the polytope $\text{conv}(\mathcal{S})$ by a system of linear inequalities $A\mathbf{x} \leq \mathbf{b}$,

then problem \mathbf{P} is reduced to solving the following linear program:

$$\max\{\mathbf{c}^T \mathbf{x} : A\mathbf{x} \leq \mathbf{b}\}.$$

The polyhedral approach studies the polyhedral structure of $\text{conv}(\mathcal{S})$, and solves \mathbf{P} by solving a linear optimization problem. Recall that solving a linear optimization problem has been proved to be polynomial (Khachiyan [1979, 1980], Karmarkar [1984]).

However, in general it is difficult to obtain the complete linear description of the polytope $\text{conv}(\mathcal{S})$. Additionally, if the problem is \mathcal{NP} -hard, it is not likely to find such description. Moreover, even if the complete description of $\text{conv}(\mathcal{S})$ is known, it may contain an exponential number of inequalities. It makes it impossible to solve the problem using the algorithms for general linear optimization problems. The *cutting-plane algorithm* has been introduced in order to solve problems in such scenario. It allows us to obtain an optimal solution to a problem by solving a sequence of linear programs, each of which contains a polynomial number of inequalities.

A key component of the cutting-plane algorithm is the so-called *separation problem* and the computational equivalence of optimization and separation (Grötschel et al. [1981], Karp and Papadimitriou [1982], Padberg and Rao [1982]). More details can be found in Schrijver [1986b], Grötschel et al. [1988], Padberg [1999] and Korte and Vygen [2012].

Definition 1.3.1. *Let P be a polyhedron in \mathbb{R}^n . Given a vector $\mathbf{x} \in \mathbb{R}^n$, the separation problem for P is to decide whether \mathbf{x} is in P and if not, give an inequality that is valid for P but violated by \mathbf{x} .*

Optimization and separation over a polyhedron has been proved to be computationally equivalent (Grötschel et al. [1981], Karp and Papadimitriou [1982], Padberg and Rao [1982]), as stated in the following theorem.

Theorem 1.3.2. *Given a polyhedron P in \mathbb{R}^n , the optimization problem over P is polynomially solvable if and only if the separation problem for P is polynomially solvable. \square*

The cutting-plane algorithm is presented hereafter. Let $P' \subseteq \mathbb{R}^n$ be a polyhedron defined by a family of known valid inequalities for $\text{conv}(\mathcal{S})$, such that the separation problem for P' can be solved in polynomial time. Often limited by the resource, only a part of the valid inequalities can be added initially in the cutting-plane algorithm. Denote P^0 the polyhedron defined by a subset of inequalities defining P' .

Algorithm 1.1 shows a general cutting-plane algorithm that solves optimization problems over P' starting with the polyhedron P^0 .

The valid inequalities generated by this algorithm are often called *cutting planes* or *cuts*. By generating cuts, the cutting-plane algorithm produces a series of tighter outer-

Algorithm 1.1: General cutting-plane algorithm

Input : Polyhedron P^0 and vector \mathbf{c} .
Output: $\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in P'\}$.
begin
1 *Initialization.* Set $t = 0$.
 while *no optimal solution is found* **do**
2 Solve the linear program: $\max\{\mathbf{c}^T \mathbf{x} : \mathbf{x} \in P^t\}$ to get \mathbf{x}^t .
3 Solve the separation problem for P' with respect to \mathbf{x}^t .
 if *a valid inequality $\mathbf{a}^T \mathbf{x} \leq b$ for P' is found with $\mathbf{a}^T \mathbf{x}^t > b$* **then**
4 Set $P^{t+1} = P^t \cap \{\mathbf{x} : \mathbf{a}^T \mathbf{x} \leq b\}$.
5 Set $t = t + 1$.
 else
6 Stop.

approximations of P' . However, if P' is not a complete description of $\text{conv}(\mathcal{S})$, the solution obtained by the cutting-plane algorithm is sometimes not optimal with respect to the problem (1.5).

In the context of integer optimization, the branch-and-bound algorithm is often combined with the cutting-plane algorithm. The combination of these two approaches is called a *branch-and-cut* algorithm. In a branch-and-cut algorithm, cutting planes are generated through out the enumeration tree in an attempt to obtain tighter dual bounds at each branch-and-bound node. Such branch-and-cut algorithm usually reduces the number of nodes in the enumeration tree.

Another effective approach that helps to generate valid inequalities for an integral polytope is the Chvátal-Gomory procedure. In fact, given a polyhedron, the Chvátal-Gomory procedure is a well-known method to derive the description of its integral hull, see Gomory [1958, 1960], Chvátal [1973].

More precisely, let $X \in \mathbb{Z}^n$ be the set of integer solutions for an integer optimization problem. Let $P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\}$ be a polyhedron that satisfies $X = P \cap \mathbb{Z}^n$. Denote the columns of $A \in \mathbb{R}^{m \times n}$ by $\{\mathbf{a}^1, \dots, \mathbf{a}^n\}$. Given a non-negative vector $\boldsymbol{\mu} \in \mathbb{R}_+^m$, the inequality

$$\sum_{i=1}^n \boldsymbol{\mu}^T \mathbf{a}^i x_i \leq \boldsymbol{\mu}^T \mathbf{b}$$

is valid for X . Furthermore, inequalities

$$\sum_{i=1}^n \lfloor \boldsymbol{\mu}^T \mathbf{a}^i \rfloor x_i \leq \boldsymbol{\mu}^T \mathbf{b}$$

and

$$\sum_{i=1}^n \lfloor \mu^T \mathbf{a}^i \rfloor x_i \leq \lfloor \mu^T \mathbf{b} \rfloor$$

are both valid for X , among which the last inequality is stronger than the other two. This simple procedure is proved to be sufficient to generate all valid inequalities for X .

Theorem 1.3.3. *Each valid inequality for X can be obtained by applying the Chvátal-Gomory procedure a finite number of times.*

The minimum number of times this procedure required to obtain a certain inequality is called the Chvátal-Gomory rank of that inequality. Accordingly, the Chvátal-Gomory rank of a given polyhedron P equals the minimum number of times the procedure required to obtain the integral hull of P . However, even in \mathbb{R}^2 , there exist polyhedra with arbitrarily large Chvátal-Gomory rank. Therefore, using the Chvátal-Gomory procedure to derive the integral hull of a given polyhedron is often costly and inefficient in practice.

1.4 Graph theory

The definitions and notation in graph theory used in this dissertation are mainly taken from Diestel [2000] and Schrijver [2003].

A *graph* G consists of a set of *nodes* V and a set of *edges* E , such that each edge in E has its both *extremities*, in V , that is $G = (V, E)$. Given a graph G , its node set can be referred to as $V(G)$, and its edge set as $E(G)$. If a node $v \in V$ is an extremity of an edge $e \in E$, we say that e is *at* or *incident* with v and v is *incident* with e . An edge e with extremities u, v can also be denoted by uv . Two distinct nodes $u, v \in V$ are *adjacent*, or *neighbors*, if they are the extremities of an edge $e = uv \in E$. A graph is *simple* if there do not exist two distinct edges that have the same extremities, and there does not exist an edge in it with both extremities being the same node. All graphs mentioned in this dissertation are considered simple.

If a graph $G' = (V', E')$ satisfies $V' \subseteq V$, $E' \subseteq E$, then G' is a *subgraph* of G , or we can also say that G *contains* G' .

Given a graph $G = (V, E)$ and a node set $S \subseteq V$, let $E[S]$ be the set composed of all the edges having both extremities in S , and let $G[S] = (S, E[S])$ be the subgraph of G induced by S . Accordingly, for an edge set $F \subseteq E$, let $V[F]$ be the set composed of all the nodes having at least one incident edge in F , and $G[F] = (V[F], F)$.

The *degree* of a node $v \in V$ is equal to the number of its neighbors and is denoted as $\deg(v)$. A node of degree 0 is said to be *isolated*. Let $N(v)$ denote the set of neighbors of a node v , and $\delta(v)$ the set of edges incident with v .

A *path* is a non-empty graph $G_p = (V_p, E_p)$ of the form

$$\begin{aligned} V_p &= \{v_0, v_1, \dots, v_k\}, \\ E_p &= \{v_0v_1, v_1v_2, \dots, v_{k-1}v_k\}, \end{aligned}$$

where the v_i are all distinct, $i = 1, \dots, k$. The *length* of a path equals to the number of its edges. Note that the length of a path can be 0.

The two nodes of degree one in V_p are called the *ends* of P and the two edges at these nodes are the *end edges* of P . If r is an end of P and e is the end edge of P at the other end of P , P is called an *re-path* and is denoted by P_{re} . If r and t are the two ends of P , P is called an *rt-path* and is denoted by P_{rt} .

If the path $G_p = (V_p, E_p)$ has length at least 2, then the graph $C = (V_p, E_p \cup \{v_kv_0\})$ is called a *cycle*. A graph G is *acyclic* if it does not contain any cycle.

A non-empty graph is called *connected* if there exists at least one path between any two of its nodes. A maximal connected subgraph of a graph G is called a *connected component* of G .

An acyclic graph is a *forest*, if it is also connected it is then a *tree*. A *cactus graph* is a connected graph such that each edge belongs to at most one cycle. A *complete graph* is a graph that all the nodes are pairwise adjacent. A complete graph of n nodes is often denoted K^n . Define a series composition as replacing an edge by two edges in series, and a parallel composition as replacing an edge by two edges in parallel. A *series-parallel graph* is a graph such that each of its 2-connected components can be generated from K^2 with a sequence of series and parallel compositions.

Given an undirected graph $G = (V, E)$, the *graphic matroid* (E, \mathcal{I}) satisfies that the independent sets in \mathcal{I} are the subsets of E that induce forests of G .

Given $G = (V, E)$ with a specific node, the *root* $r \in V$, a *rooted-tree* $T = (U, F)$ (also called an *r-tree*) is a tree of G which contains r if $U \neq \emptyset$. T is also said to be *rooted at* r . Notice that (\emptyset, \emptyset) and $(\{r\}, \emptyset)$ are both considered r -trees. The *tree order* associated with an r -tree $T = (U, F)$ is a partial order \leq over V that is defined as follows. For any two distinct nodes $u, v \in U$, we define $u \leq v$, if u belongs to the only rv -path in T . For example, in Figure 1.2 one has $r \leq v_1 \leq v_3$, $r \leq v_2$, $v_2 \leq v_4$ and $v_2 \leq v_5$. The *up-closure* and *down-closure* of v , denoted $\lfloor v \rfloor$ and $\lceil v \rceil$ respectively, are defined as $\lfloor v \rfloor = \{u \in U : v \leq u\}$, $\lceil v \rceil = \{u \in U : u \leq v\}$. For instance, in Figure 1.2, the up-closure and down-closure of v_2 are $\lfloor v_2 \rfloor = \{v_4, v_5\}$, $\lceil v_2 \rceil = \{r\}$, respectively.

A node $v_a \in V$ is an *articulation node*, or a *one-node cutset*, in $G = (V, E)$, if there exist two distinct sets $S_1, S_2 \subseteq V$ such that each path between any $u \in S_1$ and any $v \in S_2$

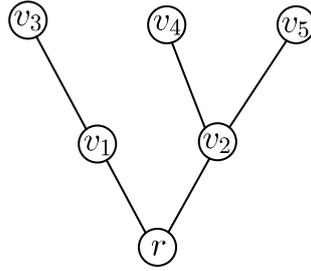


Figure 1.2: Tree order in a rooted-tree

contains v_a . Accordingly, an edge $e_b \in E$ is a *bridge* in $G = (V, E)$, if there exist two distinct sets $S_1, S_2 \subseteq V$ such that each path between any $u \in S_1$ and any $v \in S_2$ contains e_b . Such articulation node or bridge is said to *separate* S_1 and S_2 .

A *1-clique-sum*, or a *1-sum*, of two graphs G_1, G_2 is formed by gluing these two graphs at a node. Note that the shared node is then an articulation node in the resulting graph.

A graph $G = (V, E)$ is called *k-connected*, if $|V| > k$ and for each $S \subseteq V$ with $|S| < k$ the subgraph $G' = G[V \setminus S]$ is connected. Accordingly, a graph $G = (V, E)$ is called *k-edge-connected* if $|V| > k$ and for each $F \subseteq E$ with $|F| < k$, the subgraph $G' = (V, E \setminus F)$ is connected.

A set $\pi = \{S_1, \dots, S_k\}$ of non-empty subsets of V is called a *partition* of V if the sets in π are pairwise disjoint and their union is V . If $\{S, V \setminus S\}$ is a partition of the node set V , the set composed of all the edges having one extremity in each of S and $V \setminus S$ is called a *cut*, and denoted $\delta_G(S)$ or $\delta_G(V \setminus S)$. For any $S_1, S_2 \subseteq V$ with $S_1 \cap S_2 = \emptyset$, let $\delta(S_1, S_2)$ denote set of edges having one extremity in each set S_1 and S_2 , that is, $\delta(S_1, S_2) = \delta_G(S_1) \cap \delta_G(S_2)$. The subscript G may be omitted in all the notation if the underlying graph is clear from the context.

For the sake of simplification, given a graph $G = (V, E)$ and a node set $U \subseteq V$, the subgraph $G[V \setminus U]$ of G after removing U is also denoted $G \setminus U$. Similarly, for some $F \subseteq E$, one has $G \setminus F = (V, E \setminus F)$. Additionally, $G \setminus e$ is also used as a simplification of $G \setminus \{e\}$.

Chapter 2

Maximum bounded rooted-tree problem

In the field of content delivery service, one of the major issues is to find a way to deliver the content (e.g. documents, software, video streams) to the users effectively and efficiently. Peer-to-peer (P2P) architecture, as it allows users to both provide and use resources, can reduce the setup and running cost and increase greatly the scalability of the network. Hence, it is widely used in content delivery networks. Moreover, the under-provisioning issue is often addressed respecting large P2P networks, and the upload capacities of devices are commonly recognized as the bottlenecks in these networks. Therefore, the problem can be described as follows. A tree is desired by the service provider to deliver the content from a server to as many users as possible, with respect to the limit of the amount of data can be sent by each device in the network. This problem has been addressed as a combinatorial optimization problem called the MBrT problem. Moreover, if the upload limits, or so-called capacity constraints, of the devices are neglected, one obtains another problem called the MrT problem. These two problems are the main focus of this dissertation.

This chapter first defines these two problems. For each problem we present the complexity result and the associated polyhedra, and propose a formulation for each polytope. Finally, a literature review on related problems and their applications are presented.

2.1 Introduction

Firstly, the definition of the Maximum r -Tree Problem is given below.

Definition 2.1.1 (The MrT problem). *Given an undirected graph $G = (V, E)$, a node $r \in V$, an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$, and a node-price vector $\mathbf{p} \in \mathbb{R}^V$, the Maximum*

r -Tree (MrT) problem consists of finding an r -tree $T = (U, F)$ of G with maximum value

$$f(T) = w(F) + p(U).$$

When each node of G is required to satisfy some degree requirement, the Maximum Bounded r -Tree problem can be defined as follows.

Definition 2.1.2 (The MBrT problem). *Given an undirected graph $G = (V, E)$, a node $r \in V$, a capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$ with $c_v \geq 1$ for any node $v \in V$, an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$, and a node-price vector $\mathbf{p} \in \mathbb{R}^V$, the Maximum Bounded r -Tree (MBrT) problem consists of finding an r -tree T of G with maximum value $f(T)$ and such that the following capacity requirements*

$$|\delta_T(v)| \leq c_v \quad \forall v \in V$$

are satisfied.

An r -tree of G that satisfies such capacity requirements is then called a *bounded r -tree*.

The MrT problem can be seen as a relaxed MBrT problem from leaving out the capacity constraints. On the other hand, if we keep the capacity constraints of the MBrT problem but get rid of the constraints associated with the r -tree properties, we then obtain the following Maximum Simple b -Matching problem.

Definition 2.1.3 (The MSbM problem). *Given an undirected graph $G = (V, E)$, a capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$ with $c_v \geq 1$ for any node $v \in V$, and an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$, the Maximum Simple b -Matching (MSbM) problem consists of finding a set of edges $F \subseteq E$ with maximum value $w(F)$ and such that the following capacity requirements*

$$|\delta_{G[F]}(v)| \leq c_v \quad \forall v \in V$$

are satisfied.

An edge set F of G that satisfies the capacity requirements is called a *b -matching*.

Despite these three aforementioned problems are closely related, only the MSbM problem has been well studied in the literature, whereas the MrT problem and the MBrT problem have not received much attention.

The complexity result concerning each problem is given below.

Using a similar approach as in Didi-Biha et al. [2015], the MrT problem can be proved to be \mathcal{NP} -hard by polynomially reducing the Minimum Steiner Tree (MST) problem to the

MrT problem. The MST problem is well-known to be \mathcal{NP} -hard (Karp [1972]). Hereafter we present the polynomial-time reduction from the MST problem to the MrT problem.

Theorem 2.1.4. *The MST problem is polynomially reducible to the MrT problem.*

Proof. Given an undirected graph $G = (V, E)$ and a set $T \subseteq V$ of *terminals*, a Steiner tree is a tree (U, F) of G that spans all the terminals, that is, $T \subseteq U$. Let $\mathbf{w} \in \mathbb{R}^E$ be an edge-weight vector. The MST problem consists of finding a Steiner tree (U, F) with minimum value $w(F)$.

Consider an instance of the MST problem, with $T = \{v_1, \dots, v_t\} \subseteq V, t \geq 3$, and $\mathbf{w} \in \mathbb{R}_+^E$. We construct a graph $G' = (V', E')$ that is defined as follows

$$\begin{aligned} V' &= V \cup \{s_i : i = 1, \dots, t-1\}, \\ E' &= E \cup \{v_i s_i : i = 1, \dots, t-1\}. \end{aligned}$$

We can now construct an instance of MrT problem on G' by picking out $v_t \in T$ as the root node and defining an edge-weight vector $\mathbf{w}' \in \mathbb{R}_+^{E'}$ such that

$$w'_e = \begin{cases} -w_e & \text{if } e \in E, \\ M & \text{if } e \in E' \setminus E, \end{cases}$$

where M is a sufficiently large positive number. Besides, all the node weights are set to 0. Then one looks for a maximum r -tree of G' rooted at v_t . This construction can be done in polynomial time.

To show that an optimal solution of MrT problem on G' yields an optimal solution of the MST problem on G , let $G'[F']$ be a maximum r -tree of G' with $F' \subseteq E'$. Because of the large weights on the edges $v_i s_i, i = 1, \dots, t-1$, it is straightforward to see that F' contains all the edges $v_i s_i$, for $i = 1, \dots, t-1$, which gives us $T \setminus \{v_t\} \subseteq V'[F']$. In addition, as v_t is the root and $F' \neq \emptyset$, one also has $v_t \in V'[F']$. Therefore,

$$T \subseteq V'[F'].$$

Moreover, as one has $\delta(s_i) = \{v_i s_i\}$ for any $i \in \{1, \dots, t-1\}$, then $v_i s_i$ is a leaf edge and s_i is a leaf of the r -tree $G'[F']$. Let $F = F' \cap E$. It can be seen that $G[F]$ is connected and $v_i \in V[F]$ for $i \in \{1, \dots, t-1\}$. Meanwhile, since it is known that $v_t \in F'$ and $F' \cap \delta(v_t) \subseteq F$, one has $v_t \in V[F]$. Thus,

$$T \subseteq V[F].$$

Furthermore, $G[F]$ is a subgraph of acyclic graph $G'[F']$, which indicates that it is also

acyclic. Therefore, $G[F]$ is a Steiner tree of G with weight

$$w(F) = -w'(F) = -(w'(F') - M(t - 1)).$$

Conversely, given any Steiner tree of G , it is known that it contains all nodes in T . In addition, by the construction of G' , it can be seen that $v_i s_i$ is the only edge incident with s_i in G' . Thus, by extending each node v_i , for $i = 1, \dots, t - 1$, to an edge $v_i s_i$, one then obtains an r -tree of G' . In other words, each Steiner tree of G can be associated with an r -tree of G' that contains all the edges $v_i s_i$, for $i = 1, \dots, t - 1$.

Therefore any maximum r -tree $G'[F']$ of G' corresponds to a minimum Steiner tree $G[F]$ of G . \square

As a result of this polynomial reduction and the \mathcal{NP} -hardness of the MST problem, one then has that the MrT problem is \mathcal{NP} -hard.

Corollary 2.1.5. *The MrT problem is \mathcal{NP} -hard.* \square

As for the MBrT problem, if $p_v = 0$ for any $v \in V$ and $w_e = \lambda$ for any $e \in E$, where $\lambda \in \mathbb{R}$ is positive, it is then called a Maximum Size Bounded Rooted-Tree (MSBrT) problem. The MSBrT problem has been proved to be \mathcal{NP} -hard, see Chakareski et al. [2009], Kerivin et al. [2014]. Consequently, the following theorem holds.

Theorem 2.1.6. *The MBrT problem is \mathcal{NP} -hard.* \square

Alternatively, the \mathcal{NP} -hardness of the MBrT problem can also be deduced from the \mathcal{NP} -hardness of the MrT problem, as the latter is a relaxation of the former.

On the other hand, the MSbM problem has been shown to be polynomially solvable. Several polynomial-time algorithms have been proposed in the literature (Pulleyblank [1973], Marsh [1979], Gabow [1983], Anstee [1987], Gerards [1995]).

2.2 Associated polyhedra

In this section we define the polyhedra associated with the three problems introduced in Definitions 2.1.1 - 2.1.3, and propose a formulation for each polytope.

Consider any graph $G = (V, E)$, the *incidence vector* \mathbf{y}^U of a node subset U of V is defined as

$$y_v^U = \begin{cases} 1 & \text{if } v \in U, \\ 0 & \text{if } v \notin U. \end{cases}$$

Accordingly, the incidence vector \mathbf{x}^F of an edge subset F of E is defined as

$$x_e^F = \begin{cases} 1 & \text{if } e \in F, \\ 0 & \text{if } e \notin F. \end{cases}$$

The incident vector of a subgraph (U, F) of G is represented by $\begin{bmatrix} \mathbf{x}^F \\ \mathbf{y}^U \end{bmatrix}$.

The convex hull of the incidence vectors of r -trees of G is called the *Extended r -Tree Polytope* and is denoted by

$$R_{xy}(G, r) = \text{conv}(\{ \begin{bmatrix} \mathbf{x}^F \\ \mathbf{y}^U \end{bmatrix} \in \{0, 1\}^E \times \{0, 1\}^V : (U, F) \text{ is an } r\text{-tree} \}).$$

The following formulation for the extended r -tree polytope is proposed by Goemans [1994].

Let $P_{xy}(G, r)$ be the polyhedron of \mathbb{R}^{E+V} defined by

$$x(E) - y(V \setminus \{r\}) = 0, \tag{2.1}$$

$$x(E[S]) - y(S \setminus \{v\}) \leq 0 \quad \forall v \in S \subseteq V, |S| \geq 2, \tag{2.2}$$

$$y_r \leq 1, \tag{2.3}$$

$$x_e \geq 0 \quad \forall e \in E. \tag{2.4}$$

Equation (2.1) is called the *r -tree equation*. Inequalities (2.2) are called the *extended subtour elimination inequalities*. Inequalities (2.3) and (2.4) are the trivial *upper bound inequalities* and *non-negativity inequalities*.

This formulation has been proved to completely describe the extended r -tree polytope on series-parallel graphs (Goemans [1994]), that is, the following theorem holds.

Theorem 2.2.1. *Let G be a series-parallel graph. $R_{xy}(G, r) = P_{xy}(G, r)$. \square*

Alternatively, an r -tree can also be represented only by its edges. When focusing on edges only, we say that an edge set $F \subseteq E$ induces an r -tree if the graph $G_F = (V[F] \cup \{r\}, F)$ is an r -tree. It can be noted that the empty set induces the r -tree $G_\emptyset = (\{r\}, \emptyset)$.

The convex hull of the incidence vectors of edge sets inducing r -trees of G is called the *r -Tree Polytope* and is denoted by

$$R_x(G, r) = \text{conv}(\{ \mathbf{x}^F \in \{0, 1\}^E : G_F \text{ is an } r\text{-tree} \}).$$

Notice that the r -tree (\emptyset, \emptyset) is only our concern in the discussion with both node- and edge-indexed variables, as there is no way to distinguish it from $(\{r\}, \emptyset)$ with only edge-indexed variables.

Let $P_x(G, r)$ be the polyhedron defined by the linear system composed of following inequalities

$$x_e - x(\delta(S)) \leq 0 \quad \forall e \in E[S], r \in \bar{S}, \quad (2.5)$$

$$x(E[S]) \leq |S| - 1 \quad \forall S \subseteq V, |S| \geq 3, \quad (2.6)$$

$$x_e \leq 1 \quad \forall e \in E, \quad (2.7)$$

$$x_e \geq 0 \quad \forall e \in E. \quad (2.8)$$

Inequalities (2.5) are called the *connectivity inequalities*. Inequalities (2.6) are called the *subtour elimination inequalities*. Inequalities (2.7) and (2.8) are called the *box inequalities*.

It can be proved that $P_x(G, r)$ is a formulation for the polytope $R_x(G, r)$, that is,

Proposition 2.2.2. $R_x(G, r) \cap \mathbb{Z}^E = P_x(G, r) \cap \mathbb{Z}^E$.

Proof. Inequalities (2.5) ensure that each edge is connected with the root node. Inequalities (2.6) ensure the acyclicity. \square

Similarly, the bounded r -tree polyhedra can be defined as follows.

The convex hull of the incidence vectors of bounded r -trees of G is called the *Extended Bounded r -Tree Polytope* and is denoted by

$$B_{xy}(G, r, \mathbf{c}) = \text{conv}(\{ \begin{bmatrix} \mathbf{x}^F \\ \mathbf{y}^U \end{bmatrix} \in \{0, 1\}^{E+V} : (U, F) \text{ is a bounded } r\text{-tree} \}).$$

Recall that a bounded r -tree is an r -tree that satisfies the capacity requirements, which can be presented as the following *extended capacity inequalities*:

$$x(\delta(v)) - c_v y_v \leq 0 \quad \forall v \in V, \quad (2.9)$$

Therefore, a formulation $P_{xy}(G, r, \mathbf{c})$ for the extended bounded r -tree polytope can be obtained by incorporating the extended capacity inequalities into the formulation $P_{xy}(G, r)$, that is,

$$P_{xy}(G, r, \mathbf{c}) = P_{xy}(G, r) \cap \{ [\mathbf{x}] \in \mathbb{R}^{E+V} : [\mathbf{x}] \text{ satisfies (2.9)} \}.$$

This formulation used the inequalities from the formulation for the extended r -tree polytope, on top of which the *extended capacity inequalities* (2.9) are added to enforce the capacity requirements.

Proposition 2.2.3. $P_{xy}(G, r, \mathbf{c})$ is a formulation for $B_{xy}(G, r, \mathbf{c})$, that is,

$$B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V} = P_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}.$$

Proof. According to the formulation for the extended r -tree polytope, any integral vector that satisfies inequalities (2.2), (2.3) and (2.4) induces an r -tree. Moreover, with inequalities (2.9) being satisfied, any integral vector in $P_{xy}(G, r, \mathbf{c})$ induces a bounded r -tree. Conversely, any incidence vector of a bounded r -tree of G clearly satisfies all the inequalities (2.1) - (2.9). \square

Notice that for the inequalities (2.2), by setting $S = \{u, v\}$ for any $uv = e \in E$, one gets the following *subgraph inequalities*:

$$x_e - y_v \leq 0 \quad \forall v \in V, e \in \delta(v). \quad (2.10)$$

It is also worth noting that the subgraph inequalities (2.10) together with inequalities (2.3) and (2.4) ensure that the following inequalities hold.

$$\begin{aligned} x_e &\leq 1 \quad \forall e \in E, \\ y_v &\geq 0 \quad \forall v \in V. \end{aligned}$$

The convex hull of the incidence vectors of edge sets inducing bounded r -trees of G is called the *Bounded r -Tree Polytope* and is denoted by

$$B_x(G, r, \mathbf{c}) = \text{conv}(\{\mathbf{x}^F \in \{0, 1\}^E : G_F \text{ is a bounded } r\text{-tree}\}).$$

Similarly to the r -tree case, we need to incorporate the following *capacity inequalities*

$$x(\delta(v)) \leq c_v \quad \forall v \in V \quad (2.11)$$

into $P_x(G, r)$ to obtain a formulation for the bounded r -tree polytope, that is,

$$P_x(G, r, \mathbf{c}) = P_x(G, r) \cap \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x} \text{ satisfies (2.11)}\}.$$

Proposition 2.2.4. $P_x(G, r, \mathbf{c})$ is a formulation for $B_x(G, r, \mathbf{c})$, that is,

$$B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E = P_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E.$$

Proof. Recall that for the r -Tree polytope, it has been proved that any integral vector that satisfies inequalities (2.5), (2.6), (2.7) and (2.8) induces an r -tree. If such integral vector also satisfies inequalities (2.11), then the induced graph is also bounded. Therefore, any integral vector in $P_x(G, r, \mathbf{c})$ induces a bounded r -tree. Conversely, any incidence vector of the edge set of a bounded r -tree of G clearly satisfies all the inequalities (2.5), (2.6), (2.7), (2.8) and (2.11). \square

The convex hull of the incidence vectors of b -matchings in G is called the *Simple b -Matching Polytope* and is denoted by

$$M_x(G, \mathbf{c}) = \text{conv}(\{\mathbf{x}^F \in \{0, 1\}^E : G_F \text{ is bounded by } \mathbf{c}\}).$$

The complete description of the simple b -matching polytope was given by Schrijver [2003] (see also Edmonds [1965], Pulleyblank [1973] and Edmonds [1975]).

Let $P_x(G, \mathbf{c})$ be the polyhedron of \mathbb{R}^E defined by

$$x(E[S]) + x(F) \leq \left\lfloor \frac{1}{2}c(S) + |F| \right\rfloor \quad \forall S \subseteq V, F \subseteq \delta(S) \text{ with } c(S) + |F| \text{ odd}, \quad (2.12)$$

$$x(\delta(v)) \leq c_v \quad \forall v \in V, \quad (2.13)$$

$$0 \leq x_e \leq 1 \quad \forall e \in E. \quad (2.14)$$

Inequalities (2.12) are known as the *blossom inequalities*. Schrijver [2003] proved that (2.12)-(2.14) completely describe the simple b -matching polytope as stated in the following theorem (see also Edmonds [1965]).

Theorem 2.2.5. $M_x(G, \mathbf{c}) = P_x(G, \mathbf{c})$. □

Moreover, Schrijver [2003] showed that if we discard the condition of $c(S) + |F|$ being odd for the blossom inequalities (2.12), the system is then TDI (see also Pulleyblank [1981]).

Any bounded r -tree of a graph G is also an r -tree of G , and any edge set of a bounded r -tree of G is also a b -matching of G . Hence, one has the following proposition.

Proposition 2.2.6. *Given an undirected graph $G = (V, E)$, a root node $r \in V$, and a capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$ with $c_v \geq 1$ for any node $v \in V$, then the following relations among the polyhedra hold.*

$$\begin{aligned} B_{xy}(G, r, \mathbf{c}) &\subseteq R_{xy}(G, r), \\ B_x(G, r, \mathbf{c}) &\subseteq R_x(G, r), \\ B_x(G, r, \mathbf{c}) &\subseteq M_x(G, \mathbf{c}). \quad \square \end{aligned}$$

An immediate consequence of this proposition is stated as the corollary below.

Corollary 2.2.7. *Any inequality valid for $R_{xy}(G, r)$ is also valid for $B_{xy}(G, r, \mathbf{c})$. Any inequality that is valid for either $R_x(G, r)$ or $M_x(G, \mathbf{c})$ is also valid for $B_x(G, r, \mathbf{c})$. □*

It is worth noting that, for the MBrT problem, one set of nodes that stand out in both the algorithm development and the polyhedral study are the nodes with unit capacity.

It can be noticed that these nodes (except r) are not providing any connectivity to the other nodes, in other words they can only appear as leaves in any bounded r -tree. Let (2.15) denote the set of these nodes.

$$O := \{v \in V \setminus \{r\} : c_v = 1\}. \quad (2.15)$$

Nodes in O have a significant impact on the dimension analysis as well as the facet-defining conditions of valid inequalities for the polyhedra associated with MBrT problem.

As a result of the existence of O , it can be noticed that even if G is connected, there might still be some nodes or edges in G such that they cannot be reached by any bounded r -tree of G . We say that a node v (edge e , respectively) of G is *unreachable* from root r if there does not exist any path in G containing r and v (e , respectively) that does not have any node in O as an internal node. Let V_u and E_u be the sets composed of the unreachable nodes and edges of G , respectively. Any incidence vector of a bounded r -tree of G straightforwardly satisfies the following equations

$$x_e = 0 \quad \forall e \in E_u, \quad (2.16)$$

$$y_v = 0 \quad \forall v \in V_u. \quad (2.17)$$

In fact, for polyhedral study, these unreachable nodes and edges are insignificant. Let $G_c = (V \setminus V_u, E \setminus E_u)$. A characterization of $B_{xy}(G_c, r, \mathbf{c}')$ ($B_x(G_c, r, \mathbf{c}')$, respectively) yields a characterization of $B_{xy}(G, r, \mathbf{c})$ ($B_x(G, r, \mathbf{c})$, respectively), where \mathbf{c}' is the restriction of \mathbf{c} to $V \setminus V_u$.

In order to prove it, we give a more generalized result considering any polytope with some fixed variables.

Given two polytopes

$$P = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} \leq \mathbf{b}\},$$

$$P' = \left\{ \begin{bmatrix} \mathbf{x} \\ \mathbf{x}^0 \end{bmatrix} \in \mathbb{R}^{n+m} : A\mathbf{x} \leq \mathbf{b}, \mathcal{I}^0 \mathbf{x}^0 = \mathbf{b}^0 \right\},$$

where \mathcal{I}^0 is the identity matrix of size m , one has the following result.

Theorem 2.2.8. *P' is integral if and only if P is integral.*

Proof. As P is a projection of P' , according to Theorem 1.2.7, P' is integral if P is integral.

Now assume that P' is integral. Consider an arbitrary extreme point $\bar{\mathbf{x}}$ with respect to P , and let $S(\bar{\mathbf{x}})$ be the system of equations that defines it. It is straightforward to see that $\begin{bmatrix} \bar{\mathbf{x}} \\ \mathbf{x}^0 \end{bmatrix}$ is defined by the system composed of $S(\bar{\mathbf{x}})$ and $\mathcal{I}^0 \mathbf{x}^0 = \mathbf{b}^0$. Hence, $\begin{bmatrix} \bar{\mathbf{x}} \\ \mathbf{x}^0 \end{bmatrix}$ is an

extreme point with respect to P' , and it is integral as P' is integral. Thus \bar{x} is integral, and hence P is integral, which completes the proof. \square

Moreover, for graphs that contain unreachable node or edges, one can actually get rid of them in polynomial time with some search algorithms. Therefore, the rest of the dissertation is based on the following assumption.

Assumption 2.2.9.

$$\begin{aligned} V_u &= \emptyset, \\ E_u &= \emptyset. \end{aligned}$$

Denote $G_r = (V_r, E_r)$ the connected component that contains r in the graph $G[V \setminus O]$. One can deduce straightforwardly from this assumption that $G_r = G[V \setminus O]$.

2.3 State of the art

This section provides an overview of the most important known results for the aforementioned problems.

On the one hand, we show that there is a lack of intensive theoretical study of the MBrT problem in the literature, despite some related problems have been well studied.

On the other hand, we also show that in content delivery networks, especially for the live video streaming service, the underlying problem of the business procedure has been found to have a closer connection with the MBrT problem, instead of the traditionally identified max-flow problem. Thus, the quantity-focused strategy for content delivery which is in use currently should shift to a quality-over-quantity one in order to improve the overall quality of service in the dedicated networks. Besides, the algorithms that are used currently in the industry often refer to the algorithms for problems such as shortest path problem and minimum spanning tree problem. Therefore, the theoretical study and algorithms for the MBrT problem and its variants deserve more attention from the literature.

2.3.1 Degree-constrained problems

The MBrT problem is relatively new in the field of combinatorial optimization, thus hardly any research has been conducted around it. It was first addressed in Chakareski et al. [2009], where the author presented a few preliminary results and raised some open

questions. The authors proved its \mathcal{NP} -hardness, and provided a formulation for the problem. In addition, a proposed heuristic was also presented with experimental results. Several different models were proposed for the MBrT problem in Kerivin et al. [2011], including a cut-based model, a level-based model and a flow-based model. The authors proved that the linear relaxation corresponding to each model is polynomially solvable.

In Kerivin and Simon [2012], Kerivin et al. [2014], it is shown that the decision problem of the uniform-weight restriction of the MBrT problem, namely the MSBrT problem, is \mathcal{NP} -complete as 3-SAT is polynomially reducible to it. However, the authors showed that some special cases of the MBrT problem are polynomially solvable. A dynamic programming approach was proposed to solve the MSBrT problem on trees in polynomial time. In addition, on complete graphs, the author presented a polynomial-time algorithm that solves the MSBrT problem based on the trivial fact that computing a Hamiltonian path in a complete graph can be done in linear time.

As a well-studied topic in the literature, b -matching and its associated problems have a connection with the MBrT problem in the aspect of constrained node-degree. There are several different versions of b -matching problems.

The general weighted b -matching problem aims at finding a maximum-weight b -matching in a graph. It has no other constraints aside from the node-degree constraints. The associated general b -matching polytope was characterized by Edmonds [1965], while a minimal TDI system was given by Pulleyblank [1981]. When incorporating the edge-capacity constraints with the general weighted b -matching problem, one gets the so-called weighted capacitated b -matching problem. For this problem, each edge has an integer capacity which the variable associated with this edge should not exceed. If the capacity of each edge is set to 1, we obtain the MSbM problem. The simple b -matching polytope and an associated TDI system was proposed by Schrijver [2003]. The MSbM problem is the closest to the MBrT problem of all the b -matching problems, as the MBrT problem also has the unit edge-capacity constraints.

Algorithms for other versions of the MSbM problem have also been proposed in the literature (Johnson [1979], Gabow [1983], Gondran and Minoux [1984], Gerards [1995]).

The minimum bounded degree spanning tree (MBDST) problem is closely related to the MBrT problem in the sense that node-degree constraints are enforced, yet it looks for a spanning tree instead of a rooted tree and all the nodes have the same degree upper bound $k \geq 2$.

In Goemans [2006], the MBDST problem is considered, providing a crucial point that the support graph of the solution to the linear relaxation is proved to have the so-called *laminar property*. This laminar property is exploited to build a matroid, and with the

help of matroid intersection (Edmonds [1970, 1971], Lawler [1975]), the author proposed a polynomial-time approximation algorithm. The algorithm is able to obtain a suboptimal solution where the degree of each node is bounded by $k + 2$. In Zhao [2012], we proved that the MBrT problem does not possess the laminar property as described in Goemans [2006] for the MBDST problem, which makes it impossible to directly adapt the said approximation algorithm for the MBrT problem.

In Singh and Lau [2007], by using an extension of the iterative rounding method introduced by Jain [1998], the authors provided another polynomial-time approximation algorithm to obtain a suboptimal solution which improved the node-degree bound to $k + 1$.

Other methods such as primal separation and ant colony algorithm have also been studied on the MBDST problem, where some showed the potential in the computational aspect (Behle et al. [2007], Letchford and Lodi [2003], Raidl [2000], Bui et al. [2012], Malik [2012]).

2.3.2 Non-degree-constrained related problems

In Goemans [1994], the MrT problem and the extended r -tree polytope $R_{xy}(G, r)$ are studied in the context of the Steiner tree problem. A formulation is proposed along with the results concerning the separation problem for the constraints and the facets of the polytope. The author then proved that the formulation leads to a characterization of $R_{xy}(G, r)$ on series-parallel graphs. Additionally, the author also showed that the polyhedral study of $R_{xy}(G, r)$ can be restricted on 2-connected subgraphs by proving the feasibility of a decomposition through 1-sum. Nevertheless, to the best of our knowledge the r -Tree polytope $R_x(G, r)$ has not been considered in the literature.

The Steiner tree problem also share a characteristic with the MBrT problem and the MrT problem. More precisely, the root node in the MBrT problem and the MrT problem can be seen as a terminal. Thus a feasible solution to these two problems is always a Steiner tree. The Steiner tree problem is one of the most popular and well-studied topics in the field of combinatorial optimization. Although the Steiner tree problem is \mathcal{NP} -hard (Karp [1972]), even in grid graphs, planar graphs and bipartite graphs (Hakimi [1971], Garey and Johnson [1977, 1979]), there are special cases that have been proved to be polynomially solvable. If all nodes in the graph are terminals, it is then a minimum spanning tree problem, which can be solved by polynomial-time algorithms proposed by Kruskal [1956] and Prim [1957]. If there exist only two terminals in the graph, the problem becomes a shortest path problem, and can be solved by the algorithm of Dijkstra [1959] in polynomial time. Additionally, if the underlying graph is an outerplanar graph, a

series-parallel graph or a Halin graph, linear-time algorithms have been developed for the Steiner tree problem as well (Rardin et al. [1982], Wald and Colbourn [1982a,b], Winter [1987]). Further study on the algorithms and polyhedral structure of the Steiner tree problem can be found in Dreyfus and Wagner [1971], Lawler [1976], Wong [1984], Lucena [1993], Chopra et al. [1992], Goemans and Myung [1993], Chopra and Rao [1994a,b], Jain [1998], Didi-Biha et al. [2001].

Goemans and Myung [1993] presented several different formulations for the Steiner tree problem. In Chopra and Rao [1994a,b], the authors studied the polyhedral structure of the Steiner tree problem. Several new facet-defining inequalities were introduced, with which the authors proved that the description of the dominant of the Steiner tree polytope can be obtained. Moreover, Didi-Biha et al. [2001] proposed a new class of valid inequalities that generalizes the ones proposed by Chopra and Rao [1994b]. The authors showed that these inequalities can help to describe the dominant of the Steiner tree polytope in series-parallel graphs with the terminals satisfying certain conditions.

In Jain [1998], a 2-approximation algorithm is proposed for the generalized Steiner network problem relying on a property of the solution to the linear relaxation. It first obtains the solution to the linear relaxation with the assumption that the requirement function is weakly supermodular. Based on this assumption the author then proved that there exists at least one edge having value more than $\frac{1}{2}$ in any linear relaxation solution. The author proposed an algorithm which is able to iteratively round off the solution due to this property.

The connected subgraph problem is another problem that bears a similarity to the MBrT problem, as they both aim at finding a connected subgraph. In fact, the MBrT problem can be considered as a restriction of the connected subgraph problem. In Didi-Biha et al. [2015], the authors focused on the polyhedral aspect of the problem. A new set of facet-defining inequalities, namely the *matching-partition inequalities*, was introduced to strengthen the formulation. Although the general separation problem for these inequalities was proved to be \mathcal{NP} -hard, the authors showed that they are polynomially separable and are helpful for the polyhedral characterization on certain classes of graphs. Besides, valid inequalities and approximation algorithms concerning variants of connected subgraph problem have also been studied under different conditions in the literature (Cheriyān and Thurimella [1996], Bendali et al. [2007], Chan et al. [2008], Cheriyān and Vegh [2013], Cornaz et al. [2014]).

2.3.3 Applications in the telecommunication field

The MBrT problem originally comes from the telecommunication industry, more specifically, the content delivery service. This section introduces the context of the telecommunication problem, and some facts and techniques that are worth highlighting related to the problem.

Content delivery, including video streaming, has become one of the most popular services people are using on the Internet recently. According to the prediction of a technical report from Cisco in 2014 (Cisco [2014]), Internet Protocol (IP) video traffic will account for 79 percent of all consumer Internet traffic in 2018, up from 66 percent in 2013, and the sum of all forms of video will be in the range of 80 to 90 percent of global consumer traffic by 2018. Within the video traffic, by 2018, 67 percent of it will go through content delivery networks, up from 53 percent in 2013, see Cisco [2014]. In the case of the most famous video sharing website YouTube, in 2015, every second, about 97,000 videos are being viewed throughout the Internet.

Streaming from one or multiple servers simultaneously to this large amount of users and ensuring good quality of service at the same time is no doubt a challenge for any service provider. In order to meet the demand of video streaming, P2P architecture has been widely adopted to replace the old-fashion client-server architecture, as it has much better scalability and no requirements for any hierarchical organization or centralized control. Despite the supportive techniques that have been highly focused on in research and development, one fact that cannot be overlooked about P2P networks is the under-provisioning network resources, see Kerivin and Simon [2012], Passarella [2012], Sweha et al. [2012]. Furthermore, the bottleneck of the P2P network, as opposed to the assumption made for traditional network flow problems, are the upload capacities of the node devices instead of capacities of the links (Massoulié et al. [2007], Liu et al. [2010]).

Among all the approaches that have been developed for P2P streaming, although mesh-based approaches have shown its superiority in certain aspects, the tree-based approaches are still the most popular (Yiu et al. [2007], Padmanabhan et al. [2002], Magharei et al. [2007]). Moreover, the tree-based approaches fit better the under-provisioned context encountered in content delivery services (Kerivin and Simon [2012]).

The tree-based approaches can be further categorized into two sub-classes, the single-tree-based and multiple-tree-based approaches. They both use trees as overlay to deliver requested content from the servers to the clients. The difference between them is that multiple-tree-based approach splits the content stream into smaller sub-streams with the help of the Dynamic Adaptive Streaming over HTTP (DASH) mechanism (Stockhammer [2011]) and techniques such as Multiple Description Coding (MDC) or Layered Coding

(LC) (McCanne et al. [1996], Goyal et al. [1998], Li et al. [1998], Byers et al. [2001], Goyal [2001], Vitali [2007]). Each sub-stream is delivered through a particular delivery tree. Each peer might participate in a subset of all the trees. The quality of the content offered to each peer is positively related to the number of sub-streams received by that peer device.

As mentioned earlier, the upload capacity of the peers must be considered during the construction and the maintenance of the delivery trees. The policy adopted for the allocation of the upload capacities, and the algorithm used for the construction of the delivery trees can affect the quality of service significantly. In practice, peers in P2P networks may also fail or disconnect from the network (Banerjee et al. [2005]). To prevent or minimize the delivery disruption in this kind of situation, the content delivery tree or trees must be repaired within a short time.

Traditionally, one fundamental issue often addressed in literature about P2P streaming concerns the maximum bit-rate that can be sustained for all peers. However, due to the development of DASH and encoding techniques such as MDC and LC, the quality-over-quantity strategy is proved to be better than the quantity-focused one for modern streaming (Kerivin and Simon [2012]). Therefore, the objective now has become no longer the maximum bit-rate, but the best delivery tree or trees one can possibly find. Such trend in the telecommunication industry motivates the effort to study on problems such as the MBrT problem.

Another problem that is often focused on in the telecommunication field is the multicast tree construction problem (Oliveira et al. [2006]). Such problem often involves reducing the overall delay among the clients or reducing the overall cost on the links (Brosh and Shavitt [2004]), while looking for a spanning tree in general. In practice, algorithms that have been proposed for the multicast tree construction problem often employ some straightforward approaches such as the shortest path based approach and the minimum spanning tree based approach, or use heuristics or other techniques such as genetic algorithms and evolutionary algorithms (Tran et al. [2004], Li et al. [2004], Jurčík and Hanzalek [2005], Oliveira et al. [2006]).

Unfortunately, although there have been many research projects which have addressed the issues in content delivery networks, the under-provisioning aspect has not received much if any attention as it should otherwise have. In such context, researchers generally aimed at finding algorithms for the MBDST problem or the degree constrained Steiner tree problem with different constraints (Tran et al. [2004], Li et al. [2004], Liu et al. [2005], Jurčík and Hanzalek [2005], Oliveira et al. [2006], Cao et al. [2010]).

Therefore, hardly any work has defined explicitly the problem with regard to the under-provisioning as well as the quality-over-quantity objective. This eventually leads to the

current state of unbalance between the rising need in application and the lack of theoretical support.

2.4 Conclusion

In the field of combinatorial optimization, the MBrT problem has not been emphasized in research. It has been addressed recently with a few models and some polynomial cases. As a relaxation of the MBrT problem, the polyhedral structure and algorithms for the MSbM problem have been thoroughly studied by the literature. As for the MrT problem, there are some results obtained with respect to the extended r -tree polytope and its characterization, yet nothing has been done regarding the r -Tree polytope. Moreover, as far as we know, there has been neither intensive polyhedral study nor computational study for the MBrT problem.

There exist some related problems such as the Steiner tree problem, the MBDST problem and the connected subgraph problem, which have been focused by the literature. However, the results and algorithms developed cannot be used directly on the MBrT problem. On the contrary, applications of the MBrT problem in content delivery networks have received substantial attention from researchers, although the fact of under-provisioning issue of the networks is oftentimes overlooked.

Therefore, there is a compelling need for the study on the polyhedra associated with the MBrT problem as well as the MrT problem.

This dissertation is dedicated to the polyhedral, algorithmic and computational study of the MBrT problem and the MrT problem. The polyhedral study generally helps to provide dual bounds which are as tight as possible. Even for instances that cannot be solved quickly, tight dual bounds can help quantify the quality of heuristic-based solutions. We present some newly discovered facets of the polytopes and show that they indeed make a difference in providing tighter dual bounds in the computational test.

Chapter 3

Combinatorial algorithms for the $MBrT$ problem

As mentioned in the previous chapter, the $MBrT$ problem is \mathcal{NP} -hard in general, even for the $MSBrT$ problem where uniform edge-weights are considered. However in Kerivin et al. [2011], Kerivin and Simon [2012], Kerivin et al. [2014], the authors present polynomial-time algorithms for the $MSBrT$ problem on complete graphs and trees. The algorithm on complete graphs is based on constructing a Hamiltonian path, whereas a dynamic programming approach is used in the algorithm for trees.

In this chapter, we present polynomial-time algorithms for the $MBrT$ problem on different classes of graphs. In particular, an algorithm based on dynamic programming is proposed for the $MBrT$ problem on trees. We also propose algorithms to solve the $MBrT$ problem on cycles. We then explore the $MBrT$ problem on cactus graphs, since a cactus graph can be seen as a combination of trees and cycles. We prove that the $MBrT$ problem on a cactus graph can be decomposed into subproblems on the so-called cactus bases and trees. We prove that the $MBrT$ problem on cactus bases is polynomial, and using the decomposition property of cactus graphs, the $MBrT$ problem on cactus graphs can be solved in polynomial time. Besides, the proof of a general decomposition related to articulation nodes is given for the $MBrT$ problem.

Notably, the algorithms proposed in this chapter concern the general $MBrT$ problem with both edge weights and node weights. Nonetheless they can be converted into algorithms for any restriction of $MBrT$ problem such as the $MSBrT$ problem. Likewise, some relaxation of the $MBrT$ problem such as the MrT problem can also be solved by adapting the algorithms to the specific situation. For example, one can discard all the capacity-related parts in an algorithm for the $MBrT$ problem to obtain a corresponding algorithm for the MrT problem.

Recall that the MBrT problem is defined by an undirected connected graph $G = (V, E)$ with a root node $r \in V$, a node-capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$, an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$ and a node-price vector $\mathbf{p} \in \mathbb{R}^V$. A feasible solution to the MBrT problem is a bounded r -tree $T = (U, F)$ of G . The goal of the MBrT problem is to find a bounded r -tree T with maximal value of $f(T)$, where for any subgraph $T = (U, F)$ of G ,

$$f(T) = \sum_{v \in U} p_v + \sum_{e \in F} w_e.$$

Let $Opt_G(r)$ denote the value of $f(T)$ associated with a maximum bounded r -tree of G , where the subscript G can be omitted when the graph is clear from the context. Opt_G may also be used instead of $Opt_G(r)$ when the root is clear from the context.

3.1 On trees

If the underlying graph G is a tree, we present a dynamic-programming approach for solving the MBrT problem.

For any node $v \in V$, the set composed of all the neighbors of v in its up-closure is denoted by $N_u(v) = N(v) \cap \lfloor v \rfloor$. For any node $v \in V \setminus \{r\}$, the only neighbor of v in its down-closure is denoted by d_v , that is, $d_v \in N(v) \setminus N_u(v)$.

Given any node $v \in V$, let $g(v)$ be the value of a maximum non-empty bounded tree rooted at v of the subgraph $G[\lfloor v \rfloor]$ of G induced by the up-closure of v , where the capacity vector $\mathbf{c}^v \in \mathbb{Z}_+^{\lfloor v \rfloor}$ satisfies $c_v^v = c_v - 1$ if $v \neq r$, $c_v^v = c_v$ if $v = r$, and $c_s^v = c_s$, for $s \in \lfloor v \rfloor \setminus v$. In other words,

$$g(v) = \max\{f(T) : T \text{ is tree of } G[\lfloor v \rfloor] \text{ with } v \in V(T) \text{ and bounded by } \mathbf{c}^v\}.$$

It can also be seen that the following relation holds

$$Opt_{G[\lfloor v \rfloor]}(v) = \max\{0, g(v)\} \tag{3.1}$$

which gives $Opt_G(r) = \max\{0, g(r)\}$. Hence, the problem of calculating $Opt_G(r)$ is reduced to the problem of calculating $g(r)$.

For any leaf $v \in V \setminus \{r\}$, it is straightforward to see that

$$g(v) = p_v, \tag{3.2}$$

since the only non-empty bounded tree rooted at v is the graph reduced to v .

For any non-leaf node $v \in V$, let $N_u(v) = \{v_1, \dots, v_q\}$, we give the following technical lemma as a support of our latter results.

Lemma 3.1.1. *If a maximum non-empty bounded tree rooted at v contains v_k , $k \in \{1, \dots, q\}$, it also contains a maximum non-empty bounded tree rooted at v_k .*

Proof. Suppose otherwise that a maximum non-empty bounded tree T_v rooted at v contains a non-empty bounded tree T_{v_k} rooted at v_k which is not maximum. By replacing T_{v_k} in T_v by a maximum non-empty bounded tree rooted at v_k , one obviously obtains a non-empty bounded tree rooted at v whose weight is larger than T_v . Hence, it contradicts with the assumption. \square

For any non-leaf node $v \in V$, let

$$h(v_k) = w_{vv_k} + g(v_k)$$

for any $k \in \{1, \dots, q\}$. According to Lemma 3.1.1, the problem of calculating $g(v)$ reduces to

$$p_v + \max\left\{\sum_{v_k \in S} h(v_k) : S \subseteq N_u(v), |S| \leq c_v^v\right\}.$$

As it is a maximization problem over a uniform matroid, if $g(v_k)$ is known for all $k \in \{1, \dots, q\}$, it can be easily solved by a greedy algorithm in linear time, where at each step one selects a node v_k with the maximum non-negative $h(v_k)$ until there is no such nodes or $c_v - 1$ nodes have been chosen. Without loss of generality, suppose $h(v_1) \geq h(v_2) \geq \dots \geq h(v_i) > 0 \geq h(v_{i+1}) \geq \dots \geq h(v_q)$. Let $j = \min\{c_v^v, i\}$. The following equation holds.

$$g(v) = p_v + \sum_{k=1}^j h(v_k). \quad (3.3)$$

Algorithm 3.1 is used to obtain the value $g(r)$ using dynamic programming. It calculates the value of $g(v)$ for each node in V on G from the leaves to the root r .

The correctness of the solution obtained from Algorithm 3.1 is a direct consequence of the definition of $g(v)$ and the equation (3.3).

Proposition 3.1.2. *Algorithm 3.1 computes $g(r)$.* \square

It can be deduced that Algorithm 3.1 also computes a maximum-weight non-empty bounded r -tree. At each step of computing $g(v)$ for a non-leaf node $v \in V$, a maximum non-empty bounded tree T_v rooted at v is obtained by combining v with vv_k and T_{v_k} for all $k \in \{1, \dots, j\}$.

Algorithm 3.1: Algorithm to compute $g(r)$ on trees

Input : Tree $G = (V, E)$, $\mathbf{w} \in \mathbb{R}^E$ and $\mathbf{p} \in \mathbb{R}^V$.

Output: $g(r)$.

begin

- 1 | Set S as the set of the leaves in $V \setminus \{r\}$.
 - 2 | Compute $g(v)$ for each $v \in S$ according to (3.2).
 - | **while** $S \neq \{r\}$ **do**
 - 3 | | Set S as the node set such that for each $v \in S$, $g(v)$ is unknown and $g(v_k)$
 - | | is known for all $k \in \{1, \dots, q\}$.
 - 4 | | Compute $g(v)$ for each $v \in S$ according to (3.3).
-

Proposition 3.1.3. *Algorithm 3.1 has time complexity of $\mathcal{O}(n)$.*

Proof. For each $v \in V$ that is not a leaf, the greedy algorithm for calculating $g(v)$ has complexity of $\mathcal{O}(|N_u(v)|)$. In the algorithm, each $g(v)$ for all $v \in V$ is used once. Moreover, from the definition and calculation of $g(v)$, each of the terms w_e, p_v for any $e \in E, v \in V$ is used only once for the summation. Therefore, the algorithm has time complexity of $\mathcal{O}(n)$. \square

As an immediate consequence, the following theorem holds.

Theorem 3.1.4. *If G is a tree, the MBrT problem can be solved in the time of $\mathcal{O}(n)$.* \square

As the MBrT problem is proved to be polynomially solvable on trees, the following proposition can be developed thereby.

Corollary 3.1.5. *The MBrT problem on a graph containing only one cycle can be solved in the time of $\mathcal{O}(n^2)$.*

Proof. Given $G = (V, E)$ having only one cycle $C = (U_C, F_C)$ as a subgraph, we claim that the MBrT problem on G can be reduced to $|F_C|$ MBrT problems on trees.

As there exists only one cycle in G , each subgraph $G^e = (V, E \setminus \{e\})$, $e \in F_C$, is a tree. Therefore, one can solve the MBrT problem on each G^e , $e \in F_C$, using Algorithm 3.1. Moreover, there does not exist a bounded r -tree that contains all edges in F_C because of the acyclicity requirement. Hence, the maximum bounded r -tree on G is the maximum bounded r -tree among all the solutions obtained from each subgraph G^e , $e \in F_C$, that is,

$$Opt_G = \max\{Opt_{G^e} : e \in F_C\}.$$

Therefore the MBrT problem on G is polynomially solvable, and since this approach solves $|F_C|$ MBrT problems on trees, it has a running time of $\mathcal{O}(n^2)$. \square

3.2 On cycles

Assume now that graph G is a cycle on n nodes. Without loss of generality, let

$$\begin{aligned} V &= \{r, v_1, \dots, v_{n-1}\}, \\ E &= \{e_1, \dots, e_n\}, \end{aligned}$$

with

$$\begin{aligned} \delta(v_i) &= \{e_i, e_{i+1}\}, i = 1, \dots, n-1, \\ \delta(r) &= \{e_1, e_n\}. \end{aligned}$$

Recall that according to Assumption 2.2.9, $|O| \leq 1$, otherwise there will exist some unreachable nodes or edges.

If $c_r = 1$, the feasible solution space contains at most $2n$ integer points. If $c_r = 2$, the feasible solution space contains at most $\frac{n(n+1)}{2} + 1$ integer points. Thus the MBrT problem can be solved in polynomial time simply by going through all the possible solutions in the time of $\mathcal{O}(n^2)$. Notice that same argument also applies for the cases if $O \neq \emptyset$ holds in the two cases stated above, as the number of feasible solutions only reduces due to the existence of the unit-capacity node. Alternatively, according to Corollary 3.1.5, the MBrT problem on cycles can also be reduced to n MBrT problems on trees.

Nonetheless, there are other approaches to solve it without enumerating all the solutions or using the algorithm for trees. This section introduces linear-time algorithms for the MBrT problem on cycles.

Given any edge set $F \subseteq E$, denote by P_{rv}^F , an rv -path of $G[F]$ such that $P_{rv}^F \subseteq F, v \in V[F]$. For any $F \subseteq E$ with $r \in V[F]$, the weight of a maximum-weight nonempty bounded rv -path of $G[F]$ is denoted by

$$\varphi(F) := \max\{f(P_{rv}^F) : P_{rv}^F \neq \emptyset, v \in V[F]\}.$$

It is worth noting that according to the definition we have

$$\varphi(F) = \max\{f(T) : T \text{ is a bounded } r\text{-tree}, |\delta(r) \cap V(T)| = 1\}. \quad (3.4)$$

We first show that if F is an rv -path, $\varphi(F)$ can be obtained in linear time.

Proposition 3.2.1. *Given a path $P_{ru} \subseteq E$ between r and u of G for some $u \in V \setminus \{r\}$, a maximum-weight nonempty bounded rv -path of the subgraph $G[P_{ru}]$ can be obtained in linear time.*

Proof. Given an ru -path P_{ru} of length k , it is straightforward to see that there are exactly k nonempty rv -paths of $G[P_{ru}]$. Thus, finding a maximum-weight nonempty bounded rv -path of $G[P_{ru}]$ can be done in linear time. \square

We define two subpaths of G as follows.

$$P_1 = \begin{cases} \{e_2, e_3, \dots, e_n\} & \text{if } O = \emptyset, \\ \{e_o, e_{o+1}, \dots, e_n\} & \text{if } O = \{v_o\}, o \in \{1, \dots, n-1\}. \end{cases}$$

$$P_n = \begin{cases} \{e_1, \dots, e_{n-1}\} & \text{if } O = \emptyset, \\ \{e_1, \dots, e_o\} & \text{if } O = \{v_o\}, o \in \{1, \dots, n-1\}. \end{cases}$$

The following proposition can be derived.

Proposition 3.2.2.

$$\varphi(E) = \max\{\varphi(P_1), \varphi(P_n)\}. \quad (3.5)$$

Proof. Since graph $G = (V, E)$ is a cycle, any nonempty bounded rv -path $P_{rv}^E \subseteq E$, $v \in V$ of G is also an rv -path of either $G[P_1]$ or $G[P_n]$, the proposition thereby holds. \square

Theorem 3.2.3. *If G is a cycle and $c_r = 1$, the MBrT problem can be solved in linear time.*

Proof. If $c_r = 1$, as any bounded r -tree is an rv -path of G , one thus have

$$Opt_G = \max\{0, p_r, \varphi(E)\}.$$

Furthermore, according to Proposition 3.2.1, $\varphi(E)$ can be obtained in linear time. Hence, Opt_G can be obtained in linear time. \square

Consider now the case of $c_r = 2$.

Denote the path between v_1 and v_{n-1} without going through r by

$$P_r = E \setminus \{e_1, e_n\}.$$

Recall that we denote a uv -path of $G[P_r]$ by $P_{uv}^{P_r} \subseteq P_r$. Given a non-empty path P_{uv} between u and v , let

$$\zeta(P_{uv}) := f(G[P_{uv}]) - p_u - p_v,$$

and

$$\zeta_{Min}(P_r) := \min\{\zeta(P_{uv}^{P_r}) : |P_{uv}^{P_r}| \geq 1, |P_{uv}^{P_r} \cap \delta(O)| \geq |O|\}.$$

Notice that, for the case of $O = \emptyset$, one has

$$\zeta_{Min}(P_r) = \min\{\zeta(P_{uv}^{P_r}) : |P_{uv}^{P_r}| \geq 1\},$$

and for the case of $O \neq \emptyset$, one has

$$\zeta_{Min}(P_r) = \min\{\zeta(P_{uv}^{P_r}) : |P_{uv}^{P_r}| \geq 1, |P_{uv}^{P_r} \cap \delta(O)| \geq 1\}.$$

Additionally, it is possible that $V[P_{uv}^{P_r}] \setminus \{u, v\} = \emptyset$, but $P_{uv}^{P_r}$ cannot be empty.

The following lemma can be deduced based on the definition of $\zeta_{Min}(P_r)$.

Lemma 3.2.4.

$$f(G) - \zeta_{Min}(P_r) = \max\{f(T) : T = (U, F) \text{ is a bounded } r\text{-tree of } G, |\delta(r) \cap F| = 2\}.$$

Proof. From the definition of $\zeta_{Min}(P_r)$, it can be seen that $\zeta_{Min}(P_r) = \min\{f(G) - f(T) : T = (U, F) \text{ is a bounded } r\text{-tree of } G, |\delta(r) \cap F| = 2\}$. Since $f(G) = w(E) + p(V)$ is constant, the lemma is thereby proved. \square

Then the following result respecting the optimal value of the MBrT problem on G can be developed for the case of $c_r = 2$.

Proposition 3.2.5. *If $c_r = 2$,*

$$Opt_G = \max\{0, p_r, \varphi(E), f(G) - \zeta_{Min}(P_r)\}. \quad (3.6)$$

Proof. According to (3.4) and Lemma 3.2.4,

$$\begin{aligned} & \max\{\varphi(E), f(G) - \zeta_{Min}(P_r)\} \\ &= \max\{f(T) : T = (U, F) \text{ is a bounded } r\text{-tree of } G, 1 \leq |\delta(r) \cap F| \leq 2\}. \end{aligned}$$

Combining with the solutions (\emptyset, \emptyset) and $(\{r\}, \emptyset)$ and the fact that $|\delta(r) \cap F| \leq 2$ is satisfied for any subgraph (U, F) of G , one has

$$\max\{0, p_r, \varphi(E), f(G) - \zeta_{Min}(P_r)\} = \max\{f(T) : T \text{ is a bounded } r\text{-tree of } G\}.$$

Therefore the proposition holds. \square

The MBrT problem on G in the case of $c_r = 2$ is then reduced to the problem of finding $\varphi(E)$ and $\zeta_{Min}(P_r)$.

Recall that $\varphi(E)$ can be calculated in linear time, as it is shown in Propositions 3.2.1 and

3.2.2. We present hereafter the algorithms to obtain $\zeta_{Min}(P_r)$. Notice that, depending on whether O is empty or not, there are two situations to deal with.

The following proposition needs to be put forward as a preparation for the presentation of the algorithm for calculating $\zeta_{Min}(P_r)$ in the case of $O = \emptyset$.

Given a path $P_{s_1s_q} = \{s_1s_2, s_2s_3, \dots, s_{q-1}s_q\}$ between s_1 and s_q . Let $P_{s_i s_j}$ denote a path of $G[P_{s_1s_q}]$ between s_i and s_j with $1 \leq i < j \leq q$, as a simplification of $P_{s_i s_j}^{P_{s_1s_q}}$. Given $i \in \{1, \dots, q-1\}$, let $\zeta_{Min}^{s_i}(P_{s_i s_q})$ be the minimum value of $\zeta(P_{s_i s_j})$ with $i < j \leq q$, that is,

$$\zeta_{Min}^{s_i}(P_{s_i s_q}) = \min\{\zeta(P_{s_i s_j}) : i < j \leq q\}.$$

Proposition 3.2.6. *For any $i \in \{1, \dots, q-2\}$,*

$$\zeta_{Min}^{s_i}(P_{s_i s_q}) = \min\{w_{s_i s_{i+1}}, w_{s_i s_{i+1}} + p_{s_{i+1}} + \zeta_{Min}^{s_{i+1}}(P_{s_{i+1} s_q})\}.$$

Proof. It is straightforward to see

$$\zeta_{Min}^{s_i}(P_{s_i s_q}) \leq \min\{w_{s_i s_{i+1}}, w_{s_i s_{i+1}} + p_{s_{i+1}} + \zeta_{Min}^{s_{i+1}}(P_{s_{i+1} s_q})\},$$

as the right-hand side contains two feasible solutions of the left-hand side, that is, $w_{s_i s_{i+1}} = \zeta(\{s_i s_{i+1}\})$ and $w_{s_i s_{i+1}} + p_{s_{i+1}} + \zeta_{Min}^{s_{i+1}}(P_{s_{i+1} s_q}) = \zeta(P_{s_i s_j})$ for some j with $i+1 < j \leq q$.

On the other hand, suppose $\zeta_{Min}^{s_i}(P_{s_i s_q}) < \min\{w_{s_i s_{i+1}}, w_{s_i s_{i+1}} + p_{s_{i+1}} + \zeta_{Min}^{s_{i+1}}(P_{s_{i+1} s_q})\}$ with $i \leq q-1$. It is known that there exists some j with $i+1 < j \leq q$ and $\zeta(P_{s_i s_j}) = \zeta_{Min}^{s_i}(P_{s_i s_q})$. Then one has $\zeta(P_{s_i s_j}) = w_{s_i s_{i+1}} + p_{s_{i+1}} + \zeta(P_{s_{i+1} s_j}) < w_{s_i s_{i+1}} + p_{s_{i+1}} + \zeta_{Min}^{s_{i+1}}(P_{s_{i+1} s_q})$. It clearly contradicts the optimality of $\zeta_{Min}^{s_{i+1}}(P_{s_{i+1} s_q})$. Therefore,

$$\zeta_{Min}^{s_i}(P_{s_i s_q}) \geq \min\{w_{s_i s_{i+1}}, w_{s_i s_{i+1}} + p_{s_{i+1}} + \zeta_{Min}^{s_{i+1}}(P_{s_{i+1} s_q})\},$$

which completes the proof. □

We can also deduce the following proposition.

Proposition 3.2.7.

$$\zeta_{Min}(P_{s_1 s_q}) = \min\{\zeta_{Min}^{s_i}(P_{s_i s_q}) : i \in \{1, \dots, q-1\}\}. \quad (3.7)$$

Proof. From the definition of $\zeta_{Min}(P_{s_1 s_q})$ and $\zeta_{Min}^{s_i}(P_{s_i s_q})$, one has

$$\begin{aligned} & \zeta_{Min}(P_{s_1 s_q}) \\ &= \min\{\zeta(P_{s_i s_j}^{P_{s_1 s_q}}) : 1 \leq i < j \leq q\} \\ &= \min\{\zeta(P_{s_i s_j}^{P_{s_1 s_q}}) : i \in \{1, \dots, q-1\}, i < j \leq q\}, \end{aligned}$$

$$= \min\{\zeta_{Min}^{s_i}(P_{s_i s_q}) : i \in \{1, \dots, q-1\}\}.$$

Therefore, the lemma holds. \square

The algorithm for calculating $\zeta_{Min}(P_r)$ is based on Proposition 3.2.6 and Proposition 3.2.7. The algorithm uses a recursive process to calculate $\zeta_{Min}(P_{v_i v_{n-1}})$, for all $i \in \{1, \dots, n-2\}$, as demonstrated in Algorithm 3.2.

Algorithm 3.2: Algorithm to obtain $\zeta_{Min}(P_r)$ on cycles

Input : Cycle $G = (V, E)$, $\mathbf{w} \in \mathbb{R}^E$ and $\mathbf{p} \in \mathbb{R}^V$

Output: $\zeta_{Min}(P_r)$.

begin

```

1  Set  $\zeta_{Min}^{v_{n-2}}(P_{v_{n-2}v_{n-1}}^{P_r}) = \zeta(\{e_{n-1}\}) = w_{e_{n-1}}$ .
2  Set  $i = n - 3$ .
   while  $i \geq 1$  do
3     Set  $\zeta_{Min}^{v_i}(P_{v_i v_{n-1}}^{P_r}) = \min\{w_{e_{i+1}}, w_{e_{i+1}} + p_{v_{i+1}} + \zeta_{Min}^{i+1}(P_{v_{i+1}v_{n-1}}^{P_r})\}$ .
4     Set  $i = i - 1$ 
5   $\zeta_{Min}(P_r) = \min\{\zeta_{Min}^{v_i}(P_{v_i v_{n-1}}^{P_r}) : i \in \{1, \dots, n-2\}\}$ .
```

Proposition 3.2.8. *Algorithm 3.2 obtains the value of $\zeta_{Min}(P_r)$.*

Proof. Line 1 of the algorithm obtains

$$\zeta_{Min}^{v_{n-2}}(P_{v_{n-2}v_{n-1}}^{P_r}) = \zeta(\{e_{n-1}\}) = w_{e_{n-1}},$$

as the only nonempty path between v_{n-2} and v_{n-1} in $G[P_r]$ is $\{e_{n-1}\}$.

Line 3 calculates $\zeta_{Min}^{v_i}(P_{v_i v_{n-1}}^{P_r})$ for each $i \in \{1, \dots, n-3\}$, according to Proposition 3.2.6.

Line 5 calculates $\zeta_{Min}(P_r)$ according to Proposition 3.2.7. \square

Proposition 3.2.9. *Algorithm 3.2 has time complexity of $\mathcal{O}(n)$.*

Proof. In this algorithm, each term of w_e , p_v for any $e \in P_r$ and $v \in V \setminus \{r, v_1, v_{n-1}\}$ occurs only once. Therefore the time complexity is $\mathcal{O}(n)$. \square

Therefore, we now have the following result.

Theorem 3.2.10. *If G is a cycle with $c_r = 2$ and $O = \emptyset$, the MBrT problem can be solved in linear time.*

Proof. As a result of Propositions 3.2.1, 3.2.2, 3.2.8 and 3.2.9, both $\varphi(E)$ and $\zeta_{Min}(P_r)$ can be obtained in linear time. Therefore, according to Proposition 3.2.5, Opt_G can be obtained in linear time. \square

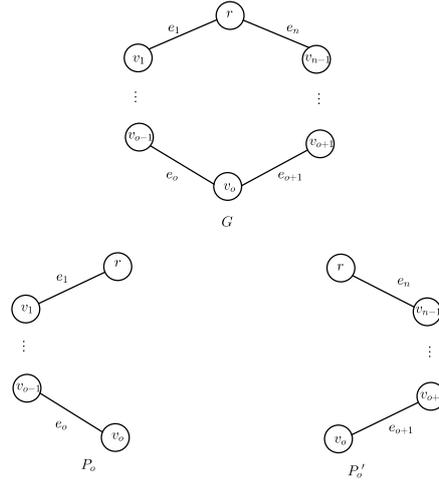


Figure 3.1: A cycle G with v_o and its subpaths P_o , P'_o

Consider the case where $c_r = 2$ and there exists a node $v_o \in V \setminus \{r\}$ with $c_{v_o} = 1$, $o \in \{1, \dots, n-1\}$. Denote $P_o = P_{rv_o}^{P_r}$ the path between r and v_o passing through e_1 , and $P'_o = P_{rv_o}^{P'_r}$ the path between r and v_o passing through e_n , as demonstrated in Figure 3.1.

In order to calculate $\zeta_{Min}(P_r)$, Algorithm 3.2 in this case should be changed to finding a minimum-weight subpath $P_{uv}^{P_r}$ of P_r which satisfies $v_o \in V[P_{uv}^{P_r}]$. Alternatively, we show this approach can be replaced by another algorithm described below.

Instead of finding a minimum-weight subpath of P_r , one can focus on finding two subpaths $P_{v_o u}^{P_o}$, $P_{v_o v}^{P'_o}$ of P_o and P'_o respectively, such that they have minimal values of $\zeta(P_{v_o u}^{P_o})$ and $\zeta(P_{v_o v}^{P'_o})$, for $u \in V[P_o] \setminus \{v_o\}$, $v \in V[P'_o] \setminus \{v_o\}$. Let

$$\zeta_o = \min\{\zeta(P_{v_o u}^{P_o}) : u \in V[P_o] \setminus \{v_o\}\},$$

and

$$\zeta'_o = \min\{\zeta(P_{v_o v}^{P'_o}) : v \in V[P'_o] \setminus \{v_o\}\}.$$

We first show that these two values lead to the optimal value of the MBrT problem.

Proposition 3.2.11. *If $c_r = 2$ and $O = \{v_o\}$, then*

$$Opt_G = \max\{0, p_r, f(G) - \zeta_o, f(G) - \zeta'_o, f(G) - \zeta_o - \zeta'_o - p_{v_o}\}.$$

Proof. Let u be a node in $V[P_o] \setminus \{v_o\}$ such that $\zeta(P_{v_o u}^{P_o}) = \zeta_o$, and let v be a node in $V[P'_o] \setminus \{v_o\}$ such that $\zeta(P_{v_o v}^{P'_o}) = \zeta'_o$.

Since $P_{rv_o}^{P'_o} \neq \emptyset$ and $P_{ru}^{P_o} \neq \emptyset$, one has

$$f((\emptyset, \emptyset)) = 0,$$

$$\begin{aligned}
f(\{\{r\}, \emptyset\}) &= p_r, \\
f(G[P_{rv_o}^{P'_o} \cup P_{ru}^{P_o}]) &= f(G) - \zeta_o, \\
f(G[P_{rv_o}^{P_o} \cup P_{rv_o}^{P'_o}]) &= f(G) - \zeta'_o, \\
f(G[P_{rv}^{P'_o} \cup P_{ru}^{P_o}]) &= f(G) - \zeta_o - \zeta'_o - p_{v_o}.
\end{aligned}$$

Hence

$$Opt_G \geq \max\{0, p_r, f(G) - \zeta_o, f(G) - \zeta'_o, f(G) - \zeta_o - \zeta'_o - p_{v_o}\},$$

because the right-hand side contains the values of feasible solutions to the MBrT problem on G .

Assume that there exists a path $P_{ss'}$ with $r \in V[P_{ss'}]$, $s \in V[P_o]$ and $s' \in V[P'_o]$, such that $f(G[P_{ss'}]) > \max\{0, f(G) - \zeta_o, f(G) - \zeta'_o, f(G) - \zeta_o - \zeta'_o - p_{v_o}\}$. It can be seen that $P_{ss'} \neq \emptyset$.

If $v_o \in V[P_{ss'}]$, without loss of generality, assume $s = v_o$. One must have $f(G[P_{ss'}]) > f(G) - \zeta'_o$ which indicates $\zeta(P_{v_o r}^{P'_o}) < \zeta'_o$ contradicts the optimality of ζ'_o .

Otherwise assume $v_o \notin V[P_{ss'}]$, then $P_{sv_o}^{P_o} \neq \emptyset$ and $P_{s'v_o}^{P'_o} \neq \emptyset$. From the assumption, one can deduce that $\zeta(P_{sv_o}^{P_o}) + \zeta(P_{s'v_o}^{P'_o}) < \zeta_o + \zeta'_o$. Thus at least one of the two inequalities $\zeta(P_{sv_o}^{P_o}) < \zeta_o$ and $\zeta(P_{s'v_o}^{P'_o}) < \zeta'_o$ holds, which forms a contradiction with the optimality of either ζ_o or ζ'_o .

Therefore, such path $P_{ss'}$ does not exist. Thus

$$Opt_G \leq \max\{0, p_r, f(G) - \zeta_o, f(G) - \zeta'_o, f(G) - \zeta_o - \zeta'_o - p_{v_o}\},$$

which concludes the proof. \square

As a result of this proposition, the MBrT problem in this situation can be reduced to the problem of finding ζ_o and ζ'_o . Similar to Proposition 3.2.1, ζ_o and ζ'_o can also be obtained in linear time, except that now v_o is the fixed end of the paths instead of r , and it looks for minimum-weight paths instead of maximum-weight paths.

Theorem 3.2.12. *If G is cycle with $c_r = 2$ and $O = \{v_o\}$, the MBrT problem can be solved in linear time.*

Proof. For ζ_o (ζ'_o , respectively), there are exactly o ($n - o$, respectively) nonempty paths of $G[P_o]$ ($G[P'_o]$, respectively) that contain o . Hence, ζ_o (ζ'_o , respectively) can be obtained in linear time, which combining with Proposition 3.2.11 completes the proof. \square

Each case on cycles is hereby given a linear-time algorithm to solve the MBrT problem accordingly.

3.3 On cactus graphs

Consider now the MBrT problem on a cactus graph $G = (V, E)$. A cactus graph can be seen as a combination of trees and cycles while having a tree hierarchy among the components of trees and cycles. Provided that the MBrT problem is polynomially solvable on both trees and cycles, it is convincing to argue that the MBrT problem on cactus graphs is also polynomially solvable.

This section provides an effective approach to decompose a cactus graph, based on which an according polynomial-time algorithm is proposed to solve the MBrT problem on it.

The following results support the decomposition of cactus graphs in later discussion.

Given a graph $G = (V, E)$ with $r \in V$ and a capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$, let $Opt_G^i(r)$ denote the maximum weight of a bounded r -tree $T = (U, F)$ of G such that $|\delta(r) \cap F| = i$, $i \leq c_r$. Opt_G^i is also used for the sake of simplification if the root node is clear from context.

Proposition 3.3.1.

$$Opt_G^1 - p_r \geq Opt_G^2 - Opt_G^1. \quad (3.8)$$

Proof. Suppose $Opt_G^1 - p_r < Opt_G^2 - Opt_G^1$. Let $T_j = (U_j, F_j)$ be a bounded r -tree of G that satisfies $|F_j \cap \delta(r)| = j$ and $f(T_j) = Opt_G^j$, $j = 1, 2$. Notice that T_2 can be seen as a combination of two r -trees having exactly one edge in $\delta(r)$. Denote these two r -trees T_2^1 and T_2^2 .

Since from the assumption, $f(T_1) - p_r < (f(T_2^1) + f(T_2^2) - p_r) - f(T_1)$ holds, one has $2f(T_1) < f(T_2^1) + f(T_2^2)$. Thus at least one of $f(T_2^1)$ and $f(T_2^2)$ is larger than $f(T_1)$, which forms a contradiction with the optimality of T_1 . \square

The following proposition can then be derived based on (3.8), which is necessary to complete the decomposition on cactus graphs.

Given a graph $G = (V, E)$ with an articulation node v_a , let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two subgraphs separated by v_a . Without loss of generality, let $r \in V_1$.

Proposition 3.3.2. *If $|\delta_{G_2}(v_a)| = 2$, let G' be the graph composed of G_1 and two edges $e'_1 = v_a v'_1, e'_2 = v_a v'_2$ taking the place of G_2 such that $p_{v'_1} = p_{v'_2} = 0$, $w_{e'_2} = Opt_{G_2}^2 - Opt_{G_2}^1$ and $w_{e'_1} = Opt_{G_2}^1 - p_{v_a}$. Then*

$$Opt_G = Opt_{G'}.$$

Proof. Firstly, any bounded r -tree of G that contains no edges in G_2 can also be seen as a bounded r -tree of G' with no edges in $\{e'_1, e'_2\}$, and vice versa.

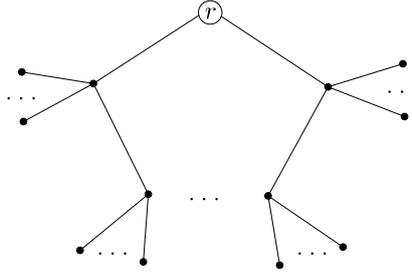


Figure 3.2: A cactus basis

For any bounded r -tree T_i of G with i edges in $\delta_{G_2}(v_a)$, $i \in \{1, 2\}$, let T'_i be the r -tree obtained by replacing the part of T_i in G_2 with e'_1 if $i = 1$, or with e'_1 and e'_2 if $i = 2$. One has $f(T'_i) \geq f(T_i)$ as a result of the optimality of $Opt_{G_2}^1$ and $Opt_{G_2}^2$. Hence $Opt_G \leq Opt_{G'}$.

Conversely, according to Proposition 3.3.1, $w_{e'_1} \geq w_{e'_2}$, thus if there exists a maximum bounded r -tree in G' that contains e'_2 , there must also exist a maximum bounded r -tree containing e'_1 . Additionally, any maximum bounded r -tree in G' containing only e'_1 or both e'_1 and e'_2 can be transformed into a bounded r -tree of G . Thus $Opt_G \geq Opt_{G'}$, and therefore completes the proof. \square

If $Opt_{G_2}^1$ and $Opt_{G_2}^2$ and the associated bounded r -trees can be obtained in polynomial time, Proposition 3.3.2 allows us to transform such subgraph G_2 into edges and hence simplify the structure of the graph.

Similarly, it is trivial to see that if $|\delta_{G_2}(v_a)| = 1$, one can transform G_2 into just one edge. The proof is immediate from the proof of Proposition 3.3.2, as one can add a virtual edge with weight $-M$, where M is a sufficiently large number. In this case $Opt_{G_2}^2 = Opt_{G_2}^1$ always holds, and the associated bounded r -trees are identical. This result is stated in the following corollary.

Corollary 3.3.3. *If $|\delta_{G_2}(v_a)| = 1$, let G' be the graph composed of G_1 and an edge $e'_1 = v_a v'_1$ taking the place of G_2 such that $p_{v'_1} = 0$ and $w_{e'_1} = Opt_{G_2}^1 - p_{v_a}$. Then*

$$Opt_G = Opt_{G'}. \quad \square$$

Based on Proposition 3.3.2 and Corollary 3.3.3 a *cactus basis* as shown in Figure 3.2 can be developed for solving the MBrT problem on cactus graphs. It is composed of a cycle where each node except the root node is incident with some pending edges. As it contains only one cycle, Corollary 3.1.5 implies that the MBrT problem on cactus bases can be solved in polynomial time. On these grounds, the MBrT problem on cactus graphs can be proved to be polynomially solvable, by reducing it to a number of subproblems on graphs with the forms of tree and cactus basis.

Denote by $H \subseteq V$, the set of *hinges*, which is the set composed of nodes that have degree at least 3 and belong to at least one cycle and r if r belongs to a cycle. A cactus graph then can be seen as a combination of cycle and tree components linked by hinges. As cactus graphs have the tree hierarchy, the notion of up-closure and down-closure can be extended to cactus graphs for the hinges. For any two distinct nodes $u, v \in H$, define $u \leq v$ if u belongs to all the rv -paths in G . The up-closure $\lceil v \rceil$ and down-closure $\lfloor v \rfloor$ of v are thereby defined as $\lceil v \rceil = \{u \in H : v \leq u\}$ and $\lfloor v \rfloor = \{u \in H : u \leq v\}$ respectively.

For each cycle $C = (U_C, F_C)$ of G , we say that it is *attached at* a hinge $h \in H$ if $h \in U_C$ and removing $\delta(h)$ disconnects all nodes in $U_C \setminus \{h\}$ from r . Denote the graph $G_C = (V_C, E_C)$ as the graph such that $F_C \subseteq E_C$, and removing F_C disconnects all nodes in $V_C \setminus U_C$ and all edges in $E_C \setminus F_C$ from r .

In addition, for some hinge $h' \in H$ such that $\lfloor h' \rfloor = \{h'\}$, and some edge $v_b h' \in \delta(h')$, such that removing $v_b h'$ disconnects some edges from r . The graph $G_{v_b} = G[\lfloor v_b \rfloor]$ with

$$V_{v_b} = \{v \in V : \text{removing } v_b h' \text{ disconnects } h' \text{ and } v\} \cup \{h'\}$$

is called a *tree branch* attached at h' associated with v_b and $v_b h'$.

Proposition 3.3.4. *The MBrT problem on a cactus graph is polynomially solvable.*

Proof. For each cycle C of G attached at hinge $h \in H$, assume that there is no other cycle components in G_C , otherwise one has to deal with them beforehand. In other words, the cycle component is only connected with a few tree branches. As the MBrT problem on trees is polynomially solvable, we can transform each tree branch G_{v_b} attached at some hinge $h' \in \lfloor h \rfloor \setminus \{h\}$ into an edge according to Corollary 3.3.3. More precisely, for each tree branch G_{v_b} attached at some hinge $h' \in \lfloor h \rfloor \setminus \{h\}$ associated with v_b , one can replace G_{v_b} by an edge $h' v'_b$ with $w_{h' v'_b} = \text{Opt}_{G_{v_b}}^1(h') - p_{h'}$ and $p_{v'_b} = 0$. Then the graph G_C is transformed into a cactus basis, and we assume that G_C is a cactus basis hereafter. Notice that if G_{v_b} contains only one edge, the transformation, although harmless, is unnecessary.

According to Corollary 3.1.5, one can obtain $\text{Opt}_{G_C}^1(h)$ and $\text{Opt}_{G_C}^2(h)$ using the following approach.

Let $\delta(h) = \{e_1, e_2\}$, and let M be a sufficiently large number. Denote by $G_e = G_C \setminus \{e\}$ the subgraph of G_C without e .

Using Algorithm 3.1, one can solve the MBrT problem on two subgraphs G_{e_1} and G_{e_2} with $c_h = 1$, and setting $w'_{e_2} = M$ in G_{e_1} and $w'_{e_1} = M$ in G_{e_2} respectively. It can be seen that

$$\text{Opt}^1(G_C)(h) = \max\{\text{Opt}_{G_{e_1}}(h) - M + w_{e_2}, \text{Opt}_{G_{e_2}}(h) - M + w_{e_1}\}.$$

This calculation can be done in the time of $\mathcal{O}(|V(G_C)|)$.

$Opt_{G_C}^2(h)$ can be obtained as follows. First, set h as the root node with $c_h = 2$, $w'_{e_1} = M$ and $w'_{e_2} = M$, and then solve the MBrT problem on G_e , for each $e \in E(C) \setminus \{e_1, e_2\}$. Then we have

$$Opt_{G_C}^2(h) = \max\{Opt_{G_e}(h) : e \in E(C) \setminus \{e_1, e_2\}\} - 2M + w_{e_1} + w_{e_2}.$$

This calculation can be done in the time of $\mathcal{O}(|C||V(G_C)|)$.

According to Proposition 3.3.2, G_C can then be replaced by two edges e'_1, e'_2 with the weights $w_{e'_1} = Opt_{G_C}^1(h) - p_h$ and $w_{e'_2} = Opt_{G_C}^2(h) - Opt_{G_C}^1(h)$.

This operation can be repeated until there is no cycles left in G . Denote the new graph by G' . According to Proposition 3.3.2, we have $Opt_G = Opt_{G'}$. Furthermore, as G' is tree, the MBrT problem on it can be solved in polynomial time according to Theorem 3.1.4. Therefore, one can solve the MBrT problem by solving a series of subproblems on cactus bases and trees. Algorithm 3.3 demonstrates the algorithm without providing the details of algorithms on the subgraphs. \square

Algorithm 3.3: Algorithm for the MBrT problem on cactus

Input : Cactus $G = (V, E)$, $\mathbf{w} \in \mathbb{R}^E$ and $\mathbf{p} \in \mathbb{R}^V$

Output: Solution to the MBrT problem on G .

begin

while *there exists some cycle C attached at a hinge h such that G_C only contains one cycle* **do**

if G_C *is already transformed into a cactus basis* **then**

1 | Transform G_C into two edges according to Proposition 3.3.2.

else if G_C *contains some tree branch G_{v_b} attached at a hinge h'* **then**

2 | Transform G_C into an edge according to Corollary 3.3.3.

3 Solve the MBrT problem on the transformed graph using Algorithm 3.1.

Proposition 3.3.5. *Algorithm 3.3 has time complexity of $\mathcal{O}(n^2)$.*

Proof. The algorithm on cactus graphs uses both the algorithm for trees and the algorithm for cactus bases. Each cycle component requires the calculation of $Opt_{G_C}^1(h)$ and $Opt_{G_C}^2(h)$ based on Algorithm 3.1, which requires $\mathcal{O}(|V(C)||V(G_C)|)$, and Algorithm 3.1 also needs to be run once on each tree branch encountered in the process. Therefore the algorithm has time complexity of $\mathcal{O}(n^2)$. \square

3.4 General graph decomposition

Besides the discussion on trees, cycles and cactus graphs, a general decomposition is proposed in this section regarding the articulation nodes.

Given a graph $G = (V, E)$ with an articulation node $v_a \in V$, let $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ be two subgraphs of G separated by v_a such that $E = E_1 \cup E_2, V = V_1 \cup V_2$. Without loss of generality, assume $r \in V_1$.

Proposition 3.4.1. *If the MBrT problem on G_1 and G_2 can be solved in polynomial time, the MBrT problem on G is also polynomially solvable.*

Proof. First it is worth noting that finding a maximum nonempty bounded r -tree of a graph can be reduced to finding a maximum bounded r -tree of a graph by setting p_r to a sufficiently large number. Hence, a maximum nonempty bounded r -tree can be found in polynomial time on both G_1 and G_2 .

There are 2 cases to be considered depending on whether $r = v_a$ holds. Notice that here we assume that $c_v \leq |\delta(v)|$ is always satisfied, otherwise one can manually set $c_v = |\delta(v)|$.

Case 1). $r = v_a$.

Let $Opt_{G_k, i}$ denote the the weight of a maximum nonempty bounded r -tree of G_j with setting $c_r = i$, for $i \in \{0, 1, \dots, c_r\}, k \in \{1, 2\}$.

For a given $i, j \in \{0, 1, \dots, c_r\}$, it is straightforward that $Opt_{G_k, i} \leq Opt_{G_k, j}$ holds if $i \leq j$, for $k \in \{1, 2\}$. As a result, for a given $i \in \{0, 1, \dots, c_r\}$, we have

$$Opt_{G_2, c_r - i} = \max\{Opt_{G_2, j} : j \in \{0, \dots, c_r - i\}\}.$$

Then the maximum nonempty bounded r -tree of G can be chosen among $c_r + 1$ combinations, which is

$$Opt'_G = \max\{Opt_{G_1, i} + Opt_{G_2, c_r - i} - p_r : i \in \{0, 1, \dots, c_r\}\}.$$

We then have the optimal value of the MBrT problem on G as

$$Opt_G = \max\{0, Opt'_G\},$$

where 0 is the weight of the empty graph.

Case 2). $r \neq v_a$.

Similarly to the previous case, let $Opt_{G_2, i}$ be the maximum weight of a nonempty bounded r -tree of G_2 with v_a being the root and $c_{v_a} = i$, for $i \in \{0, 1, \dots, c_{v_a} - 1\}$. We have

$Opt_{G_2,i} \leq Opt_{G_2,j}$, if $i \leq j$ for $i, j \in \{0, 1, \dots, c_{v_a} - 1\}$.

Let $Opt_{G_1,i,j}$ be the optimal value for the MBrT problem on G_1 with the capacity of v_a being $c_{v_a} = i$, and replacing the weight p_{v_a} of v_a by $p_{v'_a} = Opt_{G_2,j}$, for $i \in \{1, \dots, c_{v_a}\}$, $j \in \{0, \dots, c_{v_a} - 1\}$. It can be seen that $Opt_{G_1,i,j} \leq Opt_{G_1,i,k}$ if $j \leq k$, for $i \in \{1, \dots, c_{v_a}\}$, $j, k \in \{0, \dots, c_{v_a} - 1\}$.

Thus for a given $i \in \{1, \dots, c_{v_a}\}$, we have

$$Opt_{G_1,i,c_{v_a}-i} = \max\{Opt_{G_1,i,j} : j \in \{0, \dots, c_{v_a} - i\}\}.$$

It can be seen that $Opt_{G_1,i,c_{v_a}-i}$ corresponds to a bounded r -tree of G that contains at most i edges in $\delta_{G_1}(v_a)$. Hence we have

$$Opt_{G_1,i,c_{v_a}-i} \leq \max\{f(T) : T \text{ is a bounded } r\text{-tree of } G, |\delta_{G_1}(v_a) \cap E(T)| \leq i\}.$$

Assume that there exists a bounded r -tree T' of G that contains at most i edges in $\delta_{G_1}(v_a)$ such that $f(T') > Opt_{G_1,i,c_{v_a}-i}$. It can be deduced that $v_a \in V(T')$, as otherwise T' is also a bounded r -tree of G_1 that does not contain v_a , which indicates $f(T') \leq Opt_{G_1,i,c_{v_a}-i}$. Thus T' is composed of a bounded r -tree T'_1 of G_1 and a nonempty bounded tree T'_2 of G_2 such that it contains at most $c_{v_a} - i$ edges in $\delta_{G_2}(v_a)$. As $f(T'_2) \leq Opt_{G_2,c_{v_a}-i}$ holds, $f(T') \leq Opt_{G_1,i,c_{v_a}-i}$ also holds, which forms a contradiction. As a result, we have

$$Opt_{G_1,i,c_{v_a}-i} \geq \max\{f(T) : T \text{ is a bounded } r\text{-tree of } G, |\delta_{G_1}(v_a) \cap E(T)| \leq i\}.$$

Therefore, one can find the maximum weight of a bounded r -tree of G by

$$Opt_G = \max\{Opt_{G_1,i,c_{v_a}-i} : i \in \{1, \dots, c_{v_a}\}\}.$$

To summarize, the MBrT problem on G can be solved in polynomial time in both aforementioned cases. \square

3.5 Conclusion

In this chapter, we provided algorithms for the MBrT problem on trees, cycles and cactus graphs. We also proved that if the MBrT problem is polynomially solvable on two graphs it is also polynomially solvable on the 1-sum of the two graphs.

The algorithmic study provides a brief glimpse of the polyhedral structure of the problem. For instance, the decomposition for cactus graphs also reflects the possible decomposition for the polytope. Furthermore, it can also be seen how the capacity creates complexity

for us to solve the problem in different scenarios.

Moreover, since the MBrT problem is polynomially solvable on trees, cycles, and cactus graphs, it is worth studying the polyhedra on these classes of graphs, and more importantly, trying to obtain the characterization thereof.

Chapter 4

Polyhedral study on Extended Bounded r -Tree Polytope

Recall that given a graph $G = (V, E)$, a root node $r \in V$, a capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$, an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$, and a node-price vector $\mathbf{p} \in \mathbb{R}^V$, the MBrT problem consists of finding an r -tree T of G with maximum value $f(T) = w(E(T)) + p(V(T))$ and such that the degree of any node $v \in V(T)$ is bounded by c_v .

In this chapter, we focus on the extended bounded r -tree polytope $B_{xy}(G, r, \mathbf{c})$, which is the integral hull of the formulation introduced in Chapter 2, that is,

$$x(E) - y(V \setminus \{r\}) = 0, \tag{4.1}$$

$$x(E[S]) - y(S \setminus \{v\}) \leq 0 \quad \forall v \in S \subseteq V, |S| \geq 2, \tag{4.2}$$

$$x(\delta(v)) - c_v y_v \leq 0 \quad \forall v \in V, \tag{4.3}$$

$$y_r \leq 1, \tag{4.4}$$

$$x_e \geq 0 \quad \forall e \in E. \tag{4.5}$$

According to Assumption 2.2.9, G does not have any unreachable elements. We first show that its dimension is related to the blocks and unit-capacity nodes in the graph. Several sets of valid equations and inequalities are introduced during the dimension study. After that, we prove that the polytope can be decomposed with respect to articulation nodes through 1-sum. Necessary and sufficient conditions for each set of valid inequalities to be facet-defining are examined. Finally, we give the complete description of $B_{xy}(G, r, \mathbf{c})$ when the graph is a tree or a cycle, and as a result of the decomposition result, we can also characterize $B_{xy}(G, r, \mathbf{c})$ on cactus graphs.

4.1 Dimension

Before stating the dimension of $B_{xy}(G, r, \mathbf{c})$, a few lemmas related to implicit equations should be introduced in advance.

For any $v \in S \subseteq V \setminus \{r\}$, according to (4.1) and (4.2) one has

$$\begin{aligned} -x(E) + y(V \setminus \{r\}) &= 0, \\ x(E[S \setminus (O \setminus \{v\})]) - y(S \setminus (O \cup \{v\})) &\leq 0, \\ x(E[\bar{S} \setminus O]) - y(\bar{S} \setminus (O \cup \{r\})) &\leq 0. \end{aligned}$$

From (4.7) we have

$$x(\delta(v_o)) - y_{v_o} = 0 \quad \forall v_o \in O \setminus \{v\},$$

which leads to

$$x(\delta(O \setminus \{v\})) - y(O \setminus \{v\}) = 0.$$

Summing them up gives us

$$y_v - x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0. \quad (4.6)$$

Inequalities (4.6) are valid but mostly redundant, nonetheless they are used to simplify the presentation of the proof of the following lemma, and will also be useful throughout the dissertation for both the theoretical and computational developments. Note that if $|\bar{S}| = 1$ or $|S| = 1$, (4.6) defines the same face as $x(E[S]) - y(S \setminus \{v\}) \leq 0$ in the former case and $x(E[\bar{S}]) - y(\bar{S} \setminus \{r\}) \leq 0$ in the latter.

For the unit-capacity nodes but r (i.e., the nodes in O), we can deduce some equations as stated in the following lemma.

Lemma 4.1.1. *Given any node $v_o \in O$,*

$$x(\delta(v_o)) - y_{v_o} = 0 \quad (4.7)$$

is valid for $B_{xy}(G, r, \mathbf{c})$.

Proof. Since $c_{v_o} = 1$, $x(\delta(v_o)) - y_{v_o} = 0$ can be immediately obtained from (4.3) and (4.6) associated with $S = \{v_o\}$. \square

This lemma gives a brief glance at how the unit-capacity nodes affect the polytope in terms of dimension.

A *block* of a graph is defined as a maximal connected subgraph with no articulation node.

Recall that G_r is connected according to Assumption 2.2.9, and as a nonempty connected graph, G_r can be uniquely decomposed into the 1-sum union of its blocks. Note that the blocks of G_r are maximal 2-connected subgraphs or bridges.

Let $(V_1, E_1), \dots, (V_q, E_q)$, $q \geq 1$, represent the blocks of graph G_r . For any $i \in \{1, \dots, q\}$, let v_{a_i} denote the node in block (V_i, E_i) that separates all the nodes in $V_i \setminus \{v_{a_i}\}$ from r . In other words, for any node $v_i \in V_i$, $i \in \{1, \dots, q\}$, a path between r and v_i has to contain v_{a_i} . For the sake of homogeneity, we may represent r by v_{a_i} for any block (V_i, E_i) that contains r . Moreover, as nodes in O can only be leaves in any bounded r -tree of G , any bounded r -tree T of G also induces a bounded r -tree $T \setminus O$ of G_r , and this motivates our focus on the blocks of G_r rather than those of G .

We then can derive the following results with respect to the blocks and the articulation nodes of G_r .

Lemma 4.1.2. *Given a graph G_r and one of its blocks (V_i, E_i) , any rooted tree $T = (U, F)$ of G_r contains $(U \cap V_i, F \cap E_i)$ as a tree rooted at v_{a_i} .*

Proof. Direct consequence of the definition of the blocks and the articulation nodes v_{a_i} of G_r . \square

Notice that Lemma 4.1.2 applies to any graph as long as the definitions of blocks and articulation nodes are preserved.

An immediate result of (4.1) and Lemma 4.1.2 is the following corollary.

Corollary 4.1.3. *For any block (V_i, E_i) , $i \in \{1, \dots, q\}$,*

$$x(E_i) - y(V_i \setminus \{v_{a_i}\}) = 0 \quad (4.8)$$

is valid for $B_{xy}(G, r, \mathbf{c})$. \square

Since $G = (V_r \cup O, E_r \cup \delta(O))$, we can immediately derive the following linear dependency.

Corollary 4.1.4. *Equation (4.1) can be obtained as the sum of all equations (4.7) and (4.8).* \square

It is worth mentioning that for any node v in $V \setminus \{r\}$, subtracting inequality (4.2) associated with V and v from (4.1) gives the following valid inequality

$$y_v - y_r \leq 0. \quad (4.9)$$

Similarly, given any block (V_i, E_i) and any node $v \in V_i \setminus \{v_{a_i}\}$, for $i \in \{1, \dots, q\}$, from (4.8) and inequality (4.2) associated with V_i and v , the inequality

$$y_v - y_{v_{a_i}} \leq 0 \quad (4.10)$$

holds as well.

According to Lemma 4.1.1 and Corollaries 4.1.3 and 4.1.4, the dimension of $B_{xy}(G, r, \mathbf{c})$ depends on both the number of nodes in O and the number of blocks in G_r . An upper bound on the dimension of the polytope $B_{xy}(G, r, \mathbf{c})$ is given in the following proposition.

Proposition 4.1.5. $\dim B_{xy}(G, r, \mathbf{c}) \leq |E| + |V_r| - q.$

Proof. Let $S^=$ be the system composed of the equations (4.7) and (4.8). The support graphs of these $|O| + q$ equations are pairwise edge-disjoint. Since the edge sets of these support graphs are nonempty, $S^=$ then has full row-rank, that is,

$$\text{rank } S^= = |O| + q.$$

Consequently, one obtains

$$\begin{aligned} \dim B_{xy}(G, r, \mathbf{c}) &\leq |E| + |V| - |O| - q \\ &= |E| + |V_r| - q, \end{aligned}$$

where the last line comes from $\{V_r, O\}$ being a partition of V . □

The following technical lemma that was proved by Goemans [1994], and can be viewed as a specific case of Menger's Theorem (Menger [1927]), is necessary for providing a lower bound of the dimension.

Lemma 4.1.6. *Given a 2-connected graph $G = (V, E)$, for every distinct $u, v, w \in V$ there exists a path in G from u to w that does not go through v .* □

We now prove that all valid equations whose support graphs are subgraphs of G_r can be written as linear combinations of inequalities (4.8).

Proposition 4.1.7. *Then any valid equation for $B_{xy}(G, r, \mathbf{c})$ whose support graph is a subgraph of G_r is a linear combination of inequalities (4.8).*

Proof. Assume that $\alpha^T \mathbf{x} + \beta^T \mathbf{y} = \gamma$ is satisfied by all $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \in B_{xy}(G, r, \mathbf{c})$, where $\alpha_e = 0$ for any $e \notin E_r$ and $\beta_v = 0$ for any $v \notin V_r$. Since $\mathbf{0} \in B_{xy}(G, r, \mathbf{c})$, one clearly has $\gamma = 0$.

Besides, as the incidence vector of $(\{r\}, \emptyset)$ is also in $B_{xy}(G, r, \mathbf{c})$, one has

$$\beta_r = 0.$$

Consider any block (V_i, E_i) of G_r , $i \in \{1, \dots, q\}$, that contains r . For any edge $rv \in E_i$, the incidence vector of $G[\{rv\}] = (\{r, v\}, \{rv\})$ is in $B_{xy}(G, r, \mathbf{c})$, and thus

$$\alpha_{rv} + \beta_v = 0.$$

Now consider any edge $uv \in E_i \setminus \delta(r)$. Such edge exists only if this block is a 2-connected component of G_r . Recall that $O \cap V_i = \emptyset$, and hence any rv -path of $G[E_i]$ satisfies the capacity constraints. Let P_{ru} be a path of $G[E_i]$ between r and u that does not pass through v . Both P_{ru} and $P_{ru} \cup \{uv\}$ induce bounded r -trees of G . We hence deduce

$$\alpha_{uv} + \beta_v = 0.$$

Therefore from the connectivity of $G[E_i]$ we obtain

$$\begin{aligned} \alpha_e &= \lambda_i & \forall e \in E_i, \\ \beta_v &= -\lambda_i & \forall v \in V_i \setminus \{r\}, \end{aligned}$$

where λ_i is a real scalar.

Consider now any block (V_j, E_j) of G_r , $j \in \{1, \dots, q\}$, such that $r \notin V_j$. Let $P_{rv_{a_j}}$ be any path between r and v_{a_j} . For any edge $v_{a_j}v \in E_j$, both $P_{rv_{a_j}}$ and $P_{rv_{a_j}} \cup \{v_{a_j}v\}$ induce bounded r -trees of G , thus

$$\alpha_{v_{a_j}v} + \beta_v = 0.$$

Furthermore if $G[E_j]$ is a 2-connected component of G_r , consider any edge $uv \in E_j \setminus \delta(v_{a_j})$. Let $P_{v_{a_j}u}$ be a path between v_{a_j} and u that does not go through v . Both $P_{rv_{a_j}} \cup P_{v_{a_j}u}$ and $P_{rv_{a_j}} \cup P_{v_{a_j}u} \cup \{uv\}$ induce bounded r -trees of G , which leads to

$$\alpha_{uv} + \beta_v = 0.$$

Therefore from the connectivity of $G[E_j]$ we obtain

$$\begin{aligned} \alpha_e &= \lambda_j & \forall e \in E_j, \\ \beta_v &= -\lambda_j & \forall v \in V_j \setminus \{v_{a_j}\}, \end{aligned}$$

where λ_j is a real scalar.

Consequently,

$$\begin{aligned}\boldsymbol{\alpha}^T \mathbf{x} + \boldsymbol{\beta}^T \mathbf{y} &= \sum_{i=1}^q \sum_{e \in E_i} \lambda_i x(E_i) + \sum_{i=1}^q \sum_{v \in V_i \setminus \{v_{a_i}\}} (-\lambda_i) y(V_i \setminus \{v_{a_i}\}) \\ &= \sum_{i=1}^q \lambda_i (x(E_i) - y(V_i \setminus \{v_{a_i}\})).\end{aligned}$$

Therefore, $\boldsymbol{\alpha}^T \mathbf{x} + \boldsymbol{\beta}^T \mathbf{y} = 0$ is a linear combination of equations (4.8). \square

Based on Proposition 4.1.5 and Proposition 4.1.7, the dimension of $B_{xy}(G, r, \mathbf{c})$ can be stated.

Theorem 4.1.8.

$$\dim B_{xy}(G, r, \mathbf{c}) = |E| + |V_r| - q. \quad (4.11)$$

Proof. Based on Propositions 4.1.5 and 4.1.7, we have

$$B_{xy}(G_r, r, \mathbf{c}^r) = |E_r| + |V_r| - q,$$

where \mathbf{c}^r is the restriction of \mathbf{c} to G_r . In other words, one has $|E_r| + |V_r| - q + 1$ affinely independent vectors, each of which induces a bounded r -tree of G_r (and of G). Meanwhile, the connectivity of G_r also ensures that for each edge e in $\delta_G(V_r)$, there exists an re -path between r and e such that none of its internal nodes is in O . Thus, each of the $|\delta_G(V_r)|$ incidence vectors of these paths satisfies the capacity constraints. Additionally, it is straightforward to see that each of them also contains a unique element (i.e., e) with non-zero coefficient. Hence, one has $|E_r| + |V_r| - q + 1 + |\delta_G(V_r)|$ affinely independent vectors in $B_{xy}(G, r, \mathbf{c})$, which gives us

$$\begin{aligned}\dim B_{xy}(G, r, \mathbf{c}) &\geq |E_r| + |V_r| + |\delta_G(V_r)| - q \\ &= |E| + |V_r| - q,\end{aligned}$$

as $\{E_r, \delta_G(V_r)\}$ is a partition of E . Combining it with Proposition 4.1.5 completes the proof. \square

In this section the articulation nodes in G_r are crucial for the dimension of $B_{xy}(G, r, \mathbf{c})$. It is worth noting that the two sets of articulation nodes in G_r and in G do not necessarily have inclusion relation. For instance, an articulation node in G which separates only nodes in O from r is not an articulation node in G_r , whereas in a cycle G , if O is not empty, one can trivially find some articulation nodes in G_r .

Before performing a facial study of $B_{xy}(G, r, \mathbf{c})$, we show in the next section that the articulation nodes of G are the relevant ones when studying $B_{xy}(G, r, \mathbf{c})$ through a decomposition lens.

4.2 Decomposition through 1-sum

Consider a graph $G = (V, E)$ which could be written as the 1-sum of two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Let v_a denote the node that belongs to V_1 and V_2 . Note that v_a is an articulation node in G . Without loss of generality, assume $r \in V_1$.

Given a vector $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$ in \mathbb{R}^{E+V} , let $\begin{bmatrix} \mathbf{x}^i \\ \mathbf{y}^i \end{bmatrix}$ be its restriction to G_i , $i = 1, 2$. Similarly, given a capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$, let $\mathbf{c}^i \in \mathbb{Z}_+^{V_i}$, $i = 1, 2$, be the restriction of \mathbf{c} to G_i .

We hereafter show that the characterizations of $B_{xy}(G_1, r_1, \mathbf{c}^1)$ and $B_{xy}(G_2, r_2, \mathbf{c}^2)$ yield the characterization of $B_{xy}(G, r, \mathbf{c})$, where $r_1 = r$ and $r_2 = v_a$.

Denote

$$P_C(G, r, \mathbf{c}) = \left\{ \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} : \begin{bmatrix} \mathbf{x}^1 \\ \mathbf{y}^1 \end{bmatrix} \in B_{xy}(G_1, r_1, \mathbf{c}^1), \begin{bmatrix} \mathbf{x}^2 \\ \mathbf{y}^2 \end{bmatrix} \in B_{xy}(G_2, r_2, \mathbf{c}^2), \right. \\ \left. x(\delta(v_a)) - c_{v_a} y_{v_a} \leq 0 \right\}.$$

We claim that $P_C(G, r, \mathbf{c})$ and $B_{xy}(G, r, \mathbf{c})$ are equal. It can be first proved that both polytope have the same set of integral points, as stated in the following proposition.

Proposition 4.2.1.

$$B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V} = P_C(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}.$$

Proof. Consider any vector $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$ in $B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}$, that is, $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$ is the incidence vector of a bounded r -tree of G . Any bounded r -tree of G contains a tree of G_1 rooted at r_1 bounded by \mathbf{c}^1 , and according to Lemma 4.1.2, it also contains a tree of G_2 rooted at v_a and bounded by \mathbf{c}^2 . Hence $\begin{bmatrix} \mathbf{x}^1 \\ \mathbf{y}^1 \end{bmatrix}$ and $\begin{bmatrix} \mathbf{x}^2 \\ \mathbf{y}^2 \end{bmatrix}$ are in $B_{xy}(G_1, r_1, \mathbf{c}^1)$ and $B_{xy}(G_2, r_2, \mathbf{c}^2)$, respectively. Moreover, any bounded r -tree of G satisfies $x(\delta(v_a)) - c_{v_a} y_{v_a} \leq 0$ as well. Thus,

$$B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V} \subseteq P_C(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}.$$

Conversely, any vector $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$ in $P_C(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}$ induces an r -tree of G , as $y_{v_a}^1 = y_{v_a}^2$ ensures the connectivity between the two r -trees in G_1 and G_2 . This r -tree of G contains a bounded r -tree of G_i for $i \in \{1, 2\}$, thus any node v in $V \setminus \{v_a\}$ is bounded by c_v . In addition, $x(\delta(v_a)) - c_{v_a} y_{v_a} \leq 0$ is also satisfied. Therefore, $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix}$ induces a bounded r -tree of

G , meaning that it belongs to $B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}$, and hence our proof is complete. \square

Recall that $B_{xy}(G, r, \mathbf{c})$ is a convex hull of a finite subset of \mathbb{Z}^{E+V} , and then is an integral polytope. So now we just need to prove that $P_C(G, r, \mathbf{c})$ is integral.

Given any extreme point $\left[\frac{\bar{x}}{\bar{y}}\right]$ of $P_C(G, r, \mathbf{c})$, let $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ denote the linear system composed of the equations associated with the constraints of $P_C(G, r, \mathbf{c})$ binding at $\left[\frac{\bar{x}}{\bar{y}}\right]$. Since $\left[\frac{\bar{x}}{\bar{y}}\right]$ is an extreme point, it is the only solution to $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$. Let $S_i(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ be the set of equations of $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ whose support graphs are subgraphs of G_i , $i = 1, 2$. It is important to note that the equation $y_{v_a} = 0$ (or $y_{v_a} = 1$) belongs to both $S_1(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and $S_2(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ or none, and no other equations of $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ can belong to both $S_1(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and $S_2(\bar{\mathbf{x}}, \bar{\mathbf{y}})$.

Proposition 4.2.2. *Let $\left[\frac{\bar{x}}{\bar{y}}\right]$ be an extreme point of $P_C(G, r, \mathbf{c})$ with $\bar{y}_{v_a} = 0$. Then $\left[\frac{\bar{x}}{\bar{y}}\right]$ is integral.*

Proof. Assume that $\left[\frac{\bar{x}}{\bar{y}}\right]$ is fractional. Since for any node $v \in V_2 \setminus \{v_a\}$, we have $y_v \leq y_{v_a}$ by (4.10), hence $\bar{y}_v = \bar{y}_{v_a} = 0$ holds for any $v \in V_2$. Additionally, recall that a subset of the extended subtour elimination inequalities (4.2) associated with an edge and its extremity is

$$x_e - y_v \leq 0 \quad \forall v \in V, e \in \delta(v). \quad (4.12)$$

Hence, we have $x_e - y_v \leq 0$ for any $v \in V_2$, $e \in \delta(v)$, and therefore one also has $\bar{x}_e = 0$ for any $e \in E_2$. Moreover, since $\left[\frac{\bar{x}^1}{\bar{y}^1}\right] \in B_{xy}(G_1, r_1, \mathbf{c}^1)$, there must exist an integral extreme point $\left[\frac{\tilde{x}^1}{\tilde{y}^1}\right]$ of $B_{xy}(G_1, r_1, \mathbf{c}^1)$ that is a solution to $S_1(\bar{\mathbf{x}}, \bar{\mathbf{y}})$. Note that $\tilde{y}_{v_a}^1 = 0$. Combining $\left[\frac{\tilde{x}^1}{\tilde{y}^1}\right]$ and $\left[\frac{\bar{x}^2}{\bar{y}^2}\right] = \mathbf{0}$ gives us an integral point that also satisfies $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, which is a contradiction to $\left[\frac{\bar{x}}{\bar{y}}\right]$ being an extreme point. \square

Proposition 4.2.3. *Let $\left[\frac{\bar{x}}{\bar{y}}\right]$ be an extreme point of $P_C(G, r, \mathbf{c})$ with $\bar{x}(\delta(v_a)) - c_{v_a}\bar{y}_{v_a} < 0$. Then $\left[\frac{\bar{x}}{\bar{y}}\right]$ is integral.*

Proof. Assume that $\left[\frac{\bar{x}}{\bar{y}}\right]$ is fractional. Note that $\bar{y}_{v_a} > 0$ according to Proposition 4.2.2. Additionally, since $\left[\frac{\bar{x}^i}{\bar{y}^i}\right]$ satisfies $S_i(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ with $\bar{y}_{v_a}^i > 0$ for $i \in \{1, 2\}$, there exists some extreme point $\left[\frac{\tilde{x}^i}{\tilde{y}^i}\right]$ of $B_{xy}(G_i, r_i, \mathbf{c}^i)$ that also satisfies $S_i(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ with $\tilde{y}_{v_a}^i = 1$. Vector $\left[\frac{\tilde{x}^i}{\tilde{y}^i}\right]$ obviously is integral for $i \in \{1, 2\}$. Combining $\left[\frac{\tilde{x}^1}{\tilde{y}^1}\right]$ and $\left[\frac{\bar{x}^2}{\bar{y}^2}\right]$ gives us another vector that satisfies $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, which is a contradiction to $\left[\frac{\bar{x}}{\bar{y}}\right]$ being an extreme point. Therefore, $\left[\frac{\bar{x}}{\bar{y}}\right]$ is integral. \square

Proposition 4.2.4. *Let $\left[\frac{\bar{x}}{\bar{y}}\right]$ be an extreme point of $P_C(G, r, \mathbf{c})$ with $0 < \bar{y}_{v_a} < 1$. Then $\left[\frac{\bar{x}}{\bar{y}}\right]$ is integral.*

Proof. Assume that $\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$ is fractional. Note that $\bar{x}(\delta(v_a)) - c_{v_a}\bar{y}_{v_a} = 0$ according to Proposition 4.2.3. Since $\begin{bmatrix} \bar{x}^i \\ \bar{y}^i \end{bmatrix}$ satisfies $S_i(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ with $\bar{y}_{v_a}^i$ being fractional for $i \in \{1, 2\}$, there exists some extreme point $\begin{bmatrix} \tilde{x}^i \\ \tilde{y}^i \end{bmatrix}$ of $B_{xy}(G_i, r_i, \mathbf{c}^i)$ that also satisfies $S_i(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ with $\tilde{y}_{v_a}^i = 0$. Vector $\begin{bmatrix} \tilde{x}^i \\ \tilde{y}^i \end{bmatrix}$ obviously is integral. Let $\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix}$ be the vector obtained by combining $\begin{bmatrix} \tilde{x}^1 \\ \tilde{y}^1 \end{bmatrix}$ and $\begin{bmatrix} \tilde{x}^2 \\ \tilde{y}^2 \end{bmatrix}$. Moreover, as $\tilde{y}_{v_a}^1 = \tilde{y}_{v_a}^2 = 0$, and $x^i(\delta_{G_i}(v_a)) - c_{v_a}^i y_{v_a}^i \leq 0$ is valid for $B_{xy}(G_i, r_i, \mathbf{c}^i)$, $i \in \{1, 2\}$, we also have

$$\begin{aligned} \tilde{x}(\delta_G(v_a)) &= \tilde{x}^1(\delta_{G_1}(v_a)) + \tilde{x}^2(\delta_{G_2}(v_a)) \\ &\leq c_{v_a}^1 \tilde{y}_{v_a}^1 + c_{v_a}^2 \tilde{y}_{v_a}^2 \\ &= 0. \end{aligned}$$

Therefore, vector $\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix}$ is a different solution than $\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$ to $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, a contradiction to $\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$ being an extreme point. \square

Proposition 4.2.5. *Let $\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$ be an extreme point of $P_C(G, r, \mathbf{c})$ with $\bar{y}_{v_a} = 1$. Then $\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$ is integral.*

Proof. Assume that $\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$ is fractional. We claim that there does not exist $i \in \{1, 2\}$ such that $\begin{bmatrix} \bar{x}^i \\ \bar{y}^i \end{bmatrix}$ is integral. In order to prove it, assume otherwise that $\begin{bmatrix} \bar{x}^i \\ \bar{y}^i \end{bmatrix}$ is integral with $i \in \{1, 2\}$ and let $j \in \{1, 2\}$ with $j \neq i$. We know that $\bar{x}^i(\delta_{G_i}(v_a))$ is integral as well as $\bar{x}^j(\delta_{G_j}(v_a)) = c_{v_a} y_{v_a} - \bar{x}^i(\delta_{G_i}(v_a))$ is integral. Let $c' = c_{v_a} y_{v_a} - \bar{x}^i(\delta_{G_i}(v_a))$. Due to the integrality of $B_{xy}(G_j, r_j, \mathbf{c}^j)$, where \mathbf{c}^j is obtained from \mathbf{c}^j by substituting c' for $c_{v_a}^j$, there must exist some extreme point $\begin{bmatrix} \bar{x}^j \\ \bar{y}^j \end{bmatrix}$ of $B_{xy}(G_j, r_j, \mathbf{c}^j)$ satisfying $x^j(\delta_{G_j}(v_a)) = c'$ and all the equations of $S_j(\bar{\mathbf{x}}, \bar{\mathbf{y}})$. Recall that $y_{v_a} = 1$ is an equation of $S_j(\bar{\mathbf{x}}, \bar{\mathbf{y}})$. Therefore combining $\begin{bmatrix} \bar{x}^i \\ \bar{y}^i \end{bmatrix}$ and $\begin{bmatrix} \bar{x}^j \\ \bar{y}^j \end{bmatrix}$ gives an integral point that also satisfies $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, which forms a contradiction.

Therefore, $\begin{bmatrix} \bar{x}^i \\ \bar{y}^i \end{bmatrix}$ and $\begin{bmatrix} \bar{x}^j \\ \bar{y}^j \end{bmatrix}$ are both fractional. Thus there can be at most $|E_i| + |V_i| - 1$ linearly independent equations from $S_i(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, $i \in \{1, 2\}$. Without loss of generality we assume that $y_{v_a} = 1$ is included in both sets of linearly independent equations. Additionally, there is one equation (that is, $x(\delta(v_a)) - c_{v_a} y_{v_a} = 0$) of $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ that does not belong to either $S_1(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ or $S_2(\bar{\mathbf{x}}, \bar{\mathbf{y}})$. Thus we have at most

$$|E_1| + |V_1| - 1 + |E_2| + |V_2| - 1 - 1 + 1 = |E| + |V| - 1,$$

linearly independent equations from $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, which forms a contradiction to $S(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ admitting a unique solution. Thus, $\begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$ is integral. \square

Hence, the following theorem is proved by considering Propositions 4.2.1 to 4.2.5.

Theorem 4.2.6. $B_{xy}(G, r, \mathbf{c}) = P_C(G, r, \mathbf{c})$. □

It shows that if the extended bounded r -tree polytope can be characterized on two graphs then it can be characterized on their 1-sum. Before characterizing $B_{xy}(G, r, \mathbf{c})$ on trees and cycles, and as a consequence of Theorem 4.2.6 on cactus graphs, we present in the next section a facial study of $B_{xy}(G, r, \mathbf{c})$.

4.3 Facets

In this section, for each valid inequality among (4.2)-(4.5), we give necessary and sufficient conditions for it to be facet-defining of $B_{xy}(G, r, \mathbf{c})$. Actually as seen in Section 4.1, inequalities (4.6), (4.9) and (4.10) can be obtained as linear combination of inequalities (4.2) and equations (4.7), (4.8). Thus, the facial study of (4.2) covers the facial study of (4.6), (4.9) and (4.10).

For the non-negativity inequalities (4.5), it is worth noting that the following relation holds:

$$\dim\{[\begin{smallmatrix} x \\ y \end{smallmatrix}] \in B_{xy}(G, r, \mathbf{c}) : x_e = 0\} = \dim B_{xy}(G \setminus e, r, \mathbf{c}). \quad (4.13)$$

Recall that $G \setminus e = (V, E \setminus \{e\})$.

Proposition 4.3.1. *Let $e \in E$. $x_e \geq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$ if and only if the following two conditions are satisfied*

1. *there does not exist an edge $e' \in E \setminus \{e\}$ such that removing e and $\delta(O) \setminus \{e'\}$ from G separates e' and r ;*
2. *whenever e belongs to a 2-connected block (V_i, E_i) of G_r , for some $i \in \{1, \dots, q\}$, $(V_i, E_i \setminus \{e\})$ remains 2-connected.*

Proof. For the necessity, consider first that there exists an edge $e' \in E \setminus \{e\}$ such that removing e and $\delta(O) \setminus \{e'\}$ disconnects e' from r . It can be seen that the face induced by $x_e \geq 0$ is a proper subset of the face induced by $x_{e'} \geq 0$ in this case.

Suppose now that e belongs to a 2-connected block (V_i, E_i) of G_r for some $i \in \{1, \dots, q\}$, such that $(V_i, E_i \setminus \{e\})$ is not 2-connected. Let $\{E'_1, \dots, E'_{q'}\}$ be the partition of $E'_r = E_r \setminus \{e\}$ such that each E'_j , $j \in \{1, \dots, q'\}$, induces a block of $G[E'_r]$. Since $(V_i, E_i \setminus \{e\})$ is not 2-connected but (V_i, E_i) is, we have $q' \geq q + 1$. According to (4.13) and Theorem

4.1.8, one therefore has

$$\begin{aligned} \dim\{[\mathbf{x}] \in B_{xy}(G, r, \mathbf{c}) : x_e = 0\} &= \dim B_{xy}(G \setminus e, r, \mathbf{c}) \\ &= |E \setminus \{e\}| + |V_r| - q' \\ &\leq |E| + |V_r| - q - 2, \end{aligned}$$

which implies that $x_e \geq 0$ is not facet-defining.

For the sufficiency, assume that both conditions hold. In $G \setminus e$, there is no unreachable edge and the number of blocks in $G_r \setminus e$ equals q , the number of blocks in G_r . According to Theorem 4.1.8, we have

$$\begin{aligned} \dim B_{xy}(G \setminus e, r, \mathbf{c}) &= |E \setminus \{e\}| + |V_r| - q \\ &= \dim B_{xy}(G, r, \mathbf{c}) - 1, \end{aligned}$$

and thus $x_e \geq 0$ is facet-defining for $B_{xy}(G, r, \mathbf{c})$ according to (4.13). \square

The upper bound inequality (4.4) always defines a facet of $B_{xy}(G, r, \mathbf{c})$, as stated in the next proposition.

Proposition 4.3.2. $y_r \leq 1$ defines a facet of $B_{xy}(G, r, \mathbf{c})$.

Proof. As there is only one bounded r -tree without containing r , that is, the empty graph (\emptyset, \emptyset) , there must exist $\dim B_{xy}(G, r, \mathbf{c})$ affinely independent vectors that induce bounded r -trees containing r . Therefore $y_r \leq 1$ defines a facet of $B_{xy}(G, r, \mathbf{c})$. \square

Proposition 4.3.3. $x(\delta(r)) - c_r y_r \leq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$ if and only if one of the two conditions is satisfied

1. $|\delta(r)| = c_r = 1$;
2. $|\delta(r)| > c_r$.

Proof. Let $\mathcal{F} = \{[\mathbf{x}] \in B_{xy}(G, r, \mathbf{c}) : x(\delta(r)) - c_r y_r = 0\}$. Clearly we need to have $|\delta(r)| \geq c_r$ for \mathcal{F} to be nonempty.

For the necessity, assume that $|\delta(r)| = c_r \geq 2$. According to (4.12), $x_e - y_r \leq 0$ is valid for $B_{xy}(G, r, \mathbf{c})$ for any $e \in \delta(r)$. Thus any bounded r -tree of G that satisfies $x(\delta(r)) - c_r y_r = 0$ also satisfies $x_e - y_r = 0$ for $e \in \delta(r)$. The latter equation is not valid for $B_{xy}(G, r, \mathbf{c})$ since the graph reduced to r is a bounded r -tree that satisfies $x_e - y_r < 0$. As $c_r \geq 2$, the graph reduced to any edge e incident with r is a bounded r -tree that satisfies $x_e - y_r = 0$ and $x(\delta(r)) - c_r y_r < 0$. Thus \mathcal{F} is a proper subset of the proper face induced by $x_e - y_r \leq 0$ for $e \in \delta(r)$.

For the sufficiency, assume first that $|\delta(r)| = c_r = 1$. As the incidence vector of the graph reduced to r is the only vector in $B_{xy}(G, r, \mathbf{c})$ not in \mathcal{F} , there must exist $\dim B_{xy}(G, r, \mathbf{c})$ affinely independent vectors in \mathcal{F} . Therefore $x(\delta(r)) - c_r y_r \leq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$ if $|\delta(r)| = c_r = 1$.

Suppose now that $|\delta(r)| > c_r$ and $\mathcal{F} \subseteq \{\mathbf{x} \in B_{xy}(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} = b\} \subsetneq B_{xy}(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ is valid for $B_{xy}(G, r, \mathbf{c})$. As $\mathbf{0} \in \mathcal{F}$, we immediately have $b = 0$.

Let $v \in N(r)$. Consider $F \subseteq \delta(r) \setminus \delta(v)$ such that $|F| = c_r$. Clearly such set F exists. The incidence vector of the bounded r -tree $G[F]$ belongs to \mathcal{F} , and so does the incidence vector of the bounded r -tree $G[F \setminus \{e\} \cup \{rv\}]$ for any $e \in F$. Therefore, we obtain

$$a_{rv} + d_v = \lambda \quad \forall v \in N(r), \quad (4.14)$$

where $\lambda \in \mathbb{R}$. Considering again the incidence vector of $G[F]$, we have

$$\begin{aligned} d_r &= -a(F) - d(V[F] \setminus \{r\}) \\ &= -c_r \lambda. \end{aligned}$$

Consider any edge $uv \in E \setminus \delta(r)$. Without loss of generality, we suppose that $u \notin O$ and there exists a path P_{ru} in G_r between r and u such that it does not contain v and $|N(r) \cap V[P_{ru}]| = 1$. Let $F_u \subseteq \delta(r)$ with $|F_u| = c_r - 1$ and $F_u \cap P_{ru} = \emptyset$. The subgraphs $G[F \cup P_{ru}]$ and $G[F \cup P_{ru} \cup \{uv\}]$ are two bounded r -trees of G and their incidence vectors are in \mathcal{F} . Hence, it can be deduced that

$$a_{uv} + d_v = 0. \quad (4.15)$$

Note that if $uv \in E_i \setminus \delta(v_{a_i})$, $i \in \{1, \dots, q\}$, where (V_i, E_i) is a 2-connected block of G_r , then nodes u and v are interchangeable in the previous argument. Consequently, for any block (V_i, E_i) of G_r , $i \in \{1, \dots, q\}$, we have

$$\begin{aligned} d_v &= \epsilon_i & \forall v \in V_i \setminus \{v_{a_i}\}, \\ a_e &= -\epsilon_i & \forall e \in E_i \setminus \delta(r), \end{aligned}$$

where $\epsilon_i \in \mathbb{R}$. Combining this with (4.14), we deduce that for any $rv \in E_i$, for some $i \in \{1, \dots, q\}$,

$$\begin{aligned} a_{rv} &= \lambda - d_v \\ &= \lambda - \epsilon_i. \end{aligned}$$

For any node $v_o \in O$ (4.15) yields

$$\begin{aligned} d_{v_o} &= \mu_{v_o}, \\ a_e &= -\mu_{v_o} \quad \forall e \in \delta(v_o) \setminus \delta(r), \end{aligned}$$

where $\mu_{v_o} \in \mathbb{R}$.

For each edge rv_o with $v_o \in O$, from (4.14) one gets

$$\begin{aligned} a_{rv_o} &= \lambda - d_{v_o} \\ &= \lambda - \mu_{v_o}. \end{aligned}$$

It is important to note that $\{\delta(O), E_1, \dots, E_q\}$ is a partition of E and $\{\{r\}, O, V_1 \setminus \{v_{a_1}\}, \dots, V_q \setminus \{v_{a_q}\}\}$ is a partition of V . Consequently, the inequality $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ can be written as

$$\begin{aligned} -c_r \lambda y_r + \sum_{i=1}^q \sum_{e \in \delta(r) \cap E_i} (\lambda - \epsilon_i) x_e - \sum_{i=1}^q \sum_{e \in E_i \setminus \delta(r)} \epsilon_i x_e + \sum_{i=1}^q \sum_{v \in V_i \setminus \{v_{a_i}\}} \epsilon_i y_v \\ + \sum_{v_o \in O} \mu_{v_o} y_{v_o} - \sum_{v_o \in O, u \neq r} \mu_{v_o} x_{uv_o} + \sum_{v_o \in O} (\lambda - \mu_{v_o}) x_{rv_o} \leq 0, \end{aligned}$$

which is equivalent to

$$\lambda(x(\delta(r)) - c_r y_r) - \sum_{i=1}^q \epsilon_i (x(E_i) - y(V_i \setminus \{v_{a_i}\})) - \sum_{v_o \in O} \mu_{v_o} (x(\delta(v_o)) - y_{v_o}) \leq 0.$$

It is then proved that $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ is a linear combination of $x(\delta(r)) - c_r y_r \leq 0$ and equations (4.7) and (4.8).

Finally we have $\mathcal{F} \neq B_{xy}(G, r, \mathbf{c})$ as the graph reduced to r is a bounded r -tree whose incidence vector is not in \mathcal{F} . Therefore, \mathcal{F} is a facet of $B_{xy}(G, r, \mathbf{c})$. \square

Proposition 4.3.4. *Let $v \in V \setminus \{r\}$. $x(\delta(v)) - c_v y_v \leq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$ if and only if $|\delta(v)| > c_v \geq 2$.*

Proof. Let $\mathcal{F} = \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : x(\delta(v)) - c_v y_v = 0\}$. Clearly we need to have $|\delta(v)| \geq c_v$ for \mathcal{F} to be nonempty.

For the necessity, if $c_v = 1$, then we have $v \in O$, and thus any bounded r -tree of G must satisfy $x(\delta(v)) - c_v y_v = 0$ according to (4.7), that is, $\mathcal{F} = B_{xy}(G, r, \mathbf{c})$.

If $|\delta(v)| = c_v \geq 2$, we deduce from (4.12) that any bounded r -tree of G that satisfies $x(\delta(v)) - c_v y_v = 0$ also satisfies $x_e - y_v = 0$ for any edge $e \in \delta(v)$. Additionally, as

$|\delta(v)| = c_v \geq 2$, a path P_{rv} between r and v of G_r induces a bounded r -tree of G that satisfies $x_e - y_v < 0$ for some edge $e \in \delta(v)$. Hence $x_e - y_v = 0$ is not valid for $B_{xy}(G, r, \mathbf{c})$, and thus \mathcal{F} is a proper subset of the proper face defined by $x_e - y_v \leq 0$.

For the sufficiency, assume that the condition in the proposition is satisfied, and $\mathcal{F} \subseteq \{[\begin{smallmatrix} \mathbf{x} \\ \mathbf{y} \end{smallmatrix}] \in B_{xy}(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} = b\} \subsetneq B_{xy}(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ is valid for $B_{xy}(G, r, \mathbf{c})$. As the empty graph and the graph reduced to r are both bounded r -trees that satisfy $x(\delta(v)) - c_v y_v = 0$ we obtains that $b = 0$ and $d_r = 0$.

Consider any edge $uw \in E \setminus \delta(v)$. Without loss of generality, we suppose that $u \notin O$ and there exists a path P_{ru} in G_r between r and u such that it does not contain w and P_{ru} contains as few edges as possible. As $v \neq r$, $v \neq u$, we have either $|P_{ru} \cap \delta(v)| = 0$ or $|P_{ru} \cap \delta(v)| = 2$. In the former case, $G[P_{ru}]$ and $G[P_{ru} \cup \{uw\}]$ are two bounded r -trees of G and their incidence vectors are in \mathcal{F} . For the latter case, let $F_u \subseteq \delta(v)$ with $F_u \cap P_{ru} = \emptyset$ and $|F_u| = c_v - 2$. The subgraphs $G[F_u \cup P_{ru}]$ and $G[F_u \cup P_{ru} \cup \{uw\}]$ are two bounded r -trees of G and their incidence vectors are in \mathcal{F} . Hence, for both cases we have

$$a_{uw} + d_w = 0. \quad (4.16)$$

Note that if $uw \in E_i \setminus (\delta(v_{a_i}) \cup \delta(v))$, $i \in \{1, \dots, q\}$, where (V_i, E_i) is a 2-connected block of G_r , then nodes u and w are interchangeable in the previous argument. Consequently, since (4.16) only holds for edges non-incident with v , for $i \in \{1, \dots, q\}$, we deduce that

$$\begin{aligned} d_u &= \epsilon_i & \forall u \in V_i \setminus \{v_{a_i}, v\}, \\ a_e &= -\epsilon_i & \forall e \in E_i \setminus \delta(v), \end{aligned}$$

where $\epsilon_i \in \mathbb{R}$.

Without loss of generality, assume that $v \in V_j \setminus \{v_{a_j}\}$ for some block (V_j, E_j) of G_r with $j \in \{1, \dots, q\}$. It is straightforward to see that any path in G_r between r and v contains exactly (and ends with) one edge in $\delta(v) \cap E_j$.

For $u \in N(v)$, let P_{rv}^u denote a path between r and v in G_r such that $uv \notin P_{rv}^u$ and it contains as few edges as possible if such path exists (that is, if $u \neq v_{a_j}$ when block (V_j, E_j) is a bridge). Let vv' denote the edge in both $\delta(v)$ and P_{rv}^u . Consider any edge set $F_u \subseteq \delta(v) \setminus \{uv, vv'\}$ with $|F_u| = c_v - 1$. Both $G[P_{rv}^u \cup F_u]$ and $G[P_{rv}^u \cup F_u \setminus \{e\} \cup \{vu\}]$ are bounded r -trees whose incidence vectors belong to \mathcal{F} for any $e \in \delta(v) \setminus \{vv', uv\}$.

If (V_j, E_j) is 2-connected (or equivalently, $|\delta(v) \cap E_j| \geq 2$), such P_{rv}^u exists (and so does F_u) for any node in $N(v)$. Thus we have

$$d_u + a_{uv} = \lambda \quad \forall u \in N(v),$$

where $\lambda \in \mathbb{R}$. Moreover, as for any $u \in N(v)$, $G[P_{rv}^u \cup F_u]$ and $G[P_{rv}^u \setminus \delta(v)]$ are two bounded r -trees whose incidence vectors belong to \mathcal{F} , we have

$$\begin{aligned} d_v &= -(c_v \lambda - d_{v'}) \\ &= -c_v \lambda + \epsilon_j. \end{aligned}$$

Recall $v' \in N(v) \cap V[P_{rv}^u]$.

If (V_j, E_j) is a bridge (or equivalently, $|\delta(v) \cap E_j| = 1$), let $E_j = \{vv'\}$. Note that $v_{a_j} = v'$. Similar to the previous case, P_{rv}^u and F_u exist for any node in $N(v) \setminus \{v'\}$, and thus we have

$$d_u + a_{uv} = \lambda \quad \forall u \in N(v) \setminus \{v'\},$$

where $\lambda \in \mathbb{R}$. Additionally, considering $G[P_{rv}^u \cup F_u]$ and $G[P_{rv}^u \setminus \{vv'\}]$ for any $u \in N(v) \setminus \{v'\}$ gives us

$$\begin{aligned} a_{vv'} + d_v &= -(a(F_u) + d(V[F_u] \setminus \{v\})) \\ &= -(c_v - 1)\lambda. \end{aligned}$$

Thus, without loss of generality, we set

$$\begin{aligned} a_{vv'} &= \lambda - \epsilon_j, \\ d_v &= -c_v \lambda + \epsilon_j, \end{aligned}$$

where $\epsilon_j \in \mathbb{R}$.

To summarize, in both cases (block (V_j, E_j) being 2-connected or a bridge) we have

$$d_v = -c_v \lambda + \epsilon_j,$$

and also

$$a_{uv} = \lambda - d_u \quad \forall u \in N(v). \tag{4.17}$$

As a consequence, for any $u \in N(v) \cap V_i \setminus \{v_{a_i}\}$, where (V_i, E_i) is a block of G_r , we have

$$\begin{aligned} a_{uv} &= \lambda - d_u \\ &= \lambda - \epsilon_i. \end{aligned}$$

Without loss of generality, for any $v_o \in O$, let

$$d_{v_o} = \mu_{v_o},$$

where $\mu_{v_o} \in \mathbb{R}$. We then have from (4.17), for any $v_o \in N(v) \cap O$,

$$a_{vv_o} = \lambda - \mu_{v_o}.$$

Besides, from (4.16) one has for any $e \in \delta(v_o) \setminus \delta(v)$

$$a_e = -\mu_{v_o}.$$

Finally, if $r \in N(v)$, then (4.17) gives

$$\begin{aligned} a_{rv} &= \lambda - d_r \\ &= \lambda. \end{aligned}$$

Note that $\{\delta(O), E_1, \dots, E_q\}$ is a partition of E and $\{\{r\}, \{v\}, O, V_1 \setminus \{v, v_{a_1}\}, \dots, V_q \setminus \{v, v_{a_q}\}\}$ is a partition of V . Hence, all the coefficients associated with elements in V and E have been taken care of. Therefore, $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ can be rewritten as

$$\begin{aligned} & 0y_r + (-c_v\lambda + \epsilon_j)y_v + \sum_{i=1}^q \sum_{u \in V_i \setminus \{v_{a_i}\}} \epsilon_i y_u + \sum_{v_o \in O} \mu_{v_o} y_{v_o} \\ & + \sum_{i=1}^q \sum_{e \in E_i \setminus \delta(v)} (-\epsilon_i) x_e + \sum_{v_o \in O} \sum_{e \in \delta(v_o) \setminus \delta(v)} (-\mu_{v_o}) x_{vv_o} \\ & + \sum_{v_o \in O} \sum_{e \in \delta(v_o) \cap \delta(v)} (\lambda - \mu_{v_o}) x_{vv_o} + \sum_{i=1}^q \sum_{e \in E_i \cap \delta(v)} (\lambda - \epsilon_i) x_e + \sum_{e \in \delta(r) \cap \delta(v)} \lambda \leq 0, \end{aligned}$$

which is equivalent to

$$\lambda(x(\delta(v)) - c_v y_v) - \sum_{i=1}^q \epsilon_i (x(E_i) - y(V_i \setminus \{v_{a_i}\})) - \sum_{v_o \in O} \mu_{v_o} (x(\delta(v_o)) - y_{v_o}) \leq 0.$$

Hence, $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ is a linear combination of $x(\delta(v)) - c_v y_v \leq 0$ and equations (4.7) and (4.8). Finally, any path between r and v in G_r is a bounded r -tree that satisfies $x(\delta(v)) - c_v y_v < 0$, which leads to $\mathcal{F} \subsetneq B_{xy}(G, r, \mathbf{c})$. Therefore, one can conclude that $x(\delta(v)) \leq c_v y_v$ is facet-defining. \square

The extended subtour elimination inequalities (4.2) are also mentioned in Goemans [1994] for the r -tree problem and $R_{xy}(G, r)$, where the conditions are partially reflected in the

conditions for $B_{xy}(G, r, \mathbf{c})$. However, for $B_{xy}(G, r, \mathbf{c})$ the capacity constraints are considered and the facial study cannot be restricted to the blocks of G_r . As a result, the facets induced by the extended subtour elimination inequalities of $B_{xy}(G, r, \mathbf{c})$ are significantly different those of $R_{xy}(G, r)$.

The following proposition contains a condition that is taken directly from Goemans [1994] and it is still viable for $B_{xy}(G, r, \mathbf{c})$.

Proposition 4.3.5. *$x(E[S]) - y(S \setminus \{v\}) \leq 0, S \subseteq V_i$, defines a facet of $B_{xy}(G, r, \mathbf{c})$ only if $G[S]$ is connected.*

Proof. If $G[S]$ is not connected, there must exist some $u \in S \setminus \{v\}$ that is in a connected component of $G[S]$ which is different from the one containing v . Thus the face induced by $x(E[S]) - y(S \setminus \{v\}) \leq 0$ is a proper subset of a proper face induced by $x_e \geq 0$ for any $e \in \delta(u)$. \square

As a first step, we give the proofs that it is sufficient to cover all the facets induced by the extended subtour elimination inequalities by just considering a subset of them.

Firstly, we restrict the set S to such that $O \cap S \setminus \{v\} = \emptyset$.

Proposition 4.3.6. *Given $v \in S \subseteq V$ and $O \cap S \setminus \{v\} \neq \emptyset$, if $\mathcal{F} = \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : x(E[S]) - y(S \setminus \{v\}) = 0\}$ is a proper face of $B_{xy}(G, r, \mathbf{c})$, it is always a subset of a proper face defined by either $x_e \geq 0$ for some $e \in \delta(O)$, or $x(E[S']) - y(S' \setminus \{v\}) \leq 0$ with $O \cap S' \setminus \{v\} = \emptyset$.*

Proof. Assume that $v_o \in S \setminus \{v\}$ for some $v_o \in O$. If there exists $e_o \in \delta(v_o) \setminus E[S]$, \mathcal{F} is clearly a subset of the proper face induced by $x_{e_o} \geq 0$. Otherwise if $\delta(v_o) \subseteq E[S]$, then $x(E[S \setminus \{v_o\}]) - y(S \setminus \{v, v_o\}) \leq 0$ induces the same face as \mathcal{F} since $x(\delta(v_o)) - y_{v_o} = 0$ is a valid equation for $B_{xy}(G, r, \mathbf{c})$. Thus the proposition holds. \square

Secondly, we show that if $S \subseteq V \setminus O$, we can restrict to the cases such that $S \subseteq V_i$ for some $i \in \{1, \dots, q\}$.

Proposition 4.3.7. *Given $v \in S \subseteq V \setminus O$, if $\mathcal{F} = \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : x(E[S]) - y(S \setminus \{v\}) = 0\}$ is a proper face of $B_{xy}(G, r, \mathbf{c})$, it is always a subset of a proper face defined by $x(E[S']) - y(S' \setminus \{v\}) \leq 0$ where $v' \in S' \subseteq V_i \cap S$ for some $i \in \{1, \dots, q\}$.*

Proof. Note that we can assume that $G[S]$ is connected according to Proposition 4.3.5. We denote $S_i = S \cap V_i$ for $i \in \{1, \dots, q\}$. For $i \in \{1, \dots, q\}$, with $v \notin S_i$ and $E[S_i] \neq \emptyset$,

let v_i denote the node in S_i such that removing it from G_r disconnects $E[S_i]$ and v_{a_i} . Otherwise if $v \in S_i$ and $E[S_i] \neq \emptyset$, let $v_i = v$. One has the following valid inequalities

$$x(E[S_i]) - y(S_i \setminus \{v_i\}) \leq 0 \quad \forall i \in \{1, \dots, q\}, E[S_i] \neq \emptyset.$$

Notice that $\{S_1 \setminus \{v_1\}, \dots, S_q \setminus \{v_q\}\}$ is a partition of $S \setminus \{v\}$, and $\{E[S_1], \dots, E[S_q]\}$ is a partition of $E[S]$. Thus $x(E[S]) - y(S \setminus \{v\}) \leq 0$ can be written as a linear combination of these inequalities. Therefore, \mathcal{F} is a subset of the face defined by any of the aforementioned inequalities, and at least one of the faces defined by the aforementioned inequalities is proper as \mathcal{F} is proper. \square

Lastly, we show that if $v \in O$, we can restrict to the cases such that $S \setminus \{v\} \subseteq V_i$ for some $i \in \{1, \dots, q\}$.

Proposition 4.3.8. *Given $S \subseteq V$ and $S \cap O = \{v\}$, if $\mathcal{F} = \{[\frac{x}{y}] \in B_{xy}(G, r, \mathbf{c}) : x(E[S]) - y(S \setminus \{v\}) = 0\}$ is a proper face of $B_{xy}(G, r, \mathbf{c})$, it is always a subset of a proper face defined by $x(E[S']) - y(S' \setminus \{v'\}) \leq 0$ with $v' \in S'$ and $S' \setminus \{v'\} \subseteq V_i \cap S$ for some $i \in \{1, \dots, q\}$.*

Proof. We denote $S_i = S \cap V_i$ for $i \in \{1, \dots, q\}$. There exists $j \in \{1, \dots, q\}$, such that $N(v) \cap S_j \neq \emptyset$. As we clearly have

$$E[S] = E[S_j \cup \{v\}] \cup E[S \setminus S_j],$$

one has the following valid inequalities

$$\begin{aligned} x(E[S_j \cup \{v\}]) - y(S_j) &\leq 0, \\ x(E[S \setminus S_j]) - y(S \setminus (S_j \cup \{v\})) &\leq 0. \end{aligned}$$

Note that this decomposition of inequality can be applied on any $j \in \{1, \dots, q\}$, with $N(v) \cap S_j \neq \emptyset$. Thus the proposition holds. \square

According to Propositions 4.3.6 - 4.3.8, we can therefore focus on the extended subtour elimination inequalities associated with S and v such that $S \setminus \{v\} \subseteq V_i$ and $v \in V_i \cup O$, for some $i \in \{1, \dots, q\}$. In other words, the following proposition holds.

Proposition 4.3.9. *Given $v \in S \subseteq V$ with $|S| \geq 2$, if $\mathcal{F} = \{[\frac{x}{y}] \in B_{xy}(G, r, \mathbf{c}) : x(E[S]) - y(S \setminus \{v\}) = 0\}$ is a facet of $B_{xy}(G, r, \mathbf{c})$, then either $\mathcal{F} = \{[\frac{x}{y}] \in B_{xy}(G, r, \mathbf{c}) : x(E[S']) - y(S' \setminus \{v'\}) = 0\}$ for some $v' \in S'$ and $S' \setminus \{v'\} \subseteq V_i \cap S$, $i \in \{1, \dots, q\}$, or $\mathcal{F} = \{[\frac{x}{y}] \in B_{xy}(G, r, \mathbf{c}) : x_e = 0\}$ for some $e \in \delta(O)$.*

Proof. Direct result of Propositions 4.3.6 - 4.3.8. \square

We introduce a property of 2-connected graphs in the following lemma as a preparation for the facial proofs of the extended subtour elimination inequalities, for which the definition of an open ear decomposition (Whitney [1932]) is needed.

An *ear* of a graph G is either a path or a cycle, where a cycle also can be considered as a path with two ends being the same node. An *ear decomposition* of a graph G is a partition of its set of edges into a sequence of ears, such that the one or two ends of each ear belong to earlier ears in the sequence whereas the internal nodes of each ear do not belong to any earlier ear. An *open ear decomposition* is an ear decomposition in which only the first ear is a cycle. According to the work of Whitney [1932], a graph is 2-connected if and only if it has an open ear decomposition. Moreover, a 2-connected graph admits an open ear decomposition starting at any cycle of the graph. We present the following technical lemma based on this property of 2-connected graphs.

Lemma 4.3.10. *Given a 2-connected graph $G = (V, E)$ and two distinct nodes $u, v \in V$, there exists a path P_{uv} between u and v with $e \in P_{uv}$ for any $e \in E$.*

Proof. If there exists a cycle of G which contains u, v , and e , then such path obviously exists.

Otherwise suppose that there does not exist a cycle of G which contains u, v , and e . As G is 2-connected, there exists a cycle $C \subseteq E$ of G which contains u and v . Let C, P_1, \dots, P_k be an open ear decomposition of G , and denote $G_i = G_{i-1} \cup G[P_i]$ for $i \in \{1, \dots, k\}$, and $G_0 = G[C]$. For $i \in \{0, 1, \dots, k-1\}$, we call any path P such that as its ends belong to G_i and its inner nodes (if exist) do not belong to G_i a G_i -ear, .

We claim that for any edge e' of P_{i+1} , there exists a G_{i-1} -ear P' such that e' is in P' for $i \in \{1, \dots, k-1\}$. Note that P_{i+1} is a G_i -ear.

Without loss of generality, let P_i be a path between u_i and v_i for $i \in \{1, \dots, k\}$. If u_{i+1} and v_{i+1} both belong to G_{i-1} then the proof is trivial. If only one of u_{i+1} and v_{i+1} belongs to G_{i-1} , without loss of generality, suppose it is u_{i+1} . u_{i+1} is then an inner node of P_i . Combining P_{i+1} with the subpath between u_{i+1} and u_i (or v_i) of P_i gives us a G_{i-1} -ear which contains e' . If none of u_{i+1} and v_{i+1} belongs to G_{i-1} , that is, they are both inner nodes of P_i . Without loss of generality, suppose that between u_{i+1} and v_{i+1} , u_{i+1} is closer to u_i on the path P_i , and v_{i+1} is closer to v_i on P_i . Then combining P_{i+1} with the subpath between u_{i+1} and u_i of P_i and the subpath between v_{i+1} and v_i of P_i gives us a G_{i-1} -ear which contains e' .

By induction, we have that for any edge $e' \in E \setminus C$, there exists an G_0 -ear that contains e' . Thus, there exists a G_0 -ear P' that contains e , and without loss of generality, assume that the two ends of P' are u' and v' . Let $P_{u'v'}^1$ and $P_{u'v'}^2$ the two paths between u' and v' in the cycle G_0 . As there does not exist a cycle of G which contains u, v , and e , it can

be deduced that one of the two paths $P_{u'v'}^1$ and $P_{u'v'}^2$ contains u as an inner node and the other contains v as an inner node. Combining P' with the subpath between u' and u of $P_{u'v'}^1$ and the subpath between v' and v of $P_{u'v'}^2$ gives us a path between u and v that contains e . Note that here u and v , u' and v' , $P_{u'v'}^1$ and $P_{u'v'}^2$ are interchangeable without affecting the proof.

Therefore, there always exists a path P_{uv} of G between u and v with $e \in P_{uv}$. \square

As an immediate result, we also obtain the following corollary that considers a node instead of an edge.

Corollary 4.3.11. *Given a 2-connected graph $G = (V, E)$ and three distinct nodes $u, v, w \in V$, there exists a path P_{uv} between u and v with $w \in V[P_{uv}]$.*

Proof. Consider any edge $e \in \delta(w)$. According to Lemma 4.3.10, there exists a path P_{uv} of G between u and v with $e \in P_{uv}$. As a consequence, there exists a path P_{uv} between u and v with $w \in V[P_{uv}]$. \square

Another result that can be deduced directly is the following.

Corollary 4.3.12. *Given a 2-connected graph $G = (V, E)$, an edge $uv \in E$ and a node $w \in V$ with $w \neq u$ and $w \neq v$, there exists a cycle that contains uv and w .*

Proof. According to Lemma 4.3.11, there exists a path P_{uv} of G between u and v with $w \in V[P_{uv}]$. As one clearly has $uv \notin P_{uv}$, $P_{uv} \cup \{uv\}$ induces a cycle. \square

Although we are able to restrict ourselves to a subset of extended subtour elimination inequalities, they are still complicated to be dealt with as a whole. Hereafter, we split them into several even smaller subsets and present the facial study results case by case.

We first consider the extended subtour elimination inequality associated with V_i and $v \neq v_{a_i}$ for $i \in \{1, \dots, q\}$. Note that it defines the same face as $y_v - y_{v_{a_i}} \leq 0$ from (4.10) which has a simpler form.

Proposition 4.3.13. *Let $v \in V_i \setminus \{v_{a_i}\}$, $i \in \{1, \dots, q\}$. $y_v - y_{v_{a_i}} \leq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$ if and only if the following two conditions are satisfied*

1. $c_r \geq 2$ if $v_{a_i} = r$ and $|\delta(r)| \geq 2$;
2. $c_{v_{a_i}} \geq 3$ if $v_{a_i} \neq r$ and $|\delta(v_{a_i})| \geq 3$.

Proof. Let $\mathcal{F} = \{[\mathbf{x}] \in B_{xy}(G, r, \mathbf{c}) : y_v - y_{v_{a_i}} = 0\}$. For the necessity, if either condition is not satisfied, then \mathcal{F} is a proper subset of a proper face induced by $x(\delta(v_{a_i})) - c_{v_{a_i}} y_{v_{a_i}} \leq 0$.

For the sufficiency, assume that the conditions are satisfied and $\mathcal{F} \subseteq \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} = b\} \subsetneq B_{xy}(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ is valid for $B_{xy}(G, r, \mathbf{c})$. First, the empty graph is a bounded r -tree with its incidence vector in \mathcal{F} , and thus $b = 0$. If $r \neq v_{a_i}$, then the graph reduced to r is also a bounded r -tree with its incidence vector in \mathcal{F} , which gives us $d_r = 0$.

If $v_{a_i} \neq r$, let (V_j, E_j) be the block of G_r such that $v_{a_i} \in V_j \setminus \{v_{a_j}\}$.

Consider any edge $uw \in E \setminus E_i$. Without loss of generality, let $u \notin O$ and let $w \neq v_{a_p}$ if $u \in V_p$, for some $p \in \{1, \dots, q\}$. There must exist a path P_{ru} of G_r between r and u without passing w .

If $v_{a_i} \notin V[P_{ru}]$ and $w \neq v_{a_i}$, $G[P_{ru}]$ and $G[P_{ru} \cup \{uw\}]$ are bounded r -trees whose incidence vectors belong to \mathcal{F} .

If $v_{a_i} \notin V[P_{ru}]$ and $w = v_{a_i}$, $G[P_{ru}]$ and $G[P_{ru} \cup P_{v_{a_i}v} \cup \{uw\}]$ are bounded r -trees whose incidence vectors belong to \mathcal{F} , where $P_{v_{a_i}v}$ is a path between v_{a_i} and v in G_r .

If $v_{a_i} \in V[P_{ru}]$ and $E_i \cap P_{ru} = \emptyset$, then $G[P_{ru} \cup P_{v_{a_i}v}]$ and $G[P_{ru} \cup P_{v_{a_i}v} \cup \{uw\}]$ are bounded r -trees whose incidence vectors belong to \mathcal{F} , where $P_{v_{a_i}v}$ is a path between v_{a_i} and v in G_r .

If $E_i \cap P_{ru} \neq \emptyset$, let (V_k, E_k) be the block of G_r such that $v_{a_k} \in V_i \setminus \{v_{a_i}\}$ and removing v_{a_k} from G_r disconnects r and u . If $v \notin V[P_{ru}]$, one has that (V_i, E_i) is 2-connected. According to Corollary 4.3.11, there exists a path $P_{v_{a_i}v_{a_k}}$ in (V_i, E_i) such that $v \in V[P_{v_{a_i}v_{a_k}}]$. Then by substituting the subpath between v_{a_i} and v_{a_k} of P_{ru} for $P_{v_{a_i}v_{a_k}}$, one gets a path between r and u of G_r such that it contains v but not w . To summarize, there always exists a path P_{ru} between r and u of G_r such that $v \in V[P_{ru}]$ and $w \notin [P_{ru}]$. Therefore, $G[P_{ru}]$ and $G[P_{ru} \cup \{uw\}]$ are bounded r -trees whose incidence vectors belong to \mathcal{F} .

Consequently, one can summarize that for any $p \in \{1, \dots, q\}$ with $p \neq i$ we have

$$\begin{aligned} d_w &= \epsilon_p & \forall w \in V_p \setminus \{v_{a_p}, v_{a_i}\}, \\ a_e &= -\epsilon_p & \forall e \in E_p, \end{aligned}$$

where $\epsilon_p \in \mathbb{R}$. Additionally, for any $v_o \in O$, one has

$$\begin{aligned} d_{v_o} &= \mu_{v_o}, \\ a_{e_o} &= -\mu_{v_o} \quad \forall e \in \delta(v_o), \end{aligned}$$

Consider the case where (V_i, E_i) is 2-connected.

Let $uw \in E_i \cap (\delta(v_{a_i}) \cup \delta(v) \setminus \{v_{a_i}v\})$ be an edge with $w \neq v$ and $w \neq v_{a_i}$. There exists a path $P_{v_{a_i}v}$ of (V_i, E_i) without passing w . Let $P_{rv_{a_i}}$ be any path between r and v_{a_i} of G_r .

Then $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}]$ and $G[P_{rv_{a_i}} \cup P_{v_{a_i}v} \cup \{uw\}]$ are two bounded r -trees whose incidence vectors belong to \mathcal{F} .

As (V_i, E_i) is 2-connected, it contains two paths $P_{v_{a_i}v}^1$, and $P_{v_{a_i}v}^2$ such that they do not share any inner node. Consider an edge $uw \in P_{v_{a_i}v}^1 \cup P_{v_{a_i}v}^2 \setminus (\delta(v_{a_i}) \cup \delta(v))$, and without loss of generality assume $uw \in P_{v_{a_i}v}^1$. Let $P_{v_{a_i}u}^1$ denote the subpath between v_{a_i} and u of $P_{v_{a_i}v}^1$. One has that $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}^2 \cup P_{v_{a_i}u}^1]$, $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}^2 \cup P_{v_{a_i}u}^1 \cup \{uw\}]$, $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}^1 \cup P_{v_{a_i}v}^2 \setminus (P_{v_{a_i}u}^1 \cup \{uw\})]$, $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}^1 \cup P_{v_{a_i}v}^2 \setminus P_{v_{a_i}u}^1]$ are all bounded r -trees, and their incidence vectors belong to \mathcal{F} . Hence, we have

$$d_u = d_w = -a_{uw}.$$

Consequently, for any $e \in P_{v_{a_i}v}^1 \cup P_{v_{a_i}v}^2 \setminus (\delta(v_{a_i}) \cup \delta(v))$, and any $u \in V[P_{v_{a_i}v}^1] \cup V[P_{v_{a_i}v}^2] \setminus \{v_{a_i}, v\}$, we have

$$d_u = -a_e = \epsilon_i,$$

where $\epsilon_i \in \mathbb{R}$.

If $v \in N(v_{a_i})$, assume that $|P_{v_{a_i}v}^1| \geq 2$. We have that $G[P_{rv_{a_i}} \cup \{v_{a_i}v\}]$ and $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}^1]$ are two bounded r -trees whose incidence vectors belong to \mathcal{F} . This gives us

$$\begin{aligned} a_{v_{a_i}v} &= d(V[P_{v_{a_i}v}^1] \setminus \{v_{a_i}, v\}) + a(P_{v_{a_i}v}^1) \\ &= \epsilon_i. \end{aligned}$$

Consider an edge $uw \in E_i \setminus (\delta(v_{a_i}) \cup \delta(v))$ such that $w \notin V[P_{v_{a_i}v}^1] \cup V[P_{v_{a_i}v}^2]$. Let $P_{v_{a_i}u}$ be a path of G_r between v_{a_i} and u without passing through w . If $v \in V[P_{v_{a_i}u}]$, then $G[P_{rv_{a_i}} \cup P_{v_{a_i}u}]$ and $G[P_{rv_{a_i}} \cup P_{v_{a_i}u} \cup \{uw\}]$ are bounded r -trees whose incidence vectors belong to \mathcal{F} . Otherwise if $v \notin V[P_{v_{a_i}u}]$, among nodes in $V[P_{v_{a_i}v}^1] \cup V[P_{v_{a_i}v}^2]$, let u' be the closest node to u in the path $P_{v_{a_i}u}$. Note that we might have $u' = v_{a_i}$ or $u' = u$. Without loss of generality, suppose that $u' \in V[P_{v_{a_i}v}^1]$. Let $P_{v_{a_i}u'}^1$ denote the subpath of $P_{v_{a_i}v}^1$ between v_{a_i} and u' , and let $P_{u'u}$ denote the subpath of $P_{v_{a_i}u}$ between u' and u . Then $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}^2 \cup P_{v_{a_i}u'}^1 \cup P_{u'u}]$ and $G[P_{rv_{a_i}} \cup P_{v_{a_i}v}^2 \cup P_{v_{a_i}u'}^1 \cup P_{u'u} \cup \{uw\}]$ are two bounded r -trees whose incidence vectors belong to \mathcal{F} . As (V_i, E_i) is connected, we also have

$$d_u = d_w = -a_{uw} = \epsilon_i.$$

Now consider an edge $uw \in E_i \setminus (P_{v_{a_i}v}^1 \cup P_{v_{a_i}v}^2 \cup \delta(v_{a_i}) \cup \delta(v))$ such that $u \in V[P_{v_{a_i}v}^1]$, $w \in V[P_{v_{a_i}v}^2]$. Note that implicitly we have $|P_{v_{a_i}v}^1| \geq 2$, $|P_{v_{a_i}v}^2| \geq 2$. Let $P_{zz'}^l$ denote the subpath of $P_{v_{a_i}v}^l$ between any two nodes z and z' in $V[P_{v_{a_i}v}^l]$, $l \in \{1, 2\}$. $G[P_{rv_{a_i}} \cup P_{v_{a_i}u}^1 \cup \{uw\} \cup P_{uv}^2]$ and $G[P_{rv_{a_i}} \cup P_{v_{a_i}u}^1 \cup P_{v_{a_i}v}^2]$ are two bounded r -trees whose incidence vectors belong to \mathcal{F} .

Thus, we deduce that

$$\begin{aligned} a_{uw} &= a(P_{v_{a_i}w}^2) + d(V[P_{v_{a_i}w}^2] \setminus \{v_{a_i}, w\}) \\ &= -\epsilon_i. \end{aligned}$$

Note that here u and w are interchangeable in the proof.

Moreover if the block (V_i, E_i) is a bridge of G_r and $v_{a_i} = r$, we can always set $a_{v_{a_i}v} = -\epsilon_i$ with $\epsilon_i \in \mathbb{R}$. Thereby, we can now conclude that for any edge $e \in E_i$ and any node $u \in V_i \setminus \{v_{a_i}, v\}$ (if exists) we have

$$d_u = -a_e = \epsilon_i.$$

If $v_{a_i} = r$, by considering any path $P_{v_{a_i}v}$ of G_r one has

$$d_{v_{a_i}} + d_v = -a(P_{v_{a_i}v}) - d(V[P_{v_{a_i}v}] \setminus \{v_{a_i}, v\}) = \epsilon_i.$$

Without loss of generality, one gets

$$\begin{aligned} d_v &= \epsilon_i + \lambda, \\ d_{v_{a_i}} &= -\lambda, \end{aligned}$$

where $\lambda \in \mathbb{R}$.

If $v_{a_i} \neq r$, let $P_{rv_{a_i}}$ be a path between r and v_{a_i} of G_r and $P_{v_{a_i}v}$ a path between v_{a_i} and v of G_r and f be the edge in $P_{rv_{a_i}}$ and $\delta(v_{a_i})$. $G[P_{rv_{a_i}} \setminus \{f\}]$ and $G[P_{rv_{a_i}}] \cup P_{v_{a_i}v}$ are two bounded r -trees whose incidence vectors are in \mathcal{F} . We then deduce that

$$\begin{aligned} d_{v_{a_i}} + d_v &= -a_f - a(P_{v_{a_i}v}) - d(V[P_{v_{a_i}v}] \setminus \{v_{a_i}, v\}) \\ &= \epsilon_j + \epsilon_i. \end{aligned}$$

Without loss of generality, one can set

$$\begin{aligned} d_v &= \epsilon_i + \lambda, \\ d_{v_{a_i}} &= \epsilon_j - \lambda, \end{aligned}$$

where $\lambda \in \mathbb{R}$.

To summarize, in all cases $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ can be written as

$$-\lambda y_{v_{a_i}} + \lambda y_v + \sum_{p=1}^q \sum_{u \in V_p \setminus \{v_{a_p}\}} \epsilon_p y_u + \sum_{v_o \in O} \mu_{v_o} y_{v_o}$$

$$-\sum_{p=1}^q \sum_{e \in E_p} \epsilon_p x_e - \sum_{v_o \in O} \sum_{e_o \in \delta(v_o)} \mu_{v_o} x_{e_o} \leq 0,$$

that is,

$$\lambda(y_v - y_{v_{a_i}}) + \sum_{p=1}^q \epsilon_p (y(V_p \setminus \{v_{a_p}\}) - x(E_p)) + \sum_{v_o \in O} \mu_{v_o} (y_{v_o} - x(\delta(v_o))) \leq 0.$$

Thus it is a linear combination of $y_v - y_{v_{a_i}} \leq 0$ and equations (4.7) and (4.8). Finally, the incidence vector of any path between r and v_{a_i} of G_r is not in \mathcal{F} but in $B_{xy}(G, r, \mathbf{c})$. Hence \mathcal{F} is a facet of $B_{xy}(G, r, \mathbf{c})$. \square

We consider now the the extended subtour elimination inequalities associated with a single edge, that is (4.12). Note that for any edge $e \in \delta(v_o), v_o \in O, x_e - y_{v_o} \leq 0$ is redundant as $x(\delta(v_o)) - y_{v_o} = 0$ is valid from (4.7). Besides, for any edge $e = v_{a_i}v \in E_i \cap \delta(v_{a_i}), i \in \{1, \dots, q\}$, we deduce that $x_{v_{a_i}v} - y_{v_{a_i}} \leq 0$ is redundant as $y_v - y_{v_{a_i}} \leq 0$ is valid from (4.10). Additionally, if $E_i = \{v_{a_i}v\}$ for some $i \in \{1, \dots, q\}$, $x_{v_{a_i}v} - y_v \leq 0$ is redundant as $x_{v_{a_i}v} - y_v = 0$ is valid from (4.8). Thus, we focus on the case where $e = v_{a_i}v \in E_i$ with (V_i, E_i) being 2-connected, which is dealt by the following proposition.

Proposition 4.3.14. *Let $e = v_{a_i}v \in E_i \cap \delta(v_{a_i})$ be an edge in a 2-connected block (V_i, E_i) of G_r for some $i \in \{1, \dots, q\}$. $x_e - y_v \leq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$.*

Proof. Let $\mathcal{F} = \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : x_e - y_v = 0\}$. Assume that $\mathcal{F} \subsetneq \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} = b\} \subsetneq B_{xy}(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ is valid for $B_{xy}(G, r, \mathbf{c})$. First, the empty graph and the graph reduced to r are bounded r -trees with their incidence vectors in \mathcal{F} , and thus $b = 0$ and $d_r = 0$.

Consider an edge $uw \in E \setminus \{e\}$ with $u \notin O$ and $w \neq v$, and assume that there exists a path P_{ru} between r and u in G_r without passing through w . If $v \notin V[P_{ru}]$, then $G[P_{ru}]$ and $G[P_{ru} \cup \{uw\}]$ are two bounded r -trees whose incidence vectors are in \mathcal{F} . If $v \in V[P_{ru}]$, then by substituting the subpath between r and v of P_{ru} by rv , one gets another path P'_{ru} between r and u in G_r without passing through w . Hence, $G[P'_{ru}]$ and $G[P'_{ru} \cup \{uw\}]$ are two bounded r -trees whose incidence vectors are in \mathcal{F} . Thus, we obtain

$$d_w = -a_{uw}.$$

As it can be seen, for any edge $uw \in E_p \setminus (\delta(v_{a_p}) \cup \delta(v))$, for $p \in \{1, \dots, q\}$, u and w are

interchangeable in the proof. Consequently, for any $p \in \{1, \dots, q\}$, one can set

$$\begin{aligned} d_u &= \epsilon_p & \forall u \in V_p \setminus \{v, v_{a_p}\}, \\ a_f &= -\epsilon_p & \forall f \in E_p \setminus \{e\}, \end{aligned}$$

with $\epsilon_p \in \mathbb{R}$, and for any $v_o \in O$, one gets

$$\begin{aligned} d_{v_o} &= \mu_{v_o}, \\ a_{e_o} &= -\mu_{v_o} \quad \forall e_o \in \delta(v_o), \end{aligned}$$

with $\mu_{v_o} \in \mathbb{R}$. Furthermore, let $P_{rv_{a_i}}$ be a path between r and v_{a_i} in G_r . Notice that if $v_{a_i} = r$, $P_{rv_{a_i}} = \emptyset$. $G[P_{rv_{a_i}}]$ and $G[P_{rv_{a_i}} \cup \{e\}]$ are two bounded r -tree whose incidence vectors belong to \mathcal{F} . Hence, we have

$$a_e + d_v = 0,$$

and thus without loss of generality, we set

$$\begin{aligned} d_v &= \epsilon_i - \lambda, \\ a_e &= \lambda - \epsilon_i, \end{aligned}$$

with $\lambda \in \mathbb{R}$.

Thus, $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ can always be written as

$$\begin{aligned} & (\epsilon_i - \lambda)y_v + \sum_{p=1}^q \sum_{u \in V_p \setminus \{v, v_{a_p}\}} \epsilon_p y_u + \sum_{v_o \in O} \mu_{v_o} y_{v_o} \\ & + (\lambda - \epsilon_i)x_e - \sum_{p=1}^q \sum_{f \in E_p \setminus \{e\}} \epsilon_p x_f - \sum_{v_o \in O} \sum_{e_o \in \delta(v_o)} \mu_{v_o} x_{e_o} \leq 0, \end{aligned}$$

that is,

$$\lambda(x_e - y_v) + \sum_{p=1}^q \epsilon_p (y(V_p \setminus \{v_{a_p}\}) - x(E_p)) + \sum_{v_o \in O} \mu_{v_o} (y_{v_o} - x(\delta(v_o))) \leq 0.$$

Thus it is a linear combination of $x_e - y_v \leq 0$ and equations (4.7) and (4.8). Finally, the incidence vector of the graph reduced to e is not in \mathcal{F} but in $B_{xy}(G, r, \mathbf{c})$. Hence \mathcal{F} is a facet of $B_{xy}(G, r, \mathbf{c})$. \square

Now consider the edges in $\delta(O)$. As $y_v - y_r \leq 0$ is valid for any $v \in V \setminus \{r\}$ from (4.9), we deduce that $x_e - y_r \leq 0$ is redundant for any $e \in \delta(r)$. The following proposition

deals with those extended subtour elimination inequalities associated with an edge in $\delta(O) \setminus \delta(r)$.

Proposition 4.3.15. *Let $e = vv_o \in \delta(O)$ with $v \in V_i \setminus \{r\}$ for some $i \in \{1, \dots, q\}$. $x_e - y_v \leq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$ if and only if either $c_v \geq 3$ or $|\delta(v)| = 2$.*

Proof. Let $\mathcal{F} = \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : x_e - y_v = 0\}$. For the necessity, if $c_v = 2$ and $|\delta(v)| \geq 3$, then \mathcal{F} is a proper subset of a proper face induced by $x(\delta(v)) - c_v y_v \leq 0$.

For the sufficiency, assume that the condition is satisfied and $\mathcal{F} \subseteq \{[\mathbf{x}, \mathbf{y}] \in B_{xy}(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} = b\} \subsetneq B_{xy}(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ is valid for $B_{xy}(G, r, \mathbf{c})$. First, the empty graph is a bounded r -tree with its incidence vector in \mathcal{F} , and thus $b = 0$. As $v \neq r$, the graph reduced to r is also a bounded r -tree with its incidence vector in \mathcal{F} , which gives us $d_r = 0$.

Consider an edge $uw \in E \setminus \{e\}$ with $u \notin O$ and $w \neq v$, and assume that there exists a path P_{ru} between r and u in G_r without passing through w . If $v \notin V[P_{ru}]$, then $G[P_{ru}]$ and $G[P_{ru} \cup \{uw\}]$ are two bounded r -trees whose incidence vectors are in \mathcal{F} . If $v \in V[P_{ru}]$, then $G[P_{ru}] \cup \{e\}$ and $G[P_{ru} \cup \{e, uw\}]$ are two bounded r -trees whose incidence vectors are in \mathcal{F} . In both cases, we obtain

$$d_w = -a_{uw}.$$

Consequently, for any $p \in \{1, \dots, q\}$, one gets

$$\begin{aligned} d_u &= \epsilon_p & \forall u \in V_p \setminus \{v, v_{a_p}\}, \\ a_f &= -\epsilon_p & \forall f \in E_p, \end{aligned}$$

with $\epsilon_p \in \mathbb{R}$, and for any $u_o \in O$, one gets

$$\begin{aligned} d_{u_o} &= \mu_{u_o}, \\ a_{e_o} &= -\mu_{u_o} \quad \forall e_o \in \delta(u_o) \setminus \{e\}, \end{aligned}$$

with $\mu_{u_o} \in \mathbb{R}$.

Let P_{rv} be a path between r and v in G_r , and let e' be the edge in both P_{rv} and $\delta(v)$. Note that we have that $e' \in E_j$, $v \in V_j \setminus \{v_{a_j}\}$ for some $j \in \{1, \dots, q\}$. $G[P_{rv} \setminus \{e'\}]$ and $G[P_{rv} \cup \{e\}]$ are two bounded r -trees whose incidence vectors belong to \mathcal{F} . Hence, we have

$$a_{e'} + d_v + a_e + d_{v_o} = 0,$$

which leads to

$$d_v + a_e = \epsilon_j - \mu_{v_o}.$$

We can then set

$$\begin{aligned} d_v &= \epsilon_j - \lambda, \\ a_e &= \lambda - \mu_{v_o}, \end{aligned}$$

where $\lambda \in \mathbb{R}$.

Thus, $\mathbf{a}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \leq b$ can always be written as

$$\begin{aligned} (\epsilon_j - \lambda)y_v + (\lambda - \mu_{v_o})x_e + \sum_{p=1}^q \sum_{u \in V_p \setminus \{v, v_{a_p}\}} \epsilon_p y_u + \sum_{u_o \in O} \mu_{u_o} y_{u_o} \\ - \sum_{p=1}^q \sum_{f \in E_p} \epsilon_p x_f - \sum_{u_o \in O} \sum_{e_o \in \delta(u_o) \setminus \{e\}} \mu_{u_o} x_{e_o} \leq 0, \end{aligned}$$

that is,

$$\lambda(x_e - y_v) + \sum_{p=1}^q \epsilon_p (y(V_p \setminus \{v_{a_p}\}) - x(E_p)) + \sum_{u_o \in O} \mu_{u_o} (y_{u_o} - x(\delta(u_o))) \leq 0.$$

Thus it is a linear combination of $x_e - y_v \leq 0$ and equations (4.7) and (4.8). Finally, the incidence vector of any path between r and v of G_r is not in \mathcal{F} but in $B_{xy}(G, r, \mathbf{c})$. Hence \mathcal{F} is a facet of $B_{xy}(G, r, \mathbf{c})$. \square

For the rest cases of the extended subtour elimination inequalities, we give some necessary conditions for them to be facet-defining.

Proposition 4.3.16. *Let $e = uv \in E_i \setminus \delta(v_{a_i})$ for some $i \in \{1, \dots, q\}$. $x_e - y_v \leq 0$ defines a facet of $B_{xy}(G, r, \mathbf{c})$ only if $G[V_i \setminus \{u, v\}]$ is connected.*

Proof. Let $\mathcal{F} = \{[\mathbf{x}] \in B_{xy}(G, r, \mathbf{c}) : x_e - y_v = 0\}$. For the necessity, if $G[V_i \setminus \{u, v\}]$ is not connected, then there must exist some $S' \subsetneq V_i$ such that $\{u, v\} \subsetneq S'$ and \mathcal{F} is a proper subset of a proper face induced by $x(E[S']) - y(S' \setminus \{v\}) \leq 0$. \square

The following proposition deals with the extended subtour elimination inequalities that satisfy $|S| \geq 3$, $S \subseteq V_i$ for some $i \in \{1, \dots, q\}$.

Proposition 4.3.17. *$x(E[S]) - y(S \setminus \{v\}) \leq 0, S \subsetneq V_i, |S| \geq 3$, defines a facet of $B_{xy}(G, r, \mathbf{c})$ only if*

1. $G[S]$ is 2-connected;
2. $G[V_i \setminus S]$ is connected;
3. $v = v_{a_i}$ if $v_{a_i} \in S$.

Proof. Let $\mathcal{F} = \{[\frac{x}{y}] \in B_{xy}(G, r, \mathbf{c}) : x(E[S]) - y(S \setminus \{v\}) = 0\}$.

For the necessity, if $G[V_i \setminus S]$ is not connected, then there must exist some $S' \subsetneq V_i$ such that $S \subsetneq S'$ and \mathcal{F} is a proper subset of a proper face induced by $x(E[S']) - y(S' \setminus \{v\}) \leq 0$. If $G[S]$ is not 2-connected and $|S| \geq 3$, there must exist some $u \in S \setminus \{v\}$ and $e \in \delta(u) \cap E[S]$ such that \mathcal{F} is a proper subset of a proper face induced by $x_e - y_u \leq 0$. If $v \neq v_{a_i}$ and $v_{a_i} \in S$, since $S \neq V_i$, \mathcal{F} is then a proper subset of a proper face induced by $x(E[S]) - y(S \setminus \{v_{a_i}\}) \leq 0$. \square

We consider now the extended subtour elimination inequalities that satisfy $|S| \geq 3$, $S \setminus \{v\} \subsetneq V_i$ for some $i \in \{1, \dots, q\}$.

Proposition 4.3.18. $x(E[S]) - y(S \setminus \{v\}) \leq 0, S \setminus \{v\} \subsetneq V_i, |S| \geq 3, v \in O$, defines a facet of $B_{xy}(G, r, \mathbf{c})$ only if

1. $G[S]$ is 2-connected;
2. $G[V_i \setminus S]$ is connected.

Proof. Let $\mathcal{F} = \{[\frac{x}{y}] \in B_{xy}(G, r, \mathbf{c}) : x(E[S]) - y(S \setminus \{v\}) = 0\}$.

For the necessity, if $G[V_i \setminus S]$ is not connected, then there must exist some $S' \subsetneq V_i$ such that $S \subsetneq S'$ and \mathcal{F} is a proper subset of a proper face induced by $x(E[S']) - y(S' \setminus \{v\}) \leq 0$. If $G[S]$ is not 2-connected and $|S| \geq 3$, there must exist some $u \in S \setminus \{v\}$ and $e \in \delta(u) \cap E[S]$ such that \mathcal{F} is a proper subset of a proper face induced by $x_e - y_u \leq 0$. \square

After investigating the facets of $B_{xy}(G, r, \mathbf{c})$, in the next section, we show that it can be characterized on cactus graphs with the help of all the constraints introduced previously.

4.4 Characterization

In this section, we first show that the extended bounded r -tree polytope $B_{xy}(G, r, \mathbf{c})$ can be characterized on trees and cycles. As a result of Theorem 4.2.6, the characterization of $B_{xy}(G, r, \mathbf{c})$ on cactus graphs then immediately follows.

We prove the integrality of the formulation proposed in this section using the same approach as in Goemans [1994] and Lovász [1979]. Essentially, it is achieved by showing that any facet of $B_{xy}(G, r, \mathbf{c})$ is defined by one of the inequalities in the proposed formulation.

In particular, given a weight vector $\begin{bmatrix} \mathbf{w} \\ \mathbf{p} \end{bmatrix} \in \mathbb{R}^{E+V}$, denote by $\Gamma(\mathbf{w}, \mathbf{p})$ the set of incidence vectors of maximum bounded r -trees in G . We show that if none of the inequalities in the formulation is satisfied at equality by all the solutions in $\Gamma(\mathbf{w}, \mathbf{p})$, then $\begin{bmatrix} \mathbf{w} \\ \mathbf{p} \end{bmatrix}$ can be written as a linear combination of the rows in the coefficient matrix that correspond to the equations in the proposed formulation.

Let $P_{Cac}(G, r, \mathbf{c})$ denote the polytope defined by constraints (4.2) - (4.5) and (4.7), (4.8), that is,

$$P_{Cac}(G, r, \mathbf{c}) = \{ \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \in \mathbb{R}^{E+V} : \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \text{ satisfies (4.2) - (4.5) and (4.7), (4.8)} \}. \quad (4.18)$$

Notice that the equation (4.1) and inequalities (4.6), (4.9) and (4.10) are not considered since they can be obtained as linear combinations of the constraints (4.2), (4.7) and (4.8). Clearly polytope $P_{Cac}(G, r, \mathbf{c})$ is a formulation for $B_{xy}(G, r, \mathbf{c})$. We hereafter prove that on cactus graphs $P_{Cac}(G, r, \mathbf{c})$ is an ideal formulation, that is, $P_{Cac}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$.

4.4.1 On trees

In Section 4.3, we presented necessary and sufficient conditions for (4.2) - (4.5) to be facet-defining. After getting rid of the non-facet-defining inequalities from $P_{Cac}(G, r, \mathbf{c})$, we end up with the following linear optimization problem

$$\begin{aligned} \max \quad & \mathbf{w}^T \mathbf{x} + \mathbf{p}^T \mathbf{y} \\ \text{s.t.} \quad & x_{f_v} - y_v = 0 \quad \forall v \in V \setminus \{r\}, \end{aligned} \quad (4.19)$$

$$x_e - y_v \leq 0 \quad \forall v \in V, e \in \delta(v) \setminus \{f_v\}, \quad (4.20)$$

$$x(\delta(v)) - c_v y_v \leq 0 \quad \forall v \in V, v \text{ is not a leaf}, \quad (4.21)$$

$$y_r \leq 1, \quad (4.22)$$

$$x_e \geq 0 \quad \forall e \in E, e \text{ is a leaf edge}, \quad (4.23)$$

where f_v denotes the edge of path P_{rv} incident with v . Let $P_{Tree}(G, r, \mathbf{c}) = \{ \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \in \mathbb{R}^{E+V} : \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \text{ satisfies (4.19) - (4.23)} \}$. As a first step, we prove in the next proposition that $P_{Tree}(G, r, \mathbf{c})$ is a formulation for $B_{xy}(G, r, \mathbf{c})$ if G is a tree.

Proposition 4.4.1. *Let G be a tree. $P_{Tree}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V} = B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}$.*

Proof. As constraints (4.19)-(4.23) are a subset of the constraints defining $P_{Cac}(G, r, \mathbf{c})$, we have $B_{xy}(G, r, \mathbf{c}) \subseteq P_{Cac}(G, r, \mathbf{c}) \subseteq P_{Tree}(G, r, \mathbf{c})$. Consequently, $B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V} \subseteq P_{Tree}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}$.

Consider now a vector $\begin{bmatrix} \bar{\mathbf{x}} \\ \bar{\mathbf{y}} \end{bmatrix} \in P_{Tree}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}$ and its induced graph (U, F) . Clearly,

(U, F) is bounded by \mathbf{c} according to (4.21). Hence, we only need to prove that (U, F) is connected and rooted at r . For any node $v \in V \setminus \{r\}$, let P_{rv} be the unique path between r and v in G . If $\bar{y}_v = 1$ (or equivalently $\bar{x}_{f_v} = 1$ according to (4.19)), we deduce from (4.19) and (4.20) that $\bar{y}_u = 1$ for any $u \in \lceil v \rceil = V[P_{rv}]$ and $\bar{x}_e = 1$ for any $e \in E[\lceil v \rceil] = P_{rv}$. Hence for each node v in $U \setminus \{r\}$ and each edge f_v in F , $G[P_{rv}]$ is a subgraph of (U, F) . Therefore, (U, F) induces a bounded r -tree of G and $\lceil \frac{\bar{x}}{\bar{y}} \rceil \in B_{xy}(G, r, \mathbf{c}) \cap \mathbb{Z}^{E+V}$, which completes the proof. \square

We recall some notation introduced in Chapter 3 that is reused in the proof after. We denote $Opt_G(r)$ the optimal value of the MBrT problem on G with r being the root. Given any node $v \in V$, let $g(v)$ be the value of a maximum non-empty tree rooted at v of the subgraph $G[\lceil v \rceil]$ and bounded by \mathbf{c}^v , where the capacity vector $\mathbf{c}^v \in \mathbb{Z}_+^{\lceil v \rceil}$ satisfies $c_v^v = c_v - 1$ if $v \neq r$, $c_v^v = c_v$ if $v = r$, and $c_s^v = c_s$, for $s \in \lceil v \rceil \setminus v$. In other words,

$$g(v) = \max\{f(T) : T \text{ is tree of } G[\lceil v \rceil] \text{ with } v \in V(T) \text{ and bounded by } \mathbf{c}^v\}.$$

Correspondingly, for $v \in V \setminus \{r\}$, let $h(f_v) = \max\{0, w_{f_v} + g(v)\}$ be the actual gain associated with considering f_v in the solution.

Proposition 4.4.2. *Let G be a tree. $P_{Tree}(G, r, \mathbf{c})$ is integral.*

Proof. Let $A^{\leq} \lceil \frac{\mathbf{x}}{\mathbf{y}} \rceil \leq \mathbf{b}^{\leq}$ denote the system composed of inequalities in (4.20)-(4.23), and $A^= \lceil \frac{\mathbf{x}}{\mathbf{y}} \rceil = \mathbf{b}^=$ the system composed of equations in (4.19).

Let $\lceil \frac{\mathbf{w}}{\mathbf{p}} \rceil \in \mathbb{R}^{E+V}$ be a weight vector such that none of the inequalities from (4.20)-(4.23) is satisfied at equality by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$.

First of all, for any $v \in V \setminus \{r\}$, if $w_{f_v} + g(v) < 0$, then $x_e = 0$ is satisfied for any $e \in E[\lceil v \rceil] \cup \{f_v\}$ by all solutions in Γ . Thus we have

$$w_{f_v} + g(v) \geq 0 \quad \forall v \in V \setminus \{r\}. \quad (4.24)$$

As an immediate result, one has $h(f_v) = w_{f_v} + g(v)$ for all $v \in V \setminus \{r\}$.

It can be seen that $\mathbf{0} \in \Gamma(\mathbf{w}, \mathbf{p})$, as it is the only integral feasible solution with $y_r < 1$. Hence, we have $Opt_G(r) = \max\{0, g(r)\} = 0$, and thus $g(r) \leq 0$ and $p_r \leq 0$.

If $p_r < 0$, there must exist some $v \in N(r)$ with $h(f_v) > 0$, as otherwise $x_e = 0$ is satisfied for any $e \in E$ by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. Let $S = \{v \in N(r) : h(f_v) > 0\}$. If $|S| < c_r$, then $x_{f_v} - y_r = 0$ is satisfied for any $v \in S$ by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. If $|S| \geq c_r$, then $x(\delta(r)) - c_r y_r = 0$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. Thus we have

$$p_r = 0.$$

If there exists $v \in N(r)$ with $h(f_v) > 0$, one has $Opt_G(r) \geq h(f_v) + p_r > 0$ which contradicts against $Opt_G(r) = 0$. Thus we have that, $w_{f_v} + g(v) \leq 0$, for all $v \in N(r)$. Combining this with (4.24) gives us

$$w_{f_v} + g(v) = 0 \quad \forall v \in N(r).$$

We claim that given $v \in V \setminus \{r\}$, if $w_{f_v} + g(v) = 0$ then $w_{f_v} + p_v = 0$ holds and so does $w_{f_u} + g(u) = 0$ for any node $u \in N(v) \cap [v]$.

From $w_{f_v} + g(v) = 0$ and $p_v \leq g(v)$ one has $w_{f_v} + p_v \leq 0$. If $w_{f_v} + p_v < 0$, there exists some $u \in N(v) \cap [v]$ with $g(f_u) > 0$, since otherwise we would have $w_{f_v} + g(v) < 0$. Let $S = \{u \in N(v) \cap [v] : h(f_u) = w_{f_u} + g(u) > 0\}$. If $|S| < c_v^v$, then $x_{f_u} - y_v = 0$ is satisfied for any $u \in S$ by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. If $|S| \geq c_v^v$, then $x(\delta(v)) - c_v y_v = 0$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. Thus, we have $w_{f_v} + p_v = 0$. Moreover, as $w_{f_v} + g(v) = 0$ and $w_{f_v} + p_v = 0$, there does not exist $u \in N(v) \cap [v]$ with $w_{f_u} + g(u) > 0$, which leads to $w_{f_u} + g(u) = 0$.

By induction one can deduce that

$$w_{f_v} + p_v = 0 \quad \forall v \in V \setminus \{r\}.$$

Therefore, we have $[\frac{w}{p}] = \boldsymbol{\mu}^T A^=$ for some real vector $\boldsymbol{\mu}$, and hence $P_{Tree}(G, r, \mathbf{c})$ is an integral polytope. \square

Immediately, we obtain $P_{Tree}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$. Furthermore, as $P_{Cac}(G, r, \mathbf{c}) \subseteq P_{Tree}(G, r, \mathbf{c})$ and both polytopes are formulations for $B_{xy}(G, r, \mathbf{c})$, the follow theorem holds.

Theorem 4.4.3. *Let G be a tree. $P_{Cac}(G, r, \mathbf{c}) = P_{Tree}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$.* \square

For the case of trees, the nodes in O have little impact on the characterization of $B_{xy}(G, r, \mathbf{c})$, as they can only be the leaves and the associated extended capacity inequalities are redundant. Nonetheless, they play a significant role in the characterization of $B_{xy}(G, r, \mathbf{c})$ on cycles.

4.4.2 On cycles

This section presents the proof of the characterization of $B_{xy}(G, r, \mathbf{c})$ on cycles. Since in a cycle all nodes have degree of 2, hence the extended capacity inequality for any node with capacity 2 is redundant. Moreover, according to Assumption 2.2.9, we have either $O = \emptyset$ or $|O| = 1$. Thus, the characterization on cycles needs be categorized into four different situations according to the capacity of r and the existence of O .

For the case with $c_v \geq 2$ for all $v \in V$, all the r -trees of G are bounded. Hence $B_{xy}(G, r, \mathbf{c}) = R_{xy}(G, r)$. According to Theorem 2.2.1 (proved by Goemans [1994]), one has $P_{xy}(G, r) = R_{xy}(G, r)$ on series-parallel graphs, which is sufficient for the characterization of $B_{xy}(G, r, \mathbf{c})$ in this case, that is, $P_{xy}(G, r) = R_{xy}(G, r) = B_{xy}(G, r, \mathbf{c})$. Furthermore, as $P_{Cac}(G, r, \mathbf{c})$ contains all the constraints in $P_{xy}(G, r, \mathbf{c})$ and thus in $P_{xy}(G, r)$, we therefore have the following proposition.

Proposition 4.4.4. *Let G be a cycle with $c_v \geq 2$ for all $v \in V$. $P_{Cac}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$.*

Now consider the case where $c_r = 1$, and $c_v \geq 2$, for $v \in V \setminus \{r\}$. In this case, $G_r = G$ and it contains only 1 block. Hence (4.8) reduces to $x(E) - y(V \setminus \{r\}) = 0$, that is, (4.1). Moreover, as all capacities are redundant except for r , we do not have any equation form (4.7) as $O = \emptyset$, and (4.3) reduces to one inequality, that is,

$$x(\delta(r)) - y_r \leq 0. \quad (4.25)$$

Therefore, the system defining $P_{Cac}(G, r, \mathbf{c})$ can be rewritten as (4.1), (4.2), (4.4), (4.5) and (4.25).

Proposition 4.4.5. *Let G be a cycle with $c_r = 1$ and $c_v \geq 2$ for all $v \in V \setminus \{r\}$. $P_{Cac}(G, r, \mathbf{c})$ is integral.*

Proof. Let $A^{\leq} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \leq \mathbf{b}^{\leq}$ denote the system composed of inequalities in (4.2), (4.4), (4.5) and (4.25), and $A^{\dagger} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \mathbf{b}^{\dagger}$ the system composed of equations in (4.1).

Let $\begin{bmatrix} \mathbf{w} \\ \mathbf{p} \end{bmatrix} \in \mathbb{R}^{E+V}$ be a weight vector such that none of (4.2), (4.4), (4.5) and (4.25) is satisfied at equality by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$.

It can be seen that $\mathbf{0} \in \Gamma(\mathbf{w}, \mathbf{p})$, as it is the only feasible solution with $y_r < 1$. Hence, the optimal value must equal to 0, and as a result, one also has $p_r \leq 0$. Additionally, as $x(\delta(r)) - y_r \leq 0$ is not satisfied at equality by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$, it can be deduced that $p_r = 0$.

It is worth noting that each non-empty bounded r -tree in G is a rv -path for some $v \in V$. According to the assumption, there does not exist $e \in E$ and $v \in V \setminus \{r\}$ such that $e \in \delta(v)$ and $x_e - y_v = 0$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. Then there must exist a solution in $\Gamma(\mathbf{w}, \mathbf{p})$ corresponding to the rv -path P_{rv} that does not contain e . Consequently, any rv -path P_{rv} induces a solution in $\Gamma(\mathbf{w}, \mathbf{p})$ for any $v \in V \setminus \{r\}$. Thus one can deduce that $p_v + w_e = 0$ for any $e \in \delta(v)$, $v \in V \setminus \{r\}$. Due to the connectivity of G , we have that $p_v + w_e = 0$ for any $v \in V \setminus \{r\}$ and any $e \in E$. Therefore, $\begin{bmatrix} \mathbf{w} \\ \mathbf{p} \end{bmatrix} = \boldsymbol{\mu}^T A^{\dagger}$, for some real vector $\boldsymbol{\mu}$, and thus $P_{Cac}(G, r, \mathbf{c})$ is integral. \square

Consider now the case where $c_r \geq 2$ and $O = \{v_o\}$. It can be noted that G_r contains $|E| - 2$ blocks with each one of them being a bridge. Hence (4.8) reduces to the following equations

$$x_{f_v} - y_v = 0 \quad \forall v \in V \setminus \{r, v_o\}, \quad (4.26)$$

where f_v denotes the edge incident with $v \in V \setminus \{r, v_o\}$ and in the path between r and v without passing through v_o . In addition, (4.7) reduces to the following equation

$$x(\delta(v_o)) = y_{v_o}. \quad (4.27)$$

Besides, all inequalities in (4.3) are redundant. Thus, the system defining $P_{Cac}(G, r, \mathbf{c})$ can be rewritten as (4.2), (4.4), (4.5), (4.26) and (4.27). We show in this case $P_{Cac}(G, r, \mathbf{c})$ is integral as stated in the following proposition.

Proposition 4.4.6. *Let G be a cycle with $c_r \geq 2$ and $O = \{v_o\}$. $P_{Cac}(G, r, \mathbf{c})$ is integral.*

Proof. Let $A^\leq \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \leq \mathbf{b}^\leq$ denote the system composed of inequalities in (4.2), (4.4) and (4.5), and $A^\dagger \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \mathbf{b}^\dagger$ the system composed of equations in (4.26) and (4.27).

Let $\begin{bmatrix} \mathbf{w} \\ \mathbf{p} \end{bmatrix} \in \mathbb{R}^{E+V}$ be a weight vector such that none of (4.2), (4.4) and (4.5) is satisfied at equality by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. It can be seen that $\mathbf{0} \in \Gamma(\mathbf{w}, \mathbf{p})$, as it is the only feasible solution with $y_r < 1$. Hence, the optimal value equals to 0, and as a result, one also has $p_r \leq 0$.

If there exists $v \in V \setminus \{r, v_o\}$ with $w_{f_v} + p_v > 0$ and without loss of generality let $f_v = uv$, then $x_{f_v} - y_u = 0$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. Thus, we have

$$w_{f_v} + p_v \leq 0 \quad \forall v \in V \setminus \{r, v_o\}.$$

If $p_r < 0$, there must exist $e \in \delta(v_o)$ with $w_e + p_{v_o} > 0$ and $f(G[P_{re}]) = 0$ since otherwise $x_e = 0$ for $e \in E$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. We then deduce that $y_{v_o} - y_r = 0$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. Therefore, one obtains

$$p_r = 0.$$

Similarly, if there exists $w_{f_v} + p_v < 0$ for some $v \in V \setminus \{r, v_o\}$, as $x_{f_v} = 0$ is not satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$, there must exist a path P_{rv_o} with $f_v \in P_{rv_o}$ and $f(G[P_{rv_o}]) = 0$. One therefore has that $y_{v_o} - y_r = 0$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. Hence, we have

$$w_{f_v} + p_v = 0 \quad \forall v \in V \setminus \{r, v_o\}.$$

For $e \in \delta(v_o)$, as $x_e > 0$ holds for some solution in $\Gamma(\mathbf{w}, \mathbf{p})$, one has $w_e + p_{v_o} \geq 0$. And as one has $p_r = 0$ and $w_{f_v} + p_v = 0$ for any $v \in V \setminus \{r, v_o\}$, one can also deduce $w_e + p_{v_o} \leq 0$, which gives us $w_e + p_{v_o} = 0$.

To summarize, $[\frac{\mathbf{w}}{\mathbf{p}}] = \boldsymbol{\mu}^T A^=$ for some real vector $\boldsymbol{\mu}$, and thus $P_{Cac}(G, r, \mathbf{c})$ is integral. \square

Consider now the case where $c_r = 1$ and $O = \{v_o\}$. Similar to the previous case, (4.8) reduces to (4.26), and (4.7) reduces to (4.27).

Furthermore, (4.3) reduces to

$$x(\delta(r)) - y_r \leq 0. \quad (4.28)$$

Therefore, the system defining $P_{Cac}(G, r, \mathbf{c})$ can be rewritten as (4.2), (4.4), (4.5), (4.26), (4.27) and (4.28).

Proposition 4.4.7. *Let G be a cycle with $c_r = 1$ and $O = \{v_o\}$. $P_{Cac}(G, r, \mathbf{c})$ is integral.*

Proof. Let $A^{\leq}[\frac{\mathbf{x}}{\mathbf{y}}] \leq \mathbf{b}^{\leq}$ denote the system composed of inequalities in (4.2), (4.4), (4.5) and (4.28), and $A^=[\frac{\mathbf{x}}{\mathbf{y}}] = \mathbf{b}^=$ the system composed of equations in (4.26) and (4.27).

Let $[\frac{\mathbf{w}}{\mathbf{p}}] \in \mathbb{R}^{E+V}$ be a weight vector such that none of (4.2), (4.4), (4.5) and (4.28) is satisfied at equality by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$. It can be seen that $\mathbf{0} \in \Gamma(\mathbf{w}, \mathbf{p})$, as it is the only feasible solution with $y_r < 1$. Hence, the optimal value equals to 0. Moreover, as the graph reduced to r is the only solution that does not satisfy $x(\delta(r)) - y_r = 0$, its incidence vector is in also $\Gamma(\mathbf{w}, \mathbf{p})$. Hence, we have

$$p_r = 0.$$

It is worth mentioning that each non-empty bounded r -tree is a path between r and some node v such that it does not contain v_o as an inner node. For any edge $e \in E$, let P_{re} be the path between r and e such that it does not contain v_o as an inner node, and without loss of generality let $e = uv$ and $v \neq r$ is one end of P_{re} . As neither $x_e = 0$ nor $x_e - y_u = 0$ is satisfied by all solutions in $\Gamma(\mathbf{w}, \mathbf{p})$, then we deduce that P_{re} induces a maximum bounded r -tree of G for each $e \in E$. Thus we have

$$\begin{aligned} w_{f_v} + p_v &= 0 \quad \forall v \in V \setminus \{r, v_o\}, \\ w_e + p_{v_o} &= 0 \quad \forall e \in \delta(v_o), \end{aligned}$$

where f_v denotes the edge incident with v and in the path between r and v without passing through v_o .

Therefore, $[\frac{\mathbf{w}}{\mathbf{p}}] = \boldsymbol{\mu}^T A^=$ for some real vector $\boldsymbol{\mu}$, and thus $P_{Cac}(G, r, \mathbf{c})$ is integral. \square

We have thereby proved that $P_{Cac}(G, r, \mathbf{c})$ is an ideal formulation for all cases on cycles, as stated in the following theorem.

Theorem 4.4.8. *Let G be a cycle. $P_{Cac}(G, r, \mathbf{c})$ is integral.* \square

4.4.3 On cactus graphs

Combining the characterization on trees and cycles with the decomposition through 1-sum, the characterization of $B_{xy}(G, r, \mathbf{c})$ on cactus graphs can thereby be obtained.

Theorem 4.4.9. *Let G be a cactus graph. $P_{Cac}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$.*

Proof. Let G_1 and G_2 be two graphs such that their 1-sum at the node v_a is G , and $P_{Cac}(G_i, r_i, \mathbf{c}^i) = B_{xy}(G_i, r_i, \mathbf{c}^i)$ for $i = 1, 2$. Following the notation in Section 4.2, let

$$P_C(G, r, \mathbf{c}) = \left\{ \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} : \begin{bmatrix} \mathbf{x}^1 \\ \mathbf{y}^1 \end{bmatrix} \in B_{xy}(G_1, r_1, \mathbf{c}^1), \begin{bmatrix} \mathbf{x}^2 \\ \mathbf{y}^2 \end{bmatrix} \in B_{xy}(G_2, r_2, \mathbf{c}^2), \right. \\ \left. x(\delta(v_a)) - c_{v_a} y_{v_a} \leq 0 \right\}.$$

According to Theorem 4.2.6, one has $P_C(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$. Moreover, $x(\delta(v_a)) - c_{v_a} y_{v_a} \leq 0$ and all the constraints in (4.2) - (4.5) and (4.7), (4.8) with respect to $P_{Cac}(G_i, r_i, \mathbf{c}^i)$, $i = 1, 2$, are also included in the formulation with respect to $P_{Cac}(G, r, \mathbf{c})$. Hence, we have $P_{Cac}(G, r, \mathbf{c}) \subseteq P_C(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$. Finally, as $B_{xy}(G, r, \mathbf{c}) \subseteq P_{Cac}(G, r, \mathbf{c})$, we thus obtain

$$P_{Cac}(G, r, \mathbf{c}) = P_C(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c}).$$

Therefore, $P_{Cac}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c})$ holds for any graph G composed of two subgraphs which are either trees or cycles and separated by an articulation node. This deduction can be repeated as many times to obtain the same result for any graph composed of as many tree- or cycle-components separated by articulation nodes, which is also known as a cactus graph. Therefore, the theorem holds. \square

4.5 Conclusion

In this chapter, we have explored several aspects of the extended bounded r -tree polytope $B_{xy}(G, r, \mathbf{c})$. It has been shown that its dimension is related to the unit-capacity node set O and the blocks of the graph. We also showed that the polytope can be decomposed with respect to articulation nodes through 1-sum.

We introduce several sets of new valid equations and inequalities, along with the necessary and sufficient conditions for all the known inequalities be facet-defining. Note that for the extended subtour elimination inequalities only a subset of the induced facets have been identified, whereas for the rest we gave some properties and necessary conditions. The aforementioned valid constraints allow us to characterize $B_{xy}(G, r, \mathbf{c})$ on trees and cycles, and combining with the decomposition through 1-sum, $B_{xy}(G, r, \mathbf{c})$ can thus be characterized on cactus graphs.

On the other hand, an r -tree (or a bounded r -tree) can also be described alone by its edge set, while it cannot be decided if only its node set is given. Therefore, in the following two chapters, the r -tree polytope $R_x(G, r)$ and bounded r -tree polytope $B_x(G, r, \mathbf{c})$ are discussed, where only the edge-indexed variables are considered. We examine the same aspects on these two polytopes as we have done in this chapter, and show that we can also characterize them on trees and cycles.

Chapter 5

Polyhedral study on r-Tree Polytope

Recall that given a graph $G = (V, E)$, a node $r \in V$, an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$, and a node-price vector $\mathbf{p} \in \mathbb{R}^V$, the MrT problem consists of finding an r -tree T of G with maximum value $f(T) = w(E(T)) + p(V(T))$.

As a relaxation of the MBrT problem, the MrT problem has been studied by Goemans [1994]. He proposed a formulation for $R_{xy}(G, r)$, presented results concerning its facets, and gave a complete description of it on series-parallel graphs. Nonetheless, $R_x(G, r)$ has not been studied yet.

This chapter presents results on $R_x(G, r)$ which considers only the edge-indexed variables. We start with the following formulation that has been introduced in Chapter 2.

$$x_e - x(\delta(S)) \leq 0 \quad \forall e \in E[S], S \subseteq V \setminus \{r\}, \quad (5.1)$$

$$x(E[S]) \leq |S| - 1 \quad \forall S \subseteq V, |S| \geq 3, \quad (5.2)$$

$$x_e \leq 1 \quad \forall e \in E, \quad (5.3)$$

$$x_e \geq 0 \quad \forall e \in E. \quad (5.4)$$

According to Assumption 2.2.9, G is connected. With that said, we first show that $R_x(G, r)$ is full-dimensional and present some results on the known facets. We then introduce two sets of newly discovered facet-defining inequalities aside from those present in the formulation, and give necessary and sufficient conditions for all the aforementioned inequalities. Finally, we give some counter examples that suggest the infeasibility of the decomposition through 1-sum, as opposed to the case of $B_{xy}(G, r, \mathbf{c})$.

5.1 Dimension

We propose first the following lemma, which is significant for the dimension proof and facial proof of not only $R_x(G, r)$ but also $B_x(G, r, \mathbf{c})$.

Lemma 5.1.1. *Given a connected graph $G = (V, E)$ with $r \in V$, let F be any nonempty subset of edges of G . For any $e \in F$, consider an re -path P_{re} in G having as few edges as possible. The set $\{\mathbf{x}^{P_{re}} : e \in F\}$ is affinely independent.*

Proof. From the assumption one has that for any two distinct edges $e_1, e_2 \in F$, $e_1 \notin P_{re_2}$ holds if $|P_{re_1}| \geq |P_{re_2}|$.

Suppose that there exists a non-zero vector $\boldsymbol{\lambda} \in \mathbb{R}^F$ such that

$$\sum_{e \in F} (\lambda_e \mathbf{x}^{P_{re}}) = \mathbf{0}.$$

Let $F_+ = \{e \in F : \lambda_e \neq 0\}$. Without loss of generality, let e_m be an edge in F_+ such that $|P_{re_m}| \geq |P_{re}|$ holds for any $e \in F_+$. Since $|P_{re_m}| \geq |P_{re}|$, one has $e_m \notin P_{re}$ (or $x_{e_m}^{P_{re}} = 0$) for any $e \in F_+ \setminus \{e_m\}$. Therefore, $\lambda_{e_m} = 0$, which forms a contradiction with the assumption. One can thus conclude that the set $\{\mathbf{x}^{P_{re}} : e \in F\}$ is linearly independent, and thus affinely independent. \square

With Lemma 5.1.1, the dimension of $R_x(G, r)$ is given as in the following theorem.

Theorem 5.1.2. $R_x(G, r)$ is full-dimensional, that is,

$$\dim R_x(G, r) = |E|. \quad (5.5)$$

Proof. According to Lemma 5.1.1, one has $|E|$ non-zero linearly independent vectors in $R_x(G, r)$, each of which induces an re -path for a distinct $e \in E$. Combining it with the zero vector, one has $|E|+1$ affinely independent vectors in $R_x(G, r)$. Hence $\dim R_x(G, r) = |E|$. \square

After showing that $R_x(G, r)$ is full-dimensional under Assumption 2.2.9, in the next section we present a facial study of $R_x(G, r)$ for each set of inequalities presented in the formulation.

5.2 Facets

Before diving into the facial discussion of $R_x(G, r)$, it is worth mentioning that for both $R_x(G, r)$ and $B_x(G, r, \mathbf{c})$ we have discovered a few universal criteria that any facet-defining

inequality should follow. Note that the criteria for $R_x(G, r)$ can actually be extended to similar ones for $B_x(G, r, \mathbf{c})$. The following proposition shows one of them related to the bridges in the graph.

Proposition 5.2.1. *Given a valid inequality $\mathbf{a}^T \mathbf{x} \leq b$ for $R_x(G, r)$ with $a_e \geq 0, \mathbf{a} \neq \mathbf{0}, b > 0$, let $E^+ := \{e \in E \mid a_e > 0\}$. If $r \notin V[E^+]$, then $\mathbf{a}^T \mathbf{x} \leq b$ defines a facet of $R_x(G, r)$ only if there does not exist a bridge $e_b \in E$ between r and $V[E^+]$.*

Proof. Suppose there exists a bridge $e_b \in E$ between r and $V[E^+]$. For any r -tree G_F of G that satisfies $\mathbf{a}^T \mathbf{x} = b$, we claim that $e_b \in F$ holds. Assume otherwise that $e_b \notin F$. From $b > 0$ and $x^F(E^+) = b$, one deduce that $x_e^F = 1$ for some edge $e \in E^+$. Since e_b is a bridge between r and e , G_F does not contain any path between r and e . Therefore it contradicts with the assumption of G_F being an r -tree. \square

The facet-defining conditions of $x_e \geq 0$ for $R_x(G, r)$ is presented as follows.

Proposition 5.2.2. *Let $e \in E$. $x_e \geq 0$ defines a facet of $R_x(G, r)$ if and only if $G[E \setminus \{e\}]$ is connected.*

Proof. It can be trivially seen that

$$\dim\{\mathbf{x} \in R_x(G, r) : x_e = 0\} = \dim R_x(G[E \setminus \{e\}], r).$$

According to Theorem 5.1.2, $R_x(G[E \setminus \{e\}], r)$ is full-dimensional if and only if $G[E \setminus \{e\}]$ is connected. Therefore we have that $\dim\{\mathbf{x} \in R_x(G, r) : x_e = 0\} = \dim R_x(G[E \setminus \{e\}], r) = \dim R_x(G, r) - 1$ holds if and only if $G[E \setminus \{e\}]$ is connected. In other words, $x_e \geq 0$ is facet-defining if and only if $G[E \setminus \{e\}]$ is connected. \square

Notably, here we use $G[E \setminus \{e\}]$ instead of $G \setminus e$ intentionally, since $G \setminus e$ might contain some isolated node while $G[E \setminus \{e\}]$ being connected, in which case $x_e \geq 0$ is still facet-defining.

For the upper bound inequalities (5.3), we split them into two cases, for edges in $\delta(r)$ and edges not in $\delta(r)$. For the former case, we show that the associated upper bound inequality is always facet-defining.

Proposition 5.2.3. *Let $e \in \delta(r)$. $x_e \leq 1$ defines a facet of $R_x(G, r)$.*

Proof. Suppose $\mathcal{F} = \{\mathbf{x} \in R_x(G, r) : x_e = 1\} \subseteq \{\mathbf{x} \in R_x(G, r) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq R_x(G, r)$ with $\mathbf{a}^T \mathbf{x} \leq b$ being a valid inequality for $R_x(G, r)$.

As the r -tree reduced to e satisfies $x_e = 1$, we clearly have $a_e = b$. For any edge $e' \in E \setminus \{e\}$, let $P_{re'}$ be a path between r and e' . Let $e = rv$. If $e \notin P_{re'}$ and $v \in V[P_{re'}]$, there must exist another path $P'_{re'}$ with $e \in P'_{re'}$, in which case, we replace $P_{re'}$ by $P'_{re'}$.

Thus $P_{re'} \cup \{e\}$ and $P_{re'} \setminus \{e'\} \cup \{e\}$ both induce r -trees that satisfy $x_e = 1$. Hence we have $a_{e'} = 0$ for any $e' \in E \setminus \{e\}$. As a result, $\mathbf{a}^T \mathbf{x} \leq b$ can be written as $bx_e \leq b$. Finally as $\mathbf{0} \in R_x(G, r) \setminus \mathcal{F}$, we can conclude that \mathcal{F} is a facet of $R_x(G, r)$. \square

For the edges not incident with r , the general criteria stated in Proposition 5.2.1 should be applied. The necessary and sufficient conditions for the associated upper bound inequality to be facet-defining are presented in the following proposition.

Proposition 5.2.4. *Let $e = uv \in E \setminus \delta(r)$. $x_e \leq 1$ defines a facet of $R_x(G, r)$ if and only if the following two conditions are satisfied*

1. *there exist two edge-disjoint paths between r and e ;*
2. *there does not exist $w \in N(u) \cap N(v)$ such that removing uw and vw from G disconnects r and e .*

Proof. Let $\mathcal{F} = \{\mathbf{x} \in R_x(G, r) : x_e = 1\}$. For the necessity, if there is a bridge e_b in G separating r and e , any r -tree containing e must also contain e_b . Thus, \mathcal{F} is a proper subset of the proper face induced by $x_{e_b} \leq 1$. If there exists a node $w \in N(u) \cap N(v)$ such that removing uw and vw from G disconnects r and e , then \mathcal{F} is a proper subset of the proper face induced by $x(E[S]) \leq |S| - 1$ with $S = \{u, v, w\}$.

For the sufficiency, we now suppose that the conditions in the proposition are satisfied and $\mathcal{F} \subseteq \{\mathbf{x} \in R_x(G, r) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq R_x(G, r)$ with $\mathbf{a}^T \mathbf{x} \leq b$ being a valid inequality for $R_x(G, r)$.

Assume that there exists a path P_{re} with $u', v' \notin V[P_{re}]$ for $e' = u'v' \in E \setminus \{e\}$. There must exist a path $P_{we'}$ between some node $w \in V[P_{re}]$ and e' such that $P_{re} \cup P_{we'}$ induces an acyclic graph. Thus from the r -trees induced by $P_{re} \cup P_{we'}$ and $P_{re} \cup P_{we'} \setminus \{e'\}$, one gets $a_{e'} = 0$.

Assume that for $e' = u'v' \in E \setminus \{e\}$ there exists a path P_{re} with $u' \in V[P_{re}], v' \notin V[P_{re}]$, $P_{re} \cup \{e'\}$ induces an acyclic graph. Thus we have $a_{e'} = 0$ from the r -trees induced by P_{re} and $P_{re} \cup \{e'\}$.

Assume now otherwise that for $e' = u'v' \in E \setminus \{e\}$ any path between r and e contains u' and v' . Clearly, there exists a path P_{re} with $e' \in P_{re}$, and another path P'_{re} such that $P_{re} \cap P'_{re} = \{e\}$ and $u', v' \in V[P'_{re}]$. Let $P'_{u'v'}$ be the subpath of P'_{re} between u' and v' . The conditions in the proposition guarantee that there exists such P'_{re} with $|P'_{u'v'}| \geq 2$. Both P_{re} and $P_{re} \setminus \{e'\} \cup P'_{u'v'}$ induce r -trees containing e , which leads to $a_{e'} = a(P'_{u'v'})$. Moreover, for any edge $f \in P'_{u'v'}$, $P_{re} \cup P'_{u'v'} \setminus \{f\}$ also induces an r -tree containing e . Thus $a_{e'} = a_f = 0$.

To summarize, one has that $a_{e'} = 0$ for any $e' \in E \setminus \{e\}$. Finally, from the r -tree reduce to any path between r and e , one then deduce that $a_e = b$. Therefore, $\mathbf{a}^T \mathbf{x} = b$ can

be written as $bx_e = b$. As \mathcal{F} is obviously different from $R_x(G, r)$, hence it is a facet of $R_x(G, r)$. \square

Given $e \in E[S]$ with $S \subseteq V \setminus \{r\}$, the necessary and sufficient conditions for the connectivity inequalities to be facet-defining are presented in the following proposition.

Proposition 5.2.5. *Let $e = uv \in E[S]$, $S \subseteq V \setminus \{r\}$. $x_e - x(\delta(S)) \leq 0$ defines a facet of $R_x(G, r)$ if and only if the following four conditions are satisfied*

1. $G[S]$ is connected;
2. $G[\overline{S}]$ is connected;
3. there does not exist an edge $e_b \in E[S] \setminus \{e\}$ such that removing e_b from G disconnects r and e ;
4. there does not exist a node $w \in S \cap N(u) \cap N(v)$ such that removing uw, vw from G disconnects r and e .

Proof. Let $\mathcal{F} = \{\mathbf{x} \in R_x(G, r) : x_e - x(\delta(S)) = 0\}$. For the necessity, if either $G[S]$ or $G[\overline{S}]$ is not connected, there must exist some $e' \in E$ such that \mathcal{F} is a proper subset of the proper face induced by $x_{e'} \geq 0$.

If there exists an edge $e_b \in E[S] \setminus \{e\}$ such that removing e_b from G disconnects e and r , then \mathcal{F} is a proper subset of the proper face induced by $x_{e_b} - x(\delta(S)) \leq 0$.

Assume that there exists such node $w \in S \setminus \{u, v\}$ that removing uw, vw disconnects r and e . In this case, \mathcal{F} is a proper subset of the proper face induced by $x(E[\{u, v, w\}]) - (|\{u, v, w\}| - 1)x(\delta(S)) \leq 0$, which will be introduced as a new set of valid inequalities in the next section.

For the sufficiency, we suppose that all the conditions listed in the proposition are satisfied, and $\mathcal{F} \subseteq \{\mathbf{x} \in R_x(G, r) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq R_x(G, r)$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $R_x(G, r)$. First of all, we have $b = 0$ from $\mathbf{0} \in \mathcal{F}$. For any edge f in $E[\overline{S}]$, as the incidence vector of any r -tree in $G[\overline{S}]$ is in \mathcal{F} and $G[\overline{S}]$ is connected, thus one has $a_f = 0$.

Consider any edge $f = uv \in E[S] \setminus \{e\}$ such that u is not an articulation node that separates r and e . There exists a path P_{re} between r and e such that $|P_{re} \cap \delta(S)| = 1$ and it does not pass through u and thus does not contain f , and there exist a path P_f between some node in $V[P_{re}]$ and f such that $P_{re} \cup P_f$ induces an acyclic graph. It can be deduced from the two r -trees induced by $P_{re} \cup P_f$ and $P_{re} \cup P_f \setminus \{f\}$ that $a_f = 0$.

Consider now an edge $f = uv \in E[S] \setminus \{e\}$ such that u and v are both articulation nodes that separate r and e . According to the conditions, f is not a bridge that separates r and e . Then there exists a path P_{re} between r and e such that $e \notin P_{re}$ and $|P_{re} \cap \delta(S)| = 1$.

Let P_{uv} be the subpath of P_{re} between u and v . P_{re} and $P_{re} \cup \{f\} \setminus P_{uv}$ induce two r -trees whose incidence vectors are in \mathcal{F} , which leads to $a_f = a(P_{uv}) = 0$.

Now by considering $P_{rf} \cup P_{ue}$ for any $f \in \delta(S)$, where P_{rf} is a path between r and f with $|P_{rf} \cap \delta(S)| = 1$ and P_{ue} is a path of $G[S]$ between $u \in S$ and e with $f \in \delta(S) \cap \delta(u)$, we have $a_f + a_e = 0$.

Therefore, $\mathbf{a}^T \mathbf{x} \leq b$ is rewritten as $\lambda(x_e - x(\delta(S))) \leq 0$ with $\lambda \in \mathbb{R}$. Finally, we have $\mathcal{F} \neq R_x(G, r)$ considering any path between r and an edge in $\delta(S)$. One can then conclude that \mathcal{F} is a facet of $R_x(G, r)$. \square

Theorem 5.2.6. *Let $S \subseteq V, |S| \geq 3$. $x(E[S]) \leq |S| - 1$ defines a facet of $R_x(G, r)$ if and only if the following two conditions are satisfied*

1. $r \in S$;
2. $G[S]$ is 2-connected.

Proof. Let $\mathcal{F} = \{\mathbf{x} \in R_x(G, r) : x(E[S]) = |S| - 1\}$. For the necessity, if $r \notin S$, then \mathcal{F} is a proper subset of the proper face induced by $x(E[S]) - (|S| - 1)x(\delta(S)) \leq 0$. If $G[S]$ has multiple blocks, then \mathcal{F} is a proper subset of the proper face induced by $x([F_i]) \leq |V[F_i]| - 1$, for any block $(V[F_i], F_i)$ of $G[S]$.

For the sufficiency, assume that $\mathcal{F} \subseteq \{\mathbf{x} \in R_x(G, r) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq R_x(G, r)$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $R_x(G, r)$.

For any edge $e \in E \setminus E[S]$, one can construct an r -tree that satisfies $x(E[S]) = |S| - 1$ as follows. Let $F \subset E[S]$ be any edge set that induces an r -tree with $|F| = |S| - 1$. As G is connected, there must exist a path P_{ve} from some $v \in S$ to e such that $V[P_{ve}] \cap S = \{v\}$. Then $P_{ve} \cup F$ is an r -tree of $G[S]$ that satisfies $x(E[S]) = |S| - 1$. It can be deduced that $a_f = 0$ for any edge $f \in P_{ve}$ and thus $a_e = 0$ also holds for any $e \in E \setminus E[S]$.

Since $G[S]$ is 2-connected, for any distinct edge $e, f \in E[S]$, there must exist a cycle $C \subseteq E[S]$ of $G[S]$ such that it contains both e and f . In addition, 2-connectivity of $G[S]$ also ensures that there exists an edge set $F \subset E[S]$ such that it induces a spanning tree of $G[S]$ and $F \cap C = C \setminus \{e\}$. Both F and $F \cup \{e\} \setminus \{f\}$ induce r -trees that satisfy $x(E[S]) = |S| - 1$. One therefore gets $a_e = a_f$, and hence $a_e = \lambda$ for any $e \in E[S]$ and $b = (|S| - 1)\lambda$ with $\lambda \in \mathbb{R}$.

To summarize, $\mathbf{a}^T \mathbf{x} \leq b$ can be written as $\lambda x(E[S]) \leq (|S| - 1)\lambda$. Finally, as $\mathbf{0} \in R_x(G, r) \setminus \mathcal{F}$, we can conclude \mathcal{F} is a maximal proper face, that is, a facet of $R_x(G, r)$. \square

This section contributes to the facial study of all the inequalities mentioned in the formulation. However, during the course of our work on both theoretical and computational aspects, there are two sets of new inequalities that have been discovered to be facet-

defining for $R_x(G, r)$ as well. We present the results respecting these new inequalities in the next section.

5.3 New valid inequalities

In this section, we introduce two sets of new inequalities with some instances where they help to cut off some fractional points, and as well as the necessary and sufficient conditions for them to be facet-defining for $R_x(G, r)$. Note that according to the close relation between $R_x(G, r)$ and $B_x(G, r, \mathbf{c})$, these results will also be extended to $B_x(G, r, \mathbf{c})$ in the next chapter.

5.3.1 Matching-partition inequalities

Let $\pi = \{S_0, S_1, \dots, S_k\}$, $k \geq 1$, be a partition of V with $r \in S_0$ and let $M = \{e_1, \dots, e_k\}$ be a matching of G with $e_i \in E[S_i]$ for all $i \in \{1, \dots, k\}$. The pair (M, π) is called a *matching-partition* of G , a concept already considered by Didi-Biha et al. [2015] with respect to the connected subgraph problem. Denote by $\mathcal{MP}(G)$ the set composed of all the matching-partitions of G , and by E_π the set of edges having their extremities in different classes of partition π . With any matching-partition $(M, \pi) \in \mathcal{MP}(G)$, one can associate the following *matching-partition inequality*

$$x(M) - x(E_\pi) \leq 0. \quad (5.6)$$

Theorem 5.3.1. *For any $(M, \pi) \in \mathcal{MP}(G)$, inequality (5.6) is valid for $R_x(G, r)$.*

Proof. Consider any r -tree $G[F]$ with $F \subseteq E$. It can be seen that M and $F \cap M$ are both matchings of G . One needs at least $|F \cap M|$ edges in E_π to ensure that edges in $F \cap M$ and r are in the same component of $G[F]$, that is, $x^F(M) - x^F(E_\pi) \leq 0$ is satisfied. \square

Figure 5.1 demonstrates an instance where a fractional extreme point can be obtained without and cut by a matching-partition inequality. The value for each edge of the fractional extreme point \bar{x} is indicated in the figure, and \bar{x} is decided by the following

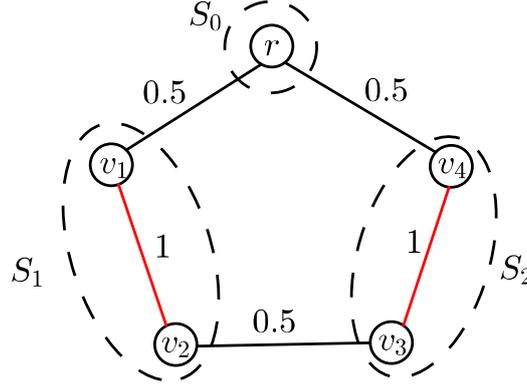


Figure 5.1: An example of matching-partition inequality cutting off a fractional extreme point

equations.

$$\begin{aligned}
 x_{v_1v_2} &= 1, \\
 x_{v_3v_4} &= 1, \\
 x_{v_1v_2} - (x_{rv_1} + x_{v_2v_3}) &= 0, \\
 x_{v_1v_2} - (x_{rv_1} + x_{rv_4}) &= 0, \\
 x_{v_3v_4} - (x_{v_2v_3} + x_{rv_4}) &= 0.
 \end{aligned}$$

At the meantime, \bar{x} violates the valid inequality $x(M) - x(E_\pi) \leq 0$ with $M = \{v_1v_2, v_3v_4\}$ and $\pi = \{S_0, S_1, S_2\}$.

Let G' be the graph obtained from G by shrinking each $S_i \in \pi$, $i \in \{0, 1, \dots, k\}$, into a node. We show that the matching-partition inequality is facet-defining under certain conditions.

Proposition 5.3.2. *Let $(M, \pi) \in \mathcal{MP}(G)$, $k \geq 2$. $x(M) - x(E_\pi) \leq 0$ defines a facet of $R_x(G, r)$ if and only if the following three conditions are satisfied*

1. $G[S_i]$ is connected, $i = 0, 1, \dots, k$;
2. G' is 2-connected;
3. there does not exist any $e \in E[S_i]$, such that removing e disconnects e_i and r .

Proof. Let $\mathcal{F} = \{x \in R_x(G, r) : x(M) - x(E_\pi) = 0\}$.

For the necessity, if $G[S_i]$ is not connected for some $i \in \{0, 1, \dots, k\}$, then \mathcal{F} is a proper subset of the proper face induced by $x_e \geq 0$ for some $e \in E$.

If there exists $S_i \in \pi$ such that $G[V \setminus S_i]$ is not connected, without loss of general-

ity, assume that $G[V \setminus S_i]$ contains a connected component induced by S_j, S_{j+1}, \dots, S_k , $j \in \{1, \dots, k\}$. One gets a new matching-partition (M'_1, π'_1) of G with $M'_1 = M \setminus \{e_j, e_{j+1}, \dots, e_k\}$ and $\pi'_1 = \pi \setminus \{S_i, S_j, S_{j+1}, \dots, S_k\} \cup \{S_i^1\}$, where $S_i^1 = S_i \cup S_j \cup S_{j+1} \cup \dots \cup S_k$. Let (M'_2, π'_2) be such that $M'_2 = M \setminus M'_1$ and $\pi'_2 = \pi \setminus \pi'_1 \cup \{S_i^2\}$, where $S_i^2 = V \setminus S_i^1 \cup S_i$. One can see that $(M'_2, \pi'_2) \in \mathcal{MP}(G)$, and moreover $E_{\pi'_1} \cap E_{\pi'_2} = \emptyset$. Hence $x(M) - x(E_\pi) \leq 0$ is a linear combination of $x(M'_1) - x(E_{\pi'_1}) \leq 0$ and $x(M'_2) - x(E_{\pi'_2}) \leq 0$, and thus it is not facet-defining.

If there exists an edge $e \in E[S_i]$, such that removing it disconnects e_i from r , then let $M' = M \setminus \{e_i\} \cup \{e\}$. \mathcal{F} is a proper subset of the proper face induced by $x(M') - x(E_\pi) \leq 0$.

For the sufficiency, we suppose that the conditions in the proposition are satisfied and $\mathcal{F} \subseteq \{\mathbf{x} \in \mathbb{R}_x(G, r) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq \mathbb{R}_x(G, r)$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $\mathbb{R}_x(G, r)$. One first has $b = 0$ because of $\mathbf{0} \in \mathcal{F}$.

As $G[S_0]$ is connected, for any edge $e \in E[S_0]$, any re -path of $G[S_0]$ is a feasible solution in \mathcal{F} . We then deduce that $a_e = 0$ for any $e \in E[S_0]$.

Let P_{re_i} be a path between r and e_i such that $|P_{re_i} \cap \delta(S_i)| = |P_{re_i} \cap \delta(S_0)| = 1$, $|P_{re_i} \cap \delta(S_j)| \in \{0, 2\}$, for all $S_j \in \pi \setminus \{S_0, S_i\}$. The conditions ensure that there must exist an edge set $F_{P_{re_i}}$ that is composed of P_{re_i} and a path from $V[P_{re_i}] \cap S_j$ to e_j for each S_j which has $|P_{re_i} \cap \delta(S_j)| = 2$ and $e_j \notin P_{re_i}$.

For any $e_i \in M$ and any $e' \in E[S_i] \setminus \{e_i\}$, $i \in \{1, \dots, k\}$, the conditions in the proposition ensure that there must exist P_{re_i} and $F_{P_{re_i}}$ with $e' \notin P_{re_i}$. Let $P_{e'}$ be the path between some node in $V[F_{P_{re_i}}] \cap S_i$ and e' . $F_{P_{re_i}}$ and $F_{P_{re_i}} \cup P_{e'}$ are two r -trees whose incidence vectors belong to \mathcal{F} . One can deduce that $a_{e'} = 0$ for any $e' \in E[S_i] \setminus \{e_i\}$.

For any $e \in \delta(S_i)$, $i \in \{1, \dots, k\}$, there must exist a path P_{re_i} and $F_{P_{re_i}}$ with $e \in P_{re_i}$ as G' is 2-connected. Let the subpath between e_i and e of P_{re_i} be P_{ee_i} . $F_{P_{re_i}}$ and $F_{P_{re_i}} \setminus P_{ee_i}$ are both r -trees that satisfies $x(M) - x(E_\pi) = 0$. Thus one has $a_e + a_{e_i} = 0$. Therefore, from the connectivity of G , one also deduce that $a_e = -a_{e_i}$ for any $e \in E_\pi$ and $e_i \in M$.

Thereby $\mathbf{a}^T \mathbf{x} \leq b$ now can be written as $\lambda x(M) - \lambda x(E_\pi) \leq 0$ with $\lambda \in \mathbb{R}$. Moreover, any path between r and an edge in E_π is an r -tree that satisfies $x(M) - x(E_\pi) < 0$. Therefore, \mathcal{F} is a facet of $\mathbb{R}_x(G, r)$. \square

It can be noticed that the matching-partition inequalities introduced here are slightly different from those proposed by Didi-Biha et al. [2015] for CSP because of the existence of the root. Moreover, For any $(M, \pi) \in \mathcal{MP}(G)$ with $E[S_0] \neq \emptyset$, a matching-partition inequality for CSP is as follows

$$x(M') - x(E_\pi) \leq 1,$$

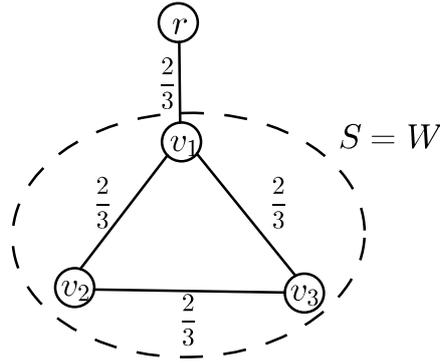


Figure 5.2: An example of an acyclicity-connectivity inequality cutting off a fractional extreme point

where $M' = M \cup \{e_0\}$ with $e_0 \in E[S_0]$. It can be obtained simply from the combination of $x(M) - x(E_\pi) \leq 0$ and $x_{e_0} \leq 1$.

Note that the connectivity inequalities can be seen as a restriction of matching-partition inequalities to the case of $|M| = 1$. Nonetheless, the matching-partition inequalities consider the connectivity of multiple parts of the graph instead of only 2 parts.

Aside from the matching-partition inequalities, we have discovered another set of inequalities that are also facet-defining for $R_x(G, r)$.

5.3.2 Acyclicity-connectivity inequalities

Let $W \subseteq S \subseteq V \setminus \{r\}$, $|W| \geq 2$. The *acyclicity-connectivity inequality* is defined as follows.

$$x(E[W]) - (|W| - 1)x(\delta(S)) \leq 0 \quad (5.7)$$

Theorem 5.3.3. *For any $W \subseteq S \subseteq V \setminus \{r\}$, $|W| \geq 2$, (5.7) is valid for $R_x(G, r)$.*

Proof. Assume that there exists an r -tree induced by an edge set $F \subseteq E$ such that $x^F(E[W]) - (|W| - 1)x^F(\delta(S)) \geq 1$. If $x^F(\delta(S)) = 0$, as $x^F(E[W]) \geq 1$, the connectivity inequality associated with W and some edge in $E[W]$ is then violated. If $x^F(\delta(S)) \geq 1$, we have $x^F(E[W]) \geq (|W| - 1)x^F(\delta(S)) + 1 \geq |W|$, and thus the acyclicity inequality associated with W is then violated. \square

Figure 5.2 shows an instance, where a fractional extreme point is cut by an acyclicity-

connectivity inequality. The extreme point is decided by

$$\begin{aligned} x(E[S]) &= |S| - 1, \\ x_{v_1v_2} - x_{rv_1} &= 0, \\ x_{v_2v_3} - x_{rv_1} &= 0, \\ x_{v_1v_3} - x_{rv_1} &= 0. \end{aligned}$$

It violates (5.7) with $W = S = \{v_1, v_2, v_3\}$.

Note that the connectivity inequalities (5.1) can also be seen as a restriction of acyclicity-connectivity inequalities with $|W| = 2$. We present the necessary and sufficient conditions for any acyclicity-connectivity inequality with $|W| \geq 3$ to be facet-defining in the following proposition.

Proposition 5.3.4. *Let $W \subseteq S \subseteq V \setminus \{r\}$ with $|W| \geq 3$. $x(E[W]) - (|W| - 1)x(\delta(S)) \leq 0$ defines a facet of $R_x(G, r)$ if and only if the following four conditions are satisfied*

1. $G[S]$ is connected;
2. $G[\overline{S}]$ is connected;
3. $G[W]$ is 2-connected;
4. there does not exist $e_b \in E[S] \setminus E[W]$ such that removing e_b disconnects S and r .

Proof. Let $\mathcal{F} = \{\mathbf{x} \in R_x(G, r) : x(E[W]) - (|W| - 1)x(\delta(S)) = 0\}$.

For the necessity, if $G[W]$ is not connected, \mathcal{F} is a proper subset of a proper faced induced by $x_e \geq 0$ for any $e \in E[W]$. If $G[S]$ is not connected, then \mathcal{F} is a proper subset of a proper faced induced by $x_e \geq 0$ for some edge $e \in \delta(v)$, and v in the connected component of $G[S]$ that does not contain W . If $G[\overline{S}]$ is not connected, then \mathcal{F} is a proper subset of a proper faced induced by $x_e \geq 0$ for some edge $e \in \delta(v)$, and v in the connected component of $G[\overline{S}]$ that does not contain r .

If there exists an edge e_b in $E[W]$ that removing e_b disconnects $G[W]$, \mathcal{F} is a proper subset of a proper faced induced by $x_{e_b} - x(\delta(S)) \leq 0$.

If there exists an articulation node v_a in W that removing $\delta(v_a)$ disconnects $G[W]$, then for any connected component $G[W_i \cup \{v_a\}]$ induced by v_a , \mathcal{F} is a proper subset of a proper faced induced by $x(E[W_i \cup \{v_a\}]) - (|W_i \cup \{v_a\}| - 1)x(\delta(S)) \leq 0$.

If there exists an edge e_b in $E[S] \setminus E[W]$ that removing e_b disconnects S from r , \mathcal{F} is a proper subset of a proper faced induced by $x(E[W]) - (|W| - 1)x(\delta(S')) \leq 0$, where $S' \subsetneq S$ and $\delta(S') = \{e_b\}$.

For the sufficiency, assume that the conditions in the proposition are satisfied and $\mathcal{F} \subseteq$

$\{\mathbf{x} \in R_x(G, r) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq R_x(G, r)$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $R_x(G, r)$. It can be first deduced that $b = 0$ from $\mathbf{0} \in \mathcal{F}$.

As $G[\overline{S}]$ is connected, for any edge $e \in E[\overline{S}]$, there exists a path P_{re} of $G[\overline{S}]$ between r and e . Any subpath of P_{re} containing r is an r -tree of G that satisfies $x(E[W]) - (|W| - 1)x(\delta(S)) = 0$, thus one has $a_e = 0$ for any $e \in E[\overline{S}]$.

For any two distinct edges $e_1, e_2 \in E[W]$, $G[W]$ is 2-connected, there must exist a cycle $C \subseteq E[W]$ that contains both e_1 and e_2 . Hence, there also exists such edge set $F \subsetneq E[W]$, such that $C \setminus \{e_1\} \subseteq F$ and F induces a spanning tree of $G[W]$. Moreover, $F' = F \setminus \{e_2\} \cup \{e_1\}$ also induces a spanning tree of $G[W]$. Combining either of these edge sets with any path P_{rv} from r to a node $v \in W$ such that $|P_{rv} \cap \delta(S)| = 1$, gives us an r -tree that satisfies $x(E[W]) - (|W| - 1)x(\delta(S)) = 0$. Therefore, we have $a_e = \lambda$, $\lambda \in \mathbb{R}$, for any edge $e \in E[W]$.

Now consider edges in $E[S] \setminus E[W]$. Since there does not exist $e_b \in E[S] \setminus E[W]$ such that removing e_b disconnects S from r , one can deduce that there must be two paths P_{rv}, P_{ru} from r to some node v or u in S such that $|P_{rv} \cap \delta(S)| = 1$, $|P_{ru} \cap \delta(S)| = 1$, and $P_{rv} \cap P_{ru} \cap E[S] \setminus E[W] = \emptyset$. Notice that here v and u could potentially be the same node. Let $P_{vS} = P_{rv} \cap E[S] \setminus E[W]$ and $P_{uS} = P_{ru} \cap E[S] \setminus E[W]$.

Without loss of generality, for any edge $e \in P_{vS}$, two situations can happen here. The first one is that there exists a subpath P of P_{vS} that contains e and at the same time $P \cup P_{uS}$ does not contain any cycle. In this case, one can deduce that any edge e' in P satisfies $a_{e'} = 0$. The second situation is that such P_e does not exist, which also means P_{uS} contains a path P_e between the two ends of e . Replacing P_e by e in P_{uS} to obtain P'_{uS} , and through the same process for the first case, one can prove that $a_{e'} = 0$ for any $e' \in P_e$, and thus $a_e = a(P_e) = 0$.

For any edge e in $E[S] \setminus (E[W] \cup P_{vS} \cup P_{uS})$, similarly, either there exists a path $P \subseteq E[S] \setminus E[W]$ that contains e and $P \cup P_{uS} \cup F$ does not contain any cycle, or P_{uS} contains a path P_e between the two ends of e , where F induces a tree that spans all nodes in W . In both cases, we have $a_e = 0$.

Consider now an edge $e \in \delta(S)$. As $E[S]$ is connected, there must exist a path P from r to a node in S such that $P \cap \delta(S) = \{e\}$, $P \cap E[W] = \emptyset$. Combining P with any F that induces a tree spanning all nodes in W , one has that $P \cup F$ induces an r -tree. Hence, $a_e + a(F) = 0$ according to the previous results, and consequently, $a_e = -a(F) = (|W| - 1)\lambda$.

To summarize, $\mathbf{a}^T \mathbf{x} \leq b$ is equivalent to $\lambda x(E[W]) - \lambda(|W| - 1)x(\delta(S)) \leq 0$, which implies that \mathcal{F} is a facet of $R_x(G, r)$. \square

Intuitively, the acyclicity-connectivity inequalities concern two aspects of the problem,

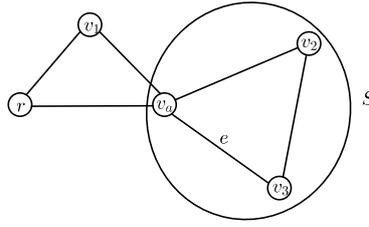


Figure 5.3: Counter example of decomposition involving 2 blocks

the acyclicity and connectivity. It ensures the acyclicity within the subgraph $G[S]$, and at the same time the connectivity between r and S if at least one edge in $E[S]$ is selected.

5.4 Decomposition

Unlike what Goemans [1994] showed for $R_{xy}(G, r)$, the polyhedral study for $R_x(G, r)$ cannot be simply restricted to 2-connected graphs based on decomposition with respect to articulation nodes. Some simple instances are presented below where such decomposition is not feasible.

Consider the graph in Figure 5.3 as an example. Let the two subgraphs separated by the articulation node v_a be $G_1 = G[\{r, v_1, v_a\}]$ and $G_2 = G[\{v_a, v_2, v_3\}]$. The incidence vectors of \emptyset , $\{rv_1\}$, $\{rv_1, v_1v_a, e\}$, $\{rv_a, e\}$, $\{rv_a, e, v_av_2\}$, $\{rv_a, e, v_av_3\}$ are affinely independent, each of which induces an r -tree and is binding at the connectivity inequality (5.1) associated with e and S , that is,

$$x_e - x(\delta(S)) \leq 0. \quad (5.8)$$

Thus it defines a facet of $R_x(G, r)$. Moreover, the support graph of inequality (5.8) contains both edges in G_1 and in G_2 . Hence if one wants to decompose G into G_1 and G_2 in polyhedral study, inequalities such as (5.8) should be added in addition to the simple combination of polytopes respecting G_1 and G_2 .

Consider again the graph in Figure 5.3. The following acyclicity-connectivity inequality (5.7) associated with S also facet-defining.

$$x(E[S]) - (|S| - 1)x(\delta(S)) \leq 0. \quad (5.9)$$

On the face induced by inequality (5.9), the incidence vectors of \emptyset , $\{rv_1\}$, $\{rv_a, v_av_3, v_3v_2\}$, $\{rv_1, v_1v_a, v_av_3, v_3v_2\}$, $\{rv_a, v_av_3, v_av_2\}$, $\{rv_a, v_1v_a, v_av_3, v_3v_2\}$ are affinely independent, and it is thus facet-defining. Notice that that (5.9) also involves edges in both G_1 and

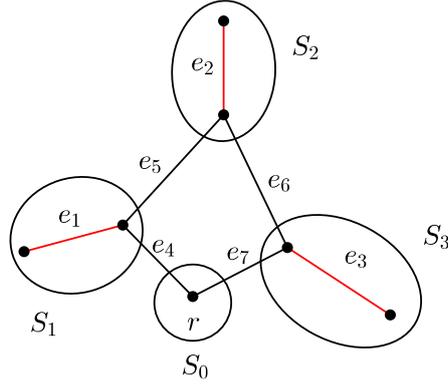


Figure 5.4: Another counter example of decomposition involving 3 blocks

G_2 .

Furthermore, consider the graph in Figure 5.4. Let (M, π) be a matching partition in the graph such that $M = \{e_1, e_2, e_3\}$ and $\pi = \{S_0, S_1, S_2, S_3\}$. We claim that the associated matching-partition inequality

$$x(M) - x(E_\pi) \leq 0 \quad (5.10)$$

defines a facet of $R_x(G, r)$.

Let the face induced by (5.10) be $\mathcal{F} = \{\mathbf{x} \in R_x(G, r) : x(M) - x(E_\pi) = 0\}$. Assume that there exists a face $\mathcal{F}' = \{\mathbf{x} \in R_x(G, r) : \mathbf{a}^T \mathbf{x} = b\}$ such that $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $R_x(G, r)$ and $\mathcal{F} \subsetneq \mathcal{F}' \subsetneq R_x(G, r)$. From $\mathbf{0} \in \mathcal{F} \subsetneq \mathcal{F}'$, one has $b = 0$. The edge sets $\{e_4, e_1\}$, $\{e_7, e_3\}$, $\{e_4, e_1, e_5, e_2\}$, $\{e_7, e_3, e_6, e_2\}$, $\{e_4, e_1, e_5, e_2, e_6, e_3\}$, $\{e_7, e_3, e_6, e_2, e_5, e_1\}$ all induce r -trees and their incidence vectors are in \mathcal{F} . One can deduce that $a_{e_4} = a_{e_5} = a_{e_6} = a_{e_7} = -a_{e_1} = -a_{e_2} = -a_{e_3}$. Hence, $\mathcal{F} = \mathcal{F}' \subsetneq R_x(G, r)$, which indicates \mathcal{F} is a facet of $R_x(G, r)$.

Therefore, (5.10) is also a facet-defining inequality for $R_x(G, r)$ whose support graph contains edges in multiple blocks separated by articulation nodes.

Such cases also exist for $B_x(G, r, c)$ since one can set the capacity on each node in V to be large enough to obtain the same polytope as $R_x(G, r)$. This fact makes the decomposition through 1-sum infeasible for both $R_x(G, r)$ and $B_x(G, r, c)$.

5.5 Conclusion

This chapter began with the discussion of the dimension of $R_x(G, r)$. It is shown that $R_x(G, r)$ is full-dimensional if G is connected. We then presented the results concerning the facets of the r -Tree Polytope. Additionally, two sets of new facet-defining inequali-

ties, namely the matching-partition inequalities and the acyclicity-connectivity inequalities, are introduced with their facet-defining conditions. We reviewed the possibility of decomposing the polytope with respect to articulation nodes. Unfortunately, a few examples have been found where several facet-defining inequalities have support graphs that are not subgraphs of blocks of G . This implies that the decomposition through 1-sum cannot be applied for either $R_x(G, r)$ or $B_x(G, r, \mathbf{c})$ as straightforwardly as in the case of $B_{xy}(G, r, \mathbf{c})$.

As mentioned in Chapter 2, any valid inequality for $R_x(G, r)$ is also valid for $B_x(G, r, \mathbf{c})$. It is also likely that these two polytopes bear similarities in some aspects such as polyhedral structure and decomposition. Thus, some of the results in this chapter can be transformed into those for $B_x(G, r, \mathbf{c})$. In the next chapter, based on the results presented in this chapter, we bring the capacity constraints into play to obtain results with regard to $B_x(G, r, \mathbf{c})$. We show that $B_x(G, r, \mathbf{c})$ can be characterized on trees and cycles with the help of some newly discovered inequalities. The characterization of $R_x(G, r)$ can thus be obtained as an immediate result.

Chapter 6

Polyhedral study on Bounded r -Tree Polytope

Recall that given a graph $G = (V, E)$, a root node $r \in V$, a capacity vector $\mathbf{c} \in \mathbb{Z}_+^V$, an edge-weight vector $\mathbf{w} \in \mathbb{R}^E$, and a node-price vector $\mathbf{p} \in \mathbb{R}^V$, the MBrT problem consists of finding an r -tree T of G with maximum value $f(T) = w(E(T)) + p(V(T))$ and such that the degree of any node $v \in V(T)$ is bounded by c_v .

As opposed to Chapter 4, this chapter provides results regarding $B_x(G, r, \mathbf{c})$, which only uses edge-indexed variables.

We start with the following formulation introduced in Chapter 2.

$$x_e - x(\delta(S)) \leq 0 \quad \forall e \in E[S], S \subseteq V \setminus \{r\}, \quad (6.1)$$

$$x(E[S]) \leq |S| - 1 \quad \forall S \subseteq V, |S| \geq 3, \quad (6.2)$$

$$x(\delta(v)) \leq c_v \quad \forall v \in V, \quad (6.3)$$

$$x_e \leq 1 \quad \forall e \in E, \quad (6.4)$$

$$x_e \geq 0 \quad \forall e \in E. \quad (6.5)$$

First, the dimension of $B_x(G, r, \mathbf{c})$ is examined. Possible approaches of decomposition for the polyhedral study are discussed with proofs. After that, some general criteria are presented for any facet-defining inequality to follow. Then, we study the necessary and sufficient conditions for each set of inequalities in the formulation to be facet-defining

Furthermore, we present several families of new valid inequalities which are facet-defining for $B_x(G, r, \mathbf{c})$. Their facet-defining conditions are also discussed. We also show that all the new inequalities proposed in this chapter can be obtained by projection from valid inequalities for $B_{xy}(G, r, \mathbf{c})$. Moreover, the bounds on the Chvátal-Gomory rank of several

sets of inequalities are investigated with respect to the original formulation. Finally, we show that incorporating certain sets of the new inequalities allows us to form a TDI system that completely describe $B_x(G, r, \mathbf{c})$ on trees and cycles.

6.1 Dimension

Recall that $G_r = (V_r, E_r)$ and $G_r = G[V \setminus O]$, and according to Assumption 2.2.9 G_r is connected. Hence, we have

$$E = E_r \cup \delta(V_r).$$

The dimension of $B_x(G, r, \mathbf{c})$ is given in the following theorem.

Theorem 6.1.1. $B_x(G, r, \mathbf{c})$ is full-dimensional, that is,

$$\dim B_x(G, r, \mathbf{c}) = |E|.$$

Proof. Given any $e \in E_r$, let P_{re} be the re -path in G_r containing as few edges as possible. According to Lemma 5.1.1, the set $\{\mathbf{x}^{P_{re}} : e \in E_r\}$ is affinely independent. Since nodes in O are not involved, each of these vectors also satisfies the capacity requirement.

For any edge $e_o = vv_o \in \delta_G(V_r)$ with $v \in V_r$ and $v_o \in O$, denote the re_o -path P_{re_o} in G such that it composed of vv_o and a path P_{rv} between r and v of G_r . It can be seen that P_{re_o} is also a bounded r -tree of G . In the set of vectors $\{\mathbf{x}^{P_{re}} : e \in E_r \cup \delta(V_r)\}$, for any $e_o \in \delta(V_r)$, $\mathbf{x}^{P_{re_o}}$ is the only vector that has $x_{e_o} = 1$. Hence one has $|E_r| + |\delta(V_r)|$ non-zero linearly independent vectors in $\{\mathbf{x}^{P_{re}} : e \in E_r \cup \delta(V_r)\}$. Finally, combining the zero vector with it gives us $|E_r| + |\delta(V_r)| + 1$ affinely independent vectors, each of which induces a bounded r -tree of G . Therefore, we have

$$\dim B_x(G, r, \mathbf{c}) \geq |\delta(V_r)| + |E_r| = |E|.$$

Combining it with the trivial fact of $\dim B_x(G, r, \mathbf{c}) \leq |E|$ completes the proof. \square

6.2 Decomposition

As it has been discussed in the Chapter 5, the decomposition with respect to the articulation nodes is unlikely to work for both $R_x(G, r)$ and $B_x(G, r, \mathbf{c})$. Nonetheless, we present in this section a few other options of decomposition one can opt for $B_x(G, r, \mathbf{c})$. Notice that most of the results developed for $B_x(G, r, \mathbf{c})$ can be used for $R_x(G, r)$, as the r -tree

problem is a relaxation of the bounded r -tree problem.

6.2.1 Decomposition at the root node

As showed in the last chapter, the general decomposition of $B_x(G, r, \mathbf{c})$ and $R_x(G, r)$ with respect to articulation nodes is proved to be infeasible due to the counter examples, or at least not as straightforward as thereof $B_{xy}(G, r, \mathbf{c})$. Nonetheless, we prove the viability of a special case of this decomposition where the articulation node is the root.

Consider a graph $G = (V, E)$ where r is an articulation node, such that $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ are two subgraphs of G separated by r , and G is a 1-sum of G_1 and G_2 . Let \mathbf{x} be a vector in \mathbb{R}^E , and \mathbf{x}^i be the restriction of \mathbf{x} to E_i , $i = 1, 2$. Let \mathbf{c}^i be the restriction of the capacity vector \mathbf{c} to V_i , $i = 1, 2$. It can be proved that the decomposition through 1-sum is feasible in this case.

Denote

$$P_R(G, r, \mathbf{c}) = \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x}^1 \in B_x(G_1, r_1, \mathbf{c}^1), \mathbf{x}^2 \in B_x(G_2, r_2, \mathbf{c}^2), x(\delta(r)) \leq c_r\}.$$

Theorem 6.2.1. $P_R(G, r, \mathbf{c}) = B_x(G, r, \mathbf{c})$.

Proof. First, an edge set $F \subseteq E$ induces a bounded r -tree of G if and only if $F \cap E_i$ induces a bounded r -tree of G_i for $i \in \{1, 2\}$ and $|F \cap \delta(r)| \leq c_r$. Hence we have

$$P_R(G, r, \mathbf{c}) \cap \mathbb{Z}^E = B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E.$$

Assume that there exists a fractional extreme point $\bar{\mathbf{x}}$ in $P_R(G, r, \mathbf{c})$. Let $S(\bar{\mathbf{x}})$ be the linear system composed of the equations associated with the constraints of $P_R(G, r, \mathbf{c})$ binding at $\bar{\mathbf{x}}$. Let $S_i(\bar{\mathbf{x}})$ denote the equations in $S(\bar{\mathbf{x}})$ that are associated with $B_x(G_i, r_i, \mathbf{c}^i)$ for $i \in \{1, 2\}$.

If $x(\delta(r)) - c_r = 0$ is not in $S(\bar{\mathbf{x}})$, then all the equations in $S(\bar{\mathbf{x}})$ are in either $S_1(\bar{\mathbf{x}})$ or $S_2(\bar{\mathbf{x}})$. From the integrality of $B_x(G_1, r_1, \mathbf{c}^1)$ and $B_x(G_2, r_2, \mathbf{c}^2)$, there must exist two integral points $\tilde{\mathbf{x}}^1 \in B_x(G_1, r_1, \mathbf{c}^1)$ and $\tilde{\mathbf{x}}^2 \in B_x(G_2, r_2, \mathbf{c}^2)$ such that $\tilde{\mathbf{x}}^i$ satisfies $S_i(\bar{\mathbf{x}})$ for $i \in \{1, 2\}$. Thus, $\begin{bmatrix} \tilde{\mathbf{x}}^1 \\ \tilde{\mathbf{x}}^2 \end{bmatrix}$ is an integral point that also satisfies $S(\bar{\mathbf{x}})$, which forms a contradiction. Hence, $x(\delta(r)) - c_r = 0$ is in $S(\bar{\mathbf{x}})$.

As $S(\bar{\mathbf{x}})$ admits a unique solution, the rank of its coefficient matrix is $|E|$. Moreover, as $x(\delta(r)) - c_r = 0$ is the only equation in $S(\bar{\mathbf{x}})$ that is not in $S_1(\bar{\mathbf{x}})$ or $S_2(\bar{\mathbf{x}})$. One has that there exists $j \in \{1, 2\}$, such that $S_j(\bar{\mathbf{x}})$ contains $|E_j|$ linearly independent equations. Without loss of generality, assume $j = 1$. According to the integrality of $B_x(G_1, r_1, \mathbf{c}^1)$, $\bar{\mathbf{x}}^1$ is integral, and thus $\bar{\mathbf{x}}^1(\delta(r))$ is integral. Furthermore, since the system composed of

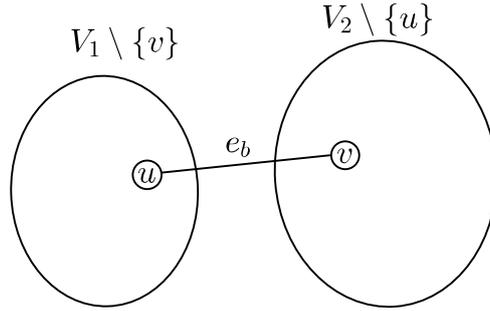


Figure 6.1: Decomposition of graph with a bridge

$x(\delta_{G_2}(r)) = c_r - \bar{x}^1(\delta(r))$ and equations in $S_2(\bar{x})$ admits a feasible solution \bar{x}^2 , there must also exist an integral solution \tilde{x}^2 which satisfies the same equations. Combining \bar{x}^1 and \tilde{x}^2 gives us an integral point that also satisfies $S(\bar{x})$, which forms a contradiction with \bar{x} being an extreme point.

Thus, $P_R(G, r, \mathbf{c})$ is integral and therefore $P_R(G, r, \mathbf{c}) = B_x(G, r, \mathbf{c})$. \square

As an immediate result, the following corollary is obtained immediately by getting rid of the capacity factor.

Corollary 6.2.2. $R_x(G, r) = \{\mathbf{x} : \mathbf{x}^1 \in R_x(G_1, r_1), \text{ and } \mathbf{x}^2 \in R_x(G_2, r_2)\}$. \square

6.2.2 Decomposition with respect to bridges

We consider now a graph containing a bridge. We show that we can decompose it to two graphs with both of them containing the bridge. Given a graph $G = (V, E)$ which contains a bridge $e_b = uv$, let E'_1 and E'_2 be the two edge sets separated by e_b . Let $E_i = E'_i \cup \{e_b\}$, $G_i = (V_i, E_i) = G[E_i]$ for $i \in \{1, 2\}$, as it is shown in Figure 6.1.

Without loss of generality, let $r_1 = r$ be the root in G_1 , and $r_2 = u$ be the root in G_2 . Note that if $v = r$, one can switch V_1 and V_2 as well as u and v in order to obtain the same setup. Let \mathbf{x} be a vector in \mathbb{R}^E , and \mathbf{x}^i be the restriction of \mathbf{x} to G_i , $i = 1, 2$. Let \mathbf{c}^i be the restriction of the capacity vector \mathbf{c} to V_i for $i = 1, 2$.

We show that if $B_x(G, r, \mathbf{c})$ can be characterized on G_1 and G_2 , it can also be characterized on G . Notice that if either $E_1 \setminus \{e_b\} = \emptyset$ or $E_2 \setminus \{e_b\} = \emptyset$, the decomposition is then meaningless. Hence we assume that $E'_i \neq \emptyset$ for $i = 1, 2$. The following theorem demonstrates the decomposition.

Theorem 6.2.3.

$$B_x(G, r, \mathbf{c}) = \{\mathbf{x} : \mathbf{x}^1 \in B_x(G_1, r_1, \mathbf{c}^1), \text{ and } \mathbf{x}^2 \in B_x(G_2, r_2, \mathbf{c}^2)\}.$$

Proof. Because of the fact that \mathbf{x}^1 and \mathbf{x}^2 have one common component x_e , one has for both sides $x_{e_b}^1 = x_{e_b}^2$.

First, if a vector $\mathbf{x} \in \mathbb{R}^E$ induces a bounded r -tree in G , then \mathbf{x}^i also induces a bounded r -tree in G_i , $i = 1, 2$. Therefore,

$$B_x(G, r, \mathbf{c}) \subseteq \{\mathbf{x} : \mathbf{x}^1 \in B_x(G_1, r_1, \mathbf{c}^1), \text{ and } \mathbf{x}^2 \in B_x(G_2, r_2, \mathbf{c}^2)\}.$$

On the other hand, if \mathbf{x}^i induces a bounded r -tree in G_i , $i = 1, 2$, and $x_{e_b}^1 = x_{e_b}^2$ also holds, then \mathbf{x} induces an r -tree in G . Additionally one has $x(\delta_G(u)) = x^1(\delta_{G_1}(u)) \leq c_u$ and $x(\delta_G(v)) = x^2(\delta_{G_2}(v)) \leq c_v$. Hence \mathbf{x} also satisfies the capacity constraints, and therefore \mathbf{x} induces a bounded r -tree in G .

Assume now that $\bar{\mathbf{x}}$ is a fractional extreme point of $\{\mathbf{x} : \mathbf{x}^1 \in B_x(G_1, r_1, \mathbf{c}^1), \text{ and } \mathbf{x}^2 \in B_x(G_2, r_2, \mathbf{c}^2)\}$. Let $S(\bar{\mathbf{x}})$ be the linear system composed of the equations associated with the constraints of $B_x(G_1, r_1, \mathbf{c}^1)$ and $B_x(G_2, r_2, \mathbf{c}^2)$ binding at $\bar{\mathbf{x}}$.

If $\bar{x}_{e_b} = 0$ ($\bar{x}_{e_b} \neq 0$, respectively), from the integrality of $B_x(G_1, r_1, \mathbf{c}^1)$ and $B_x(G_2, r_2, \mathbf{c}^2)$, there must exist two integral points $\tilde{\mathbf{x}}^1 \in B_x(G_1, r_1, \mathbf{c}^1)$ and $\tilde{\mathbf{x}}^2 \in B_x(G_2, r_2, \mathbf{c}^2)$ such that $\tilde{x}_{e_b}^1 = \tilde{x}_{e_b}^2 = 0$ ($\tilde{x}_{e_b}^1 = \tilde{x}_{e_b}^2 = 1$, respectively). Combining $\tilde{\mathbf{x}}^1$ and $\tilde{\mathbf{x}}^2$ gives us another point that also satisfies $S(\bar{\mathbf{x}})$, which forms a contradiction.

Therefore, $\{\mathbf{x} : \mathbf{x}^1 \in B_x(G_1, r_1, \mathbf{c}^1), \text{ and } \mathbf{x}^2 \in B_x(G_2, r_2, \mathbf{c}^2)\}$ is integral and thus the theorem holds. \square

The following corollary concerning $R_x(G, r)$ immediately follows after this theorem.

Corollary 6.2.4. $R_x(G, r) = \{\mathbf{x} : \mathbf{x}^1 \in R_x(G_1, r_1), \text{ and } \mathbf{x}^2 \in R_x(G_2, r_2)\}$. \square

In the next section, the facet-defining conditions for the inequalities in the proposed formulation are introduced.

6.3 Facets

In this section, we show that each set of inequalities included in the formation is indeed facet-defining. For all the inequalities (6.1)-(6.5), necessary and sufficient conditions for them to be facet-defining for $B_x(G, r, \mathbf{c})$ have been determined.

6.3.1 General results

First of all, some general criteria have also been characterized for valid inequalities to be facet-defining for $B_x(G, r, \mathbf{c})$. The following lemma describes the property of the

coefficients of the edges in $\delta(O)$ in any facet-defining inequalities.

Lemma 6.3.1. *Let $\mathbf{a}^T \mathbf{x} \leq b$ be a valid inequality for $B_x(G, r, \mathbf{c})$ that is different from $x_e \geq 0$ for some $e \in E$. $\mathbf{a}^T \mathbf{x} \leq b$ is facet-defining for $B_x(G, r, \mathbf{c})$ only if it satisfies $a_e \geq 0$ for any edge $e \in \delta(O)$.*

Proof. Assume that there exists an inequality $\mathbf{a}^T \mathbf{x} \leq b$ is facet-defining for $B_x(G, r, \mathbf{c})$ such that $a_{e_o} < 0$ for some edge $e_o \in \delta(O)$.

According to the assumption, there must exist an edge set $F \subseteq E$ such that $e_o \in F$, G_F is bounded r -tree and it satisfies $\mathbf{a}^T \mathbf{x}^F = b$. Otherwise if such F does not exist, the face induced by the inequality is included in the face defined by $x_{e_o} \geq 0$, which contradicts with the assumption of the lemma. Since e_o is incident with a node in O , it must be a leaf edge in G_F . Hence $G_{F \setminus \{e_o\}}$ is also a bounded r -tree of G . It can be deduced that $\mathbf{a}^T \mathbf{x}^{F \setminus \{e_o\}} = \mathbf{a}^T \mathbf{x}^F - a_{e_o} = b - a_{e_o} > b$. This implies inequality $\mathbf{a}^T \mathbf{x} \leq b$ is violated by the feasible solution $\mathbf{x}^{F \setminus \{e_o\}}$, which forms a contradiction with $\mathbf{a}^T \mathbf{x} \leq b$ being valid. \square

Besides, the case where the root has capacity 1 is discussed in the lemma below.

Lemma 6.3.2. *Let $c_r = 1$, and $\mathbf{a}^T \mathbf{x} \leq b$ be a valid inequality for $B_x(G, r, \mathbf{c})$ that is different from $x(\delta(r)) \leq c_r$. It is facet-defining for $B_x(G, r, \mathbf{c})$ only if $b = 0$.*

Proof. Assume that $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\}$ is a facet of $B_x(G, r, \mathbf{c})$ such that $\mathcal{F} \neq \mathcal{F}_r = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(r)) = c_r\}$ and $b \neq 0$. As $c_r = 1$, one has that any bounded r -tree of G whose edge set is nonempty satisfies $x(\delta(r)) = c_r = 1$, that is,

$$B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E \setminus \{\mathbf{0}\} \subseteq \mathcal{F}_r.$$

Since $b \neq 0$, $\mathbf{0} \notin \mathcal{F}$. Thus,

$$\mathcal{F} \subseteq B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E \setminus \{\mathbf{0}\} \subseteq \mathcal{F}_r.$$

Combining it with $\mathcal{F} \neq \mathcal{F}_r$ gives us

$$\mathcal{F} \subsetneq \mathcal{F}_r,$$

which forms a contradiction with the assumption of \mathcal{F} being a facet of $B_x(G, r, \mathbf{c})$. \square

Besides, for those inequalities having only non-negative coefficients, the following proposition can be developed.

Lemma 6.3.3. *Let $\mathbf{a}^T \mathbf{x} \leq b$ be a valid inequality for $B_x(G, r, \mathbf{c})$ with $\mathbf{a} \geq 0$, $\mathbf{a} \neq \mathbf{0}$ and $b > 0$. Denote $E^+ := \{e \in E \mid a_e > 0\}$ the set of edges with positive coefficients. If*

$r \notin V[E^+]$, then $\mathbf{a}^T \mathbf{x} \leq b$ defines a facet of $B_x(G, r, \mathbf{c})$ only if the following two conditions are satisfied

1. there does not exist a bridge $e_b \in E$ in $G[E \setminus (\delta(O) \setminus E^+)]$ between r and E^+ ;
2. there does not exist an articulation node $v_a \in V \setminus \{r\}$ in $G[E \setminus (\delta(O) \setminus E^+)]$ between r and E^+ with $c_{v_a} = 2$.

Proof. Suppose there exists a bridge $e_b \in E$ in $G[E \setminus (\delta(O) \setminus E^+)]$ between r and $V[E^+]$. Consider any $F \subseteq E$ with its incidence vector \mathbf{x}^F on the face defined by $\mathbf{a}^T \mathbf{x} \leq b$. One has $e^+ \in F$ for some $e^+ \in E^+$. Since G_F is a bounded r -tree, it must contain a path between r and e^+ , which has to include e_b . Hence $e_b \in F$, which implies the face induced by $\mathbf{a}^T \mathbf{x} \leq b$ is a proper subset of the proper face induced by $x_{e_b} \leq 1$.

Now suppose there exists an articulation node $v_a \in V \setminus \{r\}$ in $G[E \setminus (\delta(O) \setminus E^+)]$ between r and E^+ with $c_{v_a} = 2$. For any $F \subseteq E$ with its incidence vector \mathbf{x}^F on the face defined by $\mathbf{a}^T \mathbf{x} \leq b$, it can be seen that $x(\delta(v_a)) = c_{v_a}$ also holds. Thus the face induced by $\mathbf{a}^T \mathbf{x} \leq b$ is a proper subset of the proper face induced by $x(\delta(v_a)) \leq c_{v_a}$. \square

This proposition addresses the circumstance where the coefficients and the right-hand side of a valid inequality are positive. If the inequality is facet-defining, then its support graph does not contain certain substructures, specifically the bridges or articulation nodes with capacity 2.

6.3.2 Box inequalities

The condition for (6.5) to be facet-defining is relatively straightforward as stated in the following proposition.

Proposition 6.3.4. *Given an edge $e \in E$, inequality $x_e \geq 0$ defines a facet of $B_x(G, r, \mathbf{c})$ if and only if e is not a bridge between r and some edge e' in the graph $G[E \setminus (\delta(O) \setminus \{e'\})]$.*

Proof. Assume e is a bridge between r and e' in the graph $G[E \setminus (\delta(O) \setminus \{e'\})]$. Then there must exist a node set S , where $\delta(S) \setminus \delta(O) = \{e\}$ with $r \in S$ and $e' \in E[\overline{S}]$. If $x_e^* = 0$ one must have $x_{e'}^* = 0$. In another words, $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e = 0\} \subsetneq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_{e'} = 0\} \subsetneq B_x(G, r, \mathbf{c})$.

For the sufficiency, let $G_e = G[E \setminus \{e\}]$. According to the assumption there exists an re' -path between r and each edge $e' \in E \setminus \{e\}$ that also satisfies the capacity constraints. According to Lemma 5.1.1 and the proof of Theorem 6.1.1, $B_x(G_e, r, \mathbf{c})$ remains full-dimensional, which leads to $\dim\{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e = 0\} = \dim B_x(G_e, r, \mathbf{c}) = |E| - 1$. Therefore $x_e \geq 0$ defines a facet of $B_x(G, r, \mathbf{c})$. \square

Since the upper bound inequalities (6.4) have nonzero right-hand side and positive coefficients for the left-hand side, both Lemma 6.3.2 and Lemma 6.3.3 can be used here. First, based on Lemma 6.3.2 the necessary and sufficient facet-defining conditions for (6.4) associated with edges in $\delta(r)$ are given below.

Proposition 6.3.5. *Let $e = rv \in \delta(r)$. Inequality $x_e \leq 1$ defines a facet of $B_x(G, r, \mathbf{c})$ if and only if the following two conditions are satisfied*

1. $c_r \geq 2$ unless $|\delta(r)| = c_r = 1$;
2. $v \notin O$ unless $|\delta(v)| = c_v = 1$.

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e = 1\}$.

For the necessity, if $|\delta(r)| > c_r = 1$, \mathcal{F} is a proper subset of the proper face induced by $x_f \geq 0$ for any edge in $\delta(r) \setminus \{e\}$. If $v \in O$ and $|\delta(v)| > 1$, \mathcal{F} is then a proper subset of the proper face induced by $x_f \geq 0$ for any edge in $\delta(v) \setminus \{e\}$.

For the sufficiency, assume that all the conditions in the proposition are satisfied, and $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is a valid inequality for $B_x(G, r, \mathbf{c})$. Firstly, from $\mathbf{0} \notin \mathcal{F}$ we derive that $\mathcal{F} \subsetneq B_x(G, r, \mathbf{c})$.

Consider first the case where $c_r = 1$ and $\delta(r) = \{e\}$. Any non-zero integral vector in $B_x(G, r, \mathbf{c})$ is in \mathcal{F} . Hence we have $\dim \mathcal{F} = |E| - 1$, and thus \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$.

Consider now the case with $c_r \geq 2$. As the graph reduced to e induces a bounded r -tree that satisfies $x_e = 1$, we have $a_e = b$. Moreover, for any $e' \in \delta(v) \setminus \{e\}$, as $\{e, e'\}$ and $\{e\}$ both induce bounded r -trees whose incidence vectors are in \mathcal{F} , we have $a_{e'} = 0$.

Moreover, for any $e' \in E \setminus \delta(v)$, let $P_{re'}$ be a bounded path between r and e' . If $v \in V[P_{re'}]$, one can replace the subpath between r and v of $P_{re'}$ by e . We have that $P_{re'} \cup \{e\}$ and $P_{re'} \cup \{e\} \setminus \{e'\}$ induce two bounded r -trees, and their incidence vectors are in \mathcal{F} . Thus one can deduce that $a_{e'} = 0$ holds for any $e' \in E \setminus \delta(v)$.

Therefore, $\mathbf{a}^T \mathbf{x} \leq b$ can be rewritten as $bx_e \leq b$, and thus \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

Now based on Lemma 6.3.2 and Lemma 6.3.3, we deal with the inequalities in (6.4) associated with edges in $E \setminus \delta(r)$ in the following proposition.

Proposition 6.3.6. *Let $e = uv \in E \setminus \delta(r)$. Inequality $x_e \leq 1$ defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $c_r \geq 2$;
2. $e \notin \delta(O)$ unless $|\delta(v_o)| = c_{v_o} = 1$ for some $v_o \in O \cap \{u, v\}$;
3. there does not exist $w \in N(u) \cap N(v)$ such that removing uw and vw from $G[E \setminus (\delta(O) \setminus \{e\})]$ disconnects r and e ;

4. there exists a bounded path P_{rw} with $e \in P_{rw}$ if $c_w = 2$, for $w \in \{u, v\}$;
5. there exists no bridge between r and e in $G[E \setminus (\delta(O) \setminus \{e\})]$;
6. there exists no articulation node v_a with $c_{v_a} = 2$ between r and e in $G[E \setminus (\delta(O) \setminus \{e\})]$.

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e = 1\}$.

If $c_r = 1$, according to Lemma 6.3.2 it is not facet-defining for $B_x(G, r, \mathbf{c})$. If $e \in \delta(O)$ and without loss of generality let $v \in O$, then \mathcal{F} is a proper subset of the proper face induced by $x(\delta(v)) \leq c_v$ unless $|\delta(v)| = c_v = 1$. If there exists a node $w \in N(u) \cap N(v)$ such that removing uw and vw from $G[E \setminus (\delta(O) \setminus \{e\})]$ disconnects r and e , then \mathcal{F} is a proper subset of the proper face induced by $x(E[S]) \leq |S| - 1$ with $S = \{u, v, w\}$. If without loss of generality $c_u = 2$ and every bounded path between r and v has to pass through u , then \mathcal{F} is a proper subset of the proper face induced by $x(\delta(u)) \leq c_u$.

Additionally, Lemma 6.3.3 ensures the necessity of the last two conditions.

For the sufficiency, assume that all the conditions in the proposition are satisfied, and $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is a valid inequality for $B_x(G, r, \mathbf{c})$. First of all, since we clearly have $\mathbf{0} \notin \mathcal{F}$, we derive that $\mathcal{F} \subsetneq B_x(G, r, \mathbf{c})$.

Let P_1 and P_2 be two edge-disjoint bounded paths between r and e , and let

$$V_c = \{w \in V[P_1] \cap V[P_2] : w \text{ is an inner node of both } P_1 \text{ and } P_2\}.$$

According to the conditions, there exist such P_1 and P_2 with $c_w \geq 3$ for all $w \in V_c$. We have that any r -tree of $G[P_1 \cup P_2]$ that does not have any node with degree 4 is also a bounded r -tree of G . According to the proof of Proposition 5.2.4, as the graph $G[P_1 \cup P_2]$ satisfies the conditions 3 and 5, there exist $|P_1 \cup P_2|$ affinely independent vectors in $R_x(G[P_1 \cup P_2], r)$ that satisfy $x_e = 1$ such that any node in their induced graphs has degree at most 3. Thus, there exist $|P_1 \cup P_2|$ affinely independent vectors in $B_x(G, r, \mathbf{c})$ that satisfy $x_e = 1$ and $x_{e'} = 0$ for any $e' \in E \setminus (P_1 \cup P_2)$. One can then deduce that

$$\begin{aligned} a_e &= b, \\ a_{e'} &= 0 \quad \forall e' \in (P_1 \cup P_2) \setminus \{e\}. \end{aligned}$$

First we consider an edge $e' = u'v' \in E[V[P_i]] \setminus P_i$, $i \in \{1, 2\}$. Without loss of generality, let the two ends of P_i be r and v . If $u'v' \notin \delta(v)$, by substituting the subpath between u' and v' in P_1 for $u'v'$, one obtains another bounded path between r and e . As $a_{e'} = 0$ holds for any $e' \in P_1 \setminus \{e\}$, we derive $a_{u'v'} = 0$. If $v' = v$, let the subpath between u' and v of P_i be $P_i^{u'v}$. We clearly have $u' \neq u$ and $|P_i^{u'v}| \geq 2$. Hence $P_i \cup \{u'v, vu\} \setminus P_i^{u'v}$ is a

bounded path, which leads to $a_{u'v} = a(P_i^{u'v}) = 0$.

Consider an edge $u'v' \in E$ with $u' \in V[P_1] \setminus V[P_2]$ and $v' \in V[P_2] \setminus V[P_1]$. Let the subpath between r and u' of P_1 be $P_1^{ru'}$, and the subpath between r and v' of P_2 be $P_2^{rv'}$. Then $P_1^{ru'} \cup \{u'v'\} \cup (P_2 \setminus P_2^{rv'})$ is a bounded path that contains e . We then deduce that $a_{u'v'} = (P_2^{rv'}) - a(P_1^{ru'}) = 0$.

Consider now an edge $e' \in E \setminus E[V[P_1 \cup P_2]]$. If there exists a bounded path $P_{re'}$ such that $e \in P_{re'}$, we can deduce that $a_{e'} = 0$. Now suppose that there does not exist a bounded path $P_{re'}$ such that $e \in P_{re'}$.

Let $P_{re'}$ be a bounded path between r and e' . Without loss of generality let s be the node in $V[P_{re'}] \cap V[P_1]$ such that $V[P_{re'}^{se'}] \cap V[P_1] = \{s\}$ and $V[P_{re'}^{se'}] \cap V[P_2] \setminus \{s\} = \emptyset$. Let $v_c = s$ if $s \in V_c \cup \{r\}$, otherwise let v_c be the node in $V_c \cup \{r\}$ such that $V[P_1^{v_c s}] \cap (V_c \cup \{r\}) = \{v_c\}$. Then $G[P_2 \cup P_1^{v_c s} \cup P_{re'}^{se'}]$ is a bounded r -tree that contains e , and so is $G[P_2 \cup P_1^{v_c s} \cup P_{re'}^{se'} \setminus \{e'\}]$. Hence $a_{e'} = 0$.

To summarize, we have

$$a_{e'} = 0 \quad \forall e' \in E \setminus (P_1 \cup P_2).$$

Thus, $\mathbf{a}^T \mathbf{x} \leq b$ can be rewritten as $b x_e \leq b$. Moreover, as $\mathbf{0} \in B_x(G, r, \mathbf{c}) \setminus \mathcal{F}$, we conclude that \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

6.3.3 Capacity inequalities

The capacity inequalities fall into the same category as the upper bound inequalities (6.4) in terms of the left-hand side coefficients and right-hand side sign. Given $v \in V$ with $|\delta(v)| = 1$, if $c_v \geq 2$ the associated capacity inequality $x(\delta(v)) \leq c_v$ is redundant, where as if $c_v = 1$, $x(\delta(v)) \leq c_v$ and $x_e \leq 1$ are identical for some $e \in \delta(v)$, which has been discussed in the previous section as the upper bound inequalities. Thus we consider only the case with $|\delta(v)| \geq 2$.

We split the capacity inequalities into three cases, for r , nodes in O and nodes in $V \setminus (O \cup \{r\})$.

Proposition 6.3.7. *Given $|\delta(r)| \geq 2$, $x(\delta(r)) \leq c_r$ defines a facet of $B_x(G, r, \mathbf{c})$ if and only if $|\delta(r)| > c_r$.*

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(r)) = c_r\}$. Clearly, we need $|\delta(r)| \geq c_r$ to ensure that \mathcal{F} is nonempty. If $c_r = |\delta(r)| \geq 2$, then $\mathcal{F} \subsetneq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e = 1\} \subsetneq B_x(G, r, \mathbf{c})$ for any $e \in \delta(r)$.

For the sufficiency, assume that $|\delta(r)| > c_r$ and $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$. Consider an edge set F with $F \subsetneq \delta(r)$ and $|F| = c_r$. One has $\mathbf{x}^F \in \mathcal{F}$, and thus $a(F) = b$. Let e be an edge in F and e' an edge in $\delta(r) \setminus F$. We have that $\mathbf{x}^{F \cup \{e'\} \setminus \{e\}} \in \mathcal{F}$, which leads to $a_e = a_{e'}$. Hence we have

$$\begin{aligned} a_e &= \lambda \quad \forall e \in \delta(r), \\ b &= c_r \lambda, \end{aligned}$$

where $\lambda \in \mathbb{R}$.

Consider an edge $e = uv$ in $E \setminus \delta(r)$ with $u \in N(r)$. As $|\delta(r)| > c_r$, there exists $F \subseteq \delta(r) \setminus \delta(v)$ such that $ru \in F$ and $|F| = c_r$. Then both F and $F \cup \{uv\}$ induce bounded r -trees of G and their incidence vectors are in \mathcal{F} . Hence we have $a_{uv} = 0$.

Consider an edge $e = uv$ in $E \setminus \delta(r)$ with $u, v \in V \setminus N(r)$. There exists a bounded path P_{re} between r and e such that $|V[P_{re}] \cap N(r)| = 1$. There exists a set $F \subseteq \delta(r)$ such that $P_{re} \cap \delta(r) \subseteq F$ and $|F| = c_r$. Then $F \cup P_{re}$ and $F \cup P_{re} \setminus \{uv\}$ induce bounded r -trees of G and their incidence vectors are in \mathcal{F} . Hence we have $a_{uv} = 0$.

Therefore $\mathbf{a}^T \mathbf{x} \leq b$ can be rewritten as $\lambda x(\delta(r)) \leq c_r \lambda$. Combining it with $\mathbf{0} \in B_x(G, r, \mathbf{c}) \setminus \mathcal{F}$ gives us \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

We now consider the capacity inequalities associated with nodes in O .

Proposition 6.3.8. *Let $v \in O$ with $|\delta(v)| \geq 2$. $x(\delta(v)) \leq 1$ defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $c_r \geq 2$;
2. $|\delta(v)| \geq 2$;
3. *there does not exist a bridge between r and v in $G[E \setminus (\delta(O \setminus \{v\}))]$;*
4. *there does not exist an articulation node v_a with $c_{v_a} = 2$ that separates r and v in $G[E \setminus (\delta(O \setminus \{v\}))]$.*

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(v)) = 1\}$. If $c_r = 1$, as $|\delta(v)| \geq 2$, thus $\delta(v) \neq \delta(r)$, and \mathcal{F} is a proper subset of the proper face induced by $x(\delta(r)) \leq c_r$. If $c_v = |\delta(v)| \geq 2$, then $\mathcal{F} \subsetneq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e = 1\} \subsetneq B_x(G, r, \mathbf{c})$ for any $e \in \delta(v)$. Lemma 6.3.3 ensures the necessity of the last two conditions.

For the sufficiency, assume that $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$.

From the conditions we know that there exists two edge disjoint bounded paths P_1 and

P_2 between r and v , and for any $u \in V_c \setminus \{r\}$, one has $c_u \geq 3$, where V_c is defined as

$$V_c = V[P_1] \cap V[P_2] \setminus \{v\}.$$

Note that Both P_1 and P_2 induce bounded r -trees and their incidence vectors belong to \mathcal{F} .

Consider an edge $e = uw \in E[V_c]$. There must exist a subpath P_{uw} of P_i between u and w such that $|P_{uw}| \geq 2$, $i \in \{1, 2\}$. As both P_i and $P_i \cup \{uw\} \setminus P_{uw}$ induce bounded r -trees whose incidence vectors belong to \mathcal{F} , we have $a_{uw} = a(P_{uw}) = 0$. Thus,

$$a_e = 0 \quad \forall e \in E[V_c].$$

Consider an edge $e = uw \in P_1 \setminus (\delta(v) \cup E[V_c])$. There exists a subpath P_z of P_1 between some node $z \in V_c$ and uw such that $V[P_z] \cap V_c \setminus \{z\} = \emptyset$. Then $P_z \cap P_2$ induces a bounded r -tree whose incidence vector is in \mathcal{F} . We deduce that $a_{e'} = 0$ for any $e' \in P_z$. As P_1 and P_2 are interchangeable in this argument, we then have

$$a_e = 0 \quad \forall e \in P_1 \cup P_2 \setminus (\delta(v) \cup E[V_c]).$$

Now consider an edge $e \in E \setminus (\delta(v) \cup E[V_c])$. Let P_{re} be a bounded path between r and e , and let P_{se} be the subpath of P_{re} , such that $V[P_{se}] \cap V[P_1 \cup P_2] = \{s\}$. Without loss of generality, suppose that $s \in V[P_1]$. If $s \notin V_c$, let P_{ws} be the subpath of P_1 such that $w \in V_c$ and $V[P_{ws}] \cap V_c = \{w\}$, otherwise let $w = s$ and $P_{ws} = \emptyset$. We have that $P_1 \cup P_{ws} \cup P_{se}$ and $P_1 \cup P_{ws} \cup P_{se} \setminus \{e\}$ induce two bounded r -trees whose incidence vectors belong to \mathcal{F} . Therefore, we deduce

$$a_e = 0 \quad \forall e \in E \setminus (\delta(v) \cup E[V_c]).$$

For each $e \in \delta(v)$, there exists a bounded path P_{re} between r and e . We have $a_e + a(P_{re} \setminus \{e\}) = b$, which combines with $a(P_{re} \setminus \{e\}) = 0$ gives us

$$a_e = b \quad \forall e \in \delta(v).$$

Therefore $\mathbf{a}^T \mathbf{x} \leq b$ can be rewritten as $bx(\delta(v)) \leq b$, and since $\mathbf{0} \in B_x(G, r, \mathbf{c}) \setminus \mathcal{F}$, we thus conclude that \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

We conclude the discussion on the capacity inequalities with the ones associated with nodes in $V \setminus (O \cup \{r\})$.

Proposition 6.3.9. *Let $v \in V \setminus (O \cup \{r\})$ with $|\delta(v)| \geq 2$. $x(\delta(v)) \leq c_v$ defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $c_r \geq 2$;
2. $|\delta(v)| > c_v$;
3. there does not exist a bridge between r and v in $G[E \setminus \delta(O)]$;
4. there does not exist an articulation node v_a with $c_{v_a} = 2$ that separates r and v in $G[E \setminus \delta(O)]$.

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(v)) = c_v\}$. Clearly we need $c_v \leq |\delta(v)|$ to make \mathcal{F} nonempty. For the necessity, if $c_r = 1$, according to Lemma 6.3.2, it is not facet-defining. If $c_v = |\delta(v)| \geq 2$ then \mathcal{F} is a proper subset of the proper face induced by $x_e = 1$ for any $e \in \delta(v)$. Lemma 6.3.3 ensures the necessity of the rest conditions.

For the sufficiency, assume that $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$.

Let P_1 and P_2 be two edge-disjoint bounded paths between r and v , and without loss of generality assume that P_1 contains as few edges as possible and $|V[P_2] \cap N(v)| \leq 2$. Let V_c be defined as

$$V_c = \{u \in V[P_1] \cap V[P_2] : u = r \text{ or } u \text{ is an inner node of both } P_1 \text{ and } P_2\}.$$

Let F_i be a subset of $\delta(v) \setminus P_i$ such that $V[F_i] \cap V[P_i] = \{v\}$ and $|F_i| = c_v - 1$ for $i \in \{1, 2\}$. We have that $P_i \cup F_i$ induces a bounded r -tree of G and its incidence vector is in \mathcal{F} for $i \in \{1, 2\}$.

Consider any edge $e = uw \in P_i \setminus (\delta(v) \cup E[V_c])$, $i \in \{1, 2\}$, there exists a subpath $P_j^{u'e}$ of P_j between some node $u' \in V_c$ and e such that $V[P_j^{u'e}] \cap V_c = \{u'\}$ with $i \neq j \in \{1, 2\}$. Then $P_i \cup F_i \cup P_j^{u'e}$ and $P_i \cup F_i \cup P_j^{u'e} \setminus \{e\}$ both induce bounded r -trees of G and their incidence vectors belong to \mathcal{F} . Hence, $a_e = 0$.

Consider any edge $e = uw \in E[V_c]$. Note that $e \in P_1$. Let P_2^{uw} be the subpath of P_2 between u and w . $P_1 \cup F_1$ and $P_1 \cup F_1 \cup P_2^{uw} \setminus \{e\}$ induce two bounded r -trees of G and their incidence vectors belong to \mathcal{F} . Hence, $a_e = a(P_2^{uw}) = 0$. To summarize, we have

$$a_e = 0 \quad \forall e \in (P_1 \cup P_2) \setminus \delta(v).$$

For any $e_1, e_2 \in \delta(v) \setminus P_1$ there exists $F'_1 \subseteq \delta(v) \setminus P_1$ such that $|F'_1| = c_v - 2$ and $e_1, e_2 \notin F'_1$. We have that $P_1 \cup F'_1 \cup \{e_1\}$ and $P_1 \cup F'_1 \cup \{e_2\}$ induce two bounded r -trees of G and their incidence vectors belong to \mathcal{F} , which leads to $a_{e_1} = a_{e_2}$. Then as $P_1 \cup F_1$ and $P_2 \cup F_2$ are two bounded r -trees of G and their incidence vectors are in \mathcal{F} , we can deduce that

$a(P_1 \cap \delta(v)) = a_e$ for some $e \in \delta(v) \setminus P_1$. Thus, we have

$$a_e = \lambda \quad \forall e \in \delta(v),$$

where $\lambda \in \mathbb{R}$. Consequently, we deduce that

$$b = c_v \lambda.$$

Consider any edge $e = uw \in E \setminus (P_1 \cup P_2 \cup \delta(v))$ such that there exists a bounded path P_{ru} between r and u that does not pass through w and $v \in V[P_{ru}]$. Without loss of generality assume that $|V[P_{ru}] \cap N(v)| = 2$ and $|P_{ru} \cap \delta(v)| = 2$. There exists an edge set $F \subseteq \delta(v) \setminus (P_{ru} \cup \delta(w))$ with $|F| = c_v - 2$. Then $P_{ru} \cup F$ and $P_{ru} \cup F \cup \{uw\}$ are two bounded r -trees of G and their incidence vectors are in \mathcal{F} , which leads to $a_{uw} = 0$.

Consider now any edge $e = uw \in E \setminus (P_1 \cup P_2 \cup \delta(v))$ such that there exists a bounded path P_{ru} between r and u that does not pass through w and v . Without loss of generality assume that P_{ru} contains as few edges as possible. Let P_{su} be the subpath of P_{ru} , such that $V[P_{su}] \cap V[P_1 \cup P_2] = \{s\}$ with $s \in V[P_i]$, $i \in \{1, 2\}$. If $s \notin V_c$, let P_{ws} be the subpath of P_1 such that $w \in V_c$ and $V[P_{ws}] \cap V_c = \{w\}$, otherwise let $w = s$ and $P_{ws} = \emptyset$. We have that $P_i \cup F_i \cup P_{ws} \cup P_{su}$ and $P_i \cup F_i \cup P_{ws} \cup P_{su} \cup \{e\}$ induce two bounded r -trees whose incidence vectors belong to \mathcal{F} . Thus, one obtains $a_e = 0$, and we therefore have

$$a_e = 0 \quad \forall e \in E \setminus (P_1 \cup P_2 \cup \delta(v)).$$

To summarize, $\mathbf{a}^T \mathbf{x} \leq b$ can be rewritten as $\lambda x(\delta(v)) \leq c_v \lambda$, which combining with $\mathbf{0} \notin \mathcal{F}$ gives us $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$. Hence, \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

6.3.4 Connectivity inequalities

According to Lemma 6.3.1, any connectivity inequality (6.1) associated with a set $S \subseteq V \setminus \{r\}$ that satisfies $\delta(S) \cap \delta(O) \neq \emptyset$ is not facet-defining.

Proposition 6.3.10. *Let $e \in E[S]$, $S \subseteq V \setminus \{r\}$, and $e \in E[S]$. Inequality $x_e - x(\delta(S)) \leq 0$ defines a facet of $B_x(G, r, \mathbf{c})$ only if $\delta(S) \cap \delta(O) = \emptyset$.*

Proof. Assume there exists $f \in \delta(S) \cap \delta(O)$. Clearly there does not exist any bounded r -path that contains f . Hence the face induced by $x_e - x(\delta(S)) \leq 0$ is a proper subset of the face induced by $x_f \geq 0$. \square

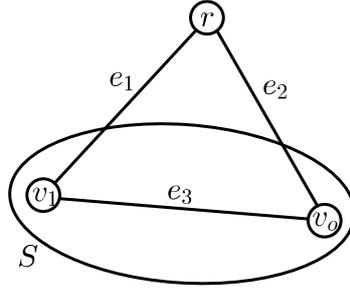


Figure 6.2: An example of inequality (6.6) exclusively defining a facet

As an extension of this proposition, the connectivity inequalities can be rewritten as the following constraint.

$$x_e - x(\delta(S) \setminus \delta(O)) \leq 0, \quad \forall e \in E[S], S \subsetneq V \setminus \{r\}. \quad (6.6)$$

Furthermore, (6.6) covers certain facets which (6.1) does not. For instance, in Figure 6.2, an inequality of (6.6) that defines a facet cannot be written in the form of (6.1), where $c_{v_o} = 1$, and $c_{v_1}, c_r \geq 2$. The inequality $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ defines a facet, as the bounded r -trees $G_\emptyset, G_{\{e_2\}}, G_{\{e_1, e_3\}}$ correspond to 3 affinely independent vectors on the induced face. However there does not exist a node set $S' \subseteq V \setminus \{r\}$ such that $\delta(S') = \delta(S) \setminus \delta(O)$ and $e \in E[S']$. This means this facet is exclusively defined by an inequality in the form of (6.6). Thus from now on, we discuss the facets induced by (6.6) instead of (6.1).

Besides the factor of unit-capacity nodes, the connectivity of the subgraphs $G[S \setminus O]$ and $G[\bar{S} \setminus O]$ should also be guaranteed.

Proposition 6.3.11. *Given $S \subseteq V \setminus \{r\}$ and $e \in E[S]$, $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ defines a facet of $B_x(G, r, \mathbf{c})$ only if $G[S \setminus O]$ is connected and $G[\bar{S} \setminus O]$ is connected.*

Proof. Suppose that $G[S \setminus O]$ ($G[\bar{S} \setminus O]$, respectively) is not connected. Let v be a node in $S \setminus O$ ($\bar{S} \setminus O$, respectively) which does not belong to the same connected component of $G[S \setminus O]$ ($G[\bar{S} \setminus O]$, respectively) as e (r , respectively). For any edge $f \in \delta(v) \setminus \delta(O)$, we have $\{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e - x(\delta(S) \setminus \delta(O)) = 0\} \subsetneq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_f = 0\} \subsetneq B_x(G, r, \mathbf{c})$. \square

If $e \in \delta(v_o)$ with $v_o \in O$, and there exists some $e' \in \delta(v_o) \cap E[S] \setminus \{e\}$, then $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ does not define a facet of $B_x(G, r, \mathbf{c})$.

Proposition 6.3.12. *Let e be an edge in $\delta(v_o) \cap E[S]$ with $v_o \in O \cap S$ and $S \subseteq V \setminus \{r\}$. Inequality $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ defines a facet of $B_x(G, r, \mathbf{c})$ only if $\delta(v_o) \cap E[S] \setminus \{e\} = \emptyset$, as stated in the following proposition.*

Proof. Suppose $e \in \delta(v_o), v_o \in O$, and $\delta(v_o) \cap E[S] \setminus \{e\} \neq \emptyset$. Consider any edge $f \in \delta(v_o) \cap E[S] \setminus \{e\}$. Clearly any bounded r -tree containing f contains at least one edge in $\delta(S) \setminus \delta(O)$ while it does not contain e , which leads to $\{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e - x(\delta(S) \setminus \delta(O)) = 0\} \subsetneq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_f = 0\} \subsetneq B_x(G, r, \mathbf{c})$. \square

There are also conditions similar to the ones described in Lemma 6.3.3, except the fact that Lemma 6.3.3 ensures the graph not to have certain substructures in G between r and some subgraph, whereas here for the connectivity inequalities, the following two conditions prevent the graph from having the same substructures in $G[S]$ between r and the edge e .

Proposition 6.3.13. *Let $S \subseteq V \setminus \{r\}$ and $e \in E[S]$. Inequality $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ defines a facet of $B_x(G, r, \mathbf{c})$, only if there does not exist an edge $e_b \in E[S]$ such that removing $\{e_b\} \cup \delta(O) \setminus \{e\}$ from G can disconnect r and e .*

Proof. Suppose that there is an edge $e_b \in E[S]$ such that removing $e_b \cup \delta(O) \setminus \{e\}$ from G can disconnect r and e . One must have a node set S' with $S' \subsetneq S$, $e \in E[S]$ and $\delta(S') \setminus \delta(O) = \{e_b\}$. Thus we have $\{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e - x(\delta(S) \setminus \delta(O)) = 0\} \subsetneq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e - x(\delta(S') \setminus \delta(O)) = 0\}$. \square

Proposition 6.3.14. *Let $S \subseteq V \setminus \{r\}$ and $e \in E[S]$. Inequality $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ defines a facet of $B_x(G, r, \mathbf{c})$ only if there does not exist a node $v_a \in S$ with $c_{v_a} = 2$ such that removing $\delta(v_a) \setminus \{e\}$ and $\delta(O) \setminus \{e\}$ from G disconnects r and e .*

Proof. Suppose that there exists a node $v_a \in S$ with $c_{v_a} = 2$ such that removing $\delta(v_a) \setminus \{e\}$ and $\delta(O) \setminus \{e\}$ from G disconnects r and e . Then $\{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e - x(\delta(S) \setminus \delta(O)) = 0\} \subsetneq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(v_a)) - c_{v_a}x(\delta(S) \setminus \delta(O)) = 0\}$. \square

Besides what has been stated previously, there is one very specific case where the connectivity inequality is not facet-defining. Figure 6.3 shows an example, with $\bar{S} = \{r\}$, $S = \{u, v, w\}$ and $E[S] = \{e, uv, vw\}$. In this case, the face induced by $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ is a proper subset of the proper face induced by $x(E[S]) - (|S| - 1)x(\delta(S) \setminus \delta(O)) \leq 0$.

Proposition 6.3.15. *Let $e = uv \in E[S]$, $S \subseteq V \setminus \{r\}$. Inequality $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ defines a facet of $B_x(G, r, \mathbf{c})$ only if there does not exist a node $w \in S \cap N(u) \cap N(v)$ that removing wu, vw and $\delta(O) \setminus \{e\}$ from G disconnects r and e .*

Proof. Assume that there exists a node $w \in S \cap N(u) \cap N(v)$ that removing wu, vw and $\delta(O) \setminus \{e\}$ from G disconnects r and e . The face induced by $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ is a proper subset of the proper face induced by $x(E[S']) - (|S| - 1)x(\delta(S) \setminus \delta(O)) \leq 0$, with $S' = \{u, v, w\}$. \square

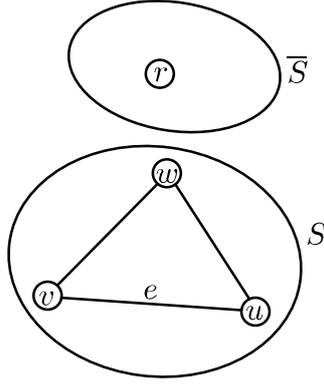


Figure 6.3: Example of a connectivity inequality with a triangle substructure

Proposition 6.3.16. *Let $S \subseteq V \setminus \{r\}$ and $e \in E[S]$. Inequality $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ then defines a facet of $B_x(G, r, \mathbf{c})$ if the conditions in Propositions 6.3.10-6.3.15 are satisfied.*

Proof. Assume the conditions in Proposition 6.3.10- 6.3.15 are satisfied and $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x_e - x(\delta(S) \setminus \delta(O)) = 0\} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$. As $G[\bar{S} \setminus O]$ is connected, for any edge $e' \in E[\bar{S}] \cup (\delta(O) \cap \delta(\bar{S} \setminus O))$ there exists a bounded path $P_{re'}$ between r and e' with $P_{re'} \cap (\delta(S) \setminus \delta(O)) = \emptyset$. One can deduce that

$$a_{e'} = 0 \quad \forall e' \in E[\bar{S}] \cup (\delta(O) \cap \delta(\bar{S} \setminus O)).$$

As $\mathbf{0} \in \mathcal{F}$ and $\mathbf{x}^{P_{re}} \in \mathcal{F}$, $a(P_{re}) = b = 0$. Let $V_c \subseteq S$ be the set of nodes such that they are inner nodes of all bounded paths between r and e . We have from the conditions that $c_v \geq 3$ for any $v \in V_c$.

Consider any edge $f = uv \in E[S] \cup (\delta(O) \cap \delta(S)) \setminus \{e\}$ such that $u \notin V_c$. There exist a bounded path P_{re} between r and e and a bounded path P_f between some node in V_c and f , such that $|P_{re} \cap \delta(S)| = 1$ and it does not pass through u , and $P_{re} \cup P_f$ induces an acyclic graph. It can be deduced from the two bounded r -trees induced by $P_{re} \cup P_f$ and $P_{re} \cup P_f \setminus \{f\}$ that $a_f = 0$.

Consider now an edge $f = uv \in E[S] \setminus \{e\}$ such that $u, v \in V_c$. According to the conditions, f is not a bridge that separates r and e . Then there must exist a bounded path P_{re} between r and e , $f \notin P_{re}$ such that $|P_{re} \cap \delta(S)| = 1$. Let the subpath of P_{re} between u and v be P_{uv} . We have that P_{re} and $P_{re} \cup \{f\} \setminus P_{uv}$ both induce bounded r -trees whose incidence vectors are in \mathcal{F} . Thus $a_f = a(P_{uv}) = 0$.

To summarize, we have

$$a_f = 0 \quad \forall f \in E[S] \setminus \{e\}.$$

Similarly for $f \in \delta(O) \cap \delta(S \setminus O)$, there exist a bounded path P_{re} between r and e and a bounded path P_f between some node in V_c and f , such that $|P_{re} \cap \delta(S)| = 1$ and $P_{re} \cup P_f$ induces an acyclic graph. The incidence vectors of the two bounded r -trees induced by $P_{re} \cup P_f$ and $P_{re} \cup P_f \setminus \{f\}$ belong to \mathcal{F} , which gives us

$$a_f = 0 \quad \forall f \in \delta(O) \cap \delta(S \setminus O).$$

For each edge $f \in \delta(S) \setminus \delta(O)$, as $G[S \setminus O]$ and $G[\bar{S} \setminus O]$ both are connected, there exists a bounded path P_{re} such that $f \in P_{re}$ and $|P_{re} \cap \delta(S) \setminus \delta(O)| = 1$. As $a(P_{re}) = b = 0$ holds, and combining with the results above, one has

$$a_f + a_e = 0 \quad \forall f \in \delta(S) \setminus \delta(O).$$

Therefore $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, which indicates \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

The facial study on the connectivity inequalities thereby concludes. The foregoing discussion suggests that there are several substructures in the support graph of an inequality to be avoided in order to make the inequality facet-defining, such as the bridges and capacity 2 articulation nodes, and there is more to explore for other valid inequalities.

6.3.5 Subtour elimination inequalities

With respect to the subtour elimination inequalities, some necessary conditions are presented as follows.

Proposition 6.3.17. *Let S be a subset of V with $|S| \geq 3$. Inequality $x(E[S]) \leq |S| - 1$ defines a facet of $B_x(G, r, \mathbf{c})$ only if*

1. $c_r \geq 2$;
2. $r \in S$;
3. $S \cap O = \emptyset$;
4. $G[S]$ is 2-connected;
5. there does not exist such node set $W \subseteq S$, such that removing W disconnects $G[S]$ into k connected components, and $|W| + k - 1 \geq c(W)$, $k \geq 2$.

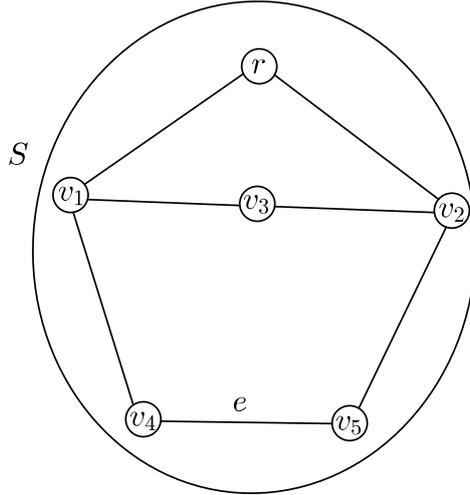


Figure 6.4: An example with a subtour elimination inequality that is not facet-defining

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(E[S]) = |S| - 1\}$.

If $c_r = 1$, according to Lemma 6.3.2, it is not facet-defining. If $r \notin S$, \mathcal{F} is a proper subset of the proper face induced by $x(E[S]) - (|S| - 1)x(\delta(S) \setminus \delta(O)) \leq 0$. If there exists $u \in S \cap O$, one has $x(\delta(u)) = c_u$ for any feasible solution in \mathcal{F} . If there exists an edge e_b in $E[S]$ that removing e_b disconnects $G[S]$, one has that \mathcal{F} is a proper subset of the proper face induced by $x_{e_b} \leq 1$. If $G[S]$ is not 2-connected, then for any block $G[S_i]$ of $G[S]$, \mathcal{F} is a proper subset of the proper face induced by $x(E[S_i]) \leq |S_i| - 1$.

If there exists a set $W \subseteq S$, such that removing $\delta(W)$ disconnects $G[S]$ into k connected components, $|W| + k - 1 \geq c(W)$, $k \geq 2$. Since any vector in the face \mathcal{F} that induces a bounded r -tree spanning all the nodes in S , at least $|W| + k - 1$ edges are needed in $\bigcup_{v \in W} \delta(v)$ to connect the $|W|$ nodes in W and $k - 1$ connected components. If $|W| + k - 1 \geq c(W)$, then either each node in W is saturated which leads to $x(\delta(v)) = c_v$ for $v \in W$ for any $x \in \mathcal{F}$. In other words, \mathcal{F} is a proper subset of the proper face induced by $x(\delta(v)) \leq c_v$ for $v \in W$. \square

Figure 6.4 demonstrates an instance which violates the last condition in the former proposition, where $W = \{v_1, v_2\}$, $c_{v_1} = c_{v_2} = 2$ and $k = 3$. It can be noticed that on the face induced by $x(E[S]) \leq |S| - 1$, one always has that $x(\delta(v_1)) = c_{v_1}$ and $x(\delta(v_2)) = c_{v_2}$ for any vector that induces a bounded r -tree. The substructure to be avoided in this example is related to nodes with arbitrary capacities, rather than only the nodes with capacity two, that create gaps among other nodes in the graph. Figure 6.5 shows, a partition of S , consisting of W, U_1, \dots, U_k , where $\delta(U_i, U_j) = \emptyset$, for any distinct $i, j \in \{1, \dots, k\}$. If $|W| + k - 1 \geq \sum_{v \in W} c_v$, the subtour elimination inequality associated with S is then not

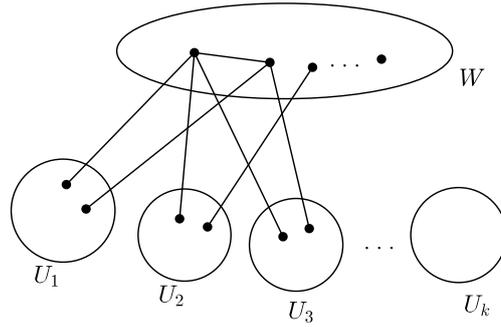


Figure 6.5: General example with an articulation set

facet-defining. Notice here r can be either in W or in any of U_i , $i = 1, \dots, k$.

It can be noted that this substructure is similar to the case with articulation node of capacity 2 mentioned earlier for previous inequalities. In fact, if $k = 2$ and $W = \{v_a\}$, $c_{v_a} = 2$, then the substructure degenerates to a capacity 2 articulation node.

In the next section we introduce several new sets of valid inequalities for $B_x(G, r, \mathbf{c})$ and study the necessary and sufficient conditions to for them be facet-defining.

6.4 New valid inequalities

Besides the inequalities mentioned in the previous section, several other families of inequalities have been found to be facet-defining for $B_x(G, r, \mathbf{c})$ either during our theoretical work or from computational simulations.

6.4.1 Matching-partition inequalities

As mentioned in Chapter 5, the matching-partition inequalities can be extended to the case of $B_x(G, r, \mathbf{c})$.

Recall that the pair (M, π) is called a matching-partition of G , where $M = \{e_1, \dots, e_k\}$ is a matching of G , and $\pi = \{S_0, S_1, \dots, S_k\}$, $k \geq 1$ is a partition of V with $r \in S_0$ such that $e_i \in E[S_i]$ for $i \in \{1, \dots, k\}$. Denote $\mathcal{MP}(G)$, the set composed of all the matching-partitions of G , and by E_π the set of edges having their extremities in different classes of partition π . With any matching-partition $(M, \pi) \in \mathcal{MP}(G)$, one can associate the following matching-partition inequality

$$x(M) - x(E_\pi \setminus \delta(O)) \leq 0. \quad (6.7)$$

It is worth noting that the matching-partition inequality for $B_x(G, r, \mathbf{c})$ need to consider

the capacity factor, specifically the nodes in O , and Lemma 6.3.1 applies here as well. The valid proof is also similar to the case of the r -tree polytope.

Theorem 6.4.1. *For any $(M, \pi) \in \mathcal{MP}(G)$, inequality (6.7) is valid for $B_x(G, r, \mathbf{c})$.*

Proof. Consider any $\mathbf{x}^F \in B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E$, if $x^F(M) - x^F(E_\pi \setminus \delta(O)) \geq 1$, the support graph G_F of \mathbf{x}^F is therefore not connected, as one needs at least $|M \cap F|$ edges among $E_\pi \setminus \delta(O)$ to connect r and edges in $M \cap F$ without violating the capacity constraint. \square

A matching-partition inequality associated with a matching-partition (M, π) that has $|M| = 1$ is also a connectivity inequality from (6.6), for which the necessary and sufficient conditions for it to be facet-defining have already been decided. We hereafter focus on the matching-partition inequalities associated with a matching of cardinality at least 2.

Let G' be the graph obtained from G by first removing O and then contracting each $S_i \in \pi$ into a node, and consequently each set $\delta(S_i, S_j) \setminus \delta(O)$, $S_i \neq S_j \in \pi$, becomes an edge in G' .

Proposition 6.4.2. *Let $(M, \pi) \in \mathcal{MP}(G)$ be a matching-partition with $|M| \geq 2$. Inequality $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$ defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $G[S_i \setminus O]$ is connected for $i \in \{0, 1, \dots, k\}$;
2. G' is 2-connected;
3. $E[S_i] \cap \delta(v_o) \setminus \{e_i\} = \emptyset$ if $e_i \in \delta(v_o)$, $v_o \in O$ for $i \in \{1, \dots, k\}$;
4. there does not exist $w \in S_i \cap N(u_i) \cap N(v_i)$ with $e_i = u_i v_i$ such that removing $\{u_i w, v_i w\} \cup \delta(O)$ from G disconnects e_i and r for $i \in \{1, \dots, k\}$;
5. there does not exist any $e \in E[S_i]$, such that removing $\{e\} \cup \delta(O)$ from G disconnects e_i and r ;
6. there does not exist any $v \in S_i$ with $c_v = 2$, such that removing $\delta(v) \cup \delta(O)$ from G disconnects e_i and r .

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(M) - x(E_\pi \setminus \delta(O)) = 0\}$.

For the necessity, if $G[S_i \setminus O]$ is not connected for some $i \in \{0, 1, \dots, k\}$, then \mathcal{F} is a proper subset of the proper face induced by $x_e \geq 0$ for some $e \in E$. If there exist $S_i, S_j \in \pi$, $i \neq j$, such that removing $\delta(S_i, S_j) \cup \delta(O)$ disconnects G , then let $\pi' = \pi \cup \{S_i \cup S_j\} \setminus \{S_i, S_j\}$. Without loss of generality, assume $i \neq 0$. Let $M' = M \setminus \{e_i\}$. Then \mathcal{F} is a proper subset of the proper face induced by $x(M') - x(E_{\pi'} \setminus \delta(O)) \leq 0$.

If there exists $S_i \in \pi$, such that it corresponds to an articulation node in G' , without loss of generality, assume that G' contains two connected components induced by $\pi_1, \pi_2 \subseteq \pi$

such that $S_i \in \pi_1$, $S_i \in \pi_2$, $\pi_1 \cup \pi_2 = \pi$ and $\delta(S_j, S_q) \setminus \delta(O) = \emptyset$ for any $S_j \in \pi_1 \setminus \{S_i\}$ and $S_q \in \pi_2 \setminus \{S_i\}$. Let

$$S_i^1 = \bigcup_{S_j \in \pi_2} S_j,$$

$$S_i^2 = \bigcup_{S_j \in \pi_1} S_j.$$

Then one can obtain two matching-partitions of G by substituting S_i for S_i^1 and S_i^2 respectively. Particularly, let (M'_1, π'_1) and (M'_2, π'_2) be two matching-partitions of G such that

$$\pi'_1 = \pi_1 \cup S_i^1 \setminus \{S_i\},$$

$$\pi'_2 = \pi_2 \cup S_i^2 \setminus \{S_i\},$$

and

$$M'_1 = \begin{cases} M \setminus E[S_i^1] & \text{if } S_0 \subseteq S_i^1, \\ M \setminus (E[S_i^1] \setminus \{e_i\}) & \text{otherwise,} \end{cases}$$

$$M'_2 = \begin{cases} M \setminus E[S_i^2] & \text{if } S_0 \subseteq S_i^2, \\ M \setminus (E[S_i^2] \setminus \{e_i\}) & \text{otherwise.} \end{cases}$$

Then $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$ is a linear combination of $x(M'_1) - x(E_{\pi'_1} \setminus \delta(O)) \leq 0$ and $x(M'_2) - x(E_{\pi'_2} \setminus \delta(O)) \leq 0$, and thus it is not facet-defining.

If $e_i \in \delta(v_o)$, $v_o \in O$ for some $i \in \{1, \dots, k\}$, and there exists some edge $e' \in E[S_i] \cap \delta(v_o) \setminus \{e_i\}$, then \mathcal{F} is a proper subset of the proper face induced by $x_{e'} \geq 0$.

If there exists $w \in S_i \cap N(u_i) \cap N(v_i)$ with $e_i = u_i v_i$ such that removing $\{u_i w, v_i w\} \cup \delta(O)$ from G disconnects e_i and r for some $i \in \{1, \dots, k\}$, then there exists a subset S' of S_i such that $e \in E[S']$ and $\delta(S') \setminus \delta(O) = \{u_i w, v_i w\} \setminus \delta(O)$. \mathcal{F} is then a proper subset of the proper face induced by $x_{e_i} - x(\delta(S') \setminus \delta(O)) \leq 0$.

If there exists an edge $e \in E[S_i]$, $i \in \{1, \dots, k\}$, such that removing it disconnects e_i from r , then let $M' = M \cup \{e\} \setminus \{e_i\}$ and \mathcal{F} is a proper subset of the proper face induced by $x(M') - x(E_\pi \setminus \delta(O)) \leq 0$.

If there exists a node $v \in S_i$ with $c_v = 2$, $i \in \{1, \dots, k\}$, such that removing $\delta(v) \cup \delta(O) \setminus \{e_i\}$ disconnects e_i from r , there must exist a set $S \subseteq S_i$ such that $v \in S$, $e_i \in E[S]$ and $\delta(S) \setminus \delta(O) \subseteq \delta(v_a) \setminus (\delta(O) \cup \{e_i\})$. Then \mathcal{F} is a proper subset of the proper face induced by $x(\delta(v)) - c_v \delta(S) \setminus \delta(O) \leq 0$.

For the sufficiency, assume $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where

$\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$. One first has $b = 0$ because of $\mathbf{0} \in \mathcal{F}$.

As $G[S_0 \setminus O]$ is connected, for any edge $e \in E[S_0] \cup (\delta(S_0 \setminus O) \cap \delta(O))$, any bounded r -path with all inner nodes in $S_0 \setminus O$ is a feasible solution whose incidence vector is in \mathcal{F} , which gives us

$$a_e = 0 \quad \forall e \in E[S_0] \cup (\delta(S_0 \setminus O) \cap \delta(O)).$$

Notice that for any $i \in \{1, \dots, k\}$, the necessary and sufficient conditions for the connectivity inequality associated with e_i and S_i are all satisfied. Moreover, from the conditions, for each edge $e \in \delta(S_i) \setminus \delta(O)$, there exists a bounded r -tree whose incidence vector is in \mathcal{F} such that it contains e . Thus, using a similar argument as for the edges in $E[S] \cup (\delta(O) \cap \delta(S)) \setminus \{e\}$ and $\delta(S) \setminus \delta(O)$ of Proposition 6.3.16, one can deduce that

$$\begin{aligned} a_e &= 0 & \forall e \in E[S_i] \cup (\delta(O) \cap \delta(S_i)) \setminus \{e_i\}, \\ a_f + a_{e_i} &= 0 & \forall f \in \delta(S_i) \setminus \delta(O). \end{aligned}$$

Furthermore, as G' is 2-connected, we have that for any $S_i, i \in \{1, \dots, k\}$, there exists $S_j, i \neq j \in \{1, \dots, k\}$ such that $\delta(S_i) \cup \delta(S_j) \setminus \delta(O) \neq \emptyset$. Consequently, we have

$$a_f + a_{e_i} = 0 \quad \forall f \in E_\pi \setminus \delta(O), e_i \in M.$$

Therefore, $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, which indicates \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

6.4.2 Acyclicity-connectivity inequalities

Let $W \subseteq S \subseteq V \setminus \{r\}, |W| \geq 2$. The *acyclicity-connectivity inequality* is defined as follows.

$$x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) \leq 0 \tag{6.8}$$

Theorem 6.4.3. *For any $W \subseteq S \subseteq V \setminus \{r\}, |W| \geq 2$, inequality (6.8) is valid for $B_x(G, r, \mathbf{c})$.*

Proof. Consider any $\mathbf{x}^* \in B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E$, and assume $x^*(E[W]) - (|W| - 1)x^*(\delta(S) \setminus \delta(O)) \geq 1$. If $x^*(\delta(S) \setminus \delta(O)) = 0$, we have that $x_e^* - x^*(\delta(S) \setminus \delta(O)) > 0$ for some edge e in $E[W]$ with $x_e^* = 1$. If $x^*(\delta(S) \setminus \delta(O)) \geq 1$, we then have $x^*(E[W]) > |W| - 1$. In both cases \mathbf{x}^* is not feasible, which is a contradiction. \square

Note that if $|W| = 2$, it becomes the connectivity inequalities (6.6). Thus we hereafter

only consider the case with $|W| \geq 3$. Some necessary conditions for (6.8) to be facet-defining is given below.

Proposition 6.4.4. *Let $W \subseteq S \subseteq V \setminus \{r\}$, $|W| \geq 3$. Inequality (6.8) defines a facet of $B_x(G, r, \mathbf{c})$ only if*

1. $G[S \setminus O], G[\bar{S} \setminus O]$ are connected;
2. $W \cap O = \emptyset$;
3. $G[W]$ is 2-connected;
4. there exists no such node $v_a \in S$ that $c_{v_a} = 2$ and removing $\delta(\{v_a\} \cup O)$ from G disconnects $W \setminus \{v_a\}$ and r ;
5. there does not exist $e_b \in E[S] \setminus E[W]$ such that removing e_b and $\delta(O)$ from G disconnects S and r .

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) = 0\}$. If $G[S \setminus O]$ is not connected, one has that \mathcal{F} is a proper subset of the proper face induced by $x_e \geq 0$ for some $e \in \delta(v)$, where $v \in S \setminus O$ is in the connected component of $G[S \setminus O]$ that does not contain W .

If $G[\bar{S} \setminus O]$ is not connected, one has that \mathcal{F} is a proper subset of the proper face induced by $x_e \geq 0$ for some $e \in \delta(v)$, where $v \in \bar{S} \setminus O$ is in the connected component of $G[\bar{S} \setminus O]$ that does not contain r .

If there exists $v_o \in S \cap O$, \mathcal{F} is then a proper subset of the proper face induced by $x(\delta(v_o) \setminus \delta(S)) - c_{v_o}x(\delta(S) \setminus \delta(O)) \leq 0$.

If $G[W]$ is not 2-connected, then for any block $G[W_i]$ of $G[W]$, \mathcal{F} is a proper subset of the proper face induced by $x(E[W_i]) - (|W_i| - 1)x(\delta(S) \setminus \delta(O)) \leq 0$.

If there exists an edge e_b in $E[S] \setminus E[W]$ that removing e_b and $\delta(O)$ from G disconnects S and r , \mathcal{F} is a proper subset of the proper face induced by $x(E[W]) - (|W| - 1)x(\delta(S') \setminus \delta(O)) \leq 0$, where $S' \subsetneq S$ and $\delta(S') \setminus \delta(O) = \{e_b\}$.

If there exists a node $v_a \in S$ such that $c_{v_a} = 2$ and removing $\delta(\{v_a\} \cup O)$ disconnects $W \setminus \{v_a\}$ and r , then \mathcal{F} is a proper subset of the proper face induced by $x(\delta(v_a)) - c_{v_a}x(\delta(S) \setminus \delta(O)) \leq 0$. \square

6.4.3 Upload capacity inequalities

During the study of the polytope, some examples are found to have fractional extreme points such as the one demonstrated in Figure 6.6. This extreme point is decided by the

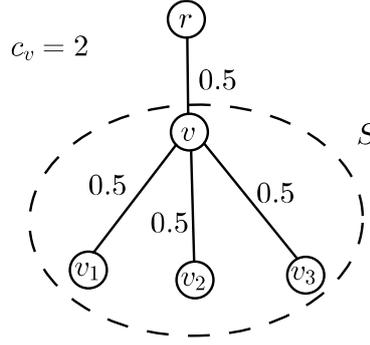


Figure 6.6: An example of upload capacity inequality cutting off a fractional extreme point

following equations.

$$\begin{aligned} x(\delta(v)) &= c_v, \\ x_{vv_1} - x_{rv} &= 0, \\ x_{vv_2} - x_{rv} &= 0, \\ x_{vv_3} - x_{rv} &= 0. \end{aligned}$$

It can be cut by the inequality $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) \leq 0$.

Let $S \subseteq V \setminus \{r\}$ with $v \in S \setminus O$. The associated *upload-capacity inequality* is

$$x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) \leq 0. \quad (6.9)$$

Theorem 6.4.5. *For any $S \subseteq V \setminus \{r\}$, $v \in S \setminus O$, inequality (6.9) is valid for $B_x(G, r, \mathbf{c})$.*

Proof. Consider any $\mathbf{x}^* \in B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E$, and assume $x^*(\delta(v)) - c_v x^*(\delta(S) \setminus \delta(O)) \geq 1$. If $x^*(\delta(S) \setminus \delta(O)) = 0$, the connectivity inequality associated with some edge in $\delta(v)$ is then violated. If $x^*(\delta(S) \setminus \delta(O)) \geq 1$, the capacity of v is then exceeded by \mathbf{x}^* . \square

Proposition 6.4.6. *Inequality (6.9) defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $|\delta(v) \setminus (\delta(S) \setminus \delta(O))| \geq c_v, |\delta(v)| \geq c_v + 1$;
2. $G[S \setminus O]$ and $G[\overline{S} \setminus O]$ are connected respectively;
3. if $\delta(v) \cap \delta(S) \setminus \delta(O) = \emptyset$ there exists no such edge $e_b \in E[S] \cup \delta(S)$ that removing $\delta(O) \cup \{e_b\}$ from G disconnects v and r ;
4. if $\delta(v) \cap \delta(S) \setminus \delta(O) = \emptyset$ there exists no such node $v_a \in S \setminus \{v\}$ that $c_{v_a} = 2$ and removing $\delta(O) \cup \delta(v_a)$ from G disconnects v and r .

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) = 0\}$.

For the necessity, if $|\delta(v) \setminus (\delta(S) \setminus \delta(O))| = c_v - 1$ or $|\delta(v)| = c_v$, then \mathcal{F} is a proper subset of the proper face induced by $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ for all $e \in \delta(v) \setminus (\delta(S) \setminus \delta(O))$. If $G[S \setminus O]$ is not connected or $G[\bar{S} \setminus O]$ is not connected, then \mathcal{F} is a proper subset of the proper face induced by $x_e \geq 0$ for some $e \in E$. If there exists an edge $e_b \in E[S] \cup \delta(S)$ such that removing $\delta(O) \cup \{e_b\}$ disconnects v and r , let S' be a subset of S such that $\delta(S') \setminus \delta(O) = \{e_b\}$, $v \in S'$. Then \mathcal{F} is a proper subset of the proper face induced by $x(\delta(v)) - c_v x(\delta(S') \setminus \delta(O)) \leq 0$. If there exists such node $v_a \in S \setminus \{v\}$ that $c_{v_a} = 2$ and removing $\delta(O) \cup \delta(v_a)$ disconnects v and r , one has that \mathcal{F} is a proper subset of the proper face induced by $x(\delta(v_a)) - c_{v_a} x(\delta(S) \setminus \delta(O)) \leq 0$.

For the sufficiency, assume $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$. One first has $b = 0$ because of $\mathbf{0} \in \mathcal{F}$.

For any edge $e \in E[\bar{S}] \cup (\delta(O) \cap \delta(\bar{S} \setminus O))$, the shortest re -path with all inner nodes in $\bar{S} \setminus O$ corresponds to a feasible solution in \mathcal{F} , from which one can deduce $a_e = 0$.

Case 1. If $\delta(v) \cap \delta(S) \setminus \delta(O) \neq \emptyset$.

For any $e \in \delta(v) \cap \delta(S) \setminus \delta(O)$, as $|\delta(v) \setminus (\delta(S) \setminus \delta(O))| \geq c_v$, the bounded r -tree induced by combining the shortest re -path with only inner nodes in \bar{S} and any $c_v - 1$ edges in $\delta(v) \setminus (\delta(S) \setminus \delta(O))$ satisfies $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) = 0$. Let $D_S = \delta(v) \setminus (\delta(S) \setminus \delta(O))$ and $D_\delta = \delta(v) \cap \delta(S) \setminus \delta(O)$. One has for any $e \in D_\delta$, $e' \in D_S$, $a_e = -(c_v - 1)a_{e'}$.

For any $e \in E[S] \setminus D_S$, one must have $F \subseteq D_S$ and a path $P_{ve} \subseteq E[S]$ from r to e , with $|F| = c_v - 1$ and $|P_{ve} \cap F| = |P_{ve} \cap \delta(v)| = 1$, which gives us $a_e = 0$.

For any $e \in \delta(S) \setminus (\delta(v) \cup \delta(O))$, there must exist a path P_{ev} from e to v and a set $F \subseteq D_S$ with $|F| = c_v$ such that $P_{re} \cup P_{ev} \cup F$ induces a bounded r -tree, where P_{re} has all inner nodes in \bar{S} . Thus $a_e = -c_v a_{e'}$ holds for any $e \in \delta(S) \setminus (\delta(v) \cup \delta(O))$ and $e' \in D_S$.

For $e \in \delta(O) \cap \delta(S)$, without loss of generality, assume $e \in \delta(v_o)$, $v_o \in O$. If $v_o \in S$, one must have an re -path with all inner nodes in \bar{S} , which leads to $a_e = 0$. If $v_o \in \bar{S}$, there must exist a path P_{ve} from v to e , an edge $e' \in D_\delta$, a path $P_{re'}$ and a set $F \subseteq D_S$, where $|F| = c_v - 1$, $|P_{ve} \cap F| = |P_{ve} \cap \delta(v)| = 1$, P_{ve} has all inner nodes in S , and $P_{re'}$ has all inner nodes in \bar{S} . $P_{re'} \cup P_{ve} \cup F$ then induces a bounded r -tree that satisfies $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) = 0$, which leads to $a_e = 0$.

Finally, One has $\lambda x(D_S) - (c_v - 1)\lambda x(D_\delta) - c_v \lambda x(\delta(S) \setminus (\delta(v) \cup \delta(O))) = 0$. As $\lambda \neq 0$, it is equivalent to $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) = 0$, which indicates \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$.

Case 2. If $\delta(v) \cap \delta(S) \setminus \delta(O) = \emptyset$.

According to the conditions, there must exist two rv -paths P_1, P_2 such that $P_1 \cap P_2 \cap (E[S] \cup \delta(S)) = \emptyset$, and $|P_1 \cap \delta(S) \setminus \delta(O)| = |P_2 \cap \delta(S) \setminus \delta(O)| = 1$, and there exists no such node $v_a \in V[P_1] \cap V[P_2] \cap S \setminus \{v\}$ with $c_{v_a} = 2$. Let $P_1 \cap \delta(S) \setminus \delta(O) = \{e_{\delta 1}\}$, $P_2 \cap \delta(S) \setminus$

$\delta(O) = \{e_{\delta 2}\}$, $P_1 \cap \delta(v) = \{e_{v1}\}$, $P_1 \cap \delta(v) = \{e_{v2}\}$, and P_{re_1}, P_{re_2} be the paths with all inner nodes in \bar{S} . As $|\delta(v)| \geq c_v + 1$, there must exist $F \subseteq \delta(v) \setminus \{e_1\}$, such that $|F| = c_v - 1$, and $P_{re_1} \cup P_1 \cup F$ induces a bounded r -tree that satisfies $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) = 0$. Without loss of generality, assume P_1 is the shortest path satisfying the aforementioned conditions, then P_1 and e_2 do not form a cycle, then by replacing any edge $e \in F$ by any $e' \in \delta(v) \setminus (F \cup \{e_1\})$, one has another bounded r -tree whose incidence vector belongs to \mathcal{F} , which gives us that for all $e \in \delta(v) \setminus \{e_1\}$, $a_e = \lambda$. If the graph induced by P_2 and e_1 does not contain a cycle, a similar argument can be applied on P_2 , which leads to for all $e \in \delta(v) \setminus (F \cup \{e_2\})$, $a_e = \lambda'$. If the graph induced by P_2 and e_1 contains a cycle, $P_{re_2} \cup P_2 \cup F' \cup \{e_1\} \setminus \{e_2\}$ induces a feasible solution in \mathcal{F} which leads to $a_{e_1} = a_{e_2}$, where $F' \subseteq \delta(v) \setminus \{e_2\}$, and $|F'| = c_v - 1$. To summarize, in both cases, for any $e \in \delta(v)$, one has $a_e = \lambda$.

For any edge $e \in E[S] \setminus (\delta(v) \cup P_1)$ or $e \in \delta(S) \cap \delta(O) \setminus \delta(v)$, based on the conditions, we claim that there must exist, a path P_{ve} from v to e with all inner nodes in S , and a set $F \subseteq \delta(v) \setminus \{e_1\}$ with $|F| = c_v - 1$, such that the graph induced by $F \cup P_{ve}$ does not contains a cycle. Moreover, $P_1 \cup F \cup P_{ve}$ induces a bounded r -tree in face \mathcal{F} , which gives us $a_e = 0$.

For any edge $e \in E[S] \cap P_1$, through a similar argument, one can prove, if the graph induced by $e \cup P_2$ does not contain a cycle, $P_2 \cup F \cup P_{ve}$ is a feasible solution in face \mathcal{F} , with some $F \subseteq \delta(v) \setminus \{e_2\}$ and $|F| = c_v - 1$. If the graph induced by $e \cup P_2$ contains a cycle, $P_2 \cup F \cup P_{ve} \setminus P_e$ induces a feasible solution in face \mathcal{F} , where $P_e \subseteq e \in E[S] \setminus (\delta(v) \cup P_1)$. Therefore $a_e = 0$.

For any edge $e \in \delta(S) \setminus (\delta(O) \cup \delta(v))$, one must have a path P_{re} from r to e with $P_{re} \setminus \{e\} \subseteq E[\bar{S}] \setminus \delta(O)$, a path P_{ev} from e to v with $P_{ev} \cap \delta(S) = \{e\}$, $|P_{ev} \cap \delta(v)| = 1$, and $P_{ev} \setminus (\delta(S) \cup \delta(v)) \subseteq E[S] \setminus \delta(v)$, a set $F \subseteq \delta(v)$ with $|F| = c_v$, $|F \cap P_{ev}| = 1$, such that $P_{re} \cup P_{ev} \cup F$ induces a bounded r -tree in face \mathcal{F} . Combining with aforementioned results, one has $a_e = -c_v \lambda$ for any $e \in \delta(S) \setminus (\delta(O) \cup \delta(v))$.

To conclude, the face $\{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\}$ and \mathcal{F} are identical, which proves that \mathcal{F} is maximal. \square

The upload capacity of nodes in O has to be expressed slightly different from other nodes. For any $S \subseteq V \setminus \{r\}$, $v_o \in S \cap O$, let $U = \{v \in S \setminus O : \delta(v) \cap \delta(S) \neq \emptyset\}$. The upload capacity inequality for $v_o \in O$ is presented as follows.

$$x(\delta(v_o) \setminus \delta(S)) - x(\delta(S) \setminus \delta(O)) \leq 0 \quad (6.10)$$

If $|\delta(v_o) \setminus \delta(S)| = 1$, it becomes the connectivity inequalities (6.6), therefore we consider only the case $|\delta(v_o) \setminus \delta(S)| \geq 2$ hereafter.

Theorem 6.4.7. *For any $v_o \in S \cap O$, $S \subseteq V \setminus \{r\}$ with $|\delta(v_o) \setminus \delta(S)| \geq 2$, inequality (6.10) is valid for $B_x(G, r, \mathbf{c})$.*

Proof. Consider any $\mathbf{x}^* \in B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E$, and assume $x^*(\delta(v_o) \setminus \delta(S)) - x^*(\delta(S) \setminus \delta(O)) \geq 1$. If $x^*(\delta(S) \setminus \delta(O)) = 0$, the connectivity inequality associated with S and some edge in $\delta(v_o) \setminus \delta(S)$ is then violated. If $x^*(\delta(S) \setminus \delta(O)) \geq 1$, the capacity constraint of v_o is then violated by \mathbf{x}^* . \square

Proposition 6.4.8. *For any $v_o \in O$, $S \subseteq V \setminus \{r\}$ with $|\delta(v_o) \setminus \delta(S)| \geq 2$, inequality (6.10) defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $G[S \setminus O]$ is connected, $G[\bar{S} \setminus O]$ is connected;
2. there does not exist an edge $e \in E[S]$ such that removing $e \cup \delta(O \setminus v_o)$ from G disconnects r and v_o ;
3. there does not exist a node $v \in S \setminus \{v_o\}$ with $c_v = 2$ such that removing $\delta(v) \cup \delta(O \setminus v_o)$ from G disconnects r and v_o ;
4. if $E[S \setminus O] \setminus E[U] \neq \emptyset$ there exists $v \in U$ with $c_v \geq 3$.

Proof. Let $\mathcal{F} = \{x(\delta(v_o) \setminus \delta(S)) - x(\delta(S) \setminus \delta(O)) = 0\}$.

For the necessity, if either $G[S \setminus O]$ or $G[\bar{S} \setminus O]$ is not connected, then \mathcal{F} is a proper subset of the proper face induced by $x_e \leq 0$ for some $e \in E$.

If there exists an edge $e_b \in E[S]$ such that removing $e_b \cup \delta(O \setminus v_o)$ disconnects r and v_o , then with a node set $S' \subsetneq S$ that satisfies $\delta(S') \setminus \delta(O) = e_b$, \mathcal{F} is a proper subset of the proper face induced by $x(\delta(v_o) \setminus \delta(S')) - x(\delta(S') \setminus \delta(O)) \leq 0$.

If there exists a node $v_a \in S$ such that removing $\delta(v_a) \cup \delta(O \setminus \{v_o\})$ disconnects r and v_o , and $c_{v_a} = 2$, then any feasible solution in \mathcal{F} also satisfies $x(\delta(v_a)) - c_{v_a}x(\delta(S) \setminus \delta(O)) = 0$.

If $E[S \setminus O] \setminus E[U] \neq \emptyset$, and there does not exist $v \in U$, such that $c_v \geq 3$, then \mathcal{F} is a proper subset of the proper face induced by $x_e \leq 0$ for some $e \in E[S \setminus O] \setminus E[U]$.

For the sufficiency, assume $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$. One first has $b = 0$ because of $\mathbf{0} \in \mathcal{F}$.

For any edge $e \in E[\bar{S}] \cup (\delta(O) \cap \delta(\bar{S} \setminus O)) \cup (\delta(v_o) \cap \delta(S))$, a bounded re -path with all inner nodes in $S \setminus O$ exists and corresponds to a feasible solution in \mathcal{F} , from which one can deduce $a_e = 0$.

If $\delta(v_o) \setminus \delta(S) = E[S]$, one can deduce $a_{e_0} = -a_e$ for any $e_0 \in \delta(v_o) \setminus \delta(S)$, and $e \in \delta(S) \setminus \delta(O)$.

If $\delta(v_o) \setminus \delta(S) \neq E[S]$, for any edge $e \in E[S \setminus O] \setminus E[U]$, Conditions (2), (3), (4) ensure that there must exist a node $u \in U$ with $c_u \geq 3$, and two paths P_{ue} between u and e , and

P_{uv_o} between u and v_o that satisfy $V[P_{ue}] \cap V[P_{uv_o}] = \{u\}$. Therefore, we have $a_e = 0$ for any $e \in E[S \setminus O] \setminus E[U]$.

For any $e = uv \in E[U]$, let P_{ru} and P_{rv} be the two bounded paths between r and u, v respectively, such that $|P_{rv} \cap \delta(S) \setminus \delta(O)| = |P_{ru} \cap \delta(S) \setminus \delta(O)| = 1$. Such paths exist because of the definition of U . Let P_{uv_o} (P_{vv_o} , respectively) be a bounded path of $G[S]$ between u (v , respectively) and v_o . We have that $P_{ru} \cup P_{uv_o}$, $P_{rv} \cup P_{vv_o}$, $P_{rv} \cup \{uv\} \cup P_{uv_o}$ and $P_{ru} \cup \{uv\} \cup P_{vv_o}$ all induce bounded r -trees of G and their incidence vectors belong to \mathcal{F} . Thus we deduce $a_{uv} = 0$.

Furthermore, as $G[S \setminus O]$ is connected, we have for any distinct $e_1, e_2 \in \delta(v_o) \setminus \delta(S)$ there exist two paths P_{ue_1}, P_{ue_2} between any $u \in U$ and e_1, e_2 . Combining them with a bounded path P_{ru} with $P_{ru} \cap \delta(S) \setminus \delta(O) = \{uv\}$ gives us two bounded r -trees of G whose incidence vectors are \mathcal{F} . Hence we have $a_{e_1} = a_{e_2} = -a_{uv}$ for any $e_1, e_2 \in \delta(v_o) \setminus \delta(S)$ and any $uv \in \delta(S) \setminus \delta(O)$.

Therefore, one has $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) \subseteq B_x(G, r, \mathbf{c})\}$, which completes the proof. \square

Moreover, we can generalize the upload-capacity inequalities for both nodes in O and in $V \setminus (O \cup \{r\})$ as follows. Given $v \in S \subseteq V \setminus \{r\}$, the associated upload-capacity inequality is

$$x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0. \quad (6.11)$$

It can be noted that the upload-capacity inequalities consider both the capacity aspect and the connectivity between r and v .

In following sections, we introduce a family of new valid inequalities, that are similar in a few aspects. Specifically, they generally consider a set of articulation nodes with the same capacity. We start with the one set with the simplest presentation, and generalize it to a wider extent.

6.4.4 Capacity-2 inequalities

Let $S_2 = \{v \in V \setminus \{r\} : c_v = 2\}$. Given $S \subseteq S_2$, $R = V \setminus (S \cup O)$, the *capacity-2 inequality* is as follows.

$$x(\delta(O)) - x(\delta(R)) \leq 0. \quad (6.12)$$

Proposition 6.4.9. *Inequality (6.12) is valid for $B_x(G, r, \mathbf{c})$.*

Proof. Consider any edge set $F \subseteq E$ that induces a bounded r -tree of G . Let G'_F be the graph obtained from G_F by contracting R into a node r' . We have that G'_F is composed

of at most $|F \cap \delta(r')|$ edge-disjoint paths between r' and some edge $e \in F$. As each of the edges in $F \cap \delta(O)$ has to be an end edge of a path in G_F , we have that G_F contains at most $|F \cap \delta(r')|$ edges in $F \cap \delta(O)$, that is, $x^F(\delta(O)) - x^F(\delta(R)) \leq 1$. \square

Proposition 6.4.10. *Inequality (6.12) defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $G[R]$ is connected, $G[S]$ is connected;
2. for any $v \in S$, $\delta(v) \cap \delta(R) \neq \emptyset$, and $\delta(v) \cap \delta(O) \neq \emptyset$.

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(O)) - x(\delta(R)) = 0\}$.

For the necessity, if $G[R]$ is not connected, \mathcal{F} is a proper subset of the proper face induced by $x_e \leq 0$ for some $e \in E[R]$. If $G[S]$ is not connected, without loss of generality, assume it has two connected components induced by node sets S_1 and S_2 , with $S = S_1 \cup S_2$. \mathcal{F} is a proper subset of the proper face induced by either $x(\delta(O)) - x(\delta(R \cup S_1)) \leq 0$ or $x(\delta(O)) - x(\delta(R \cup S_2)) \leq 0$. If there exists $v \in S$ such that $\delta(v) \cap \delta(R) = \emptyset$, or $\delta(v) \cap \delta(O) = \emptyset$, one has that there exists $S' \subsetneq S$ such that \mathcal{F} is a proper subset of the proper face induced by $x(\delta(O)) - x(\delta(R \cup S')) \leq 0$.

For the sufficiency, assume $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$. One first has $b = 0$ because of $\mathbf{0} \in \mathcal{F}$. For any edge $e \in E[R] \cup \delta(R) \cap \delta(O)$, as $G[R]$ is connected and $R \cap O = \emptyset$, one has $a_e = 0$. For any node $v \in S$, as $\delta(v) \cap \delta(R) \neq \emptyset$, and $\delta(v) \cap \delta(O) \neq \emptyset$, one has $a_e + a_f = 0$ for any $e \in \delta(v) \cap \delta(O)$ and any $f \in \delta(v) \cap \delta(R)$. Moreover, since $G[S]$ is connected, for any edge $v_1 v_2 \in E[S]$, and any $e_1 \in \delta(v_1) \cap \delta(O)$, $e_2 \in \delta(v_2) \cap \delta(O)$, $f_1 \in \delta(v_1) \cap \delta(R)$, $f_2 \in \delta(v_2) \cap \delta(R)$, one has $a_{e_1} + a_{v_1 v_2} + a_{f_2} = 0$ and $a_{e_2} + a_{v_1 v_2} + a_{f_1} = 0$, which leads to $a_{v_1 v_2} = 0$. Consequently, $a_e + a_f = 0$ holds for any $e \in \delta(O) \cap \delta(S)$ and any $f \in \delta(R) \cap \delta(S)$. Therefore, $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\}$, which indicates \mathcal{F} is a facet of $B_x(G, r, \mathbf{c})$. \square

This set of inequalities consider the aspects of capacity and the connectivity, although in a more ambiguous way. We show in the next part that it can be extended into several other sets of valid inequalities.

6.4.5 Capacity- i inequalities

The capacity-2 inequalities can be extended to a more generalized situation associating with nodes having same arbitrary capacity. Let

$$S_i = \{v \in V \setminus \{r\} : c_v = i, i \geq 2\}.$$

Given $S \subseteq S_i$, let $R = V \setminus (S \cup O)$, and the *capacity- i inequality* is as follows.

$$x(\delta(O) \setminus \delta(R)) - (i-2)x(E[S]) - (i-1)x(\delta(R) \setminus \delta(O)) \leq 0. \quad (6.13)$$

Proposition 6.4.11. *Inequality (6.13) is valid.*

Proof. Assume \bar{x} is an integer feasible solution to the problem. Consider the edges in the solution as pipes that consume and provide resource (i.e. capacities in our case) at the same time. As the edges in $E[R]$ do not participate in inequality (6.12) by any means, it is safe to say that each edge $e \in \delta(R) \setminus \delta(O)$ provides i units of capacity as it connects one node in S_i , and consumes one unit of capacity while connecting the node. For any edge $e \in E[S]$, it consumes 2 units of capacity of nodes in S_i , and provides i unit of capacity as it can be seen as connecting to a new node in S_i . Any edge in $e \in \delta(O) \setminus \delta(R)$ consumes one unit of capacity of nodes in S_i and provides none. So the total capacity provided is $i\bar{x}(\delta(R) \setminus \delta(O)) + i\bar{x}(E[S])$, and total capacity consumed is $\bar{x}(\delta(R) \setminus \delta(O)) + 2\bar{x}(E[S]) + \bar{x}(\delta(O) \setminus \delta(R))$. Since the capacity consumed cannot exceed the capacity provided, one has $\bar{x}(\delta(O) \setminus \delta(R)) - (i-2)\bar{x}(E[S]) - (i-1)\bar{x}(\delta(R) \setminus \delta(O)) \leq 0$. \square

Proposition 6.4.12. *Inequality (6.13) defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $G[R]$ is connected, $G[S]$ is connected;
2. for any $v \in S$, $\delta(v) \cap \delta(R) \neq \emptyset$, and $\delta(v) \cap \delta(O) \neq \emptyset$, and $|\delta(v) \cap \delta(O)| \geq i-1$.

Proof. Let $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : x(\delta(O) \setminus \delta(R)) - (i-2)x(E[S]) - (i-1)x(\delta(R) \setminus \delta(O)) = 0\}$.

For the necessity, if $G[R]$ is not connected, one has $x_e = 0$ for some $e \in E[R]$. If $G[S]$ is not connected, without loss of generality, assume it has two connected components induced by node sets F_1 and F_2 , with $S_i = F_1 \cup F_2$. One has that \mathcal{F} is a proper subset of either of the proper faces induced by

$$x(\delta(O) \setminus \delta(R \cup F_1)) - (i-2)x(E[S \setminus F_1]) - (i-1)x(\delta(R \cup F_1) \setminus \delta(O)) \leq 0$$

and

$$x(\delta(O) \setminus \delta(R \cup F_2)) - (i-2)x(E[S \setminus F_2]) - (i-1)x(\delta(R \cup F_2) \setminus \delta(O)) \leq 0.$$

If there exists $v \in S$ such that $\delta(v) \cap \delta(R) = \emptyset$, or $\delta(v) \cap \delta(O) = \emptyset$, then there exists $F \subsetneq S$ such that \mathcal{F} is a proper subset of the proper faces induced by

$$x(\delta(O) \setminus \delta(R \cup F)) - (i-2)x(E[S_i \setminus F]) - (i-1)x(\delta(R \cup F) \setminus \delta(O)) \leq 0.$$

If $|\delta(v) \cap \delta(O)| \leq i - 2$, then \mathcal{F} is a proper subset of the proper face induced by

$$x(\delta(O) \setminus \delta(R \cup \{v\})) - (i - 2)x(E[S \setminus \{v\}]) - (i - 1)x(\delta(R \cup \{v\}) \setminus \delta(O)) \leq 0.$$

For the sufficiency, assume that $\mathcal{F} \subseteq \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\} \subsetneq B_x(G, r, \mathbf{c})$, where $\mathbf{a}^T \mathbf{x} \leq b$ is valid for $B_x(G, r, \mathbf{c})$. One first has $b = 0$ because of $\mathbf{0} \in \mathcal{F}$. For any edge $e \in E[R] \cup \delta(R) \cap \delta(O)$, as $G[R]$ is connected and $R \cap O = \emptyset$, one has $a_e = 0$. For any node $v \in S$, as $\delta(v) \cap \delta(R) \neq \emptyset$, $\delta(v) \cap \delta(O) \neq \emptyset$, and $|\delta(v) \cap \delta(O)| \geq i - 1$, for any $F \subseteq \delta(v) \cap \delta(O)$ with $|F| = i - 1$, one has

$$\sum_{e \in F} a_e + a_f = 0.$$

Moreover, since $G[S]$ is connected, for any edge $v_1 v_2 \in E[S]$, and any $F'_1 \subset F_1 \subset \delta(v_1) \cap \delta(O)$, $F'_2 \subset F_2 \subset \delta(v_2) \cap \delta(O)$ that satisfies $|F'_1| = |F'_2| = i - 2$ and $|F_1| = |F_2| = i - 1$ one has

$$\sum_{e_1 \in F_1} a_{e_1} + \sum_{e_2 \in F'_2} a_{e_2} + a_{v_1 v_2} + a_{f_2} = 0$$

and

$$\sum_{e_2 \in F_2} a_{e_2} + \sum_{e_1 \in F'_1} a_{e_1} + a_{v_1 v_2} + a_{f_1} = 0,$$

which leads to $a_{f_1} = -(i - 1)a_{e_1}$, $a_{f_2} = -(i - 1)a_{e_2}$ for any $e_1 \in \delta(v_1) \cap \delta(O)$, $e_2 \in \delta(v_2) \cap \delta(O)$, $f_1 \in \delta(v_1) \cap \delta(R)$, $f_2 \in \delta(v_2) \cap \delta(R)$, and $a_{v_1 v_2} = -a_{f_2} - a_{e_1} = -a_{f_1} - a_{e_2}$. Consequently, one has $(i - 1)a_e + a_f = 0$ and $(i - 2)a_e + a_h = 0$ for any $e \in \delta(O) \cap \delta(S)$, $f \in \delta(R) \cap \delta(S)$, $h \in E[S]$. Therefore, $\mathcal{F} = \{\mathbf{x} \in B_x(G, r, \mathbf{c}) : \mathbf{a}^T \mathbf{x} = b\}$, which indicates \mathcal{F} is a maximal face. \square

6.4.6 i -articulation inequalities

The capacity- i inequalities can be further generalized to the case such that we consider not only nodes adjacent with nodes in O , but also nodes which can be considered articulation nodes in the graph. Note that any node v_i adjacent to a node $v_o \in O$ can be seen as an articulation node between r and the edge $v_i v_o$.

Let the set of articulation nodes in V be

$$V_a = \{v \in V \setminus (O \cup \{r\}) : N(v) \cap O \neq \emptyset \text{ or } v \text{ is an articulation node in } G_r\}.$$

For each $v \in V_a$, let

$$D_v = \{u \in V \setminus \{r, v\} : u \in N(v) \cap O \text{ or any bounded path } P_{ru} \text{ satisfies } P_{ru} \cap \delta(v) \neq \emptyset\},$$

$$F_v = \delta(v) \cap \delta(D_v).$$

Additionally, recall

$$S_i = \{v \in V \setminus \{r\} : c_v = i\}.$$

Let S be a subset of $V_a \cap S_i$ with $i \geq 2$ such that for any distinct $u, v \in S$, $D_u \cap D_v \setminus O = \emptyset$.

Let

$$D_S = \bigcup_{v \in S} D_v,$$

$$F_S = \bigcup_{v \in S} F_v,$$

$$R = V \setminus (S \cup D_S).$$

Note that $\delta(D_S) \cap \delta(S) = F_S$. The i -articulation inequality associated with S is as follows.

$$x(F_S) - (i - 2)x(E[S]) - (i - 1)x(\delta(R) \cap \delta(S)) \leq 0 \quad (6.14)$$

Proposition 6.4.13. *Inequality (6.14) is valid.*

Proof. For any two nodes $u, v \in V_a$, one has $F_u \cap F_v = \emptyset$. Therefore, the validity of (6.14) can be proved using same argument as for (6.13). \square

Proposition 6.4.14. *Inequality (6.14) defines a facet of $B_x(G, r, \mathbf{c})$ if and only if*

1. $G[R]$ is connected, $G[S]$ is connected;
2. for any $v \in S$, $\delta(v) \cap \delta(R) \neq \emptyset$, and $\delta(v) \setminus \delta(D_S) \neq \emptyset$, and $|\delta(v) \setminus \delta(D_S)| \geq i - 1$.

Proof. The proof for i -articulation inequalities is the same as for the capacity i nodes inequalities, except that the set O is replaced by D_S whereas $\delta(O) \cap \delta(S)$ is replaced by F_S . \square

6.4.7 Tightening inequalities

Another set of inequalities that has been discovered relating to capacity-2 inequalities during the computational test is called the *tightening inequalities*.

Assume that without loss of generality $1 \leq c_v \leq |\delta(v)|$ for any $v \in V$. For any $S \subseteq V \setminus (\{r\} \cup O)$, $R = V \setminus (O \cup S)$, the tightening inequalities are as follows

$$x(\delta(O)) - x(\delta(R)) \leq \sum_{v \in S} (c_v - 2) \quad (6.15)$$

This set of inequalities has been proved to be effective to tighten the dual bound in the branch-and-cut algorithm. Besides, it can also be noticed that the capacity-2 inequalities is a special case of the tightening inequalities, where $c_v = 2$ for any node $v \in S$.

6.5 Projection from the Extended Polytope

Having the new inequalities introduced, the following sections show how can these inequalities be obtained by the projection of the valid inequalities for $B_{xy}(G, r, \mathbf{c})$ using the Fourier-Motzkin elimination. Particularly, in order to get the valid inequalities for $B_x(G, r, \mathbf{c})$, we use the linear system composed of (4.2) - (4.5) and (4.7), (4.8), which are the constraints defining $P_{Cac}(G, r, \mathbf{c})$ as in (4.18). Let $A\mathbf{x} + D\mathbf{y} \leq \mathbf{b}$ denote the system composed of (4.2) - (4.5) and (4.7), (4.8). The projection cone associated with the projection of $P_{Cac}(G, r, \mathbf{c})$ onto \mathbb{R}^E is the following

$$C = \{\boldsymbol{\nu} : \boldsymbol{\nu}^T A = 0, \boldsymbol{\nu} \geq 0\},$$

where $\boldsymbol{\nu}$ are the rays of C . We show that for each of the inequalities we introduced in the last section, there exists a ray in C such that the inequality can be obtained through projection along that ray.

6.5.1 Matching-partition inequalities

Here we show that for any $(M, \pi) \in \mathcal{MP}(G)$, the matching-partition inequality $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$ is a result of projection from the valid constraints for $B_{xy}(G, r, \mathbf{c})$. Let $W = \{s_1, \dots, s_k\}$ be such that $e_i = s_i t_i$ with $c_{s_i} \leq c_{t_i}$ for each $e_i \in M$, $i = \{1, \dots, k\}$, and $s_0 = r$. Note that from the definition $W \cap O = \emptyset$ holds.

The following steps show that inequality $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$ can be obtained from (4.1), (4.2), and (4.12).

For each $S_i \in \pi$, $i = \{0, 1, \dots, k\}$, since $W \cap O = \emptyset$, one has

$$x(E[S_i \setminus O]) - y(S_i \setminus (O \cup \{s_i\})) \leq 0.$$

Summing them up with

$$-x(E) + y(V \setminus \{r\}) = 0$$

gives us

$$\begin{aligned} -x(E) + \sum_{S_i \in \pi} x(E[S_i \setminus O]) + y(W) + y(O) &\leq 0, \\ -x(E_\pi \setminus \delta(O)) - x(\delta(O)) + y(W) + y(O) &\leq 0. \end{aligned}$$

From (4.7) we have

$$x(\delta(v_o)) - y_{v_o} = 0 \quad \forall v_o \in O,$$

which leads to

$$x(\delta(O)) - y(O) = 0.$$

Hence one gets

$$-x(E_\pi \setminus \delta(O)) + y(W) \leq 0.$$

From

$$x_{e_i} - y_{s_i} \leq 0,$$

for each $s_i \in W$, and $e_i \in M$, one has

$$x(M) - y(W) \leq 0,$$

which finally leads to

$$x(M) - x(E_\pi \setminus \delta(O)) \leq 0.$$

Thus the ray ν in the projection cone C that corresponds to the matching-partition inequality $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$ is the following

$$\nu = \begin{cases} 1, & \text{for } x(E[S_i \setminus O]) - y(S_i \setminus (O \cup \{s_i\})) \leq 0 \quad \forall i \in \{1, \dots, k\}, \\ 1, & \text{for } -x(E) + y(V \setminus \{r\}) = 0, \\ 1, & \text{for } x(\delta(v_o)) - y_{v_o} = 0 \quad \forall v_o \in O, \\ 1, & \text{for } x_{e_i} - y_{s_i} \leq 0 \quad \forall i \in \{1, \dots, k\}, \\ 0, & \text{otherwise.} \end{cases}$$

6.5.2 Upload capacity inequalities

The upload capacity constraint (6.9) associated with a node in $V \setminus (O \cup \{r\})$ can be obtained from constraints (4.1), (4.2), (4.3) and (4.7) as follows. For any $S \subset V \setminus \{r\}$

such that $S \not\subseteq O$, any $v \in S \setminus O$, according to (4.1) and (4.2) one has

$$\begin{aligned} -x(E) + y(V \setminus \{r\}) &= 0, \\ x(E[S \setminus O]) - y(S \setminus (O \cup \{v\})) &\leq 0, \\ x(E[\bar{S} \setminus O]) - y(\bar{S} \setminus (O \cup \{r\})) &\leq 0. \end{aligned}$$

From (4.7) we have

$$x(\delta(v_o)) - y_{v_o} = 0 \quad \forall v_o \in O,$$

which leads to

$$x(\delta(O)) - y(O) = 0.$$

Summing them up gives us

$$y_v - x(\delta(S) \setminus \delta(O)) \leq 0,$$

which combining with

$$x(\delta(v)) - c_v y_v \leq 0.$$

gives us

$$x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) \leq 0.$$

Then the ray ν that corresponding to the upload capacity inequality $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) \leq 0$ is as follows

$$\nu = \begin{cases} c_v, & \text{for } x(E[S \setminus O]) - y(S \setminus (O \cup \{v\})) \leq 0, \\ c_v, & \text{for } x(E[\bar{S} \setminus O]) - y(\bar{S} \setminus (O \cup \{r\})) \leq 0, \\ c_v, & \text{for } -x(E) + y(V \setminus \{r\}) = 0, \\ c_v, & \text{for } x(\delta(v_o)) - y_{v_o} = 0 \\ 1, & \text{for } x(\delta(v)) - c_v y_v \leq 0, \\ 0, & \text{otherwise.} \end{cases} \quad \forall v_o \in O,$$

Similarly, the upload capacity constraint (6.10) associated with nodes in O can also be obtained from (2.1), (2.2) and (4.7).

For any $S \subset V \setminus \{r\}$ and any $v \in S \cap O$, according to (4.1) and (4.2) one has

$$\begin{aligned} -x(E) + y(V \setminus \{r\}) &= 0, \\ x(E[S \setminus (O \setminus \{v\})]) - y(S \setminus O) &\leq 0, \end{aligned}$$

$$x(E[\bar{S} \setminus O]) - y(\bar{S} \setminus (O \cup \{r\})) \leq 0.$$

From (4.7), one also has

$$x(\delta(v_o)) - y_{v_o} = 0 \quad \forall v_o \in O \setminus \{v\},$$

which leads to

$$x(\delta(O \setminus \{v\})) - y(O \setminus \{v\}) = 0.$$

Summing them up gives us

$$y_v - x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0,$$

which combining with

$$x(\delta(v)) - y_v = 0.$$

gives us

$$x(\delta(v)) - x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0,$$

which can be written as

$$x(\delta(v) \setminus \delta(S)) - x(\delta(S) \setminus \delta(O)) \leq 0,$$

Thus the ray $\nu \in C$ corresponding to the upload capacity inequality $x(\delta(v) \setminus \delta(S)) - x(\delta(S) \setminus \delta(O)) \leq 0$ is as follows

$$\nu = \begin{cases} 1, & \text{for } x(E[S \setminus (O \setminus \{v\})]) - y(S \setminus O) \leq 0, \\ 1, & \text{for } x(E[\bar{S} \setminus O]) - y(\bar{S} \setminus (O \cup \{r\})) \leq 0, \\ 1, & \text{for } -x(E) + y(V \setminus \{r\}) = 0, \\ 1, & \text{for } x(\delta(v_o)) - y_{v_o} = 0 \\ 0, & \text{otherwise.} \end{cases} \quad \forall v_o \in O,$$

6.5.3 Acyclicity-connectivity inequalities

Projection for acyclicity-connectivity inequalities (6.8) is similar to the upload capacity inequalities.

The acyclicity-connectivity inequalities can be obtained from inequalities (4.6), (4.7) and (4.2) as follows.

Let W be a subset of $S \subseteq V \setminus \{r\}$ with $W \cap O = \emptyset$ and $|W| \geq 2$. Given a node $v \in W$,

according to (4.2) one has

$$x(E[W]) - y(W \setminus \{v\}) \leq 0.$$

In addition, for each $s \in W \setminus \{v\}$, since $W \subseteq S \subseteq V \setminus \{r\}$ and $W \cap O = \emptyset$, from (4.1), (4.2) and (4.7) one has

$$\begin{aligned} -x(E) + y(V \setminus \{r\}) &= 0, \\ x(E[S \setminus O]) - y(S \setminus (O \cup \{s\})) &\leq 0, \\ x(E[\bar{S} \setminus O]) - y(\bar{S} \setminus (O \cup \{r\})) &\leq 0, \\ x(\delta(v_o)) - y_{v_o} &= 0 \quad \forall v_o \in O, \end{aligned}$$

which leads to

$$y_s - x(\delta(S) \setminus \delta(O)) \leq 0 \quad \forall s \in W \setminus \{v\}.$$

Thereby by summing all inequalities above one has

$$x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) \leq 0.$$

Thus the ray ν in C that corresponds to the acyclicity-connectivity inequality $x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) \leq 0$ is as follows

$$\nu = \begin{cases} 1, & \text{for } x(E[W]) - y(W \setminus \{v\}) \leq 0, \\ 1, & \text{for } x(E[S \setminus O]) - y(S \setminus (O \cup \{s\})) \leq 0 \quad \forall s \in W \setminus \{v\} \\ |W| - 1, & \text{for } x(E[\bar{S} \setminus O]) - y(\bar{S} \setminus (O \cup \{r\})) \leq 0, \\ |W| - 1, & \text{for } -x(E) + y(V \setminus \{r\}) = 0, \\ |W| - 1, & \text{for } x(\delta(v_o)) - y_{v_o} = 0 \quad \forall v_o \in O, \\ 0, & \text{otherwise.} \end{cases}$$

6.5.4 Capacity- i inequalities

Recall that for any $i \geq 2$,

$$S_i = \{v \in V \setminus \{r\} : c_v = i\}.$$

Given $S \subseteq S_i$, and $R = V \setminus (S \cup O)$, and the corresponding capacity- i inequality can be obtained as follows. From (4.3),

$$\sum_{v \in S} x(\delta(v)) - iy(S) \leq 0,$$

which can be rewritten as

$$x(\delta(R, S)) + x(\delta(S, O)) + 2x(E[S]) - iy(S) \leq 0. \quad (6.16)$$

From (4.2), one has

$$x(E[R]) - y(R \setminus \{r\}) \leq 0. \quad (6.17)$$

Combining (4.1) and (4.7) gives us

$$x(E[R \cup S]) - y(R \cup S \setminus \{r\}) = 0. \quad (6.18)$$

Then one can see that as a result of (6.16), (6.17), and (6.18), one has

$$\begin{aligned} & x(\delta(R, S)) + x(\delta(S, O)) + 2x(E[S]) - iy(S) \\ & \quad + ix(E[R]) - iy(R \setminus \{r\}) \\ & \quad - ix(E[R \cup S]) + iy(R \cup S \setminus \{r\}) \leq 0, \end{aligned}$$

which is equivalent to

$$x(\delta(S, O)) - (i - 2)x(E[S]) - (i - 1)x(\delta(R, S)) \leq 0,$$

and

$$x(\delta(O) \setminus \delta(R)) - (i - 2)x(E[S]) - (i - 1)x(\delta(R) \setminus \delta(O)) \leq 0.$$

Thus the corresponding ray $\nu \in C$ to this capacity- i inequality is as follows

$$\nu = \begin{cases} i, & \text{for } x(E[R]) - y(R \setminus \{r\}) \leq 0, \\ 1, & \text{for } x(\delta(v)) - iy_v \leq 0, \quad \forall v \in S, \\ i, & \text{for } -x(E) + y(V \setminus \{r\}) = 0, \\ i, & \text{for } x(\delta(v_o)) - y_{v_o} = 0 \quad \forall v_o \in O, \\ 0, & \text{otherwise.} \end{cases}$$

Furthermore, note that the capacity- i inequalities is a special case of the i -articulation inequalities. The i -articulation inequalities can be obtained by a similar approach as the one presented above, the only adjustments need to be made is that instead of considering O and $\delta(S, O)$ one considers D_S and F_S .

6.5.5 Tightening inequalities

For the tightening inequalities, from (4.3) we have

$$\sum_{v \in S} (x(\delta(v)) - c_v y_v) \leq 0,$$

which can be rewritten as

$$x(\delta(R, S)) + x(\delta(S, O)) + 2x(E[S]) - \sum_{v \in S} c_v y_v \leq 0. \quad (6.19)$$

Inequality (6.17) still holds, and inequality (6.18) holds similarly for tightening inequalities as presented below.

$$x(E[R \cup S]) - y(R \cup S \setminus \{r\}) = 0. \quad (6.20)$$

The sum of (6.17), (6.19), and (6.20) gives us

$$\begin{aligned} & x(\delta(R, S)) + x(\delta(S, O)) + 2x(E[S]) - \sum_{v \in S} c_v y_v \\ & \quad + 2x(E[R]) - 2y(R \setminus \{r\}) \\ & \quad - 2x(E[R \cup S]) + 2y(R \cup S \setminus \{r\}) \leq 0, \end{aligned}$$

which is equal to

$$-x(\delta(R, S)) + x(\delta(S, O)) - \sum_{v \in S} (c_v - 2)y_v \leq 0.$$

Combining it with $\sum_{v \in S} (c_v - 2)(y_v - 1) \leq 0$ gives us

$$-x(\delta(R, S)) + x(\delta(S, O)) - \sum_{v \in S} (c_v - 2) \leq 0,$$

which is the tightening inequality. Thus the corresponding ray $\nu \in C$ to this tightening inequality is as follows

$$\nu = \begin{cases} 2, & \text{for } x(E[R]) - y(R \setminus \{r\}) \leq 0, \\ 1, & \text{for } x(\delta(v)) - c_v y_v \leq 0, & \forall v \in S, \\ 2, & \text{for } -x(E) + y(V \setminus \{r\}) = 0, \\ 2, & \text{for } x(\delta(v_o)) - y_{v_o} = 0 & \forall v_o \in O, \\ c_v - 2, & \text{for } y_v \leq 1, & \forall v \in S, \\ 0, & \text{otherwise.} \end{cases}$$

Thereby, we show all the new inequalities introduced earlier can be projected from the facet-defining inequalities for the extended bounded r -tree polytope.

6.6 Bounds on Chvátal-Gomory rank

We now introduce some results regarding the Chvátal-Gomory rank of the newly discovered inequalities mentioned in Section 6.4 with respect to the polytope defined by (6.1)-(6.5).

6.6.1 Matching-partition inequalities

First, given a matching-partition $(M, \pi) \in \mathcal{MP}(G)$, denote the Chvátal-Gomory rank of its associated matching-partition inequality $rk(MP_{M, \pi})$. We show that the following proposition holds.

Proposition 6.6.1. *Let $(M, \pi) \in \mathcal{MP}(G)$, then*

$$rk(MP_{M, \pi}) \leq |M| - 1.$$

Proof. In order to simplify the notation in the proof hereafter, let

$$\begin{aligned} \delta'(S_1, S_2) &= \delta(S_1, S_2) \setminus \delta(O) & \forall S_1, S_2 \subsetneq V, S_1 \cap S_2 = \emptyset, \\ \delta'(S) &= \delta(S) \setminus \delta(O) & \forall S \subsetneq V, \\ E'_\pi &= E_\pi \setminus \delta(O) & \forall (M, \pi) \in \mathcal{MP}(G). \end{aligned}$$

We start with $|M| = 1$, since it becomes the connectivity inequality (6.1), its Chvátal-Gomory rank is clearly equal to 0, that is,

$$rk(MP_{M, \pi}) = 0 \quad \forall (M, \pi) \in \mathcal{MP}(G), |M| = 1.$$

For $|M| \geq 2$, let $M = \{e_1, \dots, e_k\}$, $\pi = \{S_0, S_1, \dots, S_k\}$. For any $i \in \{1, \dots, k\}$, let $M'_i = M \setminus \{e_i\}$, and $\pi'_i = \{S_0 \cup S_i, S_1, \dots, S_{i-1}, S_{i+1}, \dots, S_k\}$. It is clear that $(M'_i, \pi'_i) \in \mathcal{MP}(G)$. Thus one has

$$\begin{aligned} x(M) - x_{e_1} - (x(E'_\pi) - x(\delta'(S_0, S_1))) &\leq 0 \\ x(M) - x_{e_2} - (x(E'_\pi) - x(\delta'(S_0, S_2))) &\leq 0 \\ &\vdots \\ x(M) - x_{e_i} - (x(E'_\pi) - x(\delta'(S_0, S_i))) &\leq 0 \end{aligned}$$

$$\begin{aligned} & \vdots \\ x(M) - x_{e_k} - (x(E'_\pi) - x(\delta'(S_0, S_k))) & \leq 0 \end{aligned}$$

Summing them up gives us

$$(k-1)x(M) - kx(E'_\pi \setminus \delta'(S_0)) - (k-1)x(\delta'(S_0)) \leq 0 \quad (6.21)$$

Similarly we can also merge any distinct S_i and S_j in $\pi \setminus \{S_0\}$ to obtain two new matching-partition whose matchings are $M \setminus \{e_i\}$ and $M \setminus \{e_j\}$ respectively. Consequently, we have the following $k(k-1)$ matching-partition inequalities

$$\begin{aligned} x(M) - x_{e_1} - (x(E'_\pi) - x(\delta'(S_1, S_2))) & \leq 0 \\ x(M) - x_{e_2} - (x(E'_\pi) - x(\delta'(S_1, S_2))) & \leq 0 \\ x(M) - x_{e_1} - (x(E'_\pi) - x(\delta'(S_1, S_3))) & \leq 0 \\ x(M) - x_{e_3} - (x(E'_\pi) - x(\delta'(S_1, S_3))) & \leq 0 \\ & \vdots \\ x(M) - x_{e_i} - (x(E'_\pi) - x(\delta'(S_i, S_j))) & \leq 0 \\ x(M) - x_{e_j} - (x(E'_\pi) - x(\delta'(S_i, S_j))) & \leq 0 \\ & \vdots \\ x(M) - x_{e_{k-1}} - (x(E'_\pi) - x(\delta'(S_{k-1}, S_k))) & \leq 0 \\ x(M) - x_{e_k} - (x(E'_\pi) - x(\delta'(S_{k-1}, S_k))) & \leq 0. \end{aligned}$$

Summing them up leads to

$$(k-1)(k-1)x(M) - (k^2 - k - 2)x(E'_\pi \setminus \delta'(S_0)) - k(k-1)x(\delta'(S_0)) \leq 0. \quad (6.22)$$

From the connectivity inequalities, one has

$$\begin{aligned} x_{e_1} - x(\delta'(S_1)) & \leq 0 \\ x_{e_2} - x(\delta'(S_2)) & \leq 0 \\ & \vdots \\ x_{e_i} - x(\delta'(S_i)) & \leq 0 \\ & \vdots \\ x_{e_k} - x(\delta'(S_k)) & \leq 0, \end{aligned}$$

which leads to

$$x(M) - 2x(E'_\pi \setminus \delta'(S_0)) - x(\delta'(S_0)) \leq 0. \quad (6.23)$$

One also has, from the box inequalities

$$kx(M) \leq k^2 \quad (6.24)$$

By summing up (6.21), (6.22), (6.23), (6.24), one has

$$(k^2 + 1)x(M) - k^2x(E'_\pi \setminus \delta'(S_0)) - k^2x(\delta'(S_0)) \leq k^2.$$

Hence, we have

$$\begin{aligned} \frac{k^2 + 1}{k^2 + 1}x(M) - \frac{k^2}{k^2 + 1}x(E'_\pi \setminus \delta'(S_0)) - \frac{k^2}{k^2 + 1}x(\delta'(S_0)) &\leq \frac{k^2}{k^2 + 1}, \\ \left\lfloor \frac{k^2 + 1}{k^2 + 1} \right\rfloor x(M) + \left\lfloor -\frac{k^2}{k^2 + 1} \right\rfloor x(E'_\pi \setminus \delta'(S_0)) + \left\lfloor -\frac{k^2}{k^2 + 1} \right\rfloor x(\delta'(S_0)) &\leq \left\lfloor \frac{k^2}{k^2 + 1} \right\rfloor, \\ x(M) - x(E'_\pi \setminus \delta'(S_0)) - x(\delta'(S_0)) &\leq 0. \end{aligned}$$

Therefore, it shows that $rk(MP_{M,\pi})$ satisfies

$$rk(MP_{M,\pi}) \leq \max\{rk(MP_{M',\pi'}) : (M', \pi') \in \mathcal{MP}(G), |M'| = |M| - 1\} + 1.$$

As we have $rk(MP_{M',\pi'}) = |M'| - 1 = 0$ for any $(M', \pi') \in \mathcal{MP}(G)$ with $|M'| = 1$, by induction, we deduce that $rk(MP_{M,\pi}) \leq |M| - 1$ for any $(M, \pi) \in \mathcal{MP}(G)$. \square

It is worth noting that one can obtain a tighter bound for some matching-partition $(M, \pi) \in \mathcal{MP}(G)$ such that G' contains multiple blocks, where G' is the graph that obtained from $G \setminus O$ by contracting $S_i \in \pi$ into a node. As it has been shown in the proof of Proposition 6.4.2, the associated inequality with such matching-partition can be written as linear combination of several matching partitions, each of which corresponds to a block of G' . Thus we have the following corollary.

Corollary 6.6.2. *Given $(M, \pi) \in \mathcal{MP}(G)$, let $q \in \mathbb{Z}^+$ be the number of nodes in the largest block in G' . We have $rk(MP_{M,\pi}) \leq q - 1$.*

Proof. According to the proof of Proposition 6.4.2, $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$ can be written as a linear combination of several matching partitions $(M_1, \pi_1), \dots, (M_j, \pi_j)$ of G , such that $q = \max\{|M_i| : i \in \{1, \dots, j\}\}$. Thus,

$$rk(MP_{M,\pi}) \leq \max\{rk(MP_{M_i,\pi_i}) : i \in \{1, \dots, j\}\}.$$

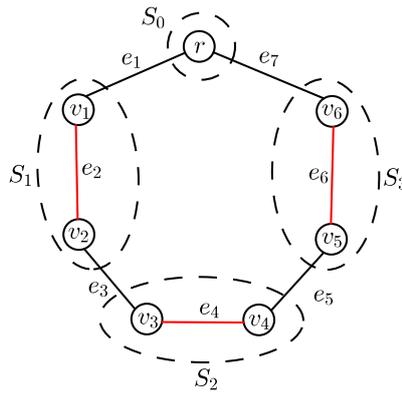


Figure 6.7: A matching-partition inequality with Chvátal-Gomory rank 2

Combining it with Proposition 6.6.1 gives us

$$\begin{aligned} rk(MP_{M,\pi}) &\leq \max\{|M_i| - 1 : i \in \{1, \dots, j\}\} \\ &= q - 1, \end{aligned}$$

which completes the proof. \square

Moreover, we can prove that there exists some matching-partition inequality whose Chvátal-Gomory rank is at least 2. Take the graph in Figure 6.7 as an instance and suppose the capacity constraints are redundant. Let $(M, \pi) \in \mathcal{MP}(G)$ be the matching-partition such that

$$\begin{aligned} M &= \{e_2, e_4, e_6\}, \\ E_\pi &= \{e_1, e_3, e_5, e_7\}. \end{aligned}$$

The system of inequalities composed of (6.1)-(6.5) then reduces to the following

$$x_e - x(\delta(S)) \leq 0 \quad \forall e \in E[S], S \subseteq V \setminus \{r\}, \quad (6.25)$$

$$x(E) \leq |E| - 1 \quad (6.26)$$

$$x_e \leq 1 \quad \forall e \in E, \quad (6.27)$$

$$-x_e \leq 0 \quad \forall e \in E. \quad (6.28)$$

The matching-partition inequality associated with (M, π) can be written as follows

$$x_{e_2} + x_{e_4} + x_{e_6} - x_{e_1} - x_{e_3} - x_{e_5} - x_{e_7} \leq 0. \quad (6.29)$$

Assume that its Chvátal-Gomory rank is 1, and let

$$\sum_{i=1}^7 a_i x_{e_i} \leq b,$$

be a linear combination of inequalities in (6.1)-(6.5) such that,

$$\begin{aligned} \lfloor a_i \rfloor &= 1 & \forall i \in \{2, 4, 6\}, \\ \lfloor a_i \rfloor &= -1 & \forall i \in \{1, 3, 5, 7\}, \\ \lfloor b \rfloor &= 0. \end{aligned}$$

Let $\lambda_{e_i, S}$ be the coefficient corresponding to the inequality in (6.25) associated with e_i and S , λ_0 be the coefficient corresponding to (6.26), λ_i^u be the coefficient of the inequality (6.27) associated with e_i , and λ_i^l be the coefficient corresponding to the inequality (6.28) associated with e_i . Note that all the coefficients should be non-negative. For any $i \in \{1, \dots, 7\}$, let

$$\begin{aligned} \mathcal{S}_i &= \{S \subseteq V \setminus \{r\} : e_i \in E[S]\}, \\ \mathcal{C}_i &= \{S \subseteq V \setminus \{r\} : e_i \in \delta(S)\}. \end{aligned}$$

Then let

$$\begin{aligned} n_1 &= \sum_{i \in \{2, 4, 6\}} \sum_{S \in \mathcal{S}_i} \lambda_{e_i, S}, \\ n_2 &= \sum_{i \in \{2, 4, 6\}} \sum_{S \in \mathcal{C}_i} \lambda_{e_i, S}, \\ n_3 &= \sum_{i \in \{1, 3, 5, 7\}} \sum_{S \in \mathcal{S}_i} \lambda_{e_i, S}, \\ n_4 &= \sum_{i \in \{1, 3, 5, 7\}} \sum_{S \in \mathcal{C}_i} \lambda_{e_i, S}. \end{aligned}$$

Straightforwardly, as each inequality in (6.25) has exactly two edges in $\delta(S)$, one has

$$2(n_1 + n_3) = n_2 + n_4,$$

which leads to

$$\begin{aligned} n_4 - n_3 &= 2n_1 + n_3 - n_2 \\ &\geq 2n_1 - n_2 \\ &\geq 2(n_1 - n_2). \end{aligned}$$

Additionally, let

$$\begin{aligned} m_1 &= \sum_{i \in \{2,4,6\}} \lambda_i^u, \\ m_2 &= \sum_{i \in \{2,4,6\}} \lambda_i^l, \\ m_3 &= \sum_{i \in \{1,3,5,7\}} \lambda_i^u, \\ m_4 &= \sum_{i \in \{1,3,5,7\}} \lambda_i^l. \end{aligned}$$

Thus we have

$$\begin{aligned} a_2 + a_4 + a_6 &= n_1 - n_2 + 3\lambda_0 + m_1 - m_2, \\ a_1 + a_3 + a_5 + a_6 &= n_3 - n_4 + 4\lambda_0 + m_3 - m_4, \\ b &= 6\lambda_0 + m_1 + m_3. \end{aligned}$$

From the assumption, one must also have

$$\begin{aligned} \frac{a_2 + a_4 + a_6}{3} &\geq 1, \\ \frac{a_1 + a_3 + a_5 + a_6}{4} &\geq -1, \\ b &< 1. \end{aligned}$$

We can derive that

$$\frac{a_2 + a_4 + a_6}{3} \geq 1 > b, \quad (6.30)$$

$$\frac{a_2 + a_4 + a_6}{3} \geq 1 \geq -\frac{a_1 + a_3 + a_5 + a_6}{4}. \quad (6.31)$$

(6.30) gives us

$$\frac{n_1 - n_2 + 3\lambda_0 + m_1 - m_2}{3} > 6\lambda_0 + m_1 + m_3,$$

and equivalently, one has

$$n_1 - n_2 - 15\lambda_0 - 2m_1 - m_2 - 3m_3 > 0. \quad (6.32)$$

On the other hand, (6.31) together with $n_4 - n_3 \geq 2(n_1 - n_2)$ gives us

$$\begin{aligned} \frac{n_1 - n_2 + 3\lambda_0 + m_1 - m_2}{3} &\geq \frac{n_4 - n_3 - 4\lambda_0 - (m_3 - m_4)}{4} \\ \frac{n_1 - n_2 + 3\lambda_0 + m_1 - m_2}{3} &\geq \frac{2(n_1 - n_2) - 4\lambda_0 - (m_3 - m_4)}{4} \\ 0 &\geq 2(n_1 - n_2) - 24\lambda_0 - 4m_1 + 4m_2 - 3m_3 + 3m_4 \\ 0 &\geq n_1 - n_2 - 12\lambda_0 - 2m_1 + 2m_2 - \frac{3}{2}m_3 + \frac{3}{2}m_4, \end{aligned}$$

which combining with

$$0 \geq -3\lambda_0 - 3m_2 - \frac{3}{2}m_3 - \frac{3}{2}m_4,$$

leads to

$$0 \geq n_1 - n_2 - 15\lambda_0 - 2m_1 - m_2 - 3m_3. \quad (6.33)$$

Therefore, (6.32) and (6.33) form a contradiction, and thus the Chvátal-Gomory rank of (6.29) is at least 2.

6.6.2 Upload capacity inequalities

Given $S \subseteq V \setminus \{r\}$ and $v \in S$, let the set of edges in $\delta(v)$ that are not in $\delta(S) \setminus \delta(O)$ be

$$D := \delta(v) \setminus (\delta(S) \setminus \delta(O)).$$

Let $UC_{|D| \leq c_v}$ represents the upload capacity inequalities that satisfy $|D| \leq c_v$.

Proposition 6.6.3. $rk(UC_{|D| \leq c_v}) \leq 1$.

Proof. For each $e_i \in D$,

$$x_{e_i} - x(\delta(S) \setminus \delta(O)) \leq 0.$$

As a result, one has

$$x(D) - |D|x(\delta(S) \setminus \delta(O)) \leq 0.$$

From the capacity inequality for v ,

$$x(D) + x(\delta(v) \setminus D) \leq c_v.$$

From these two parts, by giving them coefficients c_v and 1 respectively, one can get

$$(c_v + 1)x(D) - (c_v|D| - 1)x(\delta(v) \setminus D) - c_v|D|x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq c_v.$$

Since $|D| \leq c_v$, one can add the following part

$$\begin{aligned} -c_v(c_v - |D|)x(\delta(v) \setminus D) &\leq 0, \\ -c_v(1 + c_v - |D|)x(\delta(S) \setminus (\delta(O) \cup \delta(v))) &\leq 0. \end{aligned}$$

to obtain

$$(c_v + 1)x(D) - (c_v + 1)(c_v - 1)x(\delta(v) \setminus D) - c_v(c_v + 1)x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq c_v.$$

Finally by dividing both sides by $c_v + 1$, and round down the coefficients, one gets

$$x(D) - (c_v - 1)x(\delta(v) \setminus D) - c_v x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq \left\lfloor \frac{c_v}{c_v + 1} \right\rfloor,$$

which can be written as

$$x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0.$$

Therefore we have $rk(UC_{|D| \leq c_v}) \leq 1$. □

Moreover, if the inequality is not one of those rank 0 inequalities, then it can be restricted to $rk(UC_{|D| \leq c_v}) = 1$.

Let $F \subseteq D$, and UC' represents the following inequality

$$x(F) - (c_v - 1)x(\delta(v) \setminus D) - c_v x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq 0. \quad (6.34)$$

This inequality is valid, because of the validity of upload capacity and the non-negativity of the variables. From the Proposition 6.6.3, one can also derive the following lemma.

Lemma 6.6.4. $rk(UC'_{|F| \leq c_v}) \leq 1$,

Proof. Following a similar procedure as the proof of Proposition 6.6.3 with substituting D for F , eventually one will get

$$\begin{aligned} (c_v + 1)x(F) + x(D \setminus F) - (c_v + 1)(c_v - 1)x(\delta(v) \setminus D) \\ - c_v(c_v + 1)x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq c_v. \end{aligned}$$

Hence, after divided by $c_v + 1$ and round-down, it becomes exactly the inequality (6.34), which proves the Chvátal-Gomory rank of (6.34) is at most 1 if $|F| \leq c_v$. \square

Proposition 6.6.5. $rk(UC_{|D| \geq c_v}) \leq |D| - c_v + 1$.

Proof. According to Proposition 6.6.3, $rk(UC_{|D|=c_v}) \leq |D| - c_v + 1$.

Assume $|D| \geq c_v + 1$, from Lemma 6.6.4, one has for any $F \subseteq D$ with $|F| = c_v$, the following inequality with at most Chvátal-Gomory rank 1.

$$x(F) - (c_v - 1)x(\delta(v) \setminus D) - c_v x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq 0.$$

We then assume that, for any $F \subseteq D$, $|F| \geq c_v + 1$, the Chvátal-Gomory rank of inequality (6.34) associating with the set $F' \subseteq D$, $|F'| = |F| - 1$, is known to be at most $|F'| - c_v + 1 = |F| - c_v$.

Summing up all the possibilities of F' with each $e \in F$ being in $F \setminus F'$, results in the following inequality.

$$(|F| - 1)x(F) - (c_v - 1)|F|x(\delta(v) \setminus D) - c_v |F|x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq 0.$$

Hence, plus

$$x(D) + x(\delta(v) \setminus D) \leq c_v,$$

it leads to

$$|F|x(F) + x(D \setminus F) - ((c_v - 1)|F| - 1)x(\delta(v) \setminus D) - c_v |F|x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq c_v,$$

which, dividing both sides by $|F|$ results in

$$x(F) + \frac{1}{|F|}x(D \setminus F) - ((c_v - 1) - \frac{1}{|F|})x(\delta(v) \setminus D) - c_v x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq \frac{c_v}{|F|}.$$

Since $|F| \geq c_v + 1 \geq 2$, one has $0 < \frac{1}{|F|} < 1$, and $0 < \frac{c_v}{|F|} < 1$. Rounding this inequality down gives us

$$x(F) - (c_v - 1)x(\delta(v) \setminus D) - c_v x(\delta(S) \setminus (\delta(O) \cup \delta(v))) \leq 0.$$

Thus its Chvátal-Gomory rank is at most $|F| - c_v + 1$.

Then if $D = F$, the inequality can be written as $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) \leq 0$ instead, which indicates that its Chvátal-Gomory rank is at most $|D| - c_v + 1$. \square

Moreover, we can prove that there exists some upload capacity inequality whose Chvátal-Gomory rank is at least 2. Take the graph in Figure 6.8 as an instance and assume that

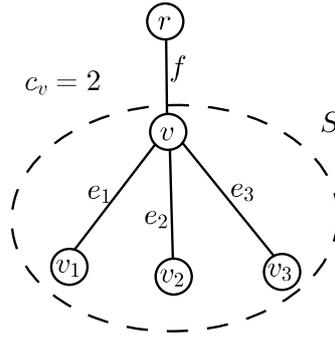


Figure 6.8: An upload capacity inequality with Chvátal-Gomory rank 2

only the capacity of v is not redundant. According to the facial study, the system of inequalities composed of (6.1)-(6.5) then reduces to the following

$$x_{e_i} - x_f \leq 0 \quad \forall i \in \{1, 2, 3\}, \quad (6.35)$$

$$x_{e_1} + x_{e_2} + x_{e_3} + x_f \leq 2 \quad (6.36)$$

$$x_f \leq 1 \quad (6.37)$$

$$-x_{e_i} \leq 0 \quad \forall i \in \{1, 2, 3\}. \quad (6.38)$$

The upload capacity inequality associated with v and $S = V \setminus \{r\}$ can be written as follows

$$x_{e_1} + x_{e_2} + x_{e_3} - x_f \leq 0. \quad (6.39)$$

Assume that its Chvátal-Gomory rank is 1. Let λ_i correspond to the coefficient to the inequality (6.35) associated with e_i , λ_c correspond to the coefficient to the (6.36), λ_f correspond to the coefficient to the (6.37), and μ_i correspond to the coefficient to the inequality (6.38) associated with e_i . We have that the following system admits some feasible solution.

$$\lambda_i + \lambda_c - \mu_i \geq 1 \quad \forall i \in \{1, 2, 3\}, \quad (6.40)$$

$$-\lambda_1 - \lambda_2 - \lambda_3 + \lambda_c + \lambda_f \geq -1, \quad (6.41)$$

$$2\lambda_c + \lambda_f < 1, \quad (6.42)$$

$$\lambda_i, \mu_i \geq 0 \quad \forall i \in \{1, 2, 3\}, \quad (6.43)$$

$$\lambda_c, \lambda_f \geq 0. \quad (6.44)$$

From (6.42) we have

$$\lambda_c < \frac{1}{2}.$$

Then from (6.40) we deduce

$$-\lambda_i + \mu_i \leq \lambda_c - 1 < -\frac{1}{2} \quad \forall i \in \{1, 2, 3\},$$

Hence, from the non-negativity of λ and μ one has

$$\begin{aligned} -\lambda_1 - \lambda_2 - \lambda_3 + \lambda_c + \lambda_f &\leq -\lambda_1 + \mu_1 - \lambda_2 + \mu_2 - \lambda_3 + \mu_3 + \lambda_c + \lambda_f \\ &= -\lambda_1 + \mu_1 - \lambda_2 + \mu_2 - (\lambda_3 + \lambda_c - \mu_3) + 2\lambda_c + \lambda_f \\ &< -\frac{1}{2} - \frac{1}{2} - 1 + 1 = -1, \end{aligned}$$

which contradicts with (6.41). Therefore, the Chvátal-Gomory rank of (6.39) is at least 2.

In the next section we show that with the help of matching-partition inequalities and the upload capacity inequalities, we can characterize $B_x(G, r, \mathbf{c})$ on cycles and trees. As we have showed in this section that there are matching-partition inequalities and upload capacity inequalities on cycles and on trees that have Chvátal-Gomory rank at least 2 with respect to the polytope defined by (6.1)-(6.5), thus the characterization can not be trivially obtained as its first Chvátal closure.

6.7 TDI-ness

6.7.1 On trees

According to the results on valid inequalities and facets presented in Sections 6.3 and 6.4, one can deduce that some of these inequalities are redundant on trees. By getting rid of these redundant inequalities we obtain the following linear program for MBrT problem on trees.

$$\begin{aligned} \max \quad & \mathbf{w}^T \mathbf{x} \\ \text{s.t.} \quad & x_e - x_{f_e} \leq 0 \quad \forall e \in E \setminus \delta(r), \end{aligned} \quad (6.45)$$

$$x(\delta(v)) - c_v x_{f_v} \leq 0 \quad \forall v \in V \setminus \{r\}, \quad (6.46)$$

$$x(\delta(r)) \leq c_r, \quad (6.47)$$

$$x_e \leq 1 \quad \forall e \in \delta(r), \quad (6.48)$$

$$x_e \geq 0 \quad \forall e \text{ is a leaf edge}, \quad (6.49)$$

where f_v denotes the edge of P_{rv} in $\delta(v)$ for $v \in V \setminus \{r\}$, and f_e denotes the edge of P_{re} adjacent to e for $e \in E \setminus \delta(r)$.

Let the polytope defined by the linear system composed of (6.45)-(6.49) be

$$P_{xT}(G, r, \mathbf{c}) = \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x} \text{ satisfies (6.45) - (6.49)}\}.$$

We hereafter show that it is a ideal formulation for $B_x(G, r, \mathbf{c})$ on trees and that the system defining $P_{xT}(G, r, \mathbf{c})$ is TDI. First of all, we show that $P_{xT}(G, r, \mathbf{c})$ is a formulation for $B_x(G, r, \mathbf{c})$ if G is a tree.

Theorem 6.7.1. *Let G be a tree. $P_{xT}(G, r, \mathbf{c}) \cap \mathbb{Z}^E = B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E$.*

Proof. Note that (6.45) is obtained from the connectivity inequalities (6.1), whereas (6.46) is obtained from the upload capacity inequalities (6.9). Hence (6.45)-(6.49) are all valid for $B_x(G, r, \mathbf{c})$. Consequently, $B_x(G, r, \mathbf{c}) \subseteq P_{xT}(G, r, \mathbf{c})$.

Consider now any vector $\mathbf{x}^F \in P_{xT}(G, r, \mathbf{c}) \cap \mathbb{Z}^E$ with $F \subseteq E$. The induced graph G_F is bounded since \mathbf{x}^F satisfies (6.46) and (6.47). For each edge $e \in F$, from (6.45), one can deduce that each edge in the path between r and e is also in F . Thus, G_F is a bounded r -tree of G , that is, $\mathbf{x}^F \in B_x(G, r, \mathbf{c}) \cap \mathbb{Z}^E$. \square

Proposition 6.7.2. *Let G be a tree. $P_{xT}(G, r, \mathbf{c})$ is integral.*

Proof. Assume that there exists a fractional extreme point $\bar{\mathbf{x}}$ of $P_x(G, r, \mathbf{c})$. Let $f_v = uv \in E$ be an edge such that \bar{x}_{f_v} is fractional, and either $u = r$ or $\bar{x}_{f_u} = 1$. Note that if $u \neq r$, we have that $x_{f_v} - x_{f_u} \leq 0$ is not binding at $\bar{\mathbf{x}}$.

According to (6.45), one has that for any edge e in $E[[v]]$, $0 \leq \bar{x}_e \leq \bar{x}_{uv} < 1$. We split the proof into two cases depending on the value of $\bar{x}(\delta(u)) - c_u$. Since $\bar{x}_{f_u} = 1$ when $u \neq r$, $\bar{x}(\delta(u)) - c_u$ corresponds to the left-hand side of (6.46) when $u \neq r$ and of (6.47) when $u = r$.

Suppose first $\bar{x}(\delta(u)) - c_u < 0$. Let $\bar{\mathbf{x}}'$ be the vector such that

$$\bar{x}'_e = \begin{cases} (1 + \epsilon)\bar{x}_e & \text{if } e \in E[[v]] \cup \{uv\}, \\ \bar{x}_e & \text{otherwise,} \end{cases}$$

where $\epsilon \in \mathbb{R} \setminus \{0\}$. Notably, we have that

$$\bar{x}'_e - \bar{x}'_{f_e} = \begin{cases} (1 + \epsilon)(\bar{x}_e - \bar{x}_{f_e}) & \text{if } e \in E[[v]], \\ \bar{x}_e - \bar{x}_{f_e} & \text{if } e \in E \setminus (E[[v]] \cup \{uv\}), \end{cases}$$

and

$$\bar{x}'(\delta(s)) - c_s \bar{x}'_{f_s} = \begin{cases} (1 + \epsilon)(\bar{x}(\delta(s)) - c_s \bar{x}_{f_s}) & \text{if } s \in [v], \\ \bar{x}(\delta(s)) - c_s \bar{x}_{f_s} & \text{if } s \in V \setminus ([v] \cup \{r, u\}). \end{cases}$$

In addition, for any $e \in E$ with $\bar{x}_e \in \{0, 1\}$, we have $\bar{x}'_e = \bar{x}_e$ from the definition of \bar{x}' . Thus if any of the inequalities (6.45), (6.46), and (6.49) is binding at \bar{x} then it is also binding at \bar{x}' . Therefore, \bar{x}' satisfies all the equations that \bar{x} does, which forms a contradiction.

Suppose now $\bar{x}(\delta(u)) - c_u = 0$. It can be deduced that there also exists another edge $uv' \in \delta(u) \setminus \{uv\}$ with $\bar{x}_{uv'}$ fractional. Since $\bar{x}_{f_u} = 1$ if $u \neq r$ we clearly have $uv' \in E[[u]]$. Consider the vector \bar{x}'' defined as

$$\bar{x}''_e = \begin{cases} (1 + \lambda)\bar{x}_e & \text{if } e \in E[[v]] \cup \{uv\}, \\ (1 - \lambda')\bar{x}_e & \text{if } e \in E[[v']] \cup \{uv'\}, \\ \bar{x}_e & \text{otherwise,} \end{cases}$$

where $\lambda, \lambda' \in \mathbb{R} \setminus \{0\}$ are such that $\lambda\bar{x}_{uv} = \lambda'\bar{x}_{uv'}$. One has that $\bar{x}''(\delta(u)) - c_u = 0$. Similarly to the previous case, we have

$$\bar{x}''_e - \bar{x}''_{f_e} = \begin{cases} (1 + \lambda)(\bar{x}_e - \bar{x}_{f_e}) & \text{if } e \in E[[v]], \\ (1 - \lambda')(\bar{x}_e - \bar{x}_{f_e}) & \text{if } e \in E[[v']], \\ \bar{x}_e - \bar{x}_{f_e} & \text{if } e \in E \setminus (E[[v]] \cup E[[v']] \cup \{uv, uv'\}), \end{cases}$$

and

$$\bar{x}''(\delta(s)) - c_s \bar{x}''_{f_s} = \begin{cases} (1 + \lambda)(\bar{x}(\delta(s)) - c_s \bar{x}_{f_s}) & \text{if } s \in [v], \\ (1 - \lambda')(\bar{x}(\delta(s)) - c_s \bar{x}_{f_s}) & \text{if } s \in [v'], \\ \bar{x}(\delta(s)) - c_s \bar{x}_{f_s} & \text{if } s \in V \setminus ([v] \cup [v'] \cup \{r, u\}). \end{cases}$$

In addition, for any $e \in E$ with $\bar{x}_e \in \{0, 1\}$, we have $\bar{x}''_e = \bar{x}_e$. Therefore, \bar{x}'' satisfies all the equations that \bar{x} does, which forms a contradiction.

Consequently, $P_{xT}(G, r, \mathbf{c})$ does not admit any fractional extreme point. \square

The next corollary immediately follows

Corollary 6.7.3. *Let G be a tree. we have*

$$P_{xT}(G, r, \mathbf{c}) = B_x(G, r, \mathbf{c}). \quad \square$$

As a result, if the capacity constraints are redundant, or in other words, in the case of $R_x(G, r)$, we have

Corollary 6.7.4. *Let G be a tree. we then have*

$$R_x(G, r) = \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x} \text{ satisfies (6.45), (6.48) and (6.49)}\}. \quad \square$$

We now prove that the linear system defining $P_{xT}(G, r, \mathbf{c})$ is TDI. Since TDI-ness is a sufficient condition for integrality, see Edmonds and Giles [1977], Theorem 6.7.1 could have been seen as a direct consequence of the next theorem.

Theorem 6.7.5. *The linear system composed of (6.45)-(6.49) is TDI.*

Proof. Consider the linear program

$$\max\{\mathbf{w}^T \mathbf{x} : \mathbf{x} \in P_{xT}(G, r, \mathbf{c})\}, \quad (6.50)$$

where $\mathbf{w} \in \mathbb{R}^E$. For any $e \in E$, let α_e be the dual variable corresponding to inequality (6.45) or (6.48) associated with e . For any $v \in V$, let β_v be the dual variable corresponding to inequality (6.46) or (6.47) associated with v . The dual linear program of (6.50) is the following

$$\begin{aligned} \min \quad & c_r \beta_r + \sum_{e \in \delta(r)} \alpha_e \\ \text{s.t.} \quad & - \sum_{e' \in \delta(v_e) \setminus \{e\}} \alpha_{e'} - (c_{v_e} - 1) \beta_{v_e} + \alpha_e + \beta_{u_e} = w_e \quad \forall e \text{ is not a leaf edge,} \end{aligned} \quad (6.51)$$

$$\alpha_e + \beta_{u_e} \geq w_e \quad \forall e \text{ is a leaf edge,} \quad (6.52)$$

$$\alpha_e, \beta_v \geq 0 \quad \forall e \in E, v \in V, \quad (6.53)$$

where for any edge $e \in E$, $e \in \delta(u_e) \cap E[[u_e]]$, and $e \in \delta(v_e) \cap E[[v_e]]$, that is, $e = u_e v_e$ and v_e is the extremity of e the furthest away from r .

We first recall some of the notation and results presented in Chapter 3. Given any node $v \in V$, let $g(v)$ be the value of a maximum bounded tree rooted at v of the subgraph $G[[v]]$ of G , where the capacity vector $\mathbf{c}^v \in \mathbb{Z}_+^{\lfloor v \rfloor}$ satisfies $c_v^v = c_v - 1$ if $v \neq r$, $c_v^v = c_v$ if $v = r$, and $c_s^v = c_s$, for $s \in \lfloor v \rfloor \setminus v$, that is,

$$g(v) = \max\{f(G_F) = w(F) : G_F \text{ is tree of } G[[v]] \text{ bounded by } \mathbf{c}^v\}.$$

For any node $v \in V$, let

$$h(v) = \begin{cases} w_{f_v} + g(v) & \text{if } v \in V \setminus \{r\}, \\ g(v) & \text{if } v = r. \end{cases}$$

According to Lemma 3.1.1, for any $v \in V$ we have

$$g(v) = \max\left\{\sum_{s \in S} h(s) : S \subseteq N(v) \cap \lfloor v \rfloor, |S| \leq c_v^v\right\}.$$

Note that since $g(v_l) = 0$ for any leaf v_l , $g(v) \geq 0$ for any $v \in V$. For each node $v \in V$, let $\{v^1, \dots, v^{t_v}, v^{t_v+1}, \dots, v^{q_v}\}$ be the set of nodes in $[v]$ adjacent to v , that is,

$$N(v) \cap [v] = \{v^1, \dots, v^{t_v}, v^{t_v+1}, \dots, v^{q_v}\}$$

with $q_v \geq 0$. Note that if v is a leaf, we have $q_v = 0$. Without loss of generality, we assume that $h(v^1) \geq h(v^2) \geq \dots \geq h(v^{t_v}) > 0 \geq h(v^{t_v+1}) \geq \dots \geq h(v^{q_v})$. Let $j_v = \min\{t_v, c_v^v\}$ for $v \in V$. We then have

$$g(v) = \sum_{k=1}^{j_v} h(v^k).$$

Moreover, a primal solution (whose objective value is $g(r)$) can be obtained using Algorithm 3.1 according to Proposition 3.1.2.

Now we present an effective approach to obtain a corresponding dual solution. The value of β can be first decided as follows

$$\beta_v = \begin{cases} h(v^{j_v}) & \text{if } v \in V, v \text{ is not a leaf and } j_v = c_v^v, \\ 0 & \text{otherwise.} \end{cases}$$

It can be seen that $\beta_v \geq 0$ for any $v \in V$.

For any edge $e = u_e v_e$, let

$$\begin{aligned} \alpha'_{u_e v_e} &= h(v_e) - \beta_{u_e}, \\ \Delta_{u_e v_e}^0 &= \begin{cases} 0 & \text{if } \alpha'_{u_e v_e} \geq 0, \\ -\alpha'_{u_e v_e} & \text{otherwise.} \end{cases} \end{aligned}$$

Notice that $\alpha'_{u_e v_e} + \Delta_{u_e v_e}^0 \geq 0$ and $\Delta_{u_e v_e}^0 \geq 0$ always hold, but it is possible that $\alpha'_{u_e v_e} < 0$. For any leaf edge $u_e v_e \in E$, we have

$$\begin{aligned} \alpha'_{u_e v_e} + \beta_{u_e} &= h(v_e) \\ &= w_{u_e v_e}. \end{aligned}$$

Consider a non-leaf node $v \in V$. By the definition of β_v , if $j_v < c_v^v$ one has $\beta_v = 0$. Thus

$$j_v \beta_v = c_v^v \beta_v \tag{6.54}$$

always holds. For any $i > j_v$, we have that $h(v^i) \leq h(v^{j_v}) = \beta_v$ if $j_v = c_v^v$, whereas

$h(v^i) \leq 0 = \beta_v$ if $j_v < c_v^v$. Hence we have $h(v^i) \leq \beta_v$ and thus

$$\alpha'_{vv^i} + \Delta_{vv^i}^0 = 0 \quad \forall i > j_v. \quad (6.55)$$

Similarly, for any $i \leq j_v$ it can be deduced from $h(v^i) \geq \beta_v$ that

$$\Delta_{vv^i}^0 = 0 \quad \forall i \leq j_v. \quad (6.56)$$

Therefore, for any non-leaf node $v \in V$, we have

$$\begin{aligned} \sum_{i=1}^{j_v} h(v^i) &= \sum_{i=1}^{j_v} (\alpha'_{vv^i} + \beta_v) + \sum_{i=j_v+1}^{q_v} (\alpha'_{vv^i} + \Delta_{vv^i}^0) \\ &= \sum_{i=1}^{j_v} (\alpha'_{vv^i} + \Delta_{vv^i}^0 + \beta_v) + \sum_{i=j_v+1}^{q_v} (\alpha'_{vv^i} + \Delta_{vv^i}^0) \\ &= c_v^v \beta_v + \sum_{i=1}^{q_v} (\alpha'_{vv^i} + \Delta_{vv^i}^0), \end{aligned}$$

where the first equality comes from the definition of α'_{vv^i} and (6.55), the second equality from (6.56), and the last equality from (6.54).

This gives us that for any non-leaf edge $e = u_e v_e \in E$, we have

$$\begin{aligned} \alpha'_{u_e v_e} + \beta_{u_e} &= h(v_e) \\ &= w_{u_e v_e} + \sum_{i=1}^{j_{v_e}} h(v_e^i) \\ &= w_{u_e v_e} + c_{v_e}^{v_e} \beta_{v_e} + \sum_{i=1}^{q_{v_e}} (\alpha'_{v_e v_e^i} + \Delta_{v_e v_e^i}^0). \end{aligned}$$

Consequently, we have

$$\begin{aligned} h(r) &= g(r) \\ &= \sum_{i=1}^{j_r} h(r^i) \\ &= c_r \beta_r + \sum_{i=1}^{q_r} (\alpha'_{rr^i} + \Delta_{rr^i}^0). \end{aligned}$$

Hereafter we construct a solution $[\frac{\alpha}{\beta}]$, based on $[\frac{\alpha' + \Delta^0}{\beta}]$, that is integral, dual-feasible and with objective value $g(r)$.

For each edge $e = u_e v_e \in E$, let $d(e)$ denote the difference between the left-hand side and right-hand side of the dual constraint associated with e . For the vector $\begin{bmatrix} \alpha' + \Delta^0 \\ \beta \end{bmatrix}$, we have

$$\begin{aligned} d(e) &= \alpha'_{u_e v_e} + \beta_{u_e} + \Delta^0_{u_e v_e} - (w_{u_e v_e} + c_{v_e}^{v_e} \beta_{v_e} + \sum_{i=1}^{q_{v_e}} (\alpha'_{v_e v_e^i} + \Delta^0_{v_e v_e^i})) \\ &= \Delta_e^0. \end{aligned}$$

Let the set of non-leaf edges which do not satisfy $d(e) = 0$ be

$$F = \{e \in E : \Delta_e^0 > 0, e \text{ is not a leaf edge}\}.$$

Now we prove that there exists a vector $\Delta^1 \in \mathbb{R}_+^E$ such that $\begin{bmatrix} \alpha' + \Delta^0 + \Delta^1 \\ \beta \end{bmatrix}$ is dual-feasible. Algorithm 6.1 computes the vector Δ^1 .

Algorithm 6.1: Algorithm on trees to obtain Δ^1

Input : Tree $G = (V, E)$ and Δ^0 .

Output: Δ^1 .

begin

```

1  Set  $\Delta^1 = \mathbf{0}$ .
   while  $F \neq \emptyset$  do
2     Take an edge  $e = u_e v_e \in F$  such that  $P_{r u_e} \cap F = \emptyset$ .
3     Pick one path  $P_{v_e v_l}$  between  $v_e$  and any leaf  $v_l \in [v_e]$ .
4     For each edge in  $e' \in P_{v_e v_l}$ 
5         set  $\Delta_{e'}^1 = \Delta_{e'}^1 + \Delta_{u_e v_e}^0$ .
6     Set  $F = F \setminus \{u_e v_e\}$ .
```

Let α denote $\alpha' + \Delta^0 + \Delta^1$. For each non-leaf edge $e = u_e v_e$, the $d(e)$ corresponding to $\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$ is

$$d(e) = \Delta_{u_e v_e}^0 + \Delta_{u_e v_e}^1 - \sum_{i=1}^{q_{v_e}} \Delta_{u_e v_e}^1.$$

Let $\mathcal{C}(e') = \{e \in E : e' \text{ belongs to } P_{v_e v_l} \text{ in Lines 4-5 of Algorithm 6.1}\}$ for any $e' \in E$. Consider any non-leaf edge $e = u_e v_e$. We have

$$\mathcal{C}(e) = \bigcup_{i=1}^{q_{v_e}} \mathcal{C}(v_e v_e^i) \setminus \{e\}.$$

The sets $\mathcal{C}(v_e v_e^i)$, $i \in \{1, \dots, q_{v_e}\}$, are pairwise disjoint. Thus

$$\begin{aligned}
d(e) &= \Delta_{u_e v_e}^0 + \Delta_{u_e v_e}^1 - \sum_{i=1}^{q_{v_e}} \Delta_{u_e v_e}^1 \\
&= \Delta_{u_e v_e}^0 + \sum_{e' \in \mathcal{C}(u_e v_e)} \Delta_{e'}^0 - \left(\sum_{i=1}^{q_{v_e}} \sum_{e' \in \mathcal{C}(v_e v_e^i)} \Delta_{e'}^0 \right) \\
&= \Delta_{u_e v_e}^0 + \sum_{e' \in \mathcal{C}(u_e v_e)} \Delta_{e'}^0 - \left(\Delta_{u_e v_e}^0 + \sum_{e' \in \mathcal{C}(u_e v_e)} \Delta_{e'}^0 \right) \\
&= 0.
\end{aligned}$$

Hence, all the equations in (6.51) are satisfied. Furthermore, for any leaf edge $e = u_e v_e \in E$, one has

$$\begin{aligned}
\alpha_{u_e v_e} + \beta_{u_e} &= \alpha_{u_e v_e} \\
&= \alpha'_{u_e v_e} + \Delta_{u_e v_e}^0 + \Delta_{u_e v_e}^1 \\
&\geq w_{u_e v_e} + \Delta_{u_e v_e}^1 \\
&\geq w_{u_e v_e},
\end{aligned}$$

which indicates that (6.52) is satisfied. For any $e \in E$, we have that $\alpha'_e + \Delta_e^0 \geq 0$ and $\Delta_e^1 \geq 0$, which leads to $\alpha_e \geq 0$. Therefore, $[\boldsymbol{\alpha}]$ is dual-feasible.

Notice that for any edge $rr^i \in \delta(r)$ as $\mathcal{C}(rr^i) = \emptyset$, we have $\Delta_{rr^i}^1 = 0$. Hence the following equation holds

$$\begin{aligned}
g(r) &= c_r \beta_r + \sum_{i=1}^{q_r} (\alpha'_{rr^i} + \Delta_{rr^i}^0) \\
&= c_r \beta_r + \sum_{i=1}^{q_r} (\alpha'_{rr^i} + \Delta_{rr^i}^0 + \Delta_{rr^i}^1) \\
&= c_r \beta_r + \sum_{i=1}^{q_r} \alpha_{rr^i}.
\end{aligned}$$

This indicates that the objective value of $[\boldsymbol{\beta}]$ is $g(r)$, and $[\boldsymbol{\beta}]$ is dual-optimal.

Finally, vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are obtained by additions and subtractions involving only the components of \boldsymbol{w} . So $[\boldsymbol{\beta}]$ is integral if \boldsymbol{w} is integral, which completes our proof. \square

6.7.2 On cycles

In this subsection, we completely characterize the polytope $B_x(G, r, \mathbf{c})$ on cycles by a TDI system. It is worth mentioning that the integrality of this polytope can also be obtained using a similar approach as for $B_{xy}(G, r, \mathbf{c})$ on trees and cycles in Section 4.4. However, for the sake of conciseness, we only present the proof based on the TDI-ness in this dissertation.

First of all, we introduce some notation.

For any edge $e \in E$, let

$$\begin{aligned}\mathcal{M}_e &= \{(M, \pi) \in \mathcal{MP}(G) : e \in M\}, \\ \mathcal{P}_e &= \{(M, \pi) \in \mathcal{MP}(G) : e \in E_\pi \setminus \delta(O)\}.\end{aligned}$$

Since G is a cycle, we can assume without loss of generality that

$$\begin{aligned}V &= \{r, v_1, \dots, v_{n-1}\}, \\ E &= \{e_1 = rv_1, e_n = rv_{n-1}\} \cup \{e_i = v_{i-1}v_i : i \in \{2, \dots, n-1\}\}.\end{aligned}$$

We consider four cases depending on the capacity of r and whether set O is empty.

6.7.2.1 Case 1. $c_r = 1$ and $O = \{v_o\}$

Note that $o \in \{1, \dots, n-1\}$. Let the two paths between r and v_o be

$$\begin{aligned}P_1 &= \{e_1, \dots, e_o\}, \\ P_2 &= \{e_{o+1}, \dots, e_n\}.\end{aligned}$$

From the connectivity inequalities (6.6), we have

$$\begin{aligned}x_{e_o} - x_{e_1} &\leq 0, \\ x_{e_{o+1}} - x_{e_n} &\leq 0,\end{aligned}$$

whereas from the capacity inequality (6.3) associated with r , we have

$$x_{e_1} + x_{e_n} \leq 1.$$

Then the capacity inequality (6.3) associated with v_o , that is, $x_{e_o} + x_{e_{o+1}} \leq 1$, can be obtained as the sum of these three inequalities, and thus is redundant. After getting rid of some redundant inequalities from (6.3)-(6.5) and (6.6) we obtain the following linear

system of inequalities

$$x_{e_i} - x_{e_{i-1}} \leq 0 \quad \forall i \in \{2, \dots, o\}, \quad (6.57)$$

$$x_{e_i} - x_{e_{i+1}} \leq 0 \quad \forall i \in \{o+1, \dots, n-1\}, \quad (6.58)$$

$$x_{e_1} + x_{e_n} \leq 1, \quad (6.59)$$

$$x_{e_i} \leq 1 \quad \forall i \in \{1, n\}. \quad (6.60)$$

$$x_{e_i} \geq 0 \quad \forall i \in \{o, o+1\}. \quad (6.61)$$

Notice that we consider (6.6) instead of (6.1) because of the existence of $v_o \in O$. Let $G' = (V', E')$ be a path obtained from G by splitting v_o into two nodes v_o^1 and v_o^2 , that is,

$$\begin{aligned} V' &= \{r, v_1, \dots, v_{o-1}, v_o^1, v_o^2, v_{o+1}, \dots, v_{n-1}\}, \\ E' &= E \setminus \delta(v_o) \cup \{v_{o-1}v_o^1, v_o^2v_{o+1}\}. \end{aligned}$$

Let $\mathbf{c}' \in \mathbb{Z}^{E'}$ be a vector such that $c_r = 1$, and $c_v \geq 2$ for any $v \in V' \setminus \{r\}$. System (6.57) - (6.61) corresponds to the system defining $P_{xT}(G', r, \mathbf{c}')$, where $v_{o-1}v_o^1$ and $v_o^2v_{o+1}$ substitute for e_o and e_{o+1} , respectively. According to Theorem 6.7.5, the system composed of (6.57) - (6.61) is TDI. Consequently the following result holds.

Corollary 6.7.6. *Let G be a cycle with $c_r = 1$ and $O = \{v_o\}$. Then we have*

$$B_x(G, r, \mathbf{c}) = \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x} \text{ satisfies (6.57) - (6.61)}\}. \quad \square$$

6.7.2.2 Case 2. $c_r \geq 2$ and $O = \{v_o\}$

In this case we follow the same notation of P_1 and P_2 as for the last case. We show that (6.3)-(6.5) and (6.6) form a TDI system and then are sufficient to characterize $B_x(G, r, \mathbf{c})$. First, after getting rid of some redundant inequalities from (6.3)-(6.5) and (6.6) we obtain the following linear program

$$\begin{aligned} \max \quad & \mathbf{w}^T \mathbf{x} \\ \text{s.t.} \quad & x_{e_i} - x_{f_{e_i}} \leq 0 \quad \forall e_i \in E \setminus \delta(r), \quad (6.62) \end{aligned}$$

$$x(\delta(v_o)) \leq 1, \quad (6.63)$$

$$x_{e_i} \leq 1 \quad \forall e_i \in \delta(r) \setminus \delta(v_o), \quad (6.64)$$

$$x_{e_i} \geq 0 \quad \forall e_i \in \delta(v_o), \quad (6.65)$$

where $f_{e_i} = e_{i-1}$ for $e_i \in P_1 \setminus \delta(r)$ and $f_{e_i} = e_{i+1}$ for $e_i \in P_2 \setminus \delta(r)$. Let the polytope that is defined by the aforementioned linear system be $P_{xC}^1(G, r, \mathbf{c})$, that is,

$$P_{xC}^1(G, r, \mathbf{c}) = \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x} \text{ satisfies (6.62) - (6.65)}\}.$$

We show that the system defining $P_{xC}^1(G, r, \mathbf{c})$ is TDI.

Theorem 6.7.7. *The linear system composed of (6.62)-(6.65) is TDI.*

Proof. Consider the linear program

$$\max\{\mathbf{w}^T \mathbf{x} : \mathbf{x} \in P_{xC}^1(G, r, \mathbf{c})\}, \quad (6.66)$$

where $\mathbf{w} \in \mathbb{R}^E$. Let α be the dual variable corresponding to inequality (6.63). For any $e \in E \setminus \delta(r)$, let β_{f_e} be the dual variable corresponding to inequality (6.62) associated with e (and then with f_e). For any $e \in \delta(r) \setminus \delta(v_o)$, let γ_e be the dual variable corresponding to inequality (6.64) associated with e . The dual linear program of (6.66) can be written as follows

$$\begin{aligned} \min \alpha + \sum_{e \in \delta(r) \setminus \delta(v_o)} \gamma_e \\ \text{s.t. } \alpha \geq w_e \quad \forall e \in \delta(r) \cap \delta(v_o), \end{aligned} \quad (6.67)$$

$$-\beta_e + \gamma_e = w_e \quad \forall e \in \delta(r) \setminus \delta(v_o), \quad (6.68)$$

$$\alpha + \beta_{f_e} \geq w_e \quad \forall e \in \delta(v_o) \setminus \delta(r), \quad (6.69)$$

$$-\beta_e + \beta_{f_e} = w_e \quad \forall e \in E \setminus (\delta(r) \cup \delta(v_o)), \quad (6.70)$$

$$\alpha, \beta_e, \gamma_e \geq 0 \quad \forall e \in E. \quad (6.71)$$

As P_1 and P_2 can be deemed to be symmetric, the values of the dual variables for edges in P_1 and P_2 can be obtained in the same way.

Let $\beta', \gamma' \in \mathbb{R}^E$ be two vectors which are obtained as follows. For $e \in \delta(r)$, let

$$\beta'_e = \max\{-w_e, 0\},$$

$$\gamma'_e = \max\{w_e, 0\}.$$

Then for each $e \in E$ such that β'_{f_e} has been calculated for f_e , we obtain β'_e and γ'_e as the following

$$\beta'_e = \max\{\beta'_{f_e} - w_e, 0\},$$

$$\gamma'_e = \max\{w_e - \beta'_{f_e}, 0\}.$$

It is straightforward to see that

$$\begin{aligned} w_e &= \gamma'_e - \beta'_e, \quad \forall e \in \delta(r), \\ w_e - \beta'_{f_e} &= \gamma'_e - \beta'_e \quad \forall e \in E \setminus \delta(r). \end{aligned}$$

Moreover, let

$$\alpha = \max\{\gamma'_{e_o}, \gamma'_{e_{o+1}}\}.$$

If $\delta(r) \cap \delta(v_o) \neq \emptyset$, that is, without loss of generality $o = 1$, one has

$$\alpha \geq \gamma'_{e_1} = \max\{w_{e_1}, 0\} \geq w_{e_1},$$

which indicates that (6.67) is satisfied. If $e_1 \in \delta(r) \setminus \delta(v_o)$, that is, $o \geq 2$, let

$$\gamma_{e_1} = \sum_{i=1}^{o-1} \gamma'_{e_i}.$$

Symmetrically, if $e_n \in \delta(r) \setminus \delta(v_o)$, that is, $o \leq n - 2$, let

$$\gamma_{e_n} = \sum_{i=o+2}^n \gamma'_{e_i}.$$

We can deduce that regardless of whether v_o is a neighbor of r or not, the following equation always holds

$$\begin{aligned} \alpha + \sum_{i=1}^{o-1} \gamma'_{e_i} + \sum_{i=o+2}^n \gamma'_{e_i} &= \begin{cases} \alpha + \gamma_{e_1} + \gamma_{e_n} & \text{if } o \in \{2, \dots, n-2\}, \\ \alpha + \gamma_{e_n} & \text{if } o = 1, \\ \alpha + \gamma_{e_1} & \text{if } o = n-1 \end{cases} \\ &= \alpha + \sum_{e \in \delta(r) \setminus \delta(v_o)} \gamma_e. \end{aligned}$$

Let

$$\beta_{e_i} = \begin{cases} \beta'_{e_i} + \sum_{j=i+1}^{o-1} \gamma'_{e_j} & \forall i \in \{1, \dots, o-1\}, \\ \beta'_{e_i} + \sum_{j=o+2}^{i-1} \gamma'_{e_j} & \forall i \in \{o+2, \dots, n\}, \\ \beta'_{e_i}, & \text{otherwise.} \end{cases}$$

Consequently for any $i \in \{2, \dots, o-1\}$, that is, $e_i \in P_1 \setminus (\delta(r) \cup \delta(v_o))$, we have

$$\begin{aligned} \beta_{e_{i-1}} - \beta_{e_i} &= \beta'_{e_{i-1}} + \sum_{j=i}^{o-1} \gamma'_{e_j} - (\beta'_{e_i} + \sum_{j=i+1}^{o-1} \gamma'_{e_j}) \\ &= \beta'_{e_{i-1}} + \gamma'_{e_i} - \beta'_{e_i} \\ &= w_{e_i}. \end{aligned}$$

Symmetrically, for any $i \in \{o+2, \dots, n\}$, that is, $e_i \in P_2 \setminus (\delta(r) \cup \delta(v_o))$, we have

$$\beta_{e_{i+1}} - \beta_{e_i} = w_{e_i}.$$

Thus (6.70) is satisfied by β . In addition for e_o we have

$$\begin{aligned} \beta_{e_{o-1}} - \beta'_{e_o} &= \beta'_{e_{o-1}} + \gamma'_{e_o} - \beta'_{e_o} \\ &= w_{e_o}, \end{aligned}$$

and

$$\begin{aligned} \beta_{e_{o+2}} - \beta'_{e_{o+1}} &= \beta'_{e_{o+2}} + \gamma'_{e_{o+1}} - \beta'_{e_{o+1}} \\ &= w_{e_{o+1}}. \end{aligned}$$

Moreover, for $e \in \delta(v_o) \setminus \delta(r)$, we have

$$\begin{aligned} \alpha + \beta_{f_e} &= \alpha + \beta'_{f_e} \\ &\geq \gamma'_e + \beta'_{f_e} \\ &= (w_e - \beta'_{f_e} + \beta'_e) + \beta'_{f_e} \\ &\geq w_e. \end{aligned}$$

Hence (6.69) is satisfied.

Furthermore, if $e_1 \in \delta(r) \setminus \delta(v_o)$, we then have

$$\begin{aligned} -\beta_{e_1} + \gamma_{e_1} &= -(\beta'_{e_1} + \sum_{i=2}^{o-1} \gamma'_{e_i}) + \sum_{i=1}^{o-1} \gamma_{e'_i} \\ &\geq -\beta'_{e_1} + \gamma'_{e_1} \\ &= w_{e_1}. \end{aligned}$$

Similarly, if $e_n \in \delta(r) \setminus \delta(v_o)$ we have

$$\begin{aligned} -\beta_{e_n} + \gamma_{e_n} &= -(\beta'_{e_n} + \sum_{i=o+2}^{n-1} \gamma'_{e_i}) + \sum_{i=o+2}^n \gamma'_{e_i} \\ &\geq -\beta'_{e_n} + \gamma'_{e_n} \\ &= w_{e_n}. \end{aligned}$$

which ensures that (6.68) is satisfied. Therefore, $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ is dual-feasible.

We now construct a primal solution based on $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$. Let j_1 be the largest number in $\{1, \dots, o-1\}$ with $\gamma'_{e_{j_1}} > 0$, that is,

$$j_1 = \max\{i \in \{1, \dots, o-1\} : \gamma'_{e_i} > 0\}.$$

Let $P_{j_1} = \emptyset$ if such j_1 does not exist, otherwise let P_{j_1} denote the subpath of P_1 between r and e_{j_1} . Similarly, j_2 be the smallest number in $\{o+1, \dots, n\}$ with $\gamma'_{e_{j_2}} > 0$, that is,

$$j_2 = \min\{i \in \{o+1, \dots, n\} : \gamma'_{e_i} > 0\}.$$

Correspondingly, let $P_{j_2} = \emptyset$ if such j_2 does not exist, otherwise let P_{j_2} denote the subpath of P_2 between r and e_{j_2} . As $\gamma'_{e_{j_1}} > 0$ and $\gamma'_{e_{j_2}} > 0$, one has $\beta'_{e_{j_1}} = 0$ and $\beta'_{e_{j_2}} = 0$. If $\alpha = 0$, let

$$E^* = P_{j_1} \cup P_{j_2}.$$

If $\alpha > 0$ and without loss of generality assume that $\alpha = \gamma'_{e_o}$, let

$$E^* = P_1 \cup P_{j_2}.$$

Clearly E^* induces a bounded r -tree of G , and thus \mathbf{x}^{E^*} is a feasible primal solution.

Furthermore, for any $j \in \{1, \dots, o\}$, we have

$$\begin{aligned} \sum_{i=1}^j w_{e_i} &= \gamma'_{e_1} - \beta'_{e_1} + \sum_{i=2}^j (\gamma'_{e_i} - \beta'_{e_i} + \beta'_{e_{i-1}}) \\ &= -\beta'_{e_1} + \sum_{i=1}^j \gamma'_{e_i}. \end{aligned}$$

Symmetrically, for any $j \in \{o+1, \dots, n\}$, we have

$$\begin{aligned} \sum_{i=j}^n w_{e_i} &= \gamma'_{e_n} - \beta'_{e_n} + \sum_{i=j}^{n-1} (\gamma'_{e_i} - \beta'_{e_i} + \beta'_{e_{i+1}}) \\ &= -\beta'_{e_n} + \sum_{i=j}^n \gamma'_{e_i}. \end{aligned}$$

Furthermore, if $\alpha = 0$, as $\beta'_{e_{j_1}} = 0$ and $\beta'_{e_{j_2}} = 0$, \mathbf{x}^{E^*} satisfies

$$\begin{aligned} \sum_{e \in E^*} w_e &= \sum_{i=1}^{j_1} w_{e_i} + \sum_{i=j_2}^n w_{e_i} \\ &= -\beta'_{e_{j_1}} + \sum_{i=1}^{j_1} \gamma'_{e_i} - \beta'_{e_{j_2}} + \sum_{i=j_2}^n \gamma'_{e_i} \\ &= \sum_{i=1}^{j_1} \gamma'_{e_i} + \sum_{i=j_2}^n \gamma'_{e_i} \\ &= \alpha + \sum_{e \in \delta(r) \setminus \delta(v_o)} \gamma_e. \end{aligned}$$

Similarly, if $\alpha > 1$, we have $\beta'_{e_o} = 0$ and $\beta'_{e_{j_2}} = 0$, and thus \mathbf{x}^{E^*} satisfies

$$\begin{aligned} \sum_{e \in E^*} w_e &= \sum_{i=1}^o w_{e_i} + \sum_{i=j_2}^n w_{e_i} \\ &= -\beta'_{e_{j_1}} + \sum_{i=1}^o \gamma'_{e_i} - \beta'_{e_{j_2}} + \sum_{i=j_2}^n \gamma'_{e_i} \\ &= \gamma'_{e_o} + \sum_{i=1}^{o-1} \gamma'_{e_i} + \sum_{i=j_2}^n \gamma'_{e_i} \\ &= \gamma'_{e_o} + \sum_{i=1}^{o-1} \gamma'_{e_i} + \sum_{i=o+2}^n \gamma'_{e_i} \\ &= \sum_{i=1}^{o-1} \gamma'_{e_i} + \sum_{i=o+2}^n \gamma'_{e_i} \\ &= \alpha + \sum_{e \in \delta(r) \setminus \delta(v_o)} \gamma_e. \end{aligned}$$

Thus, \mathbf{x}^{E^*} and $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ are both optimal. Moreover, as α , β and γ are obtained by additions and subtractions involving only the components of \mathbf{w} . So $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ is integral if \mathbf{w} is integral,

which completes our proof. \square

Consequently, we obtain the following result.

Corollary 6.7.8. *Let G be a cycle with $c_r \geq 2$ and $O = \{v_o\}$, we then have*

$$B_x(G, r, \mathbf{c}) = P_{xC}^1(G, r, \mathbf{c}). \quad \square$$

6.7.2.3 Case 3. $c_r \geq 2$ and $O = \emptyset$

In this case, all the capacity constraints (6.3) are redundant because $|\delta(v)| \leq c_v$ for all $v \in V$. The only pertinent subtour elimination inequality in (6.2) is $x(E) \leq |E| - 1$, since the other inequalities (6.2) are redundant. Thus, the primal linear program for MBrT problem whose constraints are (6.1)-(6.5) and the matching-partition inequalities (6.7) can be rewritten as follows

$$\begin{aligned} \max \quad & \mathbf{w}^T \mathbf{x} \\ \text{s.t.} \quad & x(E) \leq |E| - 1, \end{aligned} \quad (6.72)$$

$$x(M) - x(E_\pi) \leq 0 \quad \forall (M, \pi) \in \mathcal{MP}(G), \quad (6.73)$$

$$x_e \leq 1 \quad \forall e \in E, \quad (6.74)$$

$$x_e \geq 0 \quad \forall e \in E. \quad (6.75)$$

Therefore the polytope

$$P_{xC}^2(G, r, \mathbf{c}) = \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x} \text{ satisfies (6.72) - (6.75)}\}$$

is a formulation for $B_x(G, r, \mathbf{c})$ if G is a cycle, $c_r \geq 2$ and $O = \emptyset$. Hereafter we show that the system (6.72)-(6.75) is TDI.

Consider the linear program

$$\max\{\mathbf{w}^T \mathbf{x} : \mathbf{x} \in P_{xC}^2(G, r, \mathbf{c})\}, \quad (6.76)$$

where $\mathbf{w} \in \mathbb{R}^E$. Let α be the dual variable corresponding to inequality (6.72). For any matching-partition $(M, \pi) \in \mathcal{MP}(G)$, let $\beta_{(M, \pi)}$ be the dual variable corresponding to inequality (6.73) associated with (M, π) . For any $e \in E$, let γ_e be the dual variable corresponding to inequality (6.74) associated with e . The dual linear program of (6.76)

is the following

$$\begin{aligned} \min & (|E| - 1)\alpha + \sum_{e \in E} \gamma_e \\ \text{s.t.} & \alpha + \sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} - \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)} + \gamma_e \geq w_e \quad \forall e \in E, \end{aligned} \quad (6.77)$$

$$\alpha, \beta_{(M,\pi)}, \gamma_e \geq 0 \quad \forall e \in E, (M,\pi) \in \mathcal{MP}(G). \quad (6.78)$$

Given an edge-weight vector $\mathbf{w}' \in \mathbb{R}^E$, let the set of edges with positive weights be

$$E^+(\mathbf{w}') = \{e \in E : w'_e > 0\},$$

and let the set of edges with negative weights be

$$E^-(\mathbf{w}') = \{e \in E : w'_e < 0\}.$$

Let $E^0(\mathbf{w}') = E \setminus (E^+(\mathbf{w}') \cup E^-(\mathbf{w}'))$.

In order to obtain a dual-feasible solution, we introduce the following concept related to partitions of E . For the sake of homogeneity in our presentation, we allow one of the partition classes to be empty.

Definition 6.7.9. *Given a cycle $G = (V, E)$ and a vector $\mathbf{w}' \in \mathbb{R}^E$, an alternating edge partition $\rho = \{H_0, F_1, H_1, \dots, H_{q-1}, F_q\}$ of the edge set E such that*

1. $\delta(r) \cap H_0 \neq \emptyset$ if $H_0 \neq \emptyset$ and $\delta(r) \cap F_i \neq \emptyset$ for $i \in \{1, q\}$, otherwise;
2. H_i induces a connected subgraph of G and is composed of non-negative edges for $i \in \{0, \dots, q-1\}$;
3. H_i contains at least one positive edge for $i \in \{1, \dots, q-1\}$;
4. F_i induces a connected subgraph of G and is composed of non-positive edges for $i \in \{1, \dots, q\}$;
5. F_i contains at least one negative edge for $i \in \{1, \dots, q\}$;
6. $F_i \cup H_i \cup F_{i+1}$ induces a connected subgraph of G for $i \in \{1, \dots, q-1\}$.

The sets H_1, \dots, H_{q-1} (F_1, \dots, F_q , respectively) are called the positive (negative, respectively) classes of the partition.

Several examples of alternating edge partition are given in Figure 6.9, where the value by each edge is its w' -value, the solid lines represent the edges in the set H_0, \dots, H_{q-1} , and the dashed lines the edges in the set F_1, \dots, F_q .

Consider Algorithm 6.2 that creates an alternating edge partition $\rho^1(\mathbf{w}')$ based on an

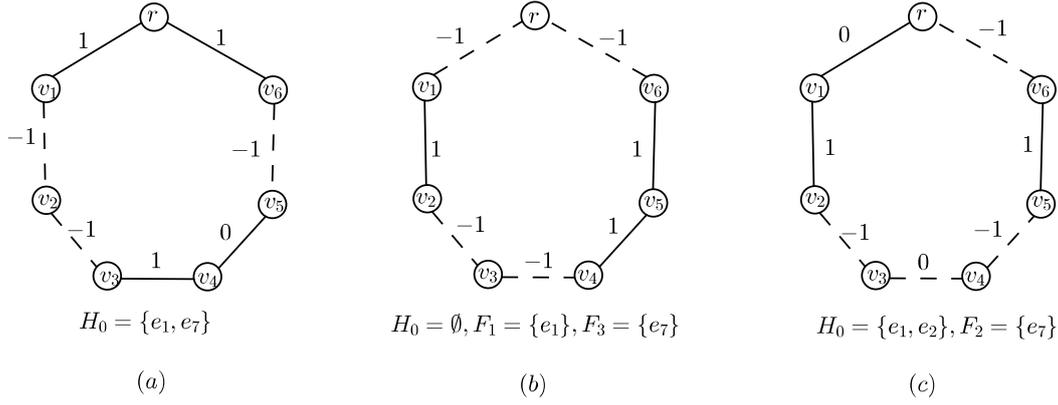


Figure 6.9: Examples of alternating edge partition

edge-weight vector $\mathbf{w}' \in \mathbb{R}^E$.

Algorithm 6.2: Algorithm to obtain an alternating edge partition on cycles

Input : Cycle $G = (V, E)$ and $\mathbf{w}' \in \mathbb{R}^E$.

Output: Alternating edge partition $\rho^1(\mathbf{w}') = \{H_0^1, F_1^1, H_1^1, \dots, H_{q^1-1}^1, F_{q^1}^1\}$.

begin

- 1 Let $H_0^1 \subseteq E$ be the edge set of the component that contains r in $G[E \setminus E^-(\mathbf{w}')$.
 - 2 Let $H_1^1, \dots, H_{p^1}^1$ be the edge sets of the components in $G[E \setminus E^-(\mathbf{w}')$ such that each of them does not contain r and contains at least one edge in $E^+(\mathbf{w}')$.
 - if** $H_0^1 \neq \emptyset$ **then**
 - 3 Let $F_1^1, \dots, F_{q^1}^1$ be the edge sets of the components in $G[E \setminus \bigcup_{i=0}^{p^1} H_i^1]$.
 - else**
 - 4 Let $F_1^0, F_2^1, \dots, F_{q^1-1}^1$ be the edge sets of the components in $G[E \setminus \bigcup_{i=0}^{p^1} H_i^1]$, where r is an inner node of the path induced by F_1^0 .
 - 5 Split F_1^0 into two sets $F_1^1, F_{q^1}^1$, each of which induces a path with one end being r .
-

The following proposition holds,

Proposition 6.7.10. *Algorithm 6.2 outputs an alternating edge partition $\rho^1(\mathbf{w}')$.*

Proof. It is trivial to see that Properties 2 and 3 of Definition 6.7.9 are guaranteed by Lines 1 and 2, whereas Properties 4 and 6 of Definition 6.7.9 are guaranteed by Line3.

Consider now Property 5 of Definition 6.7.9. Assume that there exists $i \in \{1, \dots, q^1\}$ such that $F_i^1 \cap E^-(\mathbf{w}') = \emptyset$. Then the edges in F_i^1 will be put, either in H_0^1 if F_i^1 is adjacent to H_0^1 in Line 1, or in H_j^1 for some $j \in \{1, \dots, p^1\}$ in Line 2.

Finally, we claim that $p^1 = q^1 - 1$ if $H_0^1 \neq E$, that is, the number of positive classes is always one less than the number of negative classes if $H_0^1 \neq E$. We prove this claim in two cases with $H_0^1 \neq \emptyset$ and $H_0^1 = \emptyset$.

If $H_0^1 \neq E$ and $H_0^1 \neq \emptyset$, as there do not exist two sets of $H_0^1, H_1^1, \dots, H_{p^1}^1$ that are adjacent, from Line 3 and the fact that G is a cycle, we also have that there do not exist two sets of $F_1^1, \dots, F_{q^1}^1$ that are adjacent. Thus $q^1 = p^1 + 1$.

Similarly, if $H_0^1 \neq E$ and $H_0^1 = \emptyset$, there do not exist two sets of $H_1^1, \dots, H_{p^1}^1$ that are adjacent. Then Line 4 ensures that we also have that there do not exist two sets of $F_1^0, F_2^0, \dots, F_{q^1-1}^0$ that are adjacent. Thus $q^1 - 1 = p^1$. \square

Given an edge-weight vector $\mathbf{w}^k \in \mathbb{R}^E$ and an alternating edge partition ρ^k with at least two negative classes, Algorithm 6.3 finds a matching-partition (M^k, π^k) based on ρ^k , obtains the corresponding dual variable $\beta_{(M^k, \pi^k)}$, and modifies the edge-weight vector \mathbf{w}^k into a new weight vector \mathbf{w}^{k+1} .

Algorithm 6.3: Algorithm to obtain a matching-partition and its associated dual variable based on ρ^k and \mathbf{w}^k

Input : Alternating edge partition $\rho^k = \{H_0^k, F_1^k, H_1^k, \dots, H_{q^k-1}^k, F_{q^k}^k\}$ and $\mathbf{w}^k \in \mathbb{R}^E$.

Output: Matching-partition (M^k, π^k) , $\beta_{(M^k, \pi^k)}$ and \mathbf{w}^{k+1} .

begin

- 1 Select $f_i^+ \in H_i^k \cap E^+(\mathbf{w}^k)$ for $i \in \{1, \dots, q^k - 1\}$.
 - 2 Let $M^k = \{f_1^+, \dots, f_{q^k-1}^+\}$.
 - 3 Select $f_i^- \in F_i^k \cap E^-(\mathbf{w}^k)$ for $i \in \{1, \dots, q^k\}$.
 - 4 Let π^k be a partition of V such that $E_{\pi^k} = \{f_1^-, \dots, f_{q^k}^-\}$.
 - 5 Set $\beta_{(M^k, \pi^k)} = \min\{|w_e^k| : e \in M^k \cup E_{\pi^k}\}$.
 - 6 Set $w_e^{k+1} = w_e^k - \beta_{(M^k, \pi^k)}$ for any $e \in M^k$, and $w_e^{k+1} = w_e^k + \beta_{(M^k, \pi^k)}$ for any $e \in E_{\pi^k}$.
-

Proposition 6.7.11. *Algorithm 6.3 outputs a matching-partition and \mathbf{w}^{k+1} such that $|E^0(\mathbf{w}^{k+1})| > |E^0(\mathbf{w}^k)|$.*

Proof. First, it is trivial to see that there exists a set $S_0 \subseteq V$ such that $r \in S_0$ and $\delta(S_0) = \{f_1^-, f_{q^k}^-\}$. For each $i \in \{1, \dots, q^k - 1\}$, from the definition of alternating edge partition, one has that f_i^+ is the only edge in M^k in the path between f_i^- and f_{i+1}^- that does not contain r as an inner node. Thus, for each $i \in \{1, \dots, q^k - 1\}$ one can find a set S_i with $\delta(S_i) = \{f_i^-, f_{i+1}^-\}$, and hence (M^k, π^k) is a matching-partition of G .

Furthermore, as $(M^k \cup E_{\pi^k}) \cap E^0(\mathbf{w}^k) = \emptyset$ we have $E^0(\mathbf{w}^k) \subseteq E^0(\mathbf{w}^{k+1})$. Additionally, from Lines 5 and 6, we deduce that there exists some $e \in M^k \cup E_{\pi^k}$ with $w_e^k \neq 0$ and $w_e^{k+1} = 0$. Therefore, $|E^0(\mathbf{w}^{k+1})| > |E^0(\mathbf{w}^k)|$. \square

Algorithm 6.4: Algorithm to update an alternating edge partition ρ^k into ρ^{k+1} according to an edge-weight vector \mathbf{w}^{k+1}

Input : Alternating edge partition ρ^k and $\mathbf{w}^{k+1} \in \mathbb{R}^E$.

Output: Alternating edge partition ρ^{k+1} .

begin

```

1 | Initialize  $\rho'$  with  $\rho' = \rho^k$  and let  $\rho' = \{H'_0, F'_1, H'_1, \dots, H'_{q'-1}, F'_{q'}\}$  with  $q' = q^k$ .
   | if  $E^-(\mathbf{w}^{k+1}) = \emptyset$  then
2 | | Set  $\rho^{k+1} = \{H_0^{k+1}\}$  with  $H_0^{k+1} = E$  and end the algorithm.
   | while there exists a partition class  $C \in \rho' \setminus \{H'_0\}$  with  $w^{k+1}(C) = 0$  do
   | | if there exists  $F'_i$  with  $w(F'_i) = 0$ ,  $i \in \{1, \dots, q'\}$  then
   | | | if  $i \neq q'$  then
3 | | | | Merge  $F'_i$  with  $H'_{i-1}$  and  $H'_i$  in  $\rho'$  to get a new positive partition
   | | | | class and update  $\rho'$  and  $q'$ .
   | | | else if  $i = q'$  then
4 | | | | Merge  $F'_i$  with  $H'_{i-1}$  and  $H'_0$  to get a new positive partition class and
   | | | | update  $\rho'$  and  $q'$ .
   | | | else if there exists  $H'_i$  with  $w(H'_i) = 0$ ,  $i \in \{1, \dots, q' - 1\}$  then
5 | | | | Merge  $H'_i$  with  $F'_i$  and  $F'_{i+1}$  to get a new negative partition class and
   | | | | update  $\rho'$  and  $q'$ .
6 | | Set  $\rho^{k+1} = \rho'$ .

```

We then construct another alternating edge partition ρ^{k+1} based on ρ^k and \mathbf{w}^{k+1} using Algorithm 6.4.

Proposition 6.7.12. *Algorithm 6.4 outputs an alternating edge partition ρ^{k+1} with respect to the edge-weight vector \mathbf{w}^{k+1} .*

Proof. If $E^-(\mathbf{w}^{k+1}) = \emptyset$, Line 2 ensures that $\rho^{k+1} = \{H_0^{k+1} = E\}$ is an alternating edge partition.

If $E^-(\mathbf{w}^{k+1}) \neq \emptyset$, Lines 3, 4 and 5, first ensure that there does not exist any F_i^{k+1} with $w(F_i^{k+1}) = 0$, $i \in \{1, \dots, q^{k+1}\}$, or H_i^{k+1} with $w(H_i^{k+1}) = 0$, $i \in \{1, \dots, q^{k+1} - 1\}$, and second preserve the Properties 1, 2, 4 and 6 of Definition 6.7.9 of alternating edge partition.

Therefore, ρ^{k+1} is always an alternating edge partition with respect to \mathbf{w}^{k+1} . \square

We now are ready to prove the TDI-ness of the system composed of (6.72)-(6.75).

Theorem 6.7.13. *The linear system composed of (6.72)-(6.75) is TDI.*

Proof. We claim that a pair of primal and dual solutions can be computed using a greedy approach, as presented in Algorithm 6.5.

Algorithm 6.5: Dual algorithm on cycles with $c_r \geq 2$ and $O = \emptyset$

Input : Cycle $G = (V, E)$ and $\mathbf{w} \in \mathbb{R}^E$.

Output: Dual-feasible solution $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ and primal-feasible solution \mathbf{x}^{E^*} .

begin

```

1  Set  $\alpha = \max\{0, \min\{w_e, e \in E\}\}$ .
2  Set  $w_e^1 = w_e - \alpha$  for all  $e \in E$ .
3  Create a partition of edges  $\rho^1$  based on  $\mathbf{w}^1$  using Algorithm 6.2.
4  Set  $k = 1$ .
   while  $q^k \geq 2$  do
5     Compute a matching-partition  $(M^k, \pi^k)$  and  $\beta_{(M^k, \pi^k)}$  based on  $\rho^k$  and  $\mathbf{w}^k$ ,
        and update  $\mathbf{w}^k$  into  $\mathbf{w}^{k+1}$  using Algorithm 6.3.
6     Update  $\rho^k$  into  $\rho^{k+1}$  based on  $\mathbf{w}^{k+1}$  using Algorithm 6.4.
7     Set  $k = k + 1$ 
8  Set  $\gamma_e = \max\{w_e^k, 0\}$  and  $\delta_e = \max\{-w_e^k, 0\}$ .
   if  $q^k = 0$  and  $k = 1$  then
9     | Set  $E^* = E \setminus \{f\}$  where  $f$  is an edge with  $\gamma_f = 0$ .
   else if  $q^k = 0$  and  $k \geq 2$  then
10    | Set  $E^* = E \setminus F_1^{k-1}$ .
   else if  $q^k = 1$  then
11    | Set  $E^* = E \setminus F_1^k$ .

```

Recall that Algorithm 6.3 ensures that for the k -th iteration there exists $e \in M^k \cup E_{\pi^k}$ with $w_e^{k+1} = 0$, and thus Algorithm 6.5 requires at most $n - 2$ iteration to obtain the dual solution.

Consider now the k -th iteration of Algorithm 6.5 Furthermore, Algorithm 6.4 ensures that if $k \geq 2$ and $q^k \geq 1$, then for each $i \in \{1, \dots, q^k\}$, F_i^k satisfies

$$|F^k \cap E_{\pi^{k-1}}| - |F_j^k \cap M^{k-1}| = 1.$$

If $q^k \geq 2$, we clearly have $|F^k \cap E_{\pi^k}| - |F_j^k \cap M^k| = 1 - 0 = 0$. As a result, we deduce recursively that,

$$|F^k \cap E_{\pi^j}| - |F^k \cap M^j| = 1 \quad \forall j \in \{1, \dots, k\}. \quad (6.79)$$

Additionally, if a matching-partition (M^k, π^k) is obtained at the k -th iteration of Algorithm 6.5, we have the following equations from Algorithm 6.3

$$\begin{aligned} w_e^{k+1} &= w_e^k - \beta_{(M, \pi)} & \forall e \in M^k, \\ w_e^{k+1} &= w_e^k + \beta_{(M, \pi)} & \forall e \in E_{\pi^k}, \end{aligned}$$

$$w_e^{k+1} = w_e^k \quad \forall e \in E \setminus (M^k \cup E_{\pi^k}).$$

Let ρ^l be the last alternating edge partition obtained by Algorithm 6.5, then the following equation holds

$$\begin{aligned} w_e &= \alpha + w_e^1 \\ &= \alpha + w_e^l + \sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} - \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)}. \end{aligned}$$

Moreover, from Line 8 we have

$$w_e^l = \gamma_e - \delta_e \quad \forall e \in E.$$

Hence, Algorithm 6.5 eventually enforces the following equation

$$w_e = \alpha + \gamma_e - \delta_e + \sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} - \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)}.$$

In addition, we clearly have that $\alpha \geq 0$, $\beta_{(M,\pi)} \geq 0$ for any $(M,\pi) \in \mathcal{MP}(G)$, and $\gamma_e, \delta_e \geq 0$ for any $e \in E$. Therefore, the dual solution $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ obtained by Algorithm 6.5 is feasible.

If $q^1 = 0$, from Line 9 one has $E^* = E \setminus \{f\}$ with $\gamma_f = 0$ for some $f \in E$, $\delta_e = 0$ for any $e \in E$, and $\beta_{(M,\pi)} = 0$ for any $(M,\pi) \in \mathcal{MP}(G)$. Clearly E^* induces a bounded r -tree of G , and its incidence satisfies

$$\begin{aligned} \mathbf{w}^T \mathbf{x}^{E^*} &= \sum_{e \in E^*} w_e \\ &= \sum_{e \in E^*} (\alpha + \gamma_e) \\ &= (|E| - 1)\alpha + \sum_{e \in E^*} \gamma_e \\ &= (|E| - 1)\alpha + \sum_{e \in E} \gamma_e. \end{aligned}$$

Thus, \mathbf{x}^{E^*} and $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ are optimal.

If $q^1 \geq 1$, one immediately has $\alpha = 0$. Line 9 ensures that there does not exist any edge $e \in E^*$ with $\delta_e > 0$, and there does not exist any edge $e \in E \setminus E^*$ with $\gamma_e > 0$.

According to the algorithm and (6.79), for any $\beta_{(M,\pi)} > 0$, one has $|(E \setminus E^*) \cap E_{\pi}| - |(E \setminus$

$E^*) \cap M| = 1$. As a result we have

$$|E^* \cap E_\pi| - |E^* \cap M| = 0,$$

which leads to

$$\begin{aligned} \sum_{e \in E^*} \left(\sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} - \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)} \right) &= \sum_{(M,\pi) \in \mathcal{MP}(G)} (|E^* \cap M| - |E^* \cap E_\pi|) \beta_{(M,\pi)} \\ &= 0. \end{aligned}$$

Hence we deduce

$$\begin{aligned} (|E| - 1)\alpha + \sum_{e \in E} \gamma_e &= \sum_{e \in E} \gamma_e \\ &= \sum_{e \in E^*} \gamma_e \\ &= \sum_{e \in E^*} \left(w_e - \sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} + \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)} \right) \\ &= \sum_{e \in E^*} w_e - \sum_{e \in E^*} \left(\sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} - \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)} \right) \\ &= \sum_{e \in E^*} w_e \end{aligned}$$

Therefore, \mathbf{x}^{E^*} and $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ are always feasible and optimal. Finally, vectors α , β and γ are obtained by additions and subtractions involving only the components of \mathbf{w} . So $\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$ is integral if \mathbf{w} is integral, which completes our proof. \square

As a direct consequence of Theorem 6.7.13, we have the following corollary.

Corollary 6.7.14. *Let G be a cycle with $c_r \geq 2$ and $O = \emptyset$, we then have*

$$B_x(G, r, \mathbf{c}) = P_{xC}^2(G, r, \mathbf{c}). \quad \square$$

Furthermore, in this case we clearly have $R_x(G, r) = B_x(G, r, \mathbf{c})$ as the capacity constraints are redundant. Hence we also have the following result.

Corollary 6.7.15. *Let G be a cycle. we have*

$$R_x(G, r) = P_{xC}^2(G, r, \mathbf{c}). \quad \square$$

6.7.2.4 Case 4. $c_r = 1$ and $O = \emptyset$

In this case, the proof proceeds in a similar fashion as the previous case, and we also reuse Algorithm 6.2 Algorithm 6.3, Algorithm 6.4 to help compute the dual solution.

First of all, the primal linear program for MBrT problem that contains (6.1)-(6.5) and the matching-partition inequalities (6.7) can be rewritten as follows

$$\begin{aligned} \max \quad & \mathbf{w}^T \mathbf{x} \\ \text{s.t.} \quad & x(\delta(r)) \leq 1, \end{aligned} \tag{6.80}$$

$$x(M) - x(E_\pi) \leq 0 \quad \forall (M, \pi) \in \mathcal{MP}(G), \tag{6.81}$$

$$x_e \geq 0 \quad \forall e \in E. \tag{6.82}$$

It is worth mentioning that the upper bound inequalities for $e \in E \setminus \delta(r)$ are redundant because it can be obtained by the combination of (6.80) and $x_e - x(\delta(S)) \leq 0$ with $S = V \setminus \{r\}$. Let the polytope defined by the aforementioned linear system of inequalities be

$$P_{xC}^3(G, r, \mathbf{c}) = \{\mathbf{x} \in \mathbb{R}^E : \mathbf{x} \text{ satisfies (6.80) - (6.82)}\}.$$

We give the following theorem regarding the TDI-ness of the linear system defining $P_{xC}^3(G, r, \mathbf{c})$.

Theorem 6.7.16. *The linear system composed of (6.80)-(6.82) is TDI.*

Proof. Consider the linear program

$$\max\{\mathbf{w}^T \mathbf{x} : \mathbf{x} \in P_{xC}^3(G, r, \mathbf{c})\}, \tag{6.83}$$

where $\mathbf{w} \in \mathbb{R}^E$. Let α_r be the dual variable corresponding to inequality (6.80). For any $(M, \pi) \in \mathcal{MP}(G)$, let $\beta_{(M, \pi)}$ be the dual variable corresponding to inequality (6.81) associated with (M, π) . The dual linear program of (6.83) is the following

$$\begin{aligned} \min \quad & \alpha_r \\ \text{s.t.} \quad & \sum_{(M, \pi) \in \mathcal{M}_e} \beta_{(M, \pi)} - \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)} \geq w_e \quad \forall e \in E \setminus \delta(r), \end{aligned} \tag{6.84}$$

$$\alpha_r - \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)} \geq w_e \quad \forall e \in \delta(r), \tag{6.85}$$

$$\alpha_r, \beta_{(M, \pi)} \geq 0 \quad (M, \pi) \in \mathcal{MP}(G), \tag{6.86}$$

Algorithm 6.6 computes a dual-feasible solution. As $w_e^1 = -M$ for all $e \in \delta(r)$, we have

Algorithm 6.6: Dual algorithm on cycles with $c_r = 1$ and $O = \emptyset$

Input : Cycle $G = (V, E)$ and $\mathbf{w} \in \mathbb{R}^E$.

Output: Dual-feasible solution $[\alpha_r]$ and primal-feasible solution \mathbf{x}^{E^*} .

begin

- 1 Set $w_e^1 = w_e$ for all $e \in E \setminus \delta(r)$, and $w_e^1 = -M$ for all $e \in \delta(r)$, where M is a sufficiently large number.
 - 2 Create a partition of edges ρ^1 based on \mathbf{w}^1 using Algorithm 6.2.
 - 3 Set $k = 1$.
 - while** $q^k \geq 2$ **do**
 - 4 Compute a matching-partition (M^k, π^k) and $\beta_{(M^k, \pi^k)}$ based on ρ^k and \mathbf{w}^k , and update \mathbf{w}^k into \mathbf{w}^{k+1} using Algorithm 6.3, with the following priority rules for edges in F_1^k and $F_{q^k}^k$. If there exists $e \in F_1^k \setminus e_1$ with $w_e^k < 0$, then select e to be in E_{π^k} ; if there exists $e \in F_{q^k}^k \setminus e_n$ with $w_e^k < 0$, then select e to be in E_{π^k} .
 - 5 Update ρ^k into ρ^{k+1} based on \mathbf{w}^{k+1} using Algorithm 6.4.
 - 6 Set $k = k + 1$
 - 7 Set $\gamma_e = w_e - \sum_{(M, \pi) \in \mathcal{M}_e} \beta_{(M, \pi)} + \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)}$ for all $e \in E$.
 - 8 Set $\alpha_r = \max\{0, \gamma_{e_1}, \gamma_{e_n}\}$.
-

$H_0^k = \emptyset$ for the k -th iteration. Let l -th iteration be the last iteration, then we have $\rho^l = \{F_1^l = E\}$. Hence for any $e \in E \setminus \delta(r)$ one has

$$\begin{aligned} \gamma_e &= w_e - \sum_{(M, \pi) \in \mathcal{M}_e} \beta_{(M, \pi)} + \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)} \\ &= w_e^l \\ &\leq 0. \end{aligned}$$

As a result, we have

$$w_e \leq \sum_{(M, \pi) \in \mathcal{M}_e} \beta_{(M, \pi)} - \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)} \quad \forall e \in E \setminus \delta(r).$$

Similarly, for any $e \in \delta(r)$, as $\mathcal{M}_e = \emptyset$ we have

$$\begin{aligned} w_e &= \gamma_e + \sum_{(M, \pi) \in \mathcal{M}_e} \beta_{(M, \pi)} - \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)}, \\ &= \gamma_e - \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)}, \\ &\leq \alpha_r - \sum_{(M, \pi) \in \mathcal{P}_e} \beta_{(M, \pi)}. \end{aligned}$$

Furthermore, it is clear that $\alpha_r \geq 0$ and $\beta_{(M,\pi)} \geq 0$ for any $(M,\pi) \in \mathcal{MP}(G)$. Hence, $[\overset{\alpha_r}{\beta}]$ is dual-feasible.

Now we construct a primal feasible solution E^* based on $[\overset{\alpha_r}{\beta}]$. Firstly, if $\alpha_r = 0$, then let $E^* = \emptyset$. Clearly, $[\overset{\alpha_r}{\beta}]$ and \mathbf{x}^{E^*} are both optimal.

If $\alpha_r > 0$ and $\rho^1 = \{F_1^1 = E\}$, without loss of generality, assume $\gamma_{e_n} = \alpha_r$, then let $E^* = \{e_n\}$. We have $\beta_{(M,\pi)} = 0$ for any matching-partition $(M,\pi) \in \mathcal{MP}(G)$. Thus, $\gamma_{e_n} = w_{e_n}$ and therefore

$$\mathbf{w}^T \mathbf{x}^{E^*} = w_{e_n} = \gamma_{e_n} = \alpha_r,$$

which indicates that both $[\overset{\alpha_r}{\beta}]$ and \mathbf{x}^{E^*} are optimal.

Consider now $\alpha_r > 0$ and ρ^1 contains at least one positive class. Without loss of generality, assume $\gamma_{e_n} = \alpha_r$. We clearly have $l \geq 2$, where l is the number of the last iteration. If there does not exist $e \in E \setminus \delta(r)$ with $\gamma_e < 0$, let $E^* = E \setminus F_1^{l-1}$. If there exists $e \in E \setminus \delta(r)$ with $\gamma_e < 0$, let j be the largest number in $\{2, \dots, n-1\}$ such that there exists $\gamma_{e_i} = w_{e_i}^l < 0$. Without loss of generality let $e \in F_j^{l-1} \in \rho^{l-1}$, $j \in \{1, \dots, q^{l-1}\}$. Then, let E^* be the path between e_n and H_j^{l-1} that does not contain F_j^{l-1} , that is,

$$E^* = H_j^{l-1} \cup F_{j+1}^{l-1} \cup \dots \cup H_{q^{l-1}-1}^{l-1} \cup F_{q^{l-1}}^{l-1}.$$

In either case, we have $\gamma_e = 0$ for any $e \in E^* \setminus \delta(r)$. Furthermore, similarly to the case of $c_r \geq 2$ and $O = \emptyset$, Algorithm 6.6 ensures that $|E^* \cap M| = |E^* \cap E_\pi|$ for any $(M,\pi) \in \mathcal{MP}(G)$ with $\beta_{(M,\pi)} > 0$. Hence we have

$$\begin{aligned} \mathbf{w}^T \mathbf{x}^{E^*} &= \sum_{e \in E^*} \left(\sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} - \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)} + \gamma_e \right) \\ &= \sum_{e \in E^*} \gamma_e + \sum_{e \in E^*} \left(\sum_{(M,\pi) \in \mathcal{M}_e} \beta_{(M,\pi)} - \sum_{(M,\pi) \in \mathcal{P}_e} \beta_{(M,\pi)} \right) \\ &= \gamma_{e_n} + \sum_{(M,\pi) \in \mathcal{MP}(G)} (|E^* \cap M| - |E^* \cap E_\pi|) \beta_{(M,\pi)} \\ &= \alpha_r. \end{aligned}$$

Therefore, \mathbf{x}^{E^*} and $[\overset{\alpha_r}{\beta}]$ are both feasible and optimal. Finally, α_r and components of vector β are obtained by additions and subtractions involving only the components of \mathbf{w} . Thus $[\overset{\alpha_r}{\beta}]$ is integral if \mathbf{w} is integral, which completes our proof. \square

Consequently, the next corollary follows immediately.

Corollary 6.7.17. *Let G be a cycle with $c_r = 1$ and $O = \emptyset$, we then have*

$$B_x(G, r, \mathbf{c}) = P_{xC}^3(G, r, \mathbf{c}). \quad \square$$

The discussion on TDI systems that characterize $B_x(G, r, \mathbf{c})$ on trees and cycles thereby concludes.

6.8 Conclusion

Unlike the r -tree polytope $R_x(G, r)$ or the extended bounded r -tree polytope $B_{xy}(G, r, \mathbf{c})$, the bounded r -tree polytope $B_x(G, r, \mathbf{c})$ possesses more unexplored aspects. First, we showed that $B_x(G, r, \mathbf{c})$ is also full-dimensional under the assumption that there is no unreachable elements. Meanwhile, comparing to $R_x(G, r)$, the necessary and sufficient conditions for the inequalities to be facet-defining become more complicated as we consider the capacity factor. There are more substructures to avoid in different situations in order to ensure the facet-defining property. For example, the presence of bridges, articulation nodes, especially articulation nodes with capacity 2, and triangles with specific properties prevent certain inequalities from being facet-defining. Besides, for any facet-defining inequality, there are certain general criteria concerning the coefficients of nodes in O and the right-hand side.

Furthermore, matching-partition inequalities and acyclicity-connectivity inequalities were inherited from the facial study of $R_x(G, r)$ with some adjustments made to incorporate the capacity factor. Upload capacity inequalities were discovered while studying the property of the right-hand side of the facet-defining inequalities, whereas capacity- i inequalities and i -articulation were developed based on the features of articulation nodes and nodes in O . We then showed that all these inequalities can be obtained by projection of the valid inequalities for $B_{xy}(G, r, \mathbf{c})$.

Besides, we also obtained some bounds on the Chvátal-Gomory rank of the matching-partition inequalities and the upload capacity inequalities, and showed that they are not trivially included in the first Chvátal closure even on trees and cycles. With the help of upload capacity inequalities and matching-partition inequalities, we gave for each case on trees and cycles a TDI linear system that completely characterizes $B_x(G, r, \mathbf{c})$.

In the next chapter, we focus on the computational aspect of the MBrT problem. We discuss the separation problems of all the previously introduced inequalities for both $B_{xy}(G, r, \mathbf{c})$ and $B_x(G, r, \mathbf{c})$, and study their influence via computational test.

Chapter 7

Computational study

In this chapter, we describe four branch-and-cut frameworks that were used to perform the computational simulation on the MBrT problem, and present the results obtained under several different setups. As there is no benchmark we can refer to in the literature, we compare the performances of the enhanced formulations incorporating the newly introduced constraints and the original formulations with the intention of showing the effectiveness of the new constraints on the computational aspect of the problem. The first two frameworks are based on the initial formulations of $B_x(G, r, \mathbf{c})$ and $B_{xy}(G, r, \mathbf{c})$ with CPLEX default setting. The other two frameworks correspond to enhanced formulations for $B_x(G, r, \mathbf{c})$ and $B_{xy}(G, r, \mathbf{c})$ incorporating some new constraints introduced in the previous chapters and a matheuristic (that is, a heuristic that relies on mathematical programming models, see Ball [2011], Papageorgiou et al. [2016]) to generate primal solutions. They are tested on a variety of instance sets with different graphs, different capacity settings and uniform edge weights. Finally, the results are presented and analyzed.

7.1 Methodology

In our work, the test instances have uniform edge weights, and they are selected from two different sources. The first part of the instances are fetched from SteinLib Testdata Library Koch et al. [2017]. As SteinLib files are not designed for scenario carrying information for nodes such as capacities in our case, a capacity file is generated for each of the SteinLib instances. The node number of the SteinLib instances ranges from 50 to 5200.

The rest of the instances are generated as random connected graphs with different capacity settings. Each generated instance contain a random connected graph, which is generated by creating a tree first and adding certain amount of edges into it after. The capacity of

Testset	Number	Node number	Capacity	Source	Density	Graph class
B	18	50-100	1-3	SteinLib	sparse	random
C	20	500	1-3	SteinLib	sparse	random
I320	20	320	1-3	SteinLib	mostly sparse	random
I640	16	640	1-3	SteinLib	mostly sparse	random
MSM	30	90-5181	1-3	SteinLib	sparse	grid
ran	110	10-499	1-4	generated	dense	random
rg_13	198	10-999	1-3	generated	sparse	random
rg_23	198	10-999	2-3	generated	sparse	random

Table 7.1: Options for the instances

nodes is between 1 and 4. The graphs have node number ranging from 10 to 1000. More details about the instances can be found in Table 7.1.

The instances are tested with the help of IBM ILOG CPLEX with C++ and Concert Technology. It allows us to build the model of our problem as well as insert customized cuts and heuristics in the branch-and-cut process conducted by CPLEX.

The instances are tested on servers equipped with CPUs of Intel[®] Xeon[®] E5-2670v2 with 2.50 GHz clock rate and 64 GB available random access memory (RAM). In order to compare the performance of the algorithms in a clear fashion and have better control of each cut generated in the process, the test is only run with single thread.

In this dissertation, four different frameworks are put into comparison. The first one uses a near minimal model for $B_x(G, r, \mathbf{c})$ to guarantee the feasibility of the solution and employs the default setting of CPLEX. This framework is referred to as $CPLEX_x$ hereafter. $CPLEX_x$ uses the following system of inequalities as an initial input, which necessarily has a polynomial number of constraints.

$$x_e - x(\delta(\{u, v\}) \setminus \delta(O)) \leq 0 \quad \forall e = uv \in E \setminus \delta(r), \quad (7.1)$$

$$x(E) \leq |E| - 1, \quad (7.2)$$

$$x(\delta(v)) \leq c_v \quad \forall v \in V, \quad (7.3)$$

$$0 \leq x_e \leq 1 \quad \forall e \in E. \quad (7.4)$$

In order to cut the infeasible integral solutions, $CPLEX_x$ also includes a separation algorithm for the following connectivity inequalities and the acyclicity inequalities

$$x_e - x(\delta(S) \setminus \delta(O)) \leq 0 \quad \forall e \in E[S], S \subseteq V \setminus \{r\}, \quad (7.5)$$

$$x(E[S]) \leq |S| - 1 \quad \forall S \subseteq V, |S| \geq 3. \quad (7.6)$$

The separation problem of (7.5) and (7.6) with respect to an integral solution can be

solved by some search algorithm. For (7.5) one looks for the connected component of the support graph of a solution that does not contain r , whereas for (7.6) one looks for a 2-connected block of the support graph of a solution. The cutting planes generated while cutting integral infeasible solutions are also called *lazy cuts* in CPLEX.

The second framework, namely BRT_x , also contains (7.1)- (7.4) as initial input, and separates (7.5) and (7.6) through lazy cuts. In addition, in order to help cut the infeasible fractional solutions, we also introduce some separation algorithms or cut-generation heuristics for valid inequalities to generate the so-called *user cuts*. The user cuts in BRT_x include those for (7.5), (7.6), and as well as for the following inequalities (which has been previously introduced in Section 6.4):

the matching-partition inequalities

$$x(M) - x(E_\pi \setminus \delta(O)) \leq 0 \quad \forall (M, \pi) \in \mathcal{MP}(G), \quad (7.7)$$

the acyclicity-connectivity inequalities

$$x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) \leq 0 \quad \forall W \subseteq S \subseteq V \setminus \{r\}, \quad (7.8)$$

the upload-capacity inequalities

$$x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0 \quad \forall v \in S \subseteq V \setminus \{r\}, \quad (7.9)$$

the i -articulation inequalities

$$x(F_S) - (i - 2)x(E[S]) - (i - 1)x(\delta(R) \cap \delta(S)) \leq 0 \quad \forall S \subseteq V_a \cap S_i, \quad (7.10)$$

and the tightening inequalities

$$x(\delta(O)) - x(\delta(R)) \leq \sum_{v \in S} (c_v - 2) \quad \forall S \subseteq V \setminus (\{r\} \cup O). \quad (7.11)$$

Note that none of these inequalities is mandatory to be included in the user cuts, and thus the implementation of BRT_x can vary depending on which sets of user cuts are selected, and as well as on the priorities set for different sets of user cuts.

The third framework, which is referred to as $CPLEX_{xy}$, uses a near minimal model for $B_{xy}(G, r, \mathbf{c})$ with the following system of inequalities as an initial input.

$$y_r = 1, \quad (7.12)$$

$$x(E) - y(V \setminus \{r\}) = 0, \quad (7.13)$$

$$x(\delta(v)) \leq c_v y_v \quad \forall v \in V, \quad (7.14)$$

$$0 \leq x_e \leq 1 \quad \forall e \in E. \quad (7.15)$$

$$0 \leq y_v \leq 1 \quad \forall v \in V. \quad (7.16)$$

In order to cut the infeasible integral solutions, $CPLEX_{xy}$ also need to include a separation algorithm for the following extended subtour elimination inequalities

$$x(E[S]) - y(S \setminus \{v\}) \leq 0, \quad \forall v \in S \subseteq V. \quad (7.17)$$

Particularly, we first check if there exists any edge in the solution with one of its ends not selected, that is, to separate $x_e - y_v \leq 0$ for any $e \in \delta(v), v \in V$. Then, similarly to $CPLEX_x$, one can look for a 2-connected block of the support graph of a solution to generate the other lazy cuts (7.17).

Correspondingly, the last framework, namely BRT_{xy} , also uses (7.12) - (7.16) as initial input and contains the same lazy cuts as $CPLEX_{xy}$. Besides, in the user cuts of BRT_{xy} we also include the separation of (7.17) and the following constraint

$$y_v - x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0 \quad \forall v \in S \subseteq V \setminus \{r\}. \quad (7.18)$$

Moreover, both BRT_x and BRT_{xy} include a matheuristic to generate integral feasible solutions and a preprocess to get rid of the unreachable elements of the graph.

A brief overview of the four different frameworks can be found in Table 7.2, while Table 7.3 shows their difference in the sense of initial models and cuts.

The following two sections present, for each set of inequalities, the separation problem and algorithms used to generate user cuts.

7.2 Separation problems for valid inequalities

For the separation problems, the number of some sets of the aforementioned inequalities such as box inequalities and capacity inequalities, are polynomial, and thus they can be easily separated in polynomial time. In the remainder of this section, we discuss

Framework	Node variable	User cut	Preprocess	Matheuristic
$CPLEX_x$	×	×	×	×
$CPLEX_{xy}$	✓	×	×	×
BRT_x	×	✓	✓	✓
BRT_{xy}	✓	✓	✓	✓

Table 7.2: Options for the branch-and-cut frameworks

	$CPLEX_x$	BRT_x	$CPLEX_x$	BRT_{xy}
Initial model	(7.1)- (7.4)	(7.1)- (7.4)	(7.12) - (7.16)	(7.12) - (7.16)
Lazy cuts	(7.5), (7.6)	(7.5), (7.6)	(7.17)	(7.17)
User cuts	-	(7.7)-(7.11)	-	(7.17), (7.18)

Table 7.3: Initial models and cuts of four branch-and-cut frameworks

the separation problem for each set of inequalities that are introduced in the previous chapters.

7.2.1 Connectivity inequalities

The separation problem of the connectivity inequalities (7.5), that is, $x_e - x(\delta(S) \setminus \delta(O)) \leq 0$ for $e \in E[S]$, $S \subseteq V \setminus \{r\}$, can be solved in polynomial time, as stated in the proposition below.

Proposition 7.2.1. *The separation problem of (7.5) reduces to $|V \setminus O| - 1$ minimum st -cut calculations.*

Proof. Given a vector $\mathbf{x} \in \mathbb{R}^E$, set the weight $w_{e'}$ of each edge e' in E as

$$w_{e'} = \begin{cases} 0 & \text{if } e' \in \delta(O), \\ x_{e'} & \text{otherwise.} \end{cases}$$

Note that we have $w(\delta(S)) = x(\delta(S) \setminus \delta(O))$ for any $S \subseteq V$. For any $V \setminus (\{r\} \cup O)$ let S_v denote the node set containing v that induces a min cut between r and v , and let f_v denote an edge in $\delta(v) \cap E[S_v]$ with the maximum value of x_{f_v} , that is,

$$S_v = \operatorname{argmin}_S \{w(\delta(S)) : v \in S \subseteq V \setminus \{r\}, O \subseteq S\},$$

$$f_v = \operatorname{argmax}_{e'} \{x_{e'} : e' \in \delta(v) \cap E[S_v]\}.$$

Note that for any set $S \subseteq V \setminus \{r\}$, $w(\delta(S)) = w(\delta(S \cup O))$ holds.

Correspondingly, for any $e' \in E \setminus \delta(r)$, let $S_{e'}$ denote the node set with $e' \in E[S_{e'}]$ that induces a min cut between r and e' , that is,

$$S_{e'} = \operatorname{argmin}_S \{w(\delta(S)) : e' \in E[S], S \subseteq V \setminus \{r\}\}.$$

It is trivial to see that

$$\max\{x_{e'} - x(\delta(S) \setminus \delta(O)) : e' \in E[S], S \subseteq V \setminus \{r\}\} = \max\{x_{e'} - w(S_{e'}) : e' \in E \setminus \delta(r)\}.$$

Besides, if $\delta(v) \cap E[S_v] = \emptyset$ for some $v \in V \setminus \{r\}$, then for any $e' \in \delta(v)$ we have $\delta(v) \cap \delta(O) = \emptyset$ and $e' \in \delta(S_v)$ and thus

$$x_{e'} - w(\delta(S_{e'})) \leq x_{e'} - w(\delta(S_v)) \leq 0.$$

For any edge $e' = uv \in E \setminus (\delta(r) \cup \delta(O))$ such that $\delta(u) \cap E[S_u] \neq \emptyset$ and $\delta(v) \cap E[S_v] \neq \emptyset$, we have

$$x_{e'} - w(\delta(S_{e'})) \leq \min\{x_{f_u} - w(\delta(S_u)), x_{f_v} - w(\delta(S_v))\}.$$

Moreover, for any edge $e' = uv \in \delta(O) \setminus \delta(r)$ with $v \in O$, we have

$$w(\delta(S_{e'})) = w(\delta(S_u)),$$

and hence

$$x_{e'} - w(\delta(S_{e'})) \leq x_{f_u} - w(\delta(S_u)),$$

Therefore, if there exists an inequality in (7.5) that is violated by \mathbf{x} , we have

$$\max\{x_{e'} - w(S_{e'}) : e' \in E \setminus \delta(r)\} = \max\{x_{f_v} - w(\delta(S_v)) : v \in V \setminus \{r\}, \delta(v) \cap E[S_v] \neq \emptyset\}.$$

Otherwise if there does not exist any inequality in (7.5) that is violated by \mathbf{x} , we have either $\delta(v) \cap E[S_v] = \emptyset$ for all $v \in V \setminus \{r\}$, or

$$\max\{x_{f_v} - w(\delta(S_v)) : v \in V \setminus \{r\}, \delta(v) \cap E[S_v] \neq \emptyset\} \leq \max\{x_{e'} - w(S_{e'}) : e' \in E \setminus \delta(r)\}.$$

Algorithm 7.1: Algorithm to separate connectivity inequalities

Input : $\bar{\mathbf{x}}$.

Output: Most violated constraint $x_{f_v} - x(\delta(S_v) \setminus \delta(O)) \leq 0$ if exists.

begin

- 1 Set $\lambda = 0$ and $S = V \setminus (\{r\} \cup O)$.
 - while** $S \neq \emptyset$ **do**
 - 2 Take a node $v \in S$.
 - 3 Calculate the min cut between r and v to obtain S_v and f_v (if exists).
 - if** $\delta(v) \cap E[S_v] \neq \emptyset$ **and** $x_{f_v} - w(\delta(S_v)) > \lambda$ **then**
 - 4 Set $x_{f_v} - x(\delta(S_v) \setminus \delta(O)) \leq 0$ as the most violated inequality found so far.
 - 5 Set $S = S \setminus \{v\}$.
-

To summarize, finding the most violated inequalities in (7.5) reduces to $|V \setminus (\{r\} \cup O)|$ minimum st-cut calculations. Algorithm 7.1 demonstrates the algorithm to solve the separation problem of (7.5). \square

7.2.2 Upload capacity inequalities

The separation problem of the upload capacity inequalities (7.9), that is, $x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0$ for $v \in S \subseteq V \setminus \{r\}$, can be solved in polynomial time.

Proposition 7.2.2. *The separation problem of (7.9) reduces to $|V| - 1$ minimum st-cut calculations.*

Proof. Given a node $v \in V \setminus \{r\}$ and a vector $\mathbf{x} \in \mathbb{R}^E$, let the weight of an edge $e' \in E$ be

$$w_{e'}^v = \begin{cases} 0, & \text{if } e' \in \delta(O \setminus \{v\}), \\ x_{e'}, & \text{otherwise.} \end{cases}$$

Then let S_v be the node set containing v that induces a min cut between r and v , that is,

$$S_v = \operatorname{argmin}_S \{w^v(\delta(S)) : v \in S \subseteq V \setminus \{r\}\}.$$

Then the separation problem reduces to obtaining

$$\max\{x(\delta(v)) - S_v : v \in S \subseteq V \setminus \{r\}\}.$$

Therefore, at most $|V| - 1$ minimum $s - t$ cut calculations are needed to find a most violated inequality in (7.9). \square

Moreover, similar result can be obtained for (7.18), that is, $y_v - x(\delta(S) \setminus \delta(O \setminus \{v\})) \leq 0$ for $v \in S \subseteq V \setminus \{r\}$, as stated in the following proposition.

Proposition 7.2.3. *The separation problem of (7.18) reduces to $|V| - 1$ minimum st-cut calculations.*

Proof. We follow the same notation and edge-weight vector as previously defined for the separation problem of (7.9). Clearly, we have the following relation

$$\max\{y_v - x(\delta(S) \setminus \delta(O \setminus \{v\})) : v \in S \subseteq V \setminus \{r\}\} = \max\{y_v - S_v : v \in S \subseteq V \setminus \{r\}\}.$$

Therefore, the proposition holds. \square

7.2.3 Subtour elimination inequalities

For the separation of the subtour elimination inequalities (7.6), that is, $x(E[S]) \leq |S| - 1$ for $S \subseteq V$, Padberg and Wolsey [1983] showed that it can be done in polynomial time using at most $|V| - 2$ maximum flow calculations. Moreover, we generalize their approach to an extent that allows us to separate a series of inequalities with similar structures, including not only (7.6) but also the extended elimination inequalities (7.17), that is, $x(E[S]) - y(S \setminus \{v\}) \leq 0$ for $v \in S \subseteq V$.

We first claim that the following lemma is true.

Lemma 7.2.4. *Given a vector $\begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{E+V}$, for any node $u \in V$, $m_u = \max\{x(E[S]) - y(S) : u \in S \subseteq V\}$ can be obtained with a single minimum st-cut calculation.*

Proof. This proof proceeds similarly to the proof of Padberg and Wolsey [1983]. First, we construct a new graph G^* based on G and $\begin{bmatrix} x \\ y \end{bmatrix}$ as follows. We add two nodes s and t as source and sink to G , and for each $v \in V$, we add two edges sv and vt with weights

$$w_{sv} = \max\left\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\right\},$$

$$w_{vt} = \max\left\{y_v - \frac{1}{2}x(\delta_G(v)), 0\right\}.$$

For each edge e in G , we assign the weight

$$w_e = \frac{1}{2}x_e.$$

For any $S \subseteq V$, the weight of the cut $\delta_{G^*}(S \cup \{s\})$ can be calculated as follows.

$$\begin{aligned} w(\delta_{G^*}(S \cup \{s\})) &= \sum_{v \in V \setminus S} \max\left\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\right\} \\ &\quad + \sum_{v \in S} \max\left\{y_v - \frac{1}{2}x(\delta_G(v)), 0\right\} + \frac{1}{2}x(\delta_G(S)) \\ &= \sum_{v \in V} \max\left\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\right\} + \sum_{v \in S} \left(\max\left\{y_v - \frac{1}{2}x(\delta_G(v)), 0\right\}\right. \\ &\quad \left. - \max\left\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\right\}\right) + \frac{1}{2}x(\delta_G(S)) \\ &= \sum_{v \in V} \max\left\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\right\} + \sum_{v \in S} \left(y_v - \frac{1}{2}x(\delta_G(v))\right) + \frac{1}{2}x(\delta_G(S)) \\ &= \sum_{v \in V} \max\left\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\right\} + (y(S) - x(E[S]) - \frac{1}{2}x(\delta_G(S))) \\ &\quad + \frac{1}{2}x(\delta_G(S)) \end{aligned}$$

$$= \sum_{v \in V} \max\left\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\right\} + y(S) - x(E[S]).$$

Note that the value of $\sum_{v \in V} \max\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\}$ is independent of S , and is exclusively dependent on $[\frac{x}{y}]$. Thus for a given $u \in V$, we have

$$\begin{aligned} m_u &= \max\{x(E[S]) - y(S) : u \in S \subseteq V\} \\ &= -\min\{w(\delta_{G^*}(S \cup \{s\})) : u \in S \subseteq V\} + C, \end{aligned}$$

where $C = \sum_{v \in V} \max\{\frac{1}{2}x(\delta_G(v)) - y_v, 0\}$. It can be noted that for each $u \in V$, in order to guarantee that $u \in S$, one can set $w_{su} = +\infty$. Thus one can obtain m_u with a single minimum st-cut calculation on G^* . \square

Lemma 7.2.4 can be applied on the separation problems of both the subtour elimination inequalities (7.6), that is, $x(E[S]) \leq |S| - 1$ for $S \subseteq V$, and the extended subtour elimination inequalities (7.17), that is, $x(E[S]) - y(S \setminus \{v\}) \leq 0$ for $v \in S \subseteq V$.

Proposition 7.2.5. *The separation problem of (7.6) reduces to $|V| - 2$ minimum st-cut calculations.*

Proof. Given $x \in \mathbb{R}^E$, let $y_v = 1$ for all $v \in V$. Then $y(S) = |S|$ for any $S \subseteq V$. According to Lemma 7.2.4, for a given node $v \in V$, $m_v = \max\{x(E[S]) - |S| : v \in S \subseteq V\}$ can be obtained with a single minimum st-cut calculation.

Without loss of generality, let $V = \{v_1, \dots, v_n\}$, and $S_i = \{v_i, \dots, v_n\}$ for $i \in \{1, \dots, n\}$. Let

$$m_i = \max\{x(E[S]) - |S| : v_i \in S \subseteq S_i\}.$$

m_i can be calculated using same approach as for m_v except that we set $w_{ut} = +\infty$ for any $u \in V \setminus S_i$.

Since $|S_i| \leq 2$ for $i \in \{n-1, n\}$ and (7.6) concerns only the cases with $|S| \geq 3$, one only needs to calculate $\max\{x(E[S]) - |S| : v_i \in S \subseteq S_i\}$, for $i \in \{1, \dots, n-2\}$, that is,

$$\max\{x(E[S]) - |S| : S \subseteq V, |S| \geq 3\} = \max\{m_i : i \in \{1, \dots, n-2\}\}$$

Thus, the separation problem of (7.6) can be solved with $|V| - 2$ minimum st-cut calculations. \square

Proposition 7.2.6. *The separation problem of (7.17) reduces to $|V| - 1$ minimum st-cut calculations.*

Proof. Given $\begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^{E+V}$, similarly, let $V = \{v_1, \dots, v_n\}$, $S_i = \{v_i, \dots, v_n\}$ for $i \in \{1, \dots, n\}$, and

$$m_i = \max\{x(E[S]) - y(S) : v_i \in S \subseteq S_i\}.$$

According to Lemma 7.2.4, one can obtain m_i , and

$$S_i = \operatorname{argmax}_S \{x(E[S]) - y(S) : v_i \in S \subseteq S_i\}$$

with a single minimum st-cut calculation. Additionally, as $|S_n| = 1$ and (7.17) considers node sets with at least two nodes, thus

$$\begin{aligned} & \max\{x(E[S]) - y(S \setminus \{v\}) : v \in S \subseteq V, |S| \geq 2\} \\ &= \max\{m_i + \max\{y_u : u \in S_i\} : i \in \{1, \dots, n-1\}\}. \end{aligned}$$

Therefore, $\max\{x(E[S]) - y(S \setminus \{v\}) : v \in S \subseteq V, |S| \geq 2\}$ can be obtained with $|V| - 1$ minimum st-cut calculations. \square

7.2.4 Matching-partition inequalities

As mentioned earlier, Didi-Biha et al. [2015] have studied similar matching-partition constraints for the connected subgraph problem. In order to distinguish the matching-partitions for the two problems, we call the pair (M, π) a *CSP matching-partition* of G , if $\pi = \{S_1, \dots, S_p\}$, $p \geq 1$, is a partition of V and $M = \{e_1, \dots, e_p\}$ is a matching of G with $e_i \in E[S_i]$ for all $i \in \{1, \dots, p\}$. Denote $\mathcal{MP}_{CSP}(G)$ the set of all CSP matching-partitions of G . The CSP matching-partition constraint is as follows.

$$x(M) - x(E_\pi) \leq 1 \quad \forall (M, \pi) \in \mathcal{MP}_{CSP}(G).$$

The difference between the two versions of matching-partition inequalities is that in the matching-partition for the MBrT problem, a matching edge in the partition class containing r is unnecessary. Didi-Biha et al. [2015] have proved by a polynomial transformation from multiterminal cut problem, the separation problem of the CSP matching-partition constraint is \mathcal{NP} -hard. Due to the relation between CSP and MBrT problem, we prove the \mathcal{NP} -hardness of the separation of the matching-partition inequalities by reducing the separation problem of the CSP matching-partition inequalities to it.

Let \mathbf{x} be any vector in \mathbb{R}^E , and

$$g_x^{MBrT}(M, \pi) = x(M) - x(E_\pi \setminus \delta(O)) \quad \forall (M, \pi) \in \mathcal{MP}(G),$$

$$g_x^{CSP}(M, \pi) = x(M) - x(E_\pi) \quad \forall (M, \pi) \in \mathcal{MP}_{CSP}(G).$$

The following lemma has been proven by Didi-Biha et al. [2015].

Lemma 7.2.7. *Given a vector $\mathbf{x} \in \mathbb{R}^E$, the problem of finding $\max\{g_x^{CSP}(M, \pi) : (M, \pi) \in \mathcal{MP}_{CSP}(G)\}$ is \mathcal{NP} -hard. \square*

Then we prove the \mathcal{NP} -hardness of the separation problem of the matching-partition inequalities for the MBrT problem by showing a polynomial-time reduction from the separation problem of the matching-partition inequalities for CSP to it.

Proposition 7.2.8. *The separation problem of the CSP matching-partition inequalities can be reduced to a polynomial number (at most $|V| - 1$) of separation problems of matching-partition inequalities (7.7).*

Proof. Given an instance of CSP problem on graph $G = (V, E)$, consider an arbitrary node u in V . For each $v \in N(u)$, we obtain a new graph denoted as G'_v by contracting the edge uv into a node r_v . Note we have $O = \emptyset$ for G'_v . Let (M'_v, π'_v) be a matching-partition of G'_v such that $g_x^{MBrT}(M'_v, \pi'_v)$ is maximized, that is,

$$g_x^{MBrT}(M'_v, \pi'_v) = \max\{g_x^{MBrT}(M, \pi) : (M, \pi) \in \mathcal{MP}(G'_v)\}.$$

Let (M', π') be a CSP matching-partition of G such that $g_x^{CSP}(M', \pi')$ is maximized, that is,

$$g_x^{CSP}(M', \pi') = \max\{g_x^{CSP}(M, \pi) : (M, \pi) \in \mathcal{MP}_{CSP}(G)\}.$$

Without loss of generality, for $v \in N(u)$, let S_0^v be the partition class of π'_v that contains r_v , and let $S_0'^v$ be a subset of V such that

$$S_0'^v = S_0^v \cup \{u, v\} \setminus \{r_v\}.$$

Let e_0^v be the edge in $E[S_0'^v]$ with the maximum value of $x_{e_0^v}$, that is,

$$e_0^v = \operatorname{argmax}_e \{x_e : e \in E[S_0'^v]\}.$$

We claim that the following relation holds.

$$g_x^{CSP}(M', \pi') = \max\{g_x^{MBrT}(M'_v, \pi'_v) + x_{e_0^v} : v \in N(u)\}.$$

For a given $v \in N(u)$, it can be seen that,

$$g_x^{MBrT}(M'_v, \pi'_v) + x_{e_0^v} = \max\{x(M) - x(E_\pi) : (M, \pi) \in \mathcal{MP}_{CSP}(G), uv \notin E_\pi\}.$$

Moreover, it is known that for any $(M, \pi) \in \mathcal{MP}_{CSP}(G)$ there exists some $e \in \delta(u)$, such that $e \notin E_\pi$, since otherwise u is isolated. Hence, one has

$$\begin{aligned} g_x^{CSP}(M', \pi') &= \max\{x(M) - x(E_\pi) : (M, \pi) \in \mathcal{MP}_{CSP}(G), uv \notin E_\pi, v \in N(u)\} \\ &= \max\{g_x^{MBrT}(M'_v, \pi'_v) + x_{e_0^v} : v \in N(u)\}. \end{aligned}$$

Therefore, if the matching-partition inequalities on G'_v , for $v \in N(u)$, can be separated in polynomial time, then the matching-partition inequalities for CSP on G can also be separated in polynomial time. \square

This shows that the separation of matching-partition inequalities (7.7) is at least as hard as the separation of the matching-partition inequalities for CSP, and thus its \mathcal{NP} -hardness is proved.

Proposition 7.2.9. *The separation problem of (7.7) is \mathcal{NP} -hard.* \square

Despite its \mathcal{NP} -hardness the separation problem of the matching-partition inequalities can be solved in polynomial time on cycles and trees. On trees, the only matching-partition inequalities that define facets are the connectivity inequalities, for which the separation problem is known to be polynomially solvable according to Proposition 7.2.1.

On cycles, an approach for the separation of the matching-partition inequalities is presented below.

Firstly, let G be a cycle with $O = \{v_o\}$. Let $(M, \pi) \in \mathcal{MP}(G)$ be a matching-partition with $|M| \geq 2$. From Subsections 6.4.1 and 6.7.2, we can see that the matching-partition inequalities associated with (M, π) is redundant. Hence, we can focus on the case that has $O = \emptyset$, which means $g_x^{MBrT}(M, \pi) = x(M) - x(E_\pi)$ for any $(M, \pi) \in \mathcal{MP}(G)$.

Let $G = (V, E)$ be a cycle with $O = \emptyset$. Denote by $G' = (V', E')$, the cycle obtained by extending r to an edge rr' . Given a vector $\mathbf{x} \in \mathbb{R}^E$, let \mathbf{x}' be a vector in $\mathbb{R}^{E'}$ such that

$$x'_e = \begin{cases} x_e & \text{if } e \in E, \\ L & \text{if } e = rr', \end{cases}$$

where L is a sufficiently large number that satisfies $L > x_e$ for any $e \in E$. Now we show that the following proposition holds.

Proposition 7.2.10.

$$\begin{aligned} & \max\{g_x^{MBrT}(M, \pi) : (M, \pi) \in \mathcal{MP}(G)\} \\ & = \max\{g_x^{CSP}(M', \pi') : (M', \pi') \in \mathcal{MP}_{CSP}(G')\} - L. \end{aligned}$$

Proof. We claim that for any CSP matching-partition $(M^*, \pi^*) \in \mathcal{MP}_{CSP}(G')$ such that $g_x^{CSP}(M^*, \pi^*)$ is maximized, we have $rr' \in M^*$. First, one has $rr' \notin E_{\pi^*}$, since otherwise combining the two partition classes that contain r and r' into one partition class and picking rr' as the corresponding edge in the matching gives us a better solution. As $rr' \notin E_{\pi^*}$ and $x_{rr'} > x_e$ for any $e \in E$, one must have $rr' \in M^*$.

Furthermore, it can be seen that each matching partition $(M, \pi) \in \mathcal{MP}(G)$ has a one-to-one correspondence with a CSP matching-partition $(M', \pi') \in \mathcal{MP}_{CSP}(G')$ with $rr' \in M'$, and vice versa. Hence the following relation holds.

$$\begin{aligned} & \max\{x(M) - x(E_\pi) : (M, \pi) \in \mathcal{MP}(G)\} \\ & = \max\{x(M') - x(E_{\pi'}) : (M', \pi') \in \mathcal{MP}_{CSP}(G'), rr' \in M'\} - L. \end{aligned}$$

Therefore, the proposition holds. □

According to this proposition, the separation problem of the matching-partition inequalities is polynomially solvable on cycles if the separation problem of the CSP matching-partition inequalities is polynomially solvable on cycles. Didi-Biha et al. [2015] showed that the separation problem of the CSP matching-partition inequalities is polynomially solvable on cycles. Thus the following corollary holds.

Corollary 7.2.11. *Let G be a cycle. The separation problem of (7.7) can be solved in polynomial time.* □

For the matching-partition inequalities, a few ways of generating matching-partition inequalities have been tested, among which, there are two different approaches in terms of how the matching-partitions are generated. One is to have a matching first, then construct a partition based on it, and the other approach is just the opposite. The former seems to be a natural choice, since the matching can be built in prior based on the weights on edges.

Maximum weighted matching is our first choice going into experiment, as generally the most violated matching-partition inequality tends to contain as many large-weight edges as possible in the matching. Consider the x -value of each edge as its weight, a matching-partition in this case is built based on the maximum weighted matching. The partition classes are constructed by extending each matching edge to an edge set while ensuring the weights of the edges between partition classes to be as small as possible.

The second method is based on the fact that the edges with maximal weight value should never be present in $E_\pi \setminus \delta(O)$ of a most violated matching-partition inequality. Thus one can derive a subgraph containing only such edges, and the connected components in this subgraph lead to the initial partition classes. After that, the edges with second largest weight can be processed the same way, and so forth till the partition is fully formed. This heuristic has been proved to be able to generate some inequalities that are helpful to eliminate certain fractional points.

Another method is based on the projection from the inequalities for $B_{xy}(G, r, \mathbf{c})$. One first lift a vector $\bar{x} \in \mathbb{R}^E$ to a vector $\left[\frac{\bar{x}}{\bar{y}}\right] \in \mathbb{R}^{E+V}$ by setting $\bar{y}_v = \max\{\bar{x}_e : e \in \delta(v)\}$. Then we can separate the extended subtour elimination inequalities (7.17) to find a set $S \subsetneq V$ as the first partition class. If $r \in S$, then one looks for a matching $\{e_1, e_2\}$ of $E[\bar{S}]$ and a node set $S' \subsetneq \bar{S}$ such that $x_{e_1} + x_{e_2} - x(\delta(S') \cap E[\bar{S}] \setminus \delta(O))$ is maximized. This can be achieved using the separation algorithm described in Didi-Biha et al. [2015] for the inequality $x_e + x_f - x(\delta(S)) \leq 0$ for $e \in E[S]$ and $f \in E[\bar{S}]$, which can be reduced to $\mathcal{O}(|V|)$ minimum st-cut calculations.

If $r \notin S$, we first take an edge e_1 in $E[S]$ with the maximum \bar{x} -value to be in the matching. Then similarly to the separation of the connectivity inequalities (7.5), one looks for an edge $e_2 \in E[\bar{S}] \setminus \delta(r)$ and a node set $S' \subseteq \bar{S} \setminus \{r\}$ such that $x_{e_2} - x(\delta(S') \cap E[\bar{S}] \setminus \delta(O))$ is maximized.

This cut generation heuristic is given in Algorithm 7.2, and we show that the following result holds.

Proposition 7.2.12. *Algorithm 7.2 returns a matching-partition inequality using $\mathcal{O}(|V|)$ minimum st-cut calculations.*

Proof. It is trivial to see that Algorithm 7.2 ensures that π is a partition of V , and (M, π) is in $\mathcal{MP}(G)$. Thus the matching-partition inequality it returns is valid.

Furthermore, as each of the separation algorithms for (7.17), (7.5) and the inequality $x_e + x_f - x(\delta(S)) \leq 0$ for $e \in E[S]$ and $f \in E[\bar{S}]$ can be reduced to $\mathcal{O}(|V|)$ minimum st-cut calculations, one has that Algorithm 7.2 also reduces to $\mathcal{O}(|V|)$ minimum st-cut calculations. \square

This heuristic aims at finding a matching partition with 2 matching edges, and can be extended to a heuristic that looks for a matching partition with any fixed number of matching edges by simply increase the number of iterations.

Algorithm 7.2: Cut generation algorithm for matching-partition inequalities

Input : $\bar{x} \in \mathbb{R}^E$.

Output: Matching-partition inequality $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$ that is violated by \bar{x} .

begin

```

1   Set  $\bar{y}_v = \max\{\bar{x}_e : e \in \delta(v)\}$  for all  $v \in V$ .
2   Find an extended subtour elimination inequality that is violated by  $\lceil \frac{\bar{x}}{\bar{y}} \rceil$  and is
    associated with  $S$  using the separation algorithm for (7.17).
    if  $S$  is a proper subset of  $V$  then
      if  $r \in S$  then
3       Set  $S_0 = S$ .
4       Find  $\{e_1, e_2\} \subseteq E[\bar{S}]$  and a node set  $S' \subsetneq \bar{S}$  such that
         $\bar{x}_{e_1} + \bar{x}_{e_2} - \bar{x}(\delta(S') \cap E[\bar{S}] \setminus \delta(O))$  is maximized using a similar algorithm
        as described in Didi-Biha et al. [2015] for their connectivity inequalities.
5       Set  $\pi = \{S, S', \bar{S} \setminus S'\}$  and  $M = \{e_1, e_2\}$ .
      else
6       Set  $S_1 = S$ .
7       Find an edge  $e_1$  in  $E[S_1]$  with the maximum  $\bar{x}$ -value.
8       Find an edge  $e_1 \in E[\bar{S}] \setminus \delta(r)$  and a node set  $S' \subseteq \bar{S} \setminus \{r\}$  such that
         $\bar{x}_{e_1} - \bar{x}(\delta(S') \cap E[\bar{S}] \setminus \delta(O))$  is maximized using a similar algorithm to
        the separation algorithm of (7.5).
9       Set  $\pi = \{S, S', \bar{S} \setminus S'\}$  and  $M = \{e_1, e_2\}$ .
10      Return the inequality  $x(M) - x(E_\pi \setminus \delta(O)) \leq 0$  if it is violated by  $\bar{x}$ ;

```

7.2.5 i-articulation inequalities and tightening inequalities

Recall that a special case of the i-articulation inequalities (7.10) is the following capacity-i inequalities

$$x(\delta(O) \setminus \delta(R)) - (i - 2)x(E[S]) - (i - 1)x(\delta(R) \setminus \delta(O)) \leq 0 \quad \forall S \subseteq S_i, \quad (7.19)$$

where $S_i = \{v \in V \setminus \{r\} : c_v = i, i \geq 2\}$ and $R = V \setminus (S \cup O)$.

Since the capacity-i inequalities, the i-articulation inequalities and tightening inequalities all have similar structures, their separation problems can be solved in a similar way.

The separation of capacity-i inequalities (7.19) can be done in polynomial time by calculating the minimum cut between $R^0 = V \setminus (S_i \cup O)$ and O for each possible i between 2 and $\max\{c_v : v \in V \setminus (\{r\} \cup O)\}$. Therefore, at most $n - 2$ maximum flow calculation is required to separate (7.19).

Proposition 7.2.13. *The separation problem of (7.19) reduces to at most $|V \setminus O| - 1$*

minimum st -cut calculations.

Proof. Given $\mathbf{x} \in [0, 1]^E$ and integer $2 \leq i \leq \max\{c_v : v \in V \setminus \{r\}\}$, we claim a set $S \subseteq S_i$ that induces maximum $x(\delta(O) \setminus \delta(R)) - (i - 2)x(E[S_i]) - (i - 1)x(\delta(R) \setminus \delta(O))$ with $R = V \setminus (S \cup O)$ can be found in polynomial time. As $x(\delta(O))$ is fixed for any given \mathbf{x} , the problem is equivalent to finding the minimum $x(\delta(R)) + (i - 2)(x(E[S_i]) + x(\delta(R) \setminus \delta(O)))$. We first construct a graph $G' = (V', E')$ based on G by adding two nodes the source s and the sink t to get

$$\begin{aligned} V' &= V \cup \{s, t\}, \\ E' &= E \cup \{sv : v \in R^0 \cup S_i\} \cup \{vt : v \in O\}. \end{aligned}$$

The edge weight function for E' is defined as follows. First of all, in order to ensure that in any minimum $s - t$ cut of G' , R^0 is on the side of s , and O is on the side of t , we set

$$w_e = M \quad \forall e \in \delta(t) \cup \delta(s) \cap \delta(R^0),$$

where M is a large enough number.

Let

$$f(v) = \sum_{v_j \in S_i \setminus \{v\}} x_{vv_j} \quad \forall v \in S_i.$$

We set the weights of the rest of the edges as follows

$$w_e = \begin{cases} x_e, & \text{if } e \in E \setminus E[S_i], \\ \frac{i}{2}x_e, & \text{if } e \in E[S_i], \\ \frac{i-2}{2}f(v), & \text{if } e = sv \in \delta(s) \cap \delta(S_i). \end{cases}$$

Given any $S \subseteq S_i$, let $R = V \setminus (S \cup O)$ and $R' = R \cup \{s\}$. The weight of the cut set in G' induced by R' the following

$$\begin{aligned} w(\delta_{G'}(R')) &= x(\delta_G(R)) + \frac{i-2}{2} \sum_{e \in \delta(R) \cap \delta(S)} x_e + \sum_{v \in S} w_{sv} \\ &= x(\delta_G(R)) + \frac{i-2}{2} \sum_{e \in \delta(R) \cap \delta(S)} x_e + \frac{i-2}{2} \sum_{v \in S} f(v) \\ &= x(\delta_G(R)) + (i-2)x(E[S] + x(\delta(R) \cap \delta(S))). \end{aligned}$$

Thus the minimum $s - t$ cut in G' gives us the maximum $x(\delta(O) \setminus \delta(R)) - (i - 2)x(E[S]) - (i - 1)x(\delta(R) \setminus \delta(O))$ for any given $\mathbf{x} \in [0, 1]^E$ and any integer i with $2 \leq i \leq \max\{c_v :$

$v \in V \setminus \{r\}$. Additionally, there can be at most $|V| - 2$ different values for i for each node in $V \setminus (O \cup \{r\})$. Therefore, the separation of (7.19) can be done in polynomial time with at most $|V \setminus O| - 1$ minimum st-cut calculations. \square

Similar to the capacity- i nodes inequalities, a single minimum st-cut calculation is sufficient to separate the tightening inequalities (7.11), that is, $x(\delta(O)) - x(\delta(R)) \leq \sum_{v \in S} (c_v - 2)$ for $S \subseteq V \setminus (\{r\} \cup O)$ and $R = V \setminus (S \cup O)$, as stated in the following proposition.

Proposition 7.2.14. *The separation problem of (7.11) reduces to a minimum st-cut problem.*

Proof. Given any $\mathbf{x} \in [0, 1]^E$, we claim a set $R \subset V \setminus O$ that induces maximum $x(\delta(O)) - x(\delta(R)) - \sum_{v \in S} (c_v - 2)$ can be found in polynomial time. As $x(\delta(O))$ is fixed for any given \mathbf{x} , the problem is equivalent to finding the minimum $x(\delta(R)) + \sum_{v \in S} (c_v - 2)$.

We first construct a graph $G' = (V', E')$ based on G by adding two nodes, the source s and the sink t , and a few related edges into graph G . One then has $V' = V \cup \{s, t\}$. For each node $v \in R^0 \cup V_i$, add an edge $sv \in E'$, and for each $v \in O$, add an edge $vt \in E'$, where $S^0 = V \setminus (R^0 \cup O)$, $R^0 = \{r\}$.

The edge weight function for E' is defined as follows. Let $f(v) = c_v - 2$, for any $v \in S^0$. For any edge $e \in E$, its weight $w_e = x_e$. For $e \in \delta(s) \cap \delta(R^0)$ or $e \in \delta(t)$, $w_e = M$, where M is a large enough number. For $sv \in \delta(s) \cap \delta(S^0)$, $w_{sv} = f(v)$. The minimum $s - t$ cut in G' includes the node set R^0 on the side of s and the node set O on the side of t due to the weights. Assume node set R' induces a minimum $s - t$ cut in G' , with $R' = R \cup \{s\}$, $R = R^0 \cup (S^0 \setminus S)$, $S = V \setminus (R \cup O)$. The weight of the cut set is as follows.

$$\begin{aligned} & x(\delta_G(R)) + \sum_{v \in S} w_{sv} + \sum_{v \in S^0 \setminus S} w_{vt} \\ &= x(\delta_G(R)) + \sum_{v \in S} f(v) - 0 \sum_{v \in S^0 \setminus S} f(v) \\ &= x(\delta_G(R)) + \sum_{v \in S} (c_v - 2) \end{aligned}$$

This minimum cut calculation gives us the maximum $x(\delta(O)) - x(\delta_G(R)) - \sum_{v \in S} (c_v - 2)$ for any given $\mathbf{x} \in [0, 1]^E$, and the transformation from G to G' can be done in polynomial time. Therefore, the separation of (7.11) reduces to a minimum st-cut problem. \square

Following the same notation as in Subsection 6.4.6, the separation of the i -articulation inequalities (7.10) can be done in polynomial time as well by calculating the minimum cut between $R^0 = V \setminus ((S_i \cap V_a) \cup O)$ and $D_{S_i \cap V_a}$ for each possible i between 2 and $\max\{c_v : v \in V \setminus \{r\}\}$. Therefore, at most $|V| - 2$ minimum st-cut calculations are

required to separate (7.10). The algorithm and the proof would proceed the same way as for the separation algorithm of capacity- i inequalities, and the only difference is one substitutes $D_{S_i \cap V_a}$ and $F_{S_i \cap V_a}$ for O and $\delta(O)$. Thus we give the following result directly.

Proposition 7.2.15. *The separation problem of (7.10) reduces to at most $|V \setminus O| - 1$ minimum st -cut calculations. \square*

7.2.6 Acyclicity-connectivity inequalities

For the acyclicity-connectivity inequalities (7.8), that is, $x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) \leq 0$ for $W \subseteq S \subseteq V \setminus \{r\}$, the complexity of its separation problem remains unknown. We propose the following conjecture.

Conjecture 7.2.16. *The separation problem of (7.8) is \mathcal{NP} -hard.*

A straightforward heuristic is developed based on the observation that the fractional points generated in the branch-and-cut process which violate the acyclicity-connectivity inequalities should always have a support graph with 2-connected components. Otherwise, they would be cut by the connectivity inequalities beforehand. Thus, the heuristic is based on the support graph of the given solution as follows. For each 2-connected component that does not contain r of the support graph, consider the corresponding edge set as $E[W]$ and obtain S from a minimum cut between r and W with respect to the solution. If the inequality $x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) \leq 0$ is violated, then it is added into the model.

7.3 Primal matheuristic

Let \bar{x} be a fractional solution obtained in the process of branch-and-cut with respect to BRT_x . This section presents a matheuristic that use this primal information to derive an integer feasible point, or in other words, a primal bound (or lower bound in our case).

The matheuristic is as follows.

We start with a pool of nodes $P \subseteq V$ containing only r at first, and each iteration select one node and one edge incident with the pool, until either a spanning tree has been found or there is no edge to select any more. There are a few options on how we grow the pool of nodes, and the priority one takes into consideration while choosing among the edges incident with the pool. We have decided that the information of the fractional solution \bar{x} carries should be weighed the most. Aside from that, when expanding the pool, the potential of the selected node is evaluated, where the potential of a node means if it is

selected how many other nodes can be selected further in the process. Essentially, one can compare the capacity of the nodes first, because a higher capacity often indicates more edges incident with the node can be selected. If two candidates have the same capacity, then we compare the sum of capacity of their neighbors. The details of the matheuristic are presented as in Algorithm 7.3.

Algorithm 7.3: Primal matheuristic

Input : Fractional solution $\bar{\mathbf{x}} \in \mathbb{R}^E$.
Output: Integral feasible solution $\mathbf{x}^{E^*} \in \mathbb{Z}^E$.
begin

- 1 Set $P = \{r\}$, $O' = \emptyset$, $E^* = \emptyset$.
 while *there still exists some node that is not in the pool P , and*
 $\delta(P) \setminus \delta(O') \neq \emptyset$ **do**
- 2 Among the edges in $\delta(P) \setminus \delta(O')$, choose one edge $e = uv$ according to the
 following priorities: \bar{x} value, capacity, sum of capacity of neighbors.
- 3 Set $E^* = E^* \cup \{e\}$.
- 4 Reduce the capacity of u and v by 1, and if the capacity of u (v ,
 respectively) becomes 0, add u (v , respectively) to O' .

Algorithm 7.3 essentially has the structure of Prim's algorithm for the minimum spanning tree problem, see Prim [1957]. Nonetheless it considers more factors other than the weights and has a different termination condition.

Proposition 7.3.1. *Algorithm 7.3 computes a feasible solution in time $\mathcal{O}(|V|^2)$.*

Proof. As Algorithm 7.3 starts with r in the node pool and adds exactly one node into the pool in each iteration, thus it contains at most $|V| - 1$ iterations. The complexity of Algorithm 7.3, similar to Prim's algorithm, is at worst $\mathcal{O}(|V|^2)$.

Moreover, since at each iteration, one adds an edge e to E^* such that e has exactly one end in the node pool P and both ends of e are not saturated in terms of capacity. Hence $E^* \cup \{e\}$ always induces a bounded r -tree, and thus \mathbf{x}^{E^*} is an integral feasible solution for the MBrT problem at the end of the algorithm. \square

Note that this matheuristic can also be used for any fractional solution $\left[\frac{\bar{\mathbf{x}}}{\bar{\mathbf{y}}}\right]$ obtained in BRT_{xy} by simply considering its restriction $\bar{\mathbf{x}}$. Moreover, at the last step of the candidate selection, there are several ways to modify the algorithm such as comparing other properties or changing the priority of each property.

In addition, Algorithm 7.3 only targets the uniform-weight version of the MBrT problem, that is, the MSBrT problem. When applying this matheuristic on the general MBrT problem, it should also consider the edge-weights while choosing edges in order to obtain a higher quality solution.

Other than this primal matheuristic, we also experimented with another matheuristic which was unfortunately not very effective. The idea of that matheuristic is essentially to first derive a spanning tree of the support graph of a fractional solution, and then prune the branches that violate the capacity constraints. This approach generally gives a bad primal bound because there are too much pruning of the resulting spanning tree, as it does not consider the capacity factor when constructing the spanning tree.

7.4 Properties related to optimal solutions

When dealing with uniform-weight cases, the following property has been found of the optimal solutions that one can exploit.

Proposition 7.4.1. *Given a bounded r -tree polytope $B_x(G, r, \mathbf{c})$ with $G = (V, E)$, and uniform weight vector $\mathbf{w} = \mathbf{1}^E$, let $c'_r = \min\{c_r, |\delta(r)|\}$. The following equation holds.*

$$\max\{\mathbf{w}^T \mathbf{x} : \mathbf{x} \in B_x(G, r, \mathbf{c})\} = \max\{\mathbf{w}^T \mathbf{x} : \mathbf{x} \in B_x(G, r, \mathbf{c}), x(\delta(r)) = c'_r\}.$$

Proof. Assume that all optimal solutions satisfy $x(\delta(r)) < c'_r$, and assume $\bar{\mathbf{x}}$ is one of the optimal solutions with its support graph being $G' = (V', E')$.

First, if there exists some node $v \in (V \cap N(r)) \setminus (V' \cap N(r))$, one can simply add rv to G' to obtain a better solution, and thus it forms a contradiction. Then, one has $(V \cap N(r)) \setminus (V' \cap N(r)) = \emptyset$. Knowing that, pick an arbitrary node $u \in V \cap N(r)$, one can add ru into G' , which creates one and only one cycle since G' is a tree, and the cycle contains ru . By removing the other edge in the cycle incident with u , one can obtain another r -tree G'' that satisfies the capacity constraints, as from G' to G'' , the only node with its degree increasing is r . This process can be repeated as many times until the c'_r is saturated, and therefore, one can construct a solution which satisfies $x(\delta(r)) = c'_r$. \square

This proposition allows us to reduce the problem from optimizing the objective function over the polytope $B_x(G, r, \mathbf{c})$ to optimizing the same objective function over merely a face of the polytope. In other words, the dimension of the polytope we are working on is reduced by 1. Therefore, the following equation can be added into the initial model when uniform weight is considered.

$$x(\delta(r)) = c'_r \tag{7.20}$$

This result also applies for $B_{xy}(G, r, \mathbf{c})$ when node weight is not considered.

$$\max\{\mathbf{w}^T \mathbf{x} : [\mathbf{x}] \in B_{xy}(G, r, \mathbf{c})\} = \max\{\mathbf{w}^T \mathbf{x} : [\mathbf{x}] \in B_{xy}(G, r, \mathbf{c}), x(\delta(r)) = c'_r\}.$$

In fact, it is deduced from the following two implicit equations.

$$\begin{aligned} y_r &= 1, \\ x(\delta(r)) &= c'_r y_r. \end{aligned}$$

Combining them gives us the same equation as for $B_x(G, r, \mathbf{c})$. It is also worth mentioning that $y_r = 1$ applies for any case where the root r has a non-negative weight.

7.5 Results

Note that there are plenty of options while implementing BRT_x , as we have a wide range of user cuts to select from and various options of how the branch-and-cut algorithm proceeds. For the sake of conciseness and rigorous, we only show one set of results for BRT_x , which has the most complete results and overall the best performance so far. In this implementation, we incorporated user cuts for (7.7), (7.9) and (7.11), but not for (7.8) and (7.10). Besides, the order of each set of cuts that is generated in the branch-and-cut algorithm also affects the performance. Particularly, for BRT_x we split the user cuts into two groups. We first generate cuts for the first group of inequalities, that is, the connectivity inequalities (7.5), a subset of subtour elimination inequalities (7.6), the matching-partition inequalities (7.7) and the tightening inequalities (7.11). If no cuts is found for the first group, we then generate cuts for the subtour elimination inequalities (7.6) and upload capacity inequalities (7.9).

Another factor that has influence on the performance is the timing of when and how often the primal matheuristic should be running. After a few experiments, we chose to run the matheuristic with a dynamic step. More precisely, the primal matheuristic runs every s_d iterations, where each time it is executed, we adjust the step s_d depending on how good is the solution found by the matheuristic comparing with the previous best incumbent.

Besides, we kept the CPLEX branching and node selection rules, as well as CPLEX default cuts (Gomory fractional cuts, zero-half cuts, etc.) with default setting.

We present our results on over 600 instances for each framework. Providing all the results would be overwhelming for the readers, thus for each testset we first give an overview of the performance of all four different frameworks, and then present the results of a few representative instances. For the overview, we give the overall percentage of the solved instances within a time limit of 2 hours, and compare the average upper bounds and lower bounds attained by each branch-and-cut framework for each testset.

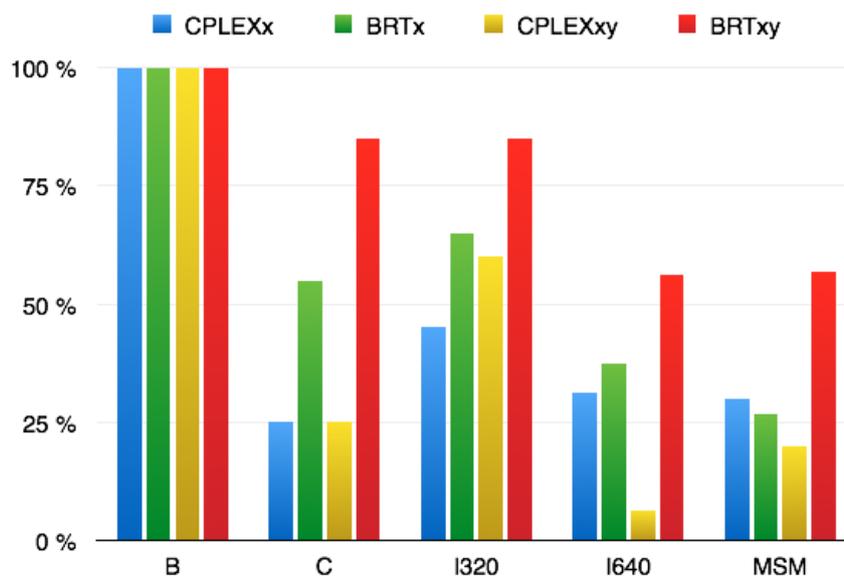


Figure 7.1: Percentage of solved instances on SteinLib testsets

The following notation is used in the forthcoming tables presenting the results.

UB : the best achievable upper bound.

LB : the best lower bound.

Gap : gap in percentage between the best lower bound and the best upper bound, which is equal to $\frac{UB-LB}{LB}$.

7.5.1 SteinLib instances

SteinLib is one of the two data sources we have for the computational test. We use the graphs of the dataset and generate a capacity files to match each of the instances. Since we only test the uniform-weight version of the MBrT problem so far, the weights in the SteinLib files are not used. Moreover, since the MBrT problem and the Steiner tree problem are quite different due to the capacity constraints in the former and the terminals in the latter, there is little to none correlation between the difficulties of the two instances that correspond to the same file. Thus, while choosing instances from the SteinLib, our intention is to make sure that the selected instances cover a wide spectrum of different properties such as node number, graph type etc.

First of all, Figure 7.1 reports the percentage of solved instances on each set of SteinLib instances. All the instances of the testset B are solved by all four frameworks within 2 hours. For testsets C and I320 BRT_x showed clear improvements comparing with $CPLEX_x$ regarding the number of times reaching optimality, whereas for testsets I640

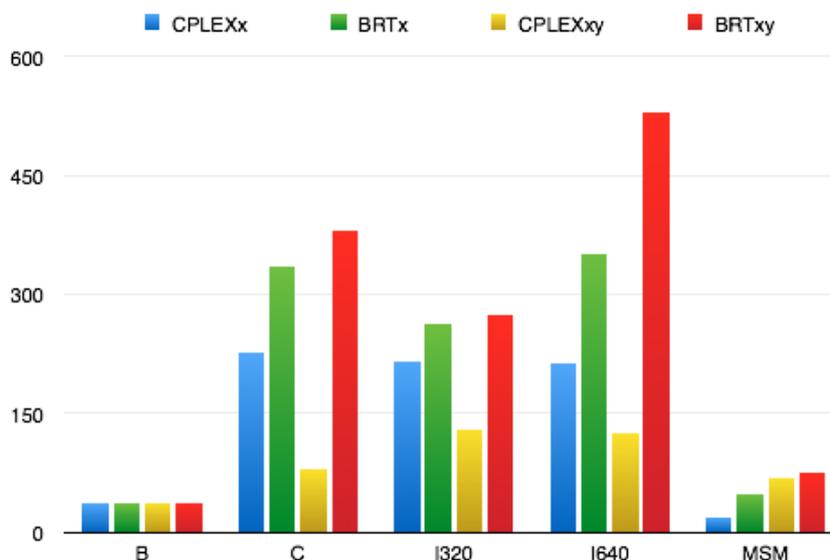


Figure 7.2: Lower bounds on SteinLib instances

and MSM their performance are close. On the other hand, BRT_{xy} always solves much more instances than $CPLEX_{xy}$ in all four other testsets than B.

Figure 7.2 and 7.3 demonstrates the differences of $CPLEX_x$, BRT_x , $CPLEX_{xy}$ and BRT_{xy} in terms of average upper bound and lower bound for the SteinLib instances. We can see that BRT_{xy} obtained the best upper bounds and lower bounds across the board, while the other three are close regarding the upper bound. However, $CPLEX_x$ and $CPLEX_{xy}$, especially $CPLEX_{xy}$, generally struggle to find a good lower bound.

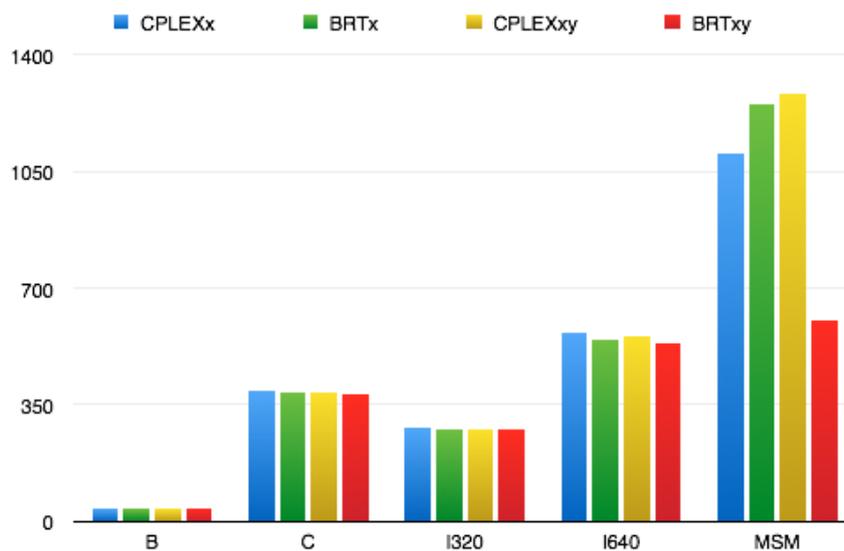


Figure 7.3: Upper bounds on SteinLib instances

Although all the 18 instances in testset B have been solved by all four frameworks within 2 hours, it generally takes $CPLEX_x$ the longest time to solve the instances. Some of the

			$CPLEX_x$				BRT_x			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
B/b01	50	63	27	27	0.00%	0.17	27	27	0.00%	0.08
B/b10	75	150	60	60	0.00%	<u>20.06</u>	60	60	0.00%	1.18
B/b11	75	150	51	51	0.00%	<u>588.01</u>	51	51	0.00%	8.84
B/b18	100	200	79	79	0.00%	<u>40.66</u>	79	79	0.00%	5.51
			$CPLEX_{xy}$				BRT_{xy}			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
B/b01	50	63	27	27	0.00%	0.02	27	27	0.00%	0.01
B/b10	75	150	60	60	0.00%	0.60	60	60	0.00%	0.06
B/b11	75	150	51	51	0.00%	0.01	51	51	0.00%	0.06
B/b18	100	200	79	79	0.00%	0.36	79	79	0.00%	0.06

Table 7.4: Test results on instances of testset B

			$CPLEX_x$				BRT_x			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
C/c04	500	625	170	188	10.59%	7200.69	172	172	0.00%	5005.52
C/c15	500	2500	<u>0</u>	488.50	-	7200.04	478	478	0.00%	4.66
C/c16	500	12500	<u>0</u>	494.50	-	7203.17	490	490	0.00%	12.72
C/c17	500	12500	499	499	0.00%	25.02	497	499	0.40%	7200.80
C/c20	500	12500	499	499	0.00%	97.38	499	499	0.00%	1.71
			$CPLEX_{xy}$				BRT_{xy}			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
C/c04	500	625	172	172	0.00%	1.83	167	178	6.59%	7200.26
C/c15	500	2500	10	478	4680%	7201.56	478	478	0.00%	20.58
C/c16	500	12500	2	490	24400%	7200.54	490	490	0.00%	140.33
C/c17	500	12500	56	499	<u>791.07%</u>	7204.38	499	499	0.00%	6.11
C/c20	500	12500	24	499	<u>1979.17%</u>	7201.85	499	499	0.00%	37.10

Table 7.5: Test results on instances of testset C

results are shown in Table 7.4. It can be seen that for instances b10, b11, b18, $CPLEX_x$ needs up to 10 minutes to solve them to optimality while the other three need less than 10 seconds.

In Table 7.4 and the tables after, we use underlined numbers to highlight the notable worst result among the four frameworks and bold-face numbers to highlight the best result.

Table 7.5 shows that $CPLEX_x$ and $CPLEX_{xy}$ can only obtain a bad lower bound on some instances comparing with their counterparts, while BRT_x and BRT_{xy} performed much better overall.

This trend continues for the testsets I320, I640 and MSM, where BRT_{xy} outperformed

			<i>CPLEX_x</i>				<i>BRT_x</i>			
Instance	V	E	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
i320-001	320	480	155	173.83	12.15%	7200.01	160	163.00	1.88%	7200.76
i320-121	320	51040	<u>0</u>	313.50	-	7200.18	308	308	0.00%	98.45
i320-201	320	480	164	179.17	9.25%	7200.44	170	170	0.00%	816.92
i320-331	320	640	265	272	2.64%	7200.02	251	271.33	8.10%	7200.04
i640-141	640	40896	639	639	0.00%	779.97	639	639	0.00%	920.94
i640-211	640	4135	<u>0</u>	629.50	-	7200.32	620	620	0.00%	25.78
i640-241	640	40896	0	625.50	-	7200.23	2	612	30500%	7209.99
i640-341	640	40896	639	639	0.00%	167.74	639	639	0.00%	138.05
msm1707	278	478	42	42	0.00%	220.60	42	42	0.00%	1.38
msm2000	898	1562	0	575.05	-	7200.10	75	755.96	907.95%	7200.29
msm4114	402	690	<u>0</u>	298.50	-	7201.04	2	2	0.00%	6030.40
msm4515	777	1358	<u>0</u>	604.50	-	7200.65	9	661.00	7244.44%	7200.49
			<i>CPLEX_{xy}</i>				<i>BRT_{xy}</i>			
Instance	V	E	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
i320-001	320	480	160	160	0.00%	3.20	160	160	1.88%	62.01
i320-121	320	51040	93	308	231.18 %	7200.48	308	308	0.00%	1.82
i320-201	320	480	170	170	0.00%	63.98	170	170	0.00%	2.95
i320-331	320	640	237	273.85	15.55%	7200.02	267	267	0.00%	13.62
i640-141	640	40896	96	639	<u>565.63%</u>	7200.32	639	639	0.00%	329.95
i640-211	640	4135	229	620	170.74 %	7200.09	620	620	0.00%	39.79
i640-241	640	40896	0	625.50	-	7200.23	612	612	0.00%	1819.88
i640-341	640	40896	7	639	<u>9028.57%</u>	7201.42	639	639	0.00%	2352.22
msm1707	278	478	42	42	0.00%	299.10	42	42	0.00%	0.20
msm2000	898	1562	75	706.02	841.36%	7200.00	174	174	0.00%	245.02
msm4114	402	690	2	316	15700%	7200.08	2	2	0.00%	2.88
msm4515	777	1358	41	<u>686.67</u>	1574.8%	7200.01	41	41	0.00%	56.86

Table 7.6: Test results on instances of testsets I320, I640 and MSM

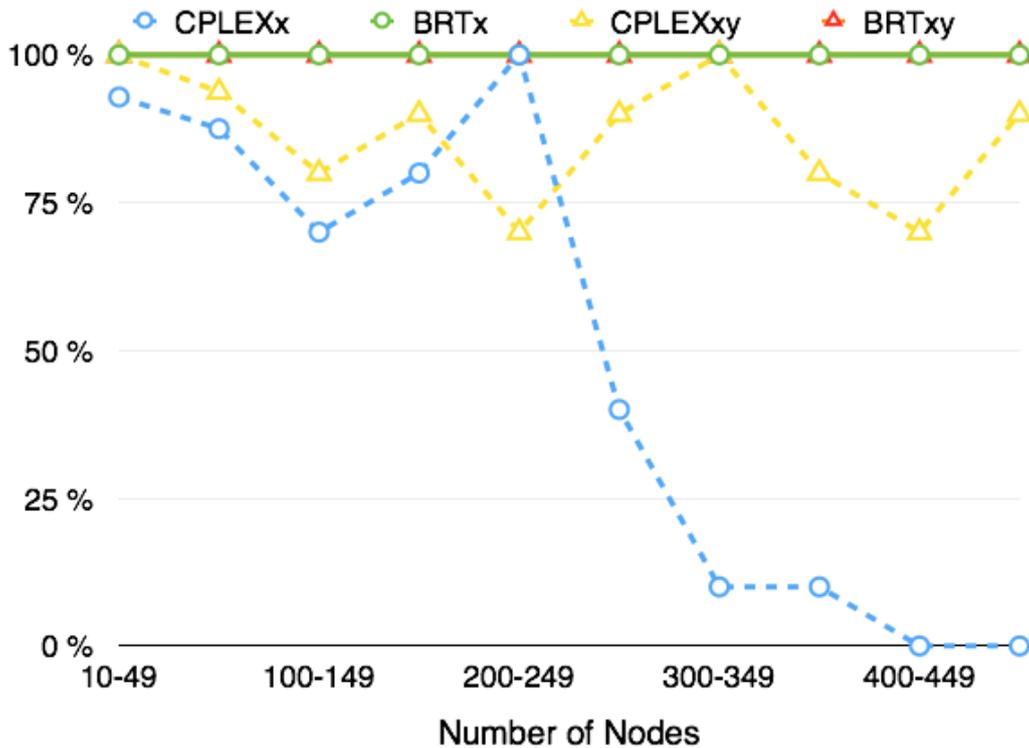


Figure 7.4: Percentage of solved instances on random dense graphs

all the other three frameworks by a considerable margin, and $CPLEX_x$ and $CPLEX_{xy}$ could not find a good lower bound on some instances, as it is shown in Table 7.6. Notably, BRT_x actually provided some worse upper bound than $CPLEX_x$ sometimes. The reason of that could be that we did not adjust the separation algorithms specifically for each testset. Instead, we use the same separation algorithms with the same setup for all the instances. Thus in the case of testset MSM, as the instances are extremely sparse, many of our separation algorithms are non-necessarily costly when going through each node in the graph, which leads to much longer iteration time and relatively bad bounds in the end.

7.5.2 Random generated instances

The first set of random generated instances contains dense graphs with node capacities between 1 to 4. We have only tested on instances with less than 500 nodes, as it is already sufficient to demonstrate the big discrepancy between the results given by $CPLEX_x$, $CPLEX_{xy}$ and those of BRT_x and BRT_{xy} .

Figure 7.4 demonstrates the difference between these two groups in terms of percentage of solved instances. It can be seen that BRT_x and BRT_{xy} were able to solve all the instances, while $CPLEX_x$ and $CPLEX_{xy}$ had difficulties on some of them. Notably, the

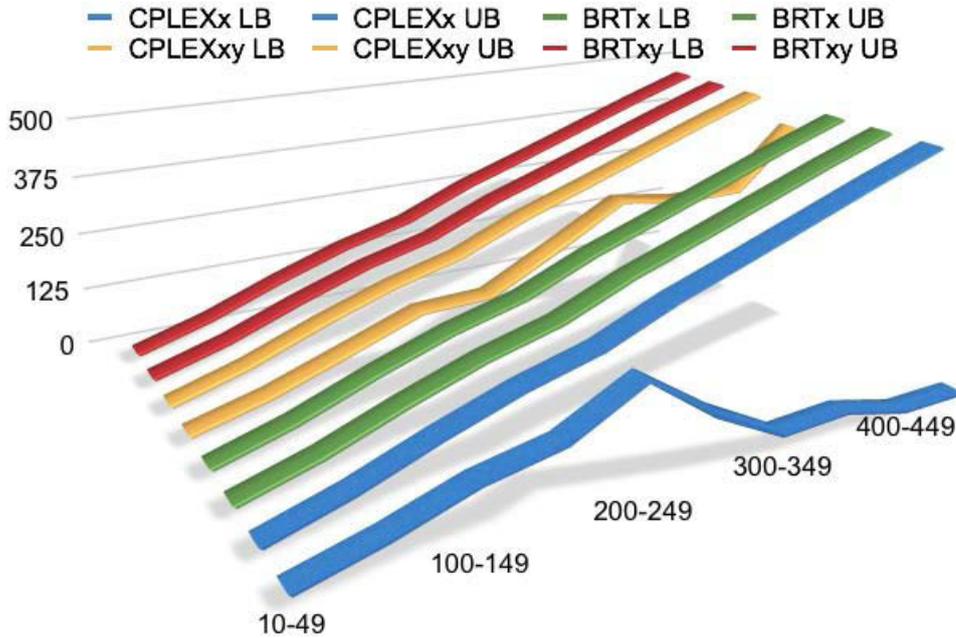


Figure 7.5: Bounds on random dense graphs

number instances which $CPLEX_x$ is able to solve decreases dramatically as the size of the graph grows.

Consequently, the bounds achieved by $CPLEX_x$ and $CPLEX_{xy}$ are much worse than those of BRT_x and BRT_{xy} , as it is shown in Figure 7.5 where the horizontal axis represents the number of nodes.

Table 7.7 provides a glimpse to the results obtained for the dense graphs. As we can see, all 6 instances listed in Table 7.7 have been solved by BRT_x and BRT_{xy} within 1 minute, whereas $CPLEX_x$ and $CPLEX_{xy}$ failed to find a good lower bound on some of them.

Moving onto the instances with sparse graphs, we first test on the instances with node capacity chosen from 1 to 3 (that is, the testset `rg_13`). As it is shown in Figure 7.6, BRT_{xy} has solved the most instances, whereas BRT_x still constantly performed better than $CPLEX_x$. However, the performance of BRT_x on this testset is close to $CPLEX_{xy}$ and is outperformed by BRT_{xy} in a convincing fashion.

Figure 7.7 shows the averages bounds obtained by the four different frameworks. It can be seen that, as opposed to the case on dense graphs, $CPLEX_{xy}$ struggles much harder to find a good lower bound than $CPLEX_x$.

Table 7.8 reports some of the results on testset `rg_13`. We can clearly see that BRT_{xy} had the best overall performance, which is slightly better than BRT_x , and much better than both $CPLEX_x$ and $CPLEX_{xy}$. Despite $CPLEX_x$ failed to solve some instances, the gaps at the end of the execution are much smaller than those of $CPLEX_{xy}$, as a

			$CPLEX_x$				BRT_x			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
39_7	39	130	29	30.50	5.17%	7200.01	29	29	0.00%	0.02
97_5	97	4196	96	96	0.00%	1.02	96	96	0.00%	0.69
139_6	139	9118	0	138	-	7200.09	138	138	0.00%	2.77
222_0	222	6356	221	221	0.00%	3983.44	221	221	0.00%	0.85
370_7	370	61606	0	369	-	7214.24	369	369	0.00%	6.89
477_7	477	65937	0	476	-	7209.92	476	476	0.00%	39.15
			$CPLEX_{xy}$				BRT_{xy}			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
39_7	39	130	29	29	0.00%	0.60	29	29	0.00%	0.01
97_5	97	4196	6	96	1500%	7200.08	96	96	0.00%	0.52
139_6	139	9118	138	138	0.00%	1.01	138	138	0.00%	0.68
222_0	222	6356	0	221	-	7200.10	221	221	0.00%	0.44
370_7	370	61606	0	369	-	7205.22	369	369	0.00%	3.48
477_7	477	65937	0	476	-	7200.97	476	476	0.00%	2.06

Table 7.7: Test results on instances on dense graphs

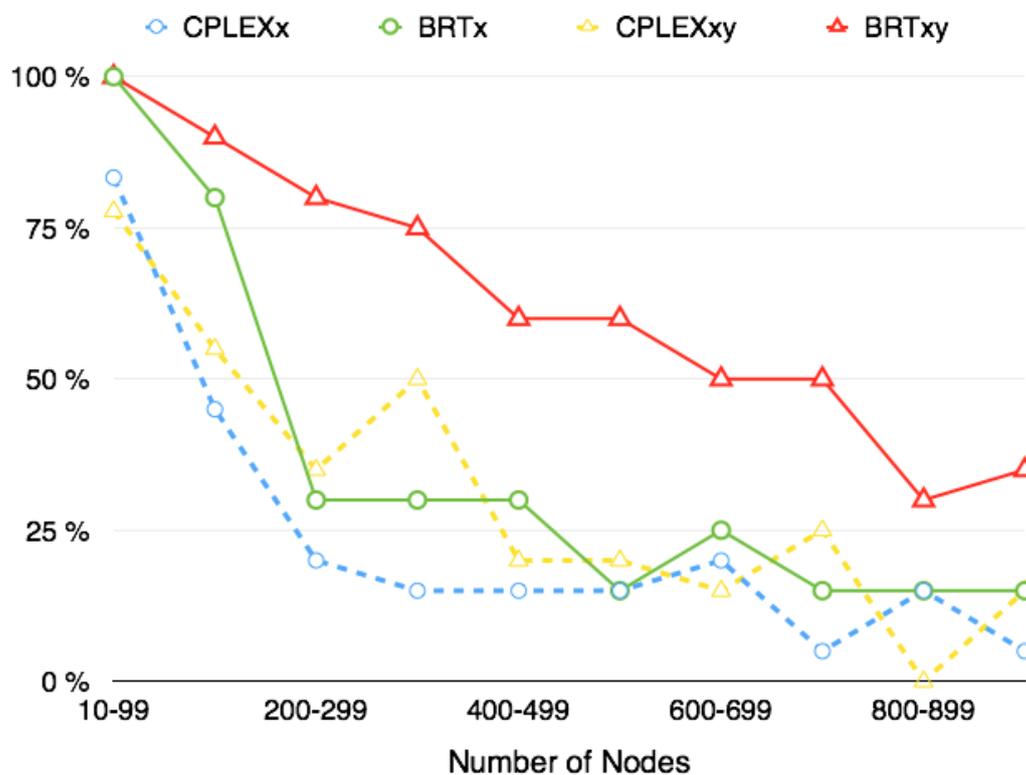


Figure 7.6: Percentage of solved instances on random sparse graphs

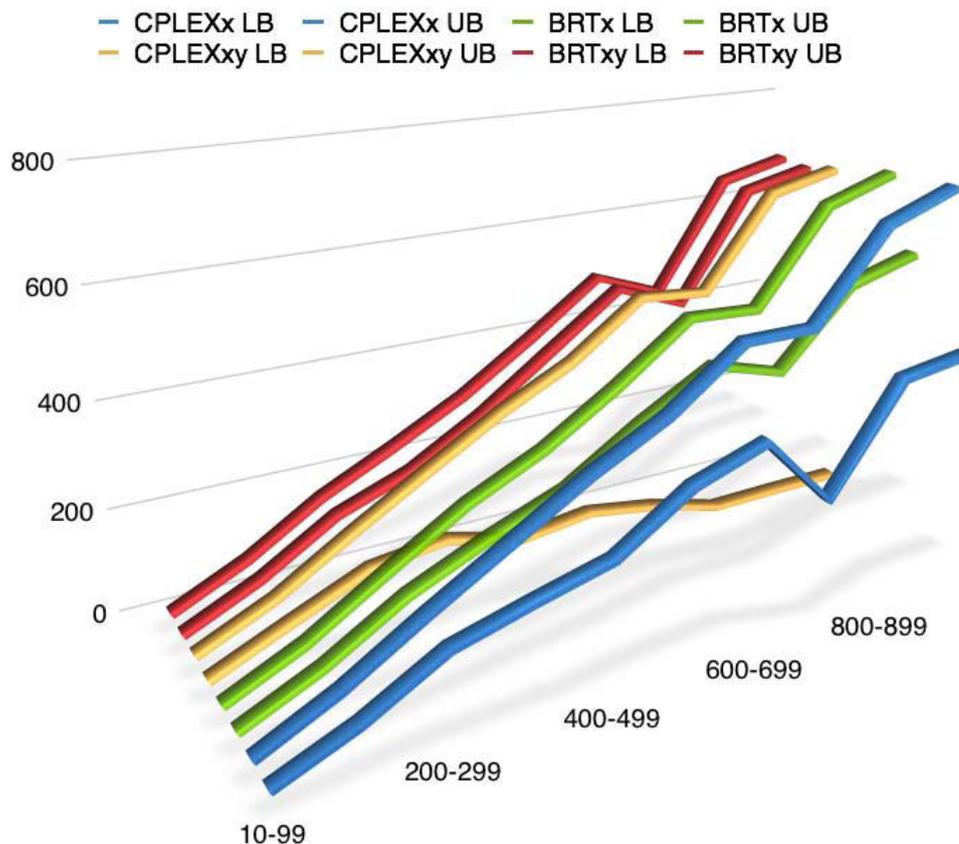


Figure 7.7: Bounds on random sparse graphs

			$CPLEX_x$				BRT_x			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
rg_13/69	69	108	1	1	0.00%	365.69	1	1	0.00%	0.39
rg_13/154	154	200	104	104	0.00%	56.71	104	104	0.00%	0.77
rg_13/348	348	970	342	347	1.46%	7200.49	343	347	1.17%	7200.43
rg_13/560	560	1088	535	535	0.00%	59.90	535	535	0.00%	81.41
rg_13/739	739	2081	730	738	1.10%	7200.30	734	738.00	0.54%	7200.01
rg_13/981	981	2408	934	935	0.11%	7201.20	935	935	0.00%	112.09
			$CPLEX_{xy}$				BRT_{xy}			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
rg_13/69	69	108	1	43	<u>4200%</u>	<u>7200.02</u>	1	1	0.00%	0.07
rg_13/154	154	200	104	104	0.00%	<u>1092.06</u>	104	104	0.00%	0.01
rg_13/348	348	970	332	345	3.92%	7200.02	343	343	0.00%	5.46
rg_13/560	560	1088	60	537	<u>795%</u>	7201.89	534	536	0.37%	7202.00
rg_13/739	739	2081	51	738	<u>1347.06%</u>	7201.45	735	735	0.00%	23.74
rg_13/981	981	2408	15	935	<u>6133.33%</u>	7200.54	935	935	0.00%	394.51

Table 7.8: Test results on instances on sparse graphs

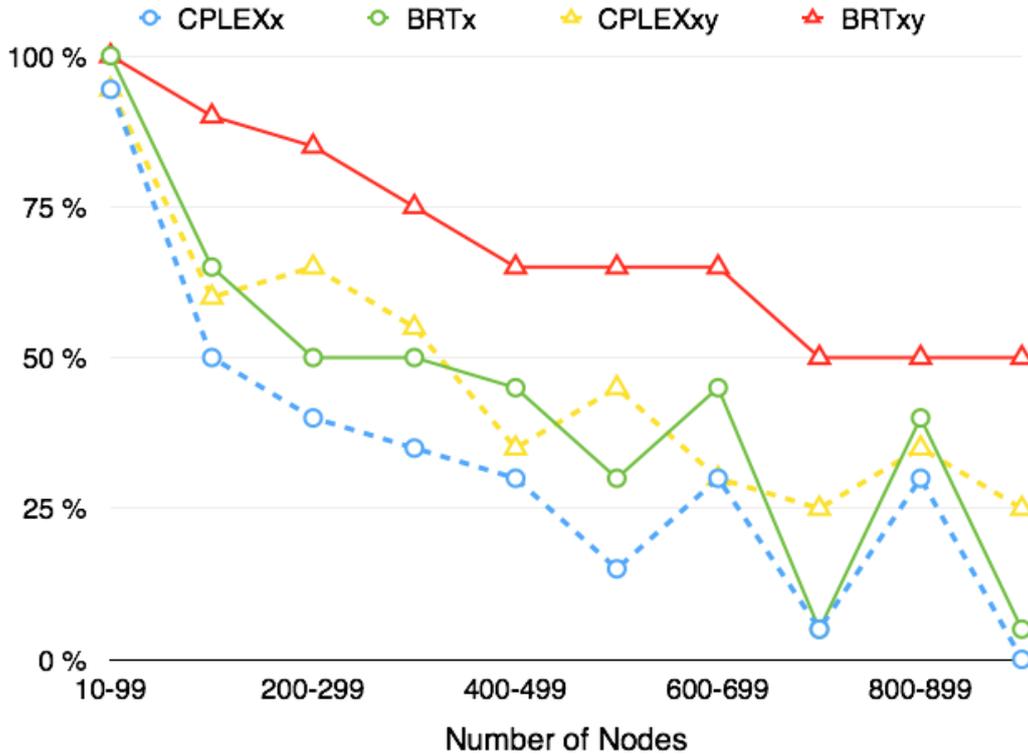


Figure 7.8: Percentage of solved instances on random sparse graphs with $O = \emptyset$

result of the bad lower bounds attained by $CPLEX_{xy}$.

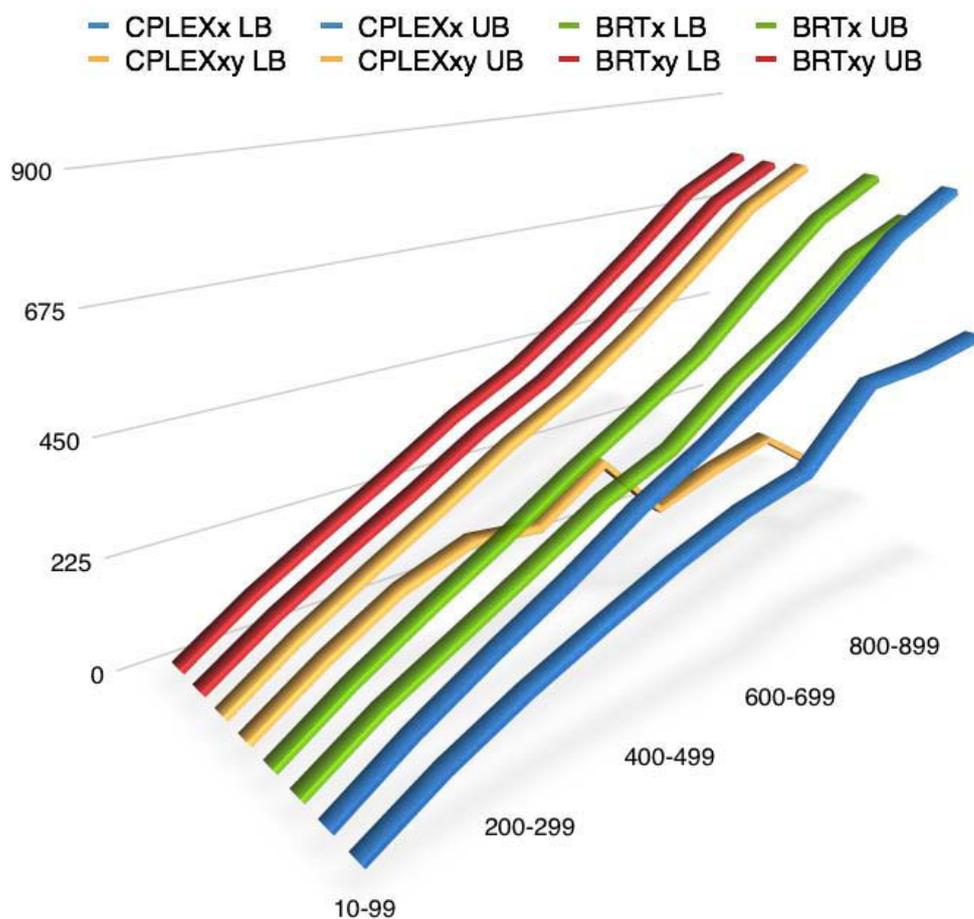
One of the two aspects that contributes to the difficulty of the $MBrT$ problem is the node capacity, and out of all the nodes, those in O seem to create most of the difficulties in our theoretical work including the study of algorithms and polyhedral structure. Thus, we created the testset `rg_23`, where all nodes in the graphs have capacity between 2 and 3 (that is, $O = \emptyset$) to see if there is any difference.

However, the results we obtained on testset `rg_23` actually present in a similar pattern to those on testset `rg_13`. As it is demonstrated in Figure 7.8, BRT_{xy} has solved the largest amount of instances, while BRT_x and $CPLEX_{xy}$ had similar performances and both outperformed $CPLEX_x$ in terms of total number of instances solved.

The average upper bounds are relatively close for all four frameworks. Nonetheless, $CPLEX_{xy}$ had the worst average lower bounds, whereas $CPLEX_x$ was significantly better than $CPLEX_{xy}$ but slightly worse than BRT_x , with BRT_{xy} consistently being the best.

Table 7.9 shows some examples of the results obtained on `rg_23`, which support our previous inferences.

Out of the over 600 instances we have tested, BRT_{xy} has solved 73.61% of them, and BRT_x 52.46%, whereas $CPLEX_{xy}$ has solved 47.70% of them, and $CPLEX_x$ 35.08%.

Figure 7.9: Bounds on random sparse graphs with $O = \emptyset$

			$CPLEX_x$				BRT_x			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
rg_23/54	54	108	52	53	1.92%	7200.08	52	52	0.00%	0.10
rg_23/122	122	292	118	121	2.54%	7200.15	118	118	0.00%	80.44
rg_23/254	254	569	249	253	1.61%	7200.09	249	253	1.61%	7200.16
rg_23/436	436	607	378	378	0.00%	292.65	378	378	0.00%	812.77
rg_23/695	695	1257	0	519	-	7201.88	509	519.00	1.96%	7202.87
rg_23/941	941	2480	912	940	3.07%	7200.59	929	940	1.18%	7200.45
			$CPLEX_{xy}$				BRT_{xy}			
Instance	$ V $	$ E $	LB	UB	Gap	Time(s)	LB	UB	Gap	Time(s)
rg_23/54	54	108	52	53	1.92%	7200.32	52	52	0.00%	0.01
rg_23/122	122	292	118	120.86	2.42%	7200.75	118	119	0.85%	7200.01
rg_23/254	254	569	249	251	0.80%	7201.14	249	249	0.00%	1.13
rg_23/436	436	607	85	385	<u>352.94%</u>	7201.32	378	378	0.00%	6.70
rg_23/695	695	1257	2	519	25850%	7200.04	519	519	0.00%	2707.97
rg_23/941	941	2480	7	940	<u>13328.57%</u>	7201.09	939	939	0.00%	2137.44

Table 7.9: Test results on instances on sparse graphs with $O = \emptyset$

This shows, to a great extent, that the overall level of strength of each framework, and how much improvement has been made by adding the newly introduced constraints and the primal matheuristic to the original formulations.

7.6 Conclusion

This chapter has described the method used to perform the computational test of different models for the MBrT problem, in order to show how can our theoretical results make a difference in the computational aspect.

We first discussed the separation problem for each set of inequalities that has been introduced earlier for both $B_x(G, r, \mathbf{c})$ and $B_{xy}(G, r, \mathbf{c})$. We showed that most of them can be separated in polynomial time and provided respective separation algorithm. We proved that the separation problem of the matching-partition inequalities is \mathcal{NP} -hard, and provided a few heuristics to generate cut planes.

With the help of CPLEX we implemented four different branch-and-cut algorithms, namely $CPLEX_x$, $CPLEX_{xy}$, BRT_x and BRT_{xy} . Particularly, $CPLEX_x$ and $CPLEX_{xy}$ are minimal models for $B_x(G, r, \mathbf{c})$ and $B_{xy}(G, r, \mathbf{c})$, whereas BRT_x and BRT_{xy} incorporate the newly discovered constraints, a preprocess and a primal matheuristic. We then tested them on a variety of instances with different properties, which were either fetched from SteinLib or generated by ourselves. Their performance are compared in the several different aspects, including overall percentage of solved instances, execution time, and upper bound and lower bound.

The results showed that in all aspects BRT_x and BRT_{xy} beat their counterparts $CPLEX_x$ and $CPLEX_{xy}$ significantly. It showed convincingly the strength of the enhanced formulations in the computational aspect. Moreover, while BRT_x and BRT_{xy} performed equally good on dense graphs, BRT_{xy} outperformed BRT_x on sparse graphs and most testsets of SteinLib instances. The performances of BRT_x and $CPLEX_{xy}$ are overall similar, except that on dense graphs BRT_x outperformed $CPLEX_{xy}$ quite significantly. Additionally, $CPLEX_x$ showed its weakness in finding good lower bounds in dense graphs, whereas $CPLEX_{xy}$ generally struggled the most to find good lower bounds in sparse graphs.

Conclusion

In this dissertation, we explored several aspects of two \mathcal{NP} -hard combinatorial optimization problems, the MBrT problem and the MrT problem. We first defined their associated polytopes, namely $B_{xy}(G, r, \mathbf{c})$, $B_x(G, r, \mathbf{c})$, $R_{xy}(G, r)$ and $R_x(G, r)$. Among these two polytopes, $R_{xy}(G, r)$ has been studied by Goemans [1994], whereas the other three have not drawn much attention from the literature. We gave a formulation for each one of the four polytopes. A literature review was given on the MBrT problem and the MrT problem as well as other related problems. We then introduced the background and the application of the MBrT problem in the telecommunication field, along with some of the recent related research topics. We showed that the study of the MBrT problem and its applications deserve more attention.

We approached the MBrT problem and the MrT problem from three directions, algorithms, polyhedra and computational test.

Firstly, we proposed several polynomial-time combinatorial algorithms for the MBrT problem on trees, cycles and cactus graphs respectively. A dynamic programming based approach is used on trees which computes the solution from the leaves to the root recursively. On cycles, we split the problem into 3 different cases, and solved them by calculating a min/max subpath of a given path. For cactus graphs, we reduced the problem to some subproblems on the so-called cactus basis and on trees, and then proved that the MBrT problem on cactus basis can be solved in polynomial time. It is worth noting that the proposed algorithms can also be applied on the MrT problem, as it is a relaxation of the MBrT problem.

After that, we presented results concerning three polytopes, $B_{xy}(G, r, \mathbf{c})$, $R_x(G, r)$ and $B_x(G, r, \mathbf{c})$. We showed that the dimension of $B_{xy}(G, r, \mathbf{c})$ is related to the number of nodes in O , which are the none-root nodes with unit-capacity, and the number of blocks in a subgraph $G_r = G \setminus O$ of G . On the other hand, $R_x(G, r)$ and $B_x(G, r, \mathbf{c})$ is proved to be full-dimensional under Assumption 2.2.9. We also showed that $B_{xy}(G, r, \mathbf{c})$ can be decomposed through 1-sum. However, such decomposition is proved to be infeasible for $R_x(G, r)$ and $B_x(G, r, \mathbf{c})$ except when the 1-sum is at the root node r . For $B_{xy}(G, r, \mathbf{c})$, we proposed several sets of new valid equations and inequalities to strengthen the formulation

according to the dimension results. For $R_x(G, r)$, we proposed two sets of new valid inequalities that can also be inherited by $B_{xy}(G, r, \mathbf{c})$ with some adjustments. Besides these two sets of valid inequalities, we presented several other sets of new valid inequalities specifically for $B_x(G, r, \mathbf{c})$. For all the proposed inequalities we discussed the necessary and sufficient conditions to be facet-defining for the respective polytope. Besides, it is also demonstrated that the newly introduced valid inequalities for $B_x(G, r, \mathbf{c})$ can all be obtained by projection of the valid constraints for $B_{xy}(G, r, \mathbf{c})$.

We then showed that with the newly introduced constraints, $B_{xy}(G, r, \mathbf{c})$ can be characterized on trees and cycles. Moreover, as a consequence of the polyhedral decomposition through 1-sum, $B_{xy}(G, r, \mathbf{c})$ can therefore be characterized on cactus graphs. As for $B_x(G, r, \mathbf{c})$, we proved that with incorporating the newly introduced matching-partition inequalities and upload capacity inequalities, one can obtain a TDI system that defines $B_x(G, r, \mathbf{c})$ on trees and each of the four cases on cycles.

For each set of inequalities introduced in the dissertation, its separation problem has been discussed. We proposed either polynomial-time separation algorithms or some heuristics if the separation problem is \mathcal{NP} -hard. With the help of CPLEX, we implemented four different branch-and-cut frameworks, namely $CPLEX_x$, $CPLEX_{xy}$, BRT_x and BRT_{xy} , based on the formulations for $B_{xy}(G, r, \mathbf{c})$ and $B_x(G, r, \mathbf{c})$. Among the four frameworks, $CPLEX_x$ and $CPLEX_{xy}$ correspond to the original formulations for $B_x(G, r, \mathbf{c})$ and $B_{xy}(G, r, \mathbf{c})$ without including any newly introduced constraints, whereas BRT_x and BRT_{xy} correspond to the enhanced formulations incorporating the newly introduced constraints as well as a preprocess and a primal matheuristic. They were tested on a variety of over 600 instances that are either from SteinLib Testdata Library or generated.

Test results showed that the performance of BRT_x and BRT_{xy} are generally better than $CPLEX_x$ and $CPLEX_{xy}$. Particularly, on SteinLib instances and sparse graphs, BRT_{xy} had the best performance with a considerable difference comparing to the other three, whereas BRT_x outperformed $CPLEX_x$. In addition, BRT_x and $CPLEX_{xy}$ performed similarly, except $CPLEX_{xy}$ struggled the most at finding a good lower bound on instances on these instances. On dense graphs, BRT_x is on par with BRT_{xy} in performance, and they are both significantly better than $CPLEX_x$ and $CPLEX_{xy}$. $CPLEX_x$ provided the worst performance on dense graphs, and its performance declines dramatically as the number of nodes in graph increases. To summarize, the computational test has shown in a convincing fashion the strength of the enhanced formulations, especially the enhanced formulation for $B_{xy}(G, r, \mathbf{c})$.

Certainly, there remain many unexplored aspects and potential extended research subjects of the MBrT problem. Hereafter we list a few directions that we reckon to be valuable and worth studying in the future.

Ins	Time(s)	t_r (s)	$N_{B\&C}$	UB	LB	Ins	Time(s)	t_r (s)	$N_{B\&C}$	UB	LB
13 ₁	0.02	0.02	0	9	9	97 ₂	0.18	0.18	0	71	71
13 ₅	0.02	0.02	0	10	10	97 ₅	4550.08	2096.17	237	83	83
17 ₆	0.04	0.04	0	12	12	98 ₃	1059.52	327.33	270	96	96
17 ₇	0.02	0.02	0	14	14	99 ₁	7200.6	7200.6	0	79.50	0
19 ₀	0.05	0.05	0	16	16	99 ₈	50.97	30.82	80	83	83
19 ₂	0.04	0.04	0	16	16	100 ₂	7200.71	7200.71	0	135.50	0
22 ₅	0.03	0.03	0	15	15	106 ₁	510.77	510.77	0	148	148
24 ₃	0.07	0.07	0	15	15	110 ₉	7200.03	1155.24	164	130.64	0
24 ₉	0.04	0.04	0	20	20	114 ₆	5788.61	5788.61	0	163	163
29 ₈	0.03	0.03	0	22	22	122 ₇	35.13	32.96	14	121	121
37 ₈	1.29	1.29	0	31	31	128 ₀	7203.56	7203.56	0	170.50	0
39 ₇	0.20	0.20	0	29	29	133 ₀	194.72	194.72	0	185	185
45 ₄	3.00	0.15	183	29	29	139 ₆	7214.45	7214.45	0	201.00	0
47 ₁	1.37	1.37	0	45	45	144 ₄	7200.38	7200.38	0	179.09	0
57 ₀	0.12	0.12	0	39	39	149 ₅	7202.4	7202.4	0	203.50	0
58 ₈	0.10	0.10	0	44	44	151 ₄	7204	7204	0	212.50	0
63 ₈	0.56	0.56	0	38	38	155 ₇	7200.57	7200.57	0	220.50	0
68 ₅	0.25	0.25	0	46	46	164 ₅	7200.11	7200.11	0	233.00	0
76 ₅	2.09	2.09	0	60	60	167 ₁	7201.56	7201.56	0	229.50	0
78 ₈	18.75	18.75	0	75	75	176 ₂	7200.44	7200.44	0	247.50	0
83 ₂	63.12	63.12	0	83	83	179 ₆	7200.99	7200.99	0	236.00	0
86 ₂	0.85	0.30	13	36	36	183 ₃	7207.66	7207.66	0	255.50	0
94 ₀	1125.92	14.27	180	88	88	189 ₄	7202.57	7202.57	0	271.50	0
95 ₃	0.06	0.06	0	43	43	190 ₅	7208.02	7208.02	0	257.50	0
95 ₈	2.15	1.31	13	51	51	198 ₃	7201.6	7201.6	0	277.50	0

Table 7.10: Test results with $BRT P_{xy}$, 10 - 199 nodes

For the algorithmic aspect, since it has been proved that the MBrT problem can be solved in polynomial time on cactus graphs, one may extend it to possibly other classes of graphs such as series-parallel graphs and outerplanar graphs. Alternatively, it is also worth trying to devise algorithms for some classes of graphs with simple and exploitable structures such as wheels and fans.

Moreover, although the integrality proof of the polyhedron defined by the proposed formulation for $B_{xy}(G, r, \mathbf{c})$ is given in this dissertation, the TDI-ness of the formulation has yet to be proved. We obtained some examples that showed the insufficiency of the current formulation to be TDI. Thus, some redundant inequalities should be included in the formulation in order to obtain a TDI system. Besides, for $B_x(G, r, \mathbf{c})$ and $R_x(G, r)$, based on the study on trees and cycles, it is tempting to claim that a complete description is obtainable on cactus graphs, or to a wider extent, on series-parallel graphs.

On the other hand, the MBrT problem can be extended to a more generalized version,

which is called the Maximum Bounded r -Tree Packing (MBrTP) problem. The MBrTP problem, instead of finding one r -tree, consists of finding a packing of $K \geq 2$ r -trees which satisfies the overall capacity constraints. The study on the MBrT problem can be considered as the first step of the study on the MBrTP problem. There also exist corresponding applications of the MBrTP problem in various fields, hence it is worth studying its algorithms and polyhedral structure.

In fact, we have done some preliminary research on the polyhedral structure and computational tests for the MBrTP problem considering both edge- and node-indexed variables. On the polyhedral aspect, we have characterized the fractional extreme points in the case where $K = 2$ and the graph G is a star with r being the central node. The said extreme points can be cut by the following two new valid inequalities

$$\begin{aligned} x^1(F_1) + x^2(F_2) + y_r^1 - y_r^2 &\leq c_r & \forall F_1, F_2 \subseteq E, |F_1| + |F_2| > c_r, |F_1| < c_r, \\ x^1(F_1) + x^2(F_2) - y_r^2 &\leq c_r - 1 & \forall F_1, F_2 \subseteq E, |F_1| + |F_2| > c_r, |F_1| = c_r - 1, \end{aligned}$$

where \mathbf{x}^i and \mathbf{y}^i contain the variables associated with edges and nodes in the i -th bounded r -tree, $i \in \{1, 2\}$. Nonetheless, we have devised a polynomial combinatorial algorithm for the MBrTP problem on trees with $K = 2$ (Vinhas de Lima [2016]).

On the other hand, as BRT_{xy} has shown the strongest performance among all the approaches we have tested, and even without any primal heuristic, it still solved the instances on dense graphs in a matter of seconds, so we have implemented a multi-tree version of it, which we refer to as $B RTP_{xy}$. We have tested it on dense graphs having nodes between 10 and 199 with $K = 2$. The simulation results are reported in Table 7.10. The column t_r contains the time $B RTP_{xy}$ took to solve the linear relaxation at the root node of the branch-and-cut tree, whereas the column $N_{B\&C}$ indicates the number of nodes that have been gone through in the branch-and-cut tree. $B RTP_{xy}$ has solved most of the instances with 10 to 99 nodes, but cannot solve most of the instances with 100 to 199 nodes within two hours. There is a significant performance deficit comparing with the performance of BRT_{xy} in the case of $K = 1$ which has solved all the instances with 10 to 199 nodes within 10 seconds.

To summarize, the results obtained so far suggest that the polyhedral structure and the computational complexity of the MBrTP problem are much more complicated than the MBrT problem, even in the presumably easiest cases (e.g., $K = 2$, on stars).

Another direction to explore is b-matching and the related polyhedra. Despite it is a well-studied topic in the literature, the extended formulation for b-matching has not been brought up as a research topic as far as we know. We have found a representation of

an extended version of the well-know blossom inequality which might be worth studying.

$$x(E[S]) + x(F) - \sum_{u \in W} \left(\left\lceil \frac{c_u}{2} \right\rceil y_u \right) \leq \left\lfloor \frac{1}{2}(c(S \setminus W) + |F|) \right\rfloor \quad \forall W \subseteq S \subseteq V.$$

Besides, as the simple b-matching problem can be seen as a relaxation of the MB r T problem, the blossom inequality is actually valid for $B_x(G, r, \mathbf{c})$. We have implemented its separation algorithm and have tested in the computational simulation. However, it did not improve the overall performance. On the contrary, in many cases it actually slowed down the process by generating too many cuts at each node of the branch-and-cut tree, eventually resulted in a large number of iterations at each node as well as an obese model for the LP solver to solve at each iteration.

Aside from the blossom inequality, we also experimented with some lifted-projection inequalities in the computational test regarding $B_x(G, r, \mathbf{c})$ (similar to the last heuristic we introduced for matching-partition inequalities in Subsection 7.2.4). More precisely, we generated cuts by first lifting a solution $\bar{\mathbf{x}} \in \mathbb{R}^E$ to a point $\left[\frac{\bar{\mathbf{x}}}{\bar{\mathbf{y}}} \right]$ in \mathbb{R}^{E+V} , finding some violated inequalities regarding $\left[\frac{\bar{\mathbf{x}}}{\bar{\mathbf{y}}} \right] \in \mathbb{R}^{E+V}$ and $B_{xy}(G, r, \mathbf{c})$, and then projecting it down to a valid inequality for $B_x(G, r, \mathbf{c})$. This approach can effectively generate cuts in the branch-and-cut process. However, similar to the blossom inequality, this approach did not improve the computational results, as it might not be as effective as the other inequalities. Nevertheless, these inequalities deserve more of our attention, as one can potentially either lift them to a tighter version, or determine their facet-defining conditions, in order to generate cuts that are more effective.

Symmetry (Ostrowski et al. [2011], Fischetti and Liberti [2012], Fischetti et al. [2017]) is another potential issue in the problem that can be further addressed. In fact for $B_x(G, r, \mathbf{c})$, the i -articulation inequalities, especially capacity- i inequalities are symmetry-breaking inequalities to some extent. More specifically, capacity- i inequalities prevent all the symmetric solutions with respect to a set of nodes that have the same capacity and are incident with nodes in O .

References

- R. P. Anstee. A polynomial algorithm for b-matchings: an alternative approach. *Information Processing Letters*, 1987.
- G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Pro-tasi. A compendium of np optimization problems. In *Complexity and Approximation: Combinatorial optimization problems and their approximability properties*. Springer, 1999.
- E. Balas. Projection, lifting and extended formulation in integer and combinatorial optimization. *Annals of Operations Research*, pages 125–161, 2005.
- M. O. Ball. Heuristics based on mathematical programming. *Surveys in Operations Research and Management Science*, page 21–38, 2011.
- S. Banerjee, S. Lee, B. Bhattacharjee, and A. Srinivasan. Resilient overlays using multi-cast. *ACM/IEEE Transactions on Networking*, 2005.
- M. Behle, M. Jünger, and F. Liers. A primal branch-and-cut algorithm for the degree-constrained minimum spanning tree problem. *Experimental Algorithms - Lecture Notes in Computer Science*, pages 379–392, 2007.
- F. Bendali, I. Diarrassouba, M. Didi-Biha, A. R. Mahjoub, and J. Mailfert. The k-edge connected subgraph problem: Valid inequalities and branch-and-cut. In *6th International Workshop on Design and Reliable Communication Networks*, pages 1–8, 2007.
- E. Brosh and Y. Shavitt. Approximation and heuristic algorithms for minimum delay application-layer multicast trees. In *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM*, pages 2697–2707, 2004.
- T. N. Bui, X. Deng, and C. M. Zrnčić. An improved ant-based algorithm for the degree-constrained minimum spanning tree problem. *IEEE Transactions on Evolutionary Computation*, 16:266–278, 2012.
- J. Byers, M. Luby, and M. Mitzenmacher. Fine-grained layered multicast. In *Proc. IEEE INFOCOM*, 2001.
- J. Cao, J. Xie, and F. Chen. Dsd-d: A distributed algorithm for constructing high-stability application-layer multicast tree. In *Fifth International Conference on Frontier of Computer Science and Technology (FCST)*, pages 122–128, 2010.

- J. Chakareski, P. Frossard, H. Kerivin, J. Leblet, and Simon G. A note on the data-driven capacity of p2p networks. Technical Report EPFL-LTS-2009-008, École Polytechnique Fédérale de lausanne, 2009.
- Y. H. Chan, W. S. Fung, L. C. Lau, and C. K. Yung. Degree bounded network design with metric costs. In *IEEE 49th Annual Symposium on Foundations of Computer Science*, pages 125–134, 2008.
- J. Cheriyan and R. Thurimella. Approximating minimum-size k-connected spanning subgraphs via matching. In *IEEE 37th Annual Symposium on Foundations of Computer Science*, pages 292–301, 1996.
- J. Cheriyan and L. A. Vegh. Approximating minimum-cost k-node connected subgraphs via independence-free graphs. In *IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 30–39, 2013.
- S. Chopra and M. R. Rao. The steiner tree problem i: Formulations, compositions and extension of facets. *Mathematical Programming*, 64:209–229, 1994a.
- S. Chopra and M. R. Rao. The steiner tree problem ii: properties and classes of facets. *Mathematical Programming*, 64:231–246, 1994b.
- S. Chopra, E. Gorres, and M. R. Rao. Solving a steiner tree problem on a graph using branch and cut. *ORSA Journal on Computing*, 4:320–335, 1992.
- V. Chvátal. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete Mathematics*, pages 305–337, 1973.
- Cisco. Cisco visual networking index: Forecast and methodology, 2013-2018. <http://bit.ly/LVhmuL>, June 2014.
- S. A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.
- D. Cornaz, Y. Magnouche, and A. R. Mahjoub. On minimal two-edge-connected graphs. In *Proceedings of International Conference on Control, Decision and Information Technologies*, pages 251–256, 2014.
- G. B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1963.
- M. Didi-Biha, H. Kerivin, and A. R. Mahjoub. Steiner trees and polyhedra. *Discrete Applied Mathematics*, 112:101–120, 2001.
- M. Didi-Biha, H. Kerivin, and P. Ng. Polyhedral study of the connected subgraph problem. *Discrete Mathematics*, pages 80–92, 2015.
- R. Diestel. *Graph Theory*. Springer-Verlag New York, 2000.
- E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

- S. E. Dreyfus and R. A. Wagner. The steiner problem in graphs. *Networks*, 1:195–207, 1971.
- J. Edmonds. Covers and packings in a family of sets. *Bulletin of the American Mathematical Society*, pages 494–499, 1962.
- J. Edmonds. Maximum matching and a polyhedron with 0, 1 vertices. *Journal of Research of the National Bureau of Standard*, 69 B:125–130, 1965.
- J. Edmonds. Submodular functions, matroids, and certain polyhedra. In *Combinatorial Structures and Their Applications*, pages 69–87, 1970.
- J. Edmonds. Matroids and the greedy algorithm. *Mathematical Programming*, pages 127–136, 1971.
- J. Edmonds. Some well-solved problems in combinatorial optimization. In B. Roy, editor, *Combinatorial Programming: Methods and Applications*, pages 285–301. Reidel, Dordrecht, 1975.
- J. Edmonds and R. Giles. A min-max relation for submodular functions on graphs. *Annals of Discrete Mathematics*, pages 185–204, 1977.
- G. Farkas. A Fourier-féle mechanikai elv alkalmazása [Hungarian]. *Mathematikai és Természettudományi Értesítő*, pages 457–472, 1894.
- M. Fischetti and L. Liberti. Orbital shrinking. In *ISCO'12 Proceedings of the Second international conference on Combinatorial Optimization*, pages 48–58. IEEE, 2012.
- M. Fischetti, L. Liberti, D. Salvagnin, and T. Walsh. Orbital shrinking. *Discrete Applied Mathematics*, pages 109–123, 2017.
- H. N. Gabow. An efficient reduction technique for degree-constrained subgraph and bidirected network flow problems. In *Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing*, pages 448–456. The Association for Computing Machinery, 1983.
- D. Gale. Optimal assignments in an ordered set: an application of matroid theory. *Journal of Combinatorial Theory*, pages 176–180, 1968.
- D. Gale, H. W. Kuhn, and A. W. Tucker. Linear programming and the theory of games. In *Activity Analysis of Production and Allocation - Proceedings of a Conference*, pages 317–329, 1951.
- M. R. Garey and D. S. Johnson. The rectilinear steiner tree problem is np-complete. *SIAM Journal on Applied Mathematics*, pages 826–834, 1977.
- M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co. New York, 1979.
- A. M. H. Gerards. Matching. In M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser, editors, *Handbooks in Operations Research and Management Science*, vol-

- ume 7, chapter 3, pages 134–224. Elsevier, Amsterdam, 1995.
- M. X. Goemans. The steiner tree polytope and related polyhedra. *Mathematical Programming*, pages 157–182, 1994.
- M. X. Goemans. Minimum bounded degree spanning trees. In *Proc. of IEEE FOCS*, pages 273–282, 2006.
- M. X. Goemans and Y. Myung. A catalog of steiner tree formulations. *Networks*, 23: 19–28, 1993.
- R. E. Gomory. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society*, pages 275–278, 1958.
- R. E. Gomory. Solving linear programming problems in integers. In *Proceedings of Symposia in Applied Mathematics X*, American Mathematical Society, pages 211–215, 1960.
- M. Gondran and M. Minoux. *Graphs and Algorithms*. Wiley, Chichester, 1984.
- V. Goyal. Multiple description coding: Compression meets the network. *IEEE Signal Processing Magazine*, 18(5):74–93, 2001.
- V. Goyal, J. Kovacevic, R. Arean, and M. Vetterli. Multiple description transform coding of images. In *Proc. IEEE International Conference on Image Processing*, 1998.
- M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, pages 169–197, 1981.
- M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer, 1988.
- S. L. Hakimi. Steiner’s problem in graphs and its implications. *Networks*, 1:113–133, 1971.
- A. J. Hoffman. A generalization of max flow-min cut. *Mathematical Programming*, pages 352–359, 1974.
- K. Jain. A factor 2 approximation algorithm for the generalized steiner network problem. In *Proceedings of 39th Annual Symposium on Foundations of Computer Science*, pages 448–457, 1998.
- E. L. Johnson. *Network Flows, Graphs and Integer Programming*. PhD thesis, Operations Research Center, University of California, Berkeley, California, 1979.
- P. Jurčík and Z. Hanzalek. Construction of the bounded application-layer multicast tree in the overlay network model by the integer linear programming. In *10th IEEE Conference on Emerging Technologies and Factory Automation*, pages 503–510, 2005.
- N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, pages 373–395, 1984.

- R. M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103. Springer, 1972.
- R. M. Karp and C. H. Papadimitriou. On linear characterizations of combinatorial optimization problems. *SIAM Journal on Computing*, page 620–632, 1982.
- H. Kerivin and G. Simon. Exploring the dark side of live streaming. Preprint, 2012.
- H. Kerivin, J. Leblet, G. Simon, and F. Zhou. Models for the maximal bounded r-tree packing problem. Preprint, 2011.
- H. Kerivin, J. Leblet, G. Simon, and F. Zhou. Maximum bounded rooted-tree packing problem. Preprint submitted to Elsevier, 2014.
- L. G. Khachiyan. A polynomial algorithm in linear programming. *Dokl. Akad. Nauk SSSR*, pages 1093–1096, 1979.
- L. G. Khachiyan. Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics*, pages 53–72, 1980.
- T. Koch, A. Martin, D. Rehfeldt, and S. Voss. Steinlib testdata library. <http://steinlib.zib.de/>, 2017.
- B. Korte and J. Vygen. *Combinatorial Optimization - Theory and Algorithms*. Springer, 2012.
- J. B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. In *Proceedings of the American Mathematical Society*, volume 7, pages 48–50, 1956.
- E. L. Lawler. Matroid intersection algorithms. *Mathematical Programming*, pages 31–56, 1975.
- E. L. Lawler. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, 1976.
- A. N. Letchford and A. Lodi. Primal separation algorithms. *4OR - A Quarterly Journal of Operations Research*, pages 209–224, 2003.
- S. Li, R. Melhem, and T. F. Znati. An efficient algorithm for constructing delay bounded minimum cost multicast trees. *Journal of Parallel and Distributed Computing*, 64: 1399–1413, 2004.
- X. Li, M. Ammar, and S. Paul. Layered video multicast with retransmission (lvmr): Evaluation of hierarchical rate control. In *Proc. IEEE INFOCOM*, 1998.
- F. Liu, J. Huang, X. Lu, and Y. Peng. An efficient distributed algorithm for constructing delay- and degree-bounded application-level multicast tree. In *8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN)*, 2005.
- S. Liu, M. Chen, S. Sengupta, M. Chiang, J. Li, and P. A. Chou. P2p streaming capacity

- under node degree bound. In *IEEE INFOCOM*, 2010.
- L. Lovász. Graph theory and integer programming. *Annals of Discrete Mathematics*, pages 141–158, 1979.
- A. Lucena. Tight bounds for the steiner problem in graphs. Technical Report TR- 21/93, Dipartimento di Informatica. Univesitat degli Studi di Pisa, Pisa, Italy, 1993.
- N. Magharei, R. Rejaie, and Yang Guo. Mesh or multiple-tree: A comparative study of live p2p streaming approaches. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications.*, pages 1424–1432. IEEE, 2007.
- M. Malik. Bees algorithm for degree-constrained minimum spanning tree problem. In *National Conference on Computing and Communication Systems (NCCCS)*, pages 1–8, 2012.
- III Marsh, A. B. *Matching Algorithms*. PhD thesis, The Johns Hopkins University, Baltimore, Maryland, 1979.
- L. Massoulié, A. Twigg, C. Gkantsidis, and P. Rodriguez. Randomized decentralized broadcasting algorithms. In *Proc. of IEEE INFOCOM*, 2007.
- S. McCanne, V. Jacobson, and M. Vettereli. Receiver-driven layered multicast. In *Proc. ACM SIGCOMM*, 1996.
- K. Menger. Zur allgemeinen kurventheorie. *Fund. Math.*, pages 96–115, 1927.
- H. Minkowski. *Geometrie der Zahlen*. Leipzig : Teubner, 1896.
- G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. Wiley-Interscience New York, 1988.
- C. A. S. Oliveira, P. M. Pardalos, and M. G. C. Resende. Optimization problems in multicast tree construction. In *Handbook of Optimization in Telecommunications*, pages 701–731. Springer, 2006.
- J. Ostrowski, J. Linderoth, F. Rossi, and F. Smriglio. Orbital branching. *Mathematical Programming*, pages 147–178, 2011.
- M. Padberg. *Linear Optimization and Extensions*. Springer, 1999.
- M. W. Padberg and M. R. Rao. Odd minimum cut-sets and b-matchings. *Mathematics of Operations Research*, pages 67–80, 1982.
- M. W. Padberg and L. A. Wolsey. Trees and cuts. *Annals of Discrete Mathematics*, pages 511–517, 1983.
- V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai. Distributing streaming media content using cooperative networking. In *Proc. ACM NOSSDAV*, 2002.

- C. H. Papadimitriou. *Computational complexity*. Addison Wesley, 1994.
- D. Papageorgiou, M. Cheon, S. Harwood, F. Trespalacios, and G. Nemhauser. Recent progress using matheuristics for strategic maritime inventory routing. Book chapter in forthcoming Springer Volume: "Modelling, Computing and Data Handling Methodologies for Maritime Transportation.", 2016.
- A. Passarella. A survey on content-centric technologies for the current internet: Cdn and p2p solutions. *Computer Communications*, 35(1):1–32, 2012.
- R. C. Prim. Shortest connection networks and some generalizations. *Bell System Technical Journal*, 36:1389–1401, 1957.
- W. R. Pulleyblank. *Faces of Matching Polyhedra*. PhD thesis, Department of Combinatorics and Optimization, Faculty of Mathematics, University of Waterloo, 1973.
- W. R. Pulleyblank. Total dual integrality and b-matchings. *Operations Research Letters*, pages 28–30, 1981.
- R. Rado. Note on independence functions. In *Proceedings of of the London Mathematical Society*, volume 7, pages 300–320, 1957.
- G. R. Raidl. An efficient evolutionary algorithm for the degree-constrained minimum spanning tree problem. In *Proceedings of the 2000 Congress on Evolutionary Computation*, pages 104–111, 2000.
- R. L. Rardin, R. G. Parker, and M. B. Richley. A polynomial algorithm for a class of steiner tree problems on graphs. Technical Report Industrial and Systems Eng. Report J-82-5, Georgia Inst. of Technology, Atlanta, 1982.
- A. Schrijver. Polyhedral proof methods in combinatorial optimization. *Discrete Applied Mathematics*, pages 111–133, 1986a.
- A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, Inc. New York, 1986b.
- A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer-Verlag Berlin Heidelberg, 2003.
- M. Singh and L. C. Lau. Approximating minimum bounded degree spanning trees to within one of optimal. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing (STOC '07)*, pages 661–670, 2007.
- E. Steinitz. Bedingt konvergente reihen und konvexe systeme. *Journal für die reine und angewandte Mathematik*, pages 1–52, 1916.
- T. Stockhammer. Dynamic adaptive streaming over http - standards and design principles. In *MMSys '11 Proceedings of the second annual ACM conference on Multimedia systems*, pages 133–144, 2011.
- R. Sweha, V. Ishakian, and A. Bestavros. Angelcast: Cloud-based peer-assisted live

- streaming using optimized multi-tree construction. In *MMSys*, 2012.
- D. A. Tran, K. A. Hua, and T. T. Do. A peer-to-peer architecture for media streaming. *IEEE Journal on Selected Areas in Communications*, 22:121–133, 2004.
- C. Vinhas de Lima. Le Problème du Packing Borné Maximal de r -Arbre: Etude polyédrale pour le cas du graphe étoile avec 2 arbres. Master's thesis, Institut Supérieur d'Informatique, de Modélisation et de leurs Applications (ISIMA), 2016.
- A. Vitali. Multiple description coding - a new technology for video streaming. *EBU Technical Review*, Oct 2007.
- J. von Leeuwen. *Handbook of theoretical computer science: Algorithms and complexity*. Elsevier & The MIT Press, 1990.
- J. von Neumann. Discussion of a maximum problem. Institute for Advanced Study, Princeton, New Jersey, November 1947.
- J. A. Wald and C. J. Colbourn. Steiner problem in outerplanar graphs. *Congressus Numerantium*, 36:15–22, 1982a.
- J. A. Wald and C. J. Colbourn. Steiner trees, partial 2-trees, and minimum ifi networks. *Networks*, 13:159–167, 1982b.
- H. Weyl. Elementare theorie der konvexen polyeder. *Commentarii Mathematici Helvetici*, pages 290–306, 1934.
- H. Whitney. Non-separable and planar graphs. *Transactions of the American Mathematical Society*, pages 339–362, 1932.
- H. Whitney. On the abstract properties of linear dependence. *American Journal of Mathematics*, pages 509–533, 1935.
- P Winter. Steiner problem in halin networks. *Discrete Applied Mathematics*, 17:281–294, 1987.
- L. A. Wolsey. *Integer Programming*. Wiley-Interscience, 1998.
- R. T. Wong. A dual ascent approach for steiner tree problems on a directed graph. *Mathematical Programming*, pages 271–287, 1984.
- W. Yiu, X. Jin, and S. Chan. Challenges and approaches in large-scale peer-to-peer media streaming. In *IEEE Multimedia*, volume 14(2), pages 50–59, 2007.
- J. Zhao. Optimal distribution of video streams in large peer-to-peer networks. Master's thesis, Institut Supérieur d'Informatique, de Modélisation et de leurs Applications (ISIMA), 2012.

Numéro d'Ordre : D.U. 2816

EDSPIC : 799

Université Clermont Auvergne
École Doctorale : Sciences Pour l'Ingénieur de
Clermont-Ferrand

THÈSE

Présentée par

Jinhua ZHAO

pour obtenir le grade de

Docteur d'Université

Spécialité : Informatique

Le Problème de l'Arbre Enraciné Borné Maximum :
Algorithmes et Polyèdres
(Résumé Français)

Soutenue publiquement le 19 juin 2017 devant le jury

M. ou Mme Mohamed DIDI-BIHA Rapporteur et examinateur
Sourour ELLOUMI Rapporteuse et examinatrice
Ali Ridha MAHJOUR Rapporteur et examinateur
Fatiha BENDALI Examinatrice
Hervé KERIVIN Directeur de Thèse
Philippe MAHEY Directeur de Thèse

Table des matières

1	Introduction	1
2	Algorithmes combinatoires pour le problème MB^rT	5
3	Etude polyédrique sur un polytope étendu pour le problème MB^rT	9
4	Etude polyédrique sur un polytope pour le problème MrT	11
5	Etude polyédrique sur un polytope pour le problème MB^rT	13
6	Étude expérimentale	15
	Conclusion	19

Chapitre 1

Introduction

Étant donné un graphe simple non orienté $G = (V, E)$ et un sommet particulier $r \in V$ appelé *racine*, un *arbre enraciné*, ou *r -arbre*, de G est soit le graphe nul (\emptyset, \emptyset) soit un arbre contenant r . Si un vecteur de capacités sur les sommets est donné, un sous-graphe de G est dit borné si le degré de chaque sommet dans le sous-graphe est inférieur ou égal à sa capacité. Soit \mathbf{w} un vecteur de poids sur les arêtes dans \mathbb{R}^E et \mathbf{p} un vecteur de profits sur les sommets dans \mathbb{R}^V . Le problème du r -arbre borné maximum (MBrT, de l'anglais Maximum Bounded r -Tree) consiste à trouver un r -arbre borné $T = (U, F)$ de G tel que $f(T) = \sum_{e \in F} w_e + \sum_{v \in U} p_v$ soit maximisé. Si la contrainte de capacité du problème MBrT est relâchée, nous obtenons le problème du r -arbre maximum (MrT, de l'anglais Maximum r -Tree). Cette thèse contribue à l'étude des problèmes MBrT et MrT.

Tout d'abord, il a été prouvé que ces deux problèmes sont \mathcal{NP} -difficiles.

Théorème 1.0.1. *Le problème MrT est \mathcal{NP} -difficile.* □

Théorème 1.0.2. *Le problème MBrT est \mathcal{NP} -difficile.* □

L'enveloppe convexe des vecteurs d'incidence des r -arbres de G s'appelle le polytope étendu pour r -arbre et est notée par

$$R_{xy}(G, r) = \text{conv}(\left\{ \begin{bmatrix} \mathbf{x}^F \\ \mathbf{y}^U \end{bmatrix} \in \{0, 1\}^E \times \{0, 1\}^V : (U, F) \text{ est un } r\text{-arbre} \right\}).$$

La formulation suivante est proposée pour le polytope étendu pour r -arbre .

On note $P_{xy}(G, r)$ le polyèdre de \mathbb{R}^{E+V} défini par

$$x(E) - y(V \setminus \{r\}) = 0, \tag{1.1}$$

$$x(E[S]) - y(S \setminus \{v\}) \leq 0 \quad \forall v \in S \subseteq V, |S| \geq 2, \tag{1.2}$$

$$y_r \leq 1, \tag{1.3}$$

$$x_e \geq 0 \quad \forall e \in E. \quad (1.4)$$

Cette formulation a été prouvée pour décrire complètement le polytope étendu pour r -arbre sur les graphes série-parallèles, donc le théorème suivant est valable.

Théorème 1.0.3. *Soit G un graphe série parallèle, $R_{xy}(G, r) = P_{xy}(G, r)$.* \square

Lorsque nous considérons uniquement les arêtes, nous disons qu'un ensemble d'arêtes $F \subseteq E$ induit un r -arbre si le graphe $G_F = (V[F] \cup \{r\}, F)$ est un r -arbre. On peut dire que l'ensemble vide induit le r -arbre $G_\emptyset = (\{r\}, \emptyset)$.

L'enveloppe convexe des vecteurs d'incidence des ensembles des arêtes induit r -arbre de G s'appelle le polytope r -arbre et est notée par

$$R_x(G, r) = \text{conv}(\{\mathbf{x}^F \in \{0, 1\}^E : G_F \text{ est un } r\text{-arbre}\}).$$

De même, les polyèdres pour r -arbre borné peuvent être définis comme suit.

L'enveloppe convexe des vecteurs d'incidence des r -arbres bornés de G s'appelle le polytope étendu pour r -arbre borné et est notée par

$$B_{xy}(G, r, \mathbf{c}) = \text{conv}(\left\{ \begin{bmatrix} \mathbf{x}^F \\ \mathbf{y}^U \end{bmatrix} \in \{0, 1\}^{E+V} : (U, F) \text{ est un } r\text{-arbre borné} \right\}).$$

L'enveloppe convexe des vecteurs d'incidence des ensembles des arêtes induit r -arbre borné de G s'appelle le polytope pour r -arbre borné et est notée par

$$B_x(G, r, \mathbf{c}) = \text{conv}(\{\mathbf{x}^F \in \{0, 1\}^E : G_F \text{ est un } r\text{-arbre borné}\}).$$

Il convient de noter que, pour le problème MBrT, un ensemble de noeuds qui se distinguent par le développement de l'algorithme et l'étude polyédrique sont les noeuds dotés de la capacité unité. On peut remarquer que ces noeuds (à l'exception de r) ne fournissent aucune connectivité aux autres noeuds, c'est-à-dire qu'ils ne peuvent apparaître que comme feuilles dans tout r -arbre borné. L'ensemble de ces noeuds est noté par

$$O := \{v \in V \setminus \{r\} : c_v = 1\}. \quad (1.5)$$

En raison de l'existence de O , on peut remarquer que, même si G est connecté, il peut y avoir des noeuds ou des arêtes de G de sorte qu'ils ne peuvent être atteints par un r -arbre borné de G . Nous disons qu'un noeud v (arête e , respectivement) de G est *inaccessible* depuis la racine r s'il n'existe aucun chemin dans G contenant r et v (e , respectivement) qui n'a aucun noeud dans O comme un noeud interne. On note V_u et E_u les ensembles composés des noeuds et des arêtes inaccessibles de G , respectivement. Tout vecteur d'incidence

d'un r -arbre borné de G satisfait directement aux équations suivantes

$$x_e = 0 \quad \forall e \in E_u, \tag{1.6}$$

$$y_v = 0 \quad \forall v \in V_u. \tag{1.7}$$

En fait, pour les études algorithmiques et polyédriques, les noeuds et arêtes inaccessibles sont insignifiants. Nous supposons donc que

$$V_u = \emptyset,$$

$$E_u = \emptyset.$$

Cette thèse est consacrée à l'étude polyédrique, algorithmique et numérique des problèmes $MBrT$ et MrT . Dans ce résumé, une série de résultats est présentée brièvement sur chaque aspect. De plus amples détails sont disponibles dans la version anglaise.

Chapitre 2

Algorithmes combinatoires pour le problème MBrT

Dans ce chapitre, nous présentons des algorithmes de temps polynomial pour le problème MBrT sur trois différentes catégories de graphes : arbres, cycles et cactus. Nous définissons $Opt_G(r)$ comme la valeur de $f(T)$ associée à un r -arbre borné maximum de G , où l'indice G peut être omis lorsque le graphie est dégagé du contexte. Opt_G peut également être utilisé à la place de $Opt_G(r)$ lorsque la racine est dégagée du contexte.

Si le graphe G est un arbre, nous présentons une approche de programmation dynamique pour résoudre le problème MBrT.

Étant donné tout nœud $v \in V$, on note $g(v)$ comme la valeur d'un arbre borné maximum non vide arraché à v du sous-graphe $G[[v]]$ de G induit par la fermeture-en-haut (up-closure, en anglais) de v , où le vecteur de capacité $\mathbf{c}^v \in \mathbb{Z}_+^{[v]}$ satisfait $c_v^v = c_v - 1$ si $v \neq r$, $c_v^v = c_v$ si $v = r$, et $c_s^v = c_s$ pour $s \in [v] \setminus v$. En d'autres termes,

$$g(v) = \max\{f(T) : T \text{ est un arbre de } G[[v]] \text{ avec } v \in V(T) \text{ et borné par } \mathbf{c}^v\}.$$

Nous avons

$$Opt_G(r) = \max\{0, g(r)\}.$$

Pour toute feuille $v \in V \setminus \{r\}$, il est facile de voir que

$$g(v) = p_v. \tag{2.1}$$

Pour un nœud non-feuille $v \in V$, nous définissons $N_u(v) = \{v_1, \dots, v_q\}$ et

$$h(v_k) = w_{vv_k} + g(v_k).$$

Nous avons alors

$$g(v) = p_v + \sum_{k=1}^j h(v_k). \quad (2.2)$$

Sur la base de ces résultats, l'algorithme suivant résout le problème MBrT avec la complexité de $\mathcal{O}(n)$. Ainsi, le théorème suivant tient.

Algorithm 2.1: Algorithme pour calculer $g(r)$ sur les arbres

Input : Tree $G = (V, E)$, $\mathbf{w} \in \mathbb{R}^E$ and $\mathbf{p} \in \mathbb{R}^V$.

Output: $g(r)$.

begin

- 1 Set S as the set of the leaves in $V \setminus \{r\}$.
 - 2 Compute $g(v)$ for each $v \in S$ according to (2.1).
 - while** $S \neq \{r\}$ **do**
 - 3 Set S as the node set such that for each $v \in S$, $g(v)$ is unknown and $g(v_k)$ is known for all $k \in \{1, \dots, q\}$.
 - 4 Compute $g(v)$ for each $v \in S$ according to (2.2).
-

Théorème 2.0.1. *Le problème MBrT peut être résolu en temps linéaire sur un arbre.* \square

Comme résultat immédiat, nous avons également le corollaire suivant.

Corollaire 2.0.2. *Le problème MBrT sur un graphe contenant un seul cycle peut être résolu au temps de $\mathcal{O}(n^2)$.* \square

Sur les cycles, nous devons traiter le problème MB r T dans plusieurs cas différents selon le nombre de nœuds en O et la valeur de c_r . Nous prenons le cas de $c_r = 2$ à titre d'exemple.

Désignons le chemin entre v_1 et v_{n-1} sans passer par r par

$$P_r = E \setminus \{e_1, e_n\}.$$

Un uv -chemin de $G[P_r]$ est notée par $P_{uv}^{P_r}$. Étant donné un chemin non vide P_{uv} entre u et v , nous définissons

$$\zeta(P_{uv}) := f(G[P_{uv}]) - p_u - p_v,$$

et

$$\zeta_{Min}(P_r) := \min\{\zeta(P_{uv}^{P_r}) : |P_{uv}^{P_r}| \geq 1, |P_{uv}^{P_r} \cap \delta(O)| \geq |O|\}.$$

Algorithme 2.2 résout le problème MBrT dans ce cas en calculant la valeur de $\zeta_{Min}(P_r)$.

Pour chaque cas sur les cycles, nous avons conçu un algorithme linéaire pour résoudre le problème MBrT. Pour les détails des algorithmes et des preuves, veuillez référer la version anglaise. Par conséquent, le théorème suivant est valable.

Algorithm 2.2: Algorithme pour obtenir $\zeta_{Min}(P_r)$ sur les cycles

Input : Cycle $G = (V, E)$, $\mathbf{w} \in \mathbb{R}^E$ and $\mathbf{p} \in \mathbb{R}^V$
Output: $\zeta_{Min}(P_r)$.
begin
1 Set $\zeta_{Min}^{v_{n-2}}(P_{v_{n-2}v_{n-1}}^{P_r}) = \zeta(\{e_{n-1}\}) = w_{e_{n-1}}$.
2 Set $i = n - 3$.
while $i \geq 1$ **do**
3 Set $\zeta_{Min}^{v_i}(P_{v_i v_{n-1}}^{P_r}) = \min\{w_{e_{i+1}}, w_{e_{i+1}} + p_{v_{i+1}} + \zeta_{Min}^{i+1}(P_{v_{i+1}v_{n-1}}^{P_r})\}$.
4 Set $i = i - 1$
5 $\zeta_{Min}(P_r) = \min\{\zeta_{Min}^{v_i}(P_{v_i v_{n-1}}^{P_r}) : i \in \{1, \dots, n - 2\}\}$.

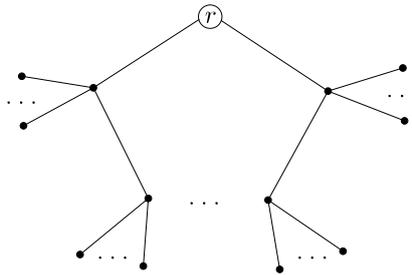


FIGURE 2.1 – Un cactus basis

Théorème 2.0.3. *Le problème MBrT peut être résolu en temps linéaire sur un cycle.* \square

Considérons maintenant le problème MBrT sur un cactus G .

Un *cactus basis* comme illustré dans la Figure 2.1 peut être développé pour résoudre le problème MBrT sur les cactus. Il est composé d'un cycle où chaque nœud, à l'exception du nœud racine, est incident avec des arêtes en attente. Comme il contient un seul cycle, Corollary 2.0.2 implique que le problème MBrT sur les cactus basis peut être résolu en temps polynomial.

Noter $Opt_G^i(r)$ comme le maximum poids d'un r -arbre borné $T = (U, F)$ de G tel que $|\delta(r) \cap F| = i$, $i \leq c_r$. Opt_G^i est également utilisé pour simplifier si le nœud racine est dégagé du contexte.

La équation suivante est valable.

$$Opt_G^1 - p_r \geq Opt_G^2 - Opt_G^1.$$

Sur la base de ces résultats, nous transformons chaque composant d'arbre en une arête et chaque composant de cycle en deux arêtes de manière récursive. Les poids de ces arêtes sont attribués selon les sous-problèmes résolus sur les composants respectifs. À la fin, le

cactus se transforme en un arbre, puis le problème MBrT peut être résolu pour obtenir la solution optimale, comme décrit dans Algorithme 2.3.

Algorithm 2.3: Algorithme pour le problème MBrT sur les cactus

Input : Cactus $G = (V, E)$, $\mathbf{w} \in \mathbb{R}^E$ and $\mathbf{p} \in \mathbb{R}^V$

Output: Solution to the MBrT problem on G .

begin

1	while <i>there exists some cycle C attached at a hinge h such that G_C only contains one cycle</i> do																		
	<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">1</td> <td style="border-left: 1px solid black; padding-left: 10px; vertical-align: top;"> if G_C <i>is already transformed into a cactus basis</i> then </td> </tr> <tr> <td></td> <td style="border-left: 1px solid black; padding-left: 10px; vertical-align: top;"> <table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into two edges.</td> </tr> </table> </td> </tr> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">2</td> <td style="border-left: 1px solid black; padding-left: 10px; vertical-align: top;"> <table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;"> else if G_C <i>contains some tree branch G_{v_b} attached at a hinge h'</i> then </td> </tr> <tr> <td></td> <td style="border-left: 1px solid black; padding-left: 10px; vertical-align: top;"> <table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into an edge.</td> </tr> </table> </td> </tr> </table> </td> </tr> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">3</td> <td style="border-left: 1px solid black; padding-left: 10px; vertical-align: top;"> <table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Solve the MBrT problem on the transformed graph using Algorithm 2.1.</td> </tr> </table> </td> </tr> </table>	1	if G_C <i>is already transformed into a cactus basis</i> then		<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into two edges.</td> </tr> </table>	┌	Transform G_C into two edges.	2	<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;"> else if G_C <i>contains some tree branch G_{v_b} attached at a hinge h'</i> then </td> </tr> <tr> <td></td> <td style="border-left: 1px solid black; padding-left: 10px; vertical-align: top;"> <table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into an edge.</td> </tr> </table> </td> </tr> </table>	┌	else if G_C <i>contains some tree branch G_{v_b} attached at a hinge h'</i> then		<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into an edge.</td> </tr> </table>	┌	Transform G_C into an edge.	3	<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Solve the MBrT problem on the transformed graph using Algorithm 2.1.</td> </tr> </table>	┌	Solve the MBrT problem on the transformed graph using Algorithm 2.1.
1	if G_C <i>is already transformed into a cactus basis</i> then																		
	<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into two edges.</td> </tr> </table>	┌	Transform G_C into two edges.																
┌	Transform G_C into two edges.																		
2	<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;"> else if G_C <i>contains some tree branch G_{v_b} attached at a hinge h'</i> then </td> </tr> <tr> <td></td> <td style="border-left: 1px solid black; padding-left: 10px; vertical-align: top;"> <table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into an edge.</td> </tr> </table> </td> </tr> </table>	┌	else if G_C <i>contains some tree branch G_{v_b} attached at a hinge h'</i> then		<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into an edge.</td> </tr> </table>	┌	Transform G_C into an edge.												
┌	else if G_C <i>contains some tree branch G_{v_b} attached at a hinge h'</i> then																		
	<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Transform G_C into an edge.</td> </tr> </table>	┌	Transform G_C into an edge.																
┌	Transform G_C into an edge.																		
3	<table style="border-collapse: collapse; width: 100%;"> <tr> <td style="width: 20px; vertical-align: top; padding-right: 10px;">┌</td> <td style="padding-left: 10px; vertical-align: top;">Solve the MBrT problem on the transformed graph using Algorithm 2.1.</td> </tr> </table>	┌	Solve the MBrT problem on the transformed graph using Algorithm 2.1.																
┌	Solve the MBrT problem on the transformed graph using Algorithm 2.1.																		

Par conséquent, nous avons le théorème suivant.

Théorème 2.0.4. *Le problème MBrT sur un cactus peut être résolu en temps polynomial.*

□

Chapitre 3

Etude polyédrique sur un polytope étendu pour le problème MBrT

La formulation initiale proposée pour ce polytope est la suivante.

$$x(E) - y(V \setminus \{r\}) = 0, \quad (3.1)$$

$$x(E[S]) - y(S \setminus \{v\}) \leq 0 \quad \forall v \in S \subseteq V, |S| \geq 2, \quad (3.2)$$

$$x(\delta(v)) - c_v y_v \leq 0 \quad \forall v \in V, \quad (3.3)$$

$$y_r \leq 1, \quad (3.4)$$

$$x_e \geq 0 \quad \forall e \in E. \quad (3.5)$$

Cependant, au cours de notre travail, deux familles de nouvelles équations valides ont été découvertes.

Étant donné un noeud $v_o \in O$,

$$x(\delta(v_o)) - y_{v_o} = 0 \quad (3.6)$$

est valable pour $B_{xy}(G, r, \mathbf{c})$.

Soit $(V_1, E_1), \dots, (V_q, E_q)$, $q \geq 1$, représentent les blocs du graphe $G_r = G[V \setminus O]$. Pour tout bloc (V_i, E_i) , $i \in \{1, \dots, q\}$,

$$x(E_i) - y(V_i \setminus \{v_{a_i}\}) = 0 \quad (3.7)$$

est valable pour $B_{xy}(G, r, \mathbf{c})$.

Avec ces équations, la dimension de $B_{xy}(G, r, \mathbf{c})$ peut être décidée comme dans le théorème suivant.

Théorème 3.0.1. $\dim B_{xy}(G, r, \mathbf{c}) = |E| + |V_r| - q.$ □

En outre, nous avons également prouvé que le polytope peut être décomposé par 1-somme. Dénoter

$$P_C(G, r, \mathbf{c}) = \left\{ \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} : \begin{bmatrix} \mathbf{x}^1 \\ \mathbf{y}^1 \end{bmatrix} \in B_{xy}(G_1, r_1, \mathbf{c}^1), \begin{bmatrix} \mathbf{x}^2 \\ \mathbf{y}^2 \end{bmatrix} \in B_{xy}(G_2, r_2, \mathbf{c}^2), \right. \\ \left. x(\delta(v_a)) - c_{v_a} y_{v_a} \leq 0 \right\}.$$

Nous prouvons que $P_C(G, r, \mathbf{c})$ et $B_{xy}(G, r, \mathbf{c})$ sont égaux. On peut prouver que les deux polytopes ont le même ensemble de points intégrés, et puis $P_C(G, r, \mathbf{c})$ n'a pas de point extrême fractionnaire. Ainsi, on a la proposition suivante.

Théorème 3.0.2. $B_{xy}(G, r, \mathbf{c}) = P_C(G, r, \mathbf{c}).$ □

En outre, nous pouvons prouver que le polytope $B_{xy}(G, r, \mathbf{c})$ peut être caractérisé sur les arbres et les cycles. Nous utilisons la formulation suivante.

$$P_{Cac}(G, r, \mathbf{c}) = \left\{ \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \in \mathbb{R}^{E+V} : \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \text{ satisfait (3.2) – (3.5) and (3.6), (3.7)} \right\}.$$

Théorème 3.0.3. *Soit G un arbre ou un cycle, $P_{Cac}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c}).$* □

En le combinant avec Théorème 3.0.2, on a le résultat suivant.

Théorème 3.0.4. *Soit G un cactus, $P_{Cac}(G, r, \mathbf{c}) = B_{xy}(G, r, \mathbf{c}).$* □

Outre les résultats susmentionnés, les conditions nécessaires et suffisantes pour que les inégalités valides soient facettes définissantes sont également examinées. Les résultats et les preuves peuvent être trouvés dans la version anglaise de cette thèse.

Chapitre 4

Etude polyédrique sur un polytope pour le problème MrT

La formulation initiale proposée pour $R_x(G, r)$ est la suivante.

$$x_e - x(\delta(S)) \leq 0 \quad \forall e \in E[S], S \subseteq V \setminus \{r\}, \quad (4.1)$$

$$x(E[S]) \leq |S| - 1 \quad \forall S \subseteq V, |S| \geq 3, \quad (4.2)$$

$$x_e \leq 1 \quad \forall e \in E, \quad (4.3)$$

$$x_e \geq 0 \quad \forall e \in E. \quad (4.4)$$

$R_x(G, r)$ est en fait en pleine dimension dans l'hypothèse d'aucun élément inaccessibles.

Théorème 4.0.1. $R_x(G, r)$ est en pleine dimension, c'est-à-dire, $\dim R_x(G, r) = |E|$. \square

En outre, nous présentons également deux familles de nouvelles inégalités valables pour $R_x(G, r)$.

Soit $\pi = \{S_0, S_1, \dots, S_k\}$, $k \geq 1$, une partition de V avec $r \in S_0$ et $M = \{e_1, \dots, e_k\}$ un couplage (ou matching, en anglais) de G avec $e_i \in E[S_i]$ pour tous $i \in \{1, \dots, k\}$. La paire (M, π) s'appelle *matching-partition* de G . Noter par $\mathcal{MP}(G)$ l'ensemble composé de toutes les partitions correspondantes de G et par E_π l'ensemble des arêtes ayant leurs extrémités dans différentes classes de partition π . Avec un matching-partition $(M, \pi) \in \mathcal{MP}(G)$, on peut associer l'*inégalité matching-partition*

$$x(M) - x(E_\pi) \leq 0. \quad (4.5)$$

Étant donné $W \subseteq S \subseteq V \setminus \{r\}$, $|W| \geq 2$, l'*inégalité acyclicité-connectivité* est définie comme suit.

$$x(E[W]) - (|W| - 1)x(\delta(S)) \leq 0 \quad (4.6)$$

Les conditions nécessaires et suffisantes pour que les inégalités valides, tant dans la formulation initiale que nouvellement découverte, soient facettes définissantes sont également examinées. Les résultats et les preuves peuvent être trouvés dans la version anglaise de cette thèse.

De plus, L'étude polyédrique pour $R_x(G, r)$ ne peut pas être simplement limitée aux graphes 2-connexes sur la base de la décomposition par 1-somme, qui peut être prouvée dans certains cas simples. Toutefois, nous montrons dans le prochain chapitre qu'il existe des décompositions possibles pour certains cas spéciaux, et $R_x(G, r)$ peut être caractérisé sur les arbres et les cycles.

Chapitre 5

Etude polyédrique sur un polytope pour le problème MB r T

La formulation initiale pour $B_x(G, r, \mathbf{c})$ est donnée comme suit.

$$x_e - x(\delta(S)) \leq 0 \quad \forall e \in E[S], S \subseteq V \setminus \{r\}, \quad (5.1)$$

$$x(E[S]) \leq |S| - 1 \quad \forall S \subseteq V, |S| \geq 3, \quad (5.2)$$

$$x(\delta(v)) \leq c_v \quad \forall v \in V, \quad (5.3)$$

$$x_e \leq 1 \quad \forall e \in E, \quad (5.4)$$

$$x_e \geq 0 \quad \forall e \in E. \quad (5.5)$$

Comme $R_x(G, r)$, nous pouvons également prouver que $B_x(G, r, \mathbf{c})$ est en pleine dimension.

Théorème 5.0.1. $B_x(G, r, \mathbf{c})$ est en pleine dimension, c'est-à-dire, $\dim B_x(G, r, \mathbf{c}) = |E|$. \square

Quelques critères généraux ont été découverts pour que les inégalités soient facettes définissantes. Nous donnons les deux lemmes suivants comme exemples.

Lemme 5.0.2. Soit une inégalité valide $\mathbf{a}^T \mathbf{x} \leq b$ pour $B_x(G, r, \mathbf{c})$ qui est différent de $x_e \geq 0$ pour certains $e \in E$. $\mathbf{a}^T \mathbf{x} \leq b$ est facette définissante pour $B_x(G, r, \mathbf{c})$ seulement si elle satisfait $a_e \geq 0$ pour tout arête $e \in \delta(O)$. \square

Lemme 5.0.3. Soit $c_r = 1$ et une inégalité valide $\mathbf{a}^T \mathbf{x} \leq b$ pour $B_x(G, r, \mathbf{c})$ qui est différent de $x(\delta(r)) \leq c_r$. Alors, $\mathbf{a}^T \mathbf{x} \leq b$ est facette définissante pour $B_x(G, r, \mathbf{c})$ seulement si $b = 0$. \square

Comme pour $R_x(G, r)$, nous avons également trouvé quelques familles de nouvelles inégalités valides pour $B_x(G, r, \mathbf{c})$. En fait, les deux familles de nouvelles inégalités pour

$R_x(G, r)$ peuvent être adaptées aux inégalités valables pour $B_x(G, r, \mathbf{c})$.

On note par E_π l'ensemble des arêtes ayant leurs extrémités dans différentes classes de un matching-partition $(M, \pi) \in \mathcal{MP}(G)$. Avec (M, π) , on peut associer l'inégalité matching-partition suivante

$$x(M) - x(E_\pi \setminus \delta(O)) \leq 0. \quad (5.6)$$

Étant donné $W \subseteq S \subseteq V \setminus \{r\}$, $|W| \geq 2$, l'inégalité acyclicité-connectivité est définie comme suit.

$$x(E[W]) - (|W| - 1)x(\delta(S) \setminus \delta(O)) \leq 0 \quad (5.7)$$

En outre, il existe quelques autres inégalités valides pour $B_x(G, r, \mathbf{c})$. Nous donnons deux exemples dans la partie suivante.

Étant donné $S \subseteq V \setminus \{r\}$ avec $v \in S \setminus O$, l'inégalité capacité de téléchargement est

$$x(\delta(v)) - c_v x(\delta(S) \setminus \delta(O)) \leq 0. \quad (5.8)$$

Nous définissons

$$S_i = \{v \in V \setminus \{r\} : c_v = i, i \geq 2\}.$$

Étant donné $S \subseteq S_i$, on note $R = V \setminus (S \cup O)$. L'inégalité capacité- i est

$$x(\delta(O) \setminus \delta(R)) - (i - 2)x(E[S]) - (i - 1)x(\delta(R) \setminus \delta(O)) \leq 0. \quad (5.9)$$

Avec les inégalités matching-partition et les inégalités capacité de téléchargement, nous montrons que $B_x(G, r, \mathbf{c})$ peut être caractérisé sur les arbres et les cycles d'un système linéaire qui est totalement dual intégré (TDI).

Théorème 5.0.4. *Sur les arbres, le système linéaire*

$$x_e - x_{f_e} \leq 0 \quad \forall e \in E \setminus \delta(r), \quad (5.10)$$

$$x(\delta(v)) - c_v x_{f_v} \leq 0 \quad \forall v \in V \setminus \{r\}, \quad (5.11)$$

$$x(\delta(r)) \leq c_r, \quad (5.12)$$

$$x_e \leq 1 \quad \forall e \in \delta(r), \quad (5.13)$$

$$x_e \geq 0 \quad \forall e \text{ est un arête de feuille}, \quad (5.14)$$

est TDI, et caractérise complètement $B_x(G, r, \mathbf{c})$. □

Des résultats similaires sont obtenus sur les cycles concernant TDI, mais sont prouvés dans quatre cas différents.

Chapitre 6

Étude expérimentale

Nous étudions les problèmes de séparation pour toutes les inégalités que nous avons trouvées jusqu'ici. Des algorithmes polynomiaux de séparation sont présentés, et lorsqu'un problème de séparation est \mathcal{NP} -difficile, nous donnons des heuristiques de séparation. Tous les résultats théoriques développés dans ce travail sont implémentés dans plusieurs algorithmes de coupes et branchements auxquels une matheuristique est également jointe pour générer rapidement des solutions réalisables. Des expérimentations intensives ont été menées via le logiciel CPLEX afin de comparer les formulations renforcées et originales. Les résultats obtenus montrent de manière convaincante la force des formulations renforcées. Puis, avec CPLEX et la technologie Concert, nous avons mis en place quatre cadres de l'algorithme de coupe, $CPLEX_x$, $CPLEX_{xy}$, BRT_x , BRT_{xy} . Tableau 6.1 donne les options choisies pour chaque cadre.

Nous testons sur les instances soit sélectionnées dans SteinLib ou générées de façon aléatoire, comme le montre le Tableau 6.2.

Dans les Figures 6.1 - 6.4, nous montrons la performance des quatre algorithmes sur chaque testset.

Sur les plus de 600 instances que nous avons testées, BRT_{xy} a résolu 73,61 % d'entre eux et BRT_x 52.46 %, tandis que $CPLEX_{xy}$ a résolu 47.70 % d'entre eux et $CPLEX_x$ 35.08 %. Cela montre, dans une large mesure, que le niveau de force global de chaque cadre

Cadre	Variable de nœud	Usercut	Prétraitement	Matheuristic
$CPLEX_x$	×	×	×	×
$CPLEX_{xy}$	✓	×	×	×
BRT_x	×	✓	✓	✓
BRT_{xy}	✓	✓	✓	✓

Tableau 6.1: Options pour les algorithmes de coupes

Testset	Nombre	Nombre de nœuds	Capacité	Source	Densité	Classe des graphes
B	18	50-100	1-3	SteinLib	clairsemé	aléatoire
C	20	500	1-3	SteinLib	clairsemé	aléatoire
I320	20	320	1-3	SteinLib	surtout clairsemé	aléatoire
I640	16	640	1-3	SteinLib	surtout clairsemé	aléatoire
MSM	30	90-5181	1-3	SteinLib	clairsemé	grille
ran	110	10-499	1-4	génére	dense	aléatoire
rg_13	198	10-999	1-3	génére	clairsemé	aléatoire
rg_23	198	10-999	2-3	génére	clairsemé	aléatoire

Tableau 6.2: Options pour les instances

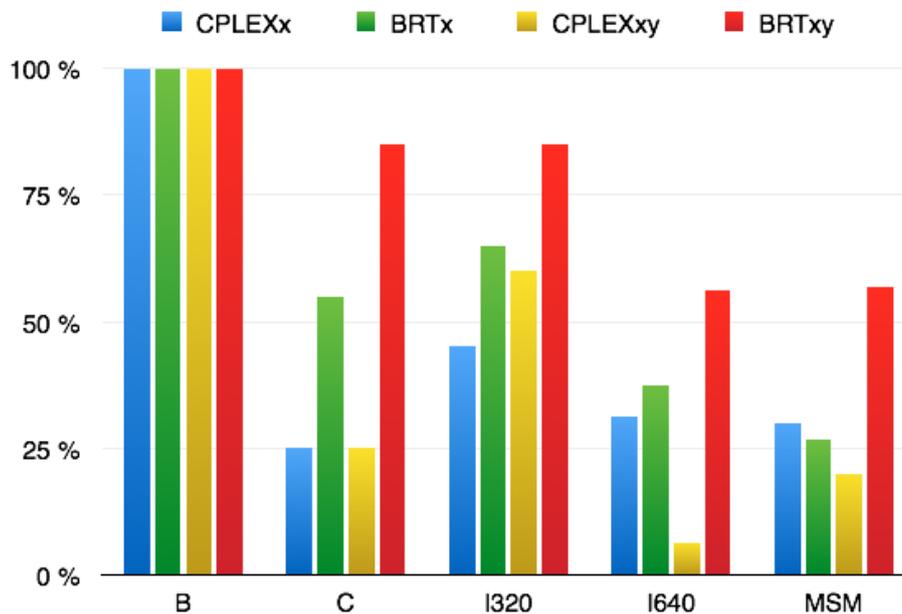


FIGURE 6.1 – Pourcentage d'instances résolues sur les testsets SteinLib

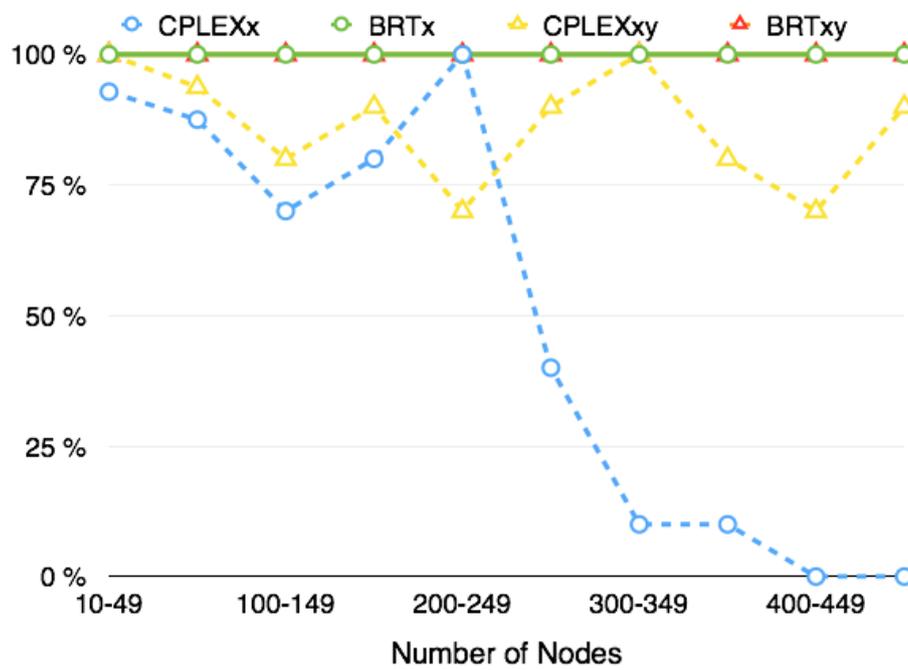


FIGURE 6.2 – Pourcentage d’instances résolues sur les graphes denses aléatoires

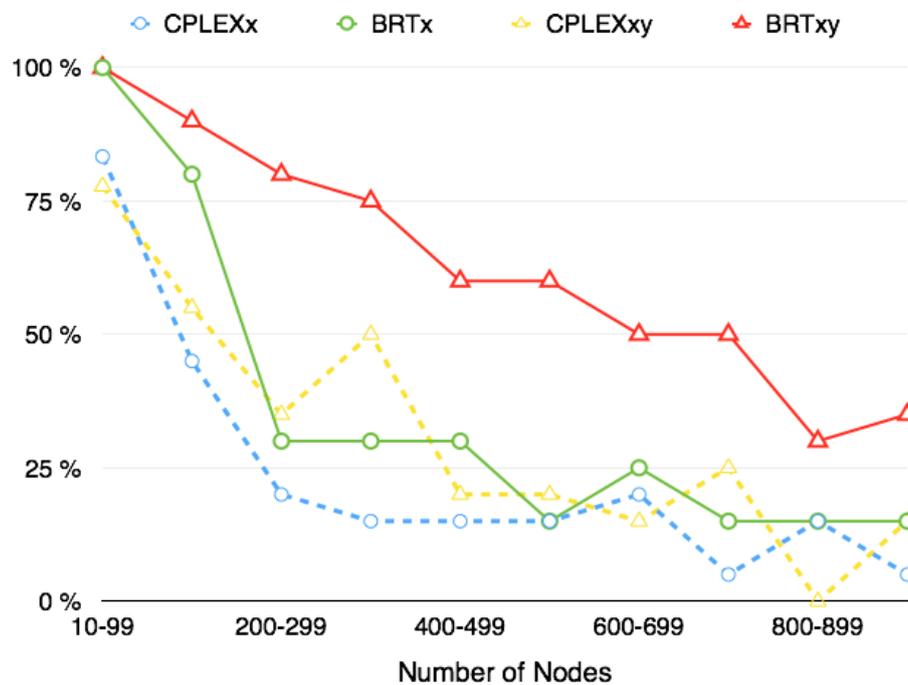


FIGURE 6.3 – Pourcentage d’instances résolues sur les graphes clairsemés aléatoires

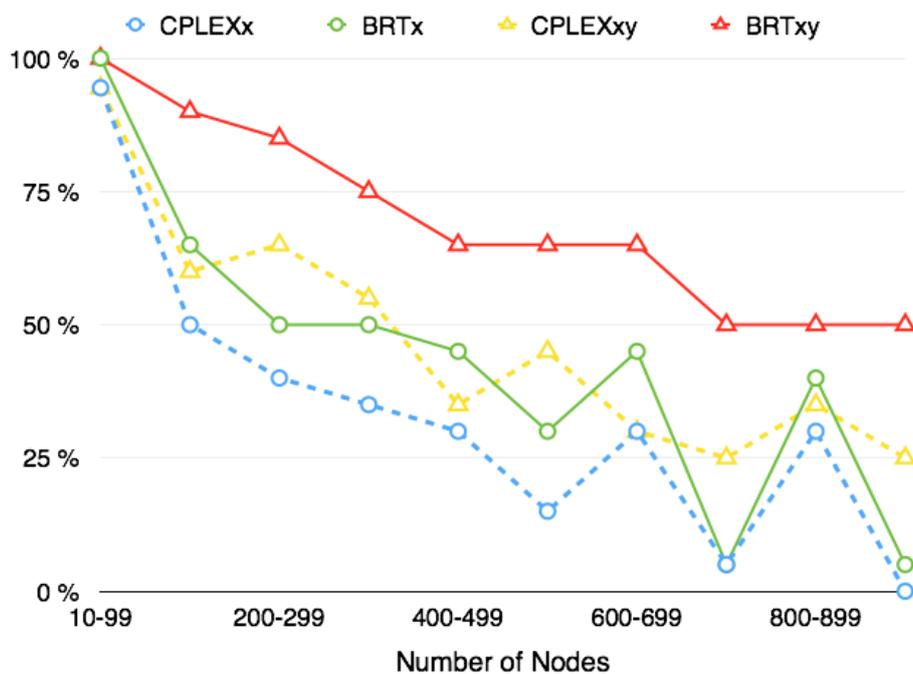


FIGURE 6.4 – Pourcentage d’instances résolues sur les graphes clairsemés aléatoires avec $O = \emptyset$

et la mesure de l’amélioration a été apportée en ajoutant les contraintes nouvellement introduites et les mathheuristiques aux formulations originales.

Conclusion

Dans cette thèse, nous avons abordé le problème $MBrT$ et le problème MrT à partir de trois directions, des algorithmes, des polyèdres et des études de calcul. Tout d'abord, nous avons proposé plusieurs algorithmes combinatoires en temps polynomial pour le problème $MBrT$ sur les arbres, les cycles et les cactus, respectivement. Ensuite, nous avons présenté des résultats concernant trois polyèdres, $B_{xy}(G, r, \mathbf{c})$, $R_x(G, r)$ et $B_x(G, r, \mathbf{c})$, y compris leur dimension, leur décomposition, les nouvelles contraintes, leurs facettes, leurs caractérisations sur certaines classes de graphes. Avec CPLEX, nous avons mis en place quatre cadres des algorithmes de coupe différents, à savoir $CPLEX_x$, $CPLEX_{xy}$, BRT_x et BRT_{xy} , en fonction des formulations pour $B_{xy}(G, r, \mathbf{c})$ et $B_x(G, r, \mathbf{c})$. Parmi les quatre cadres, $CPLEX_x$ et $CPLEX_{xy}$ correspondent aux formulations originales pour $B_x(G, r, \mathbf{c})$ et $B_{xy}(G, r, \mathbf{c})$ sans inclure les contraintes nouvellement introduites, tandis que BRT_x et BRT_{xy} correspondent aux formulations améliorées incorporant les contraintes nouvellement introduites ainsi qu'un prétraitement et une matheuristique. Le test de calcul a démontré de manière convaincante la force des formulations améliorées, en particulier la formulation améliorée pour $B_{xy}(G, r, \mathbf{c})$.

Certes, il reste beaucoup d'aspects inexplorés et des sujets potentiels de recherche du problème $MBrT$. Ensuite, nous énumérons quelques directions que nous estimons utiles et méritables d'être étudiées à l'avenir.

Pour l'aspect algorithmique, puisqu'il a été prouvé que le problème $MBrT$ peut être résolu en temps polynomial sur les cactus, on peut l'étendre à éventuellement d'autres classes de graphes tels que des graphes série parallèles et des graphes plans extérieurs. Alternativement, il vaut la peine d'essayer de concevoir des algorithmes pour certaines classes de graphes avec des structures simples et exploitables tels que les roues et les fans.

En outre, bien que la preuve d'intégralité du polyèdre défini par la formulation proposée pour $B_{xy}(G, r, \mathbf{c})$ soit donnée dans cette thèse, la totale dual-intégralité (TDI-ness, en anglais) de la formulation doit encore être prouvée. Nous avons obtenu quelques exemples montrant que l'insuffisance de la formulation actuelle était TDI. Ainsi, certaines inégalités redondantes devraient être incluses dans la formulation afin d'obtenir un système TDI. En outre, pour $B_x(G, r, \mathbf{c})$ et $R_x(G, r)$, en fonction de l'étude sur les arbres et les cycles,

il est tentant de prétendre qu'une description complète est disponible sur les cactus, ou dans une large mesure, sur les graphes série parallèles.

D'autre part, le problème $MBrT$ peut être étendu à une version plus généralisée, appelée le Problème d'emballage de r -arbre borné maximum ($MBrTP$, de l'anglais Maximum Bounded r -Tree Packing). Le problème $MBrTP$, au lieu de trouver un r -arbre, consiste à trouver un emballage de $K \geq 2$ r -arbres qui satisfait les contraintes de capacité globales. L'étude sur le problème de $MBrT$ peut être considérée comme la première étape de l'étude sur le problème de $MBrTP$. Il existe également des applications correspondantes du problème $MBrTP$ dans divers domaines, d'où il vaut la peine d'étudier ses algorithmes et sa structure polyédrique.

En fait, nous avons fait des recherches préliminaires sur la structure polyédrique et les tests de calcul pour le problème $MBrTP$ en tenant compte des variables sur les arêtes et les nœuds. Les résultats obtenus jusqu'à présent suggèrent que la structure polyédrique et la complexité computationnelle du problème $MBrTP$ sont beaucoup plus compliquées que le problème $MBrT$, même dans les cas vraisemblablement les plus faciles (e.g. $K = 2$, sur les étoiles).