The k-hop connected dominating set problem: approximation algorithms and hardness results

Rafael Santos Coelho

THESIS SUBMITTED TO

THE

INSTITUTO DE MATEMÁTICA E ESTATÍSTICA

AT

UNIVERSIDADE DE SÃO PAULO

FOR THE DEGREE

OF

DOCTOR OF SCIENCE

Doctoral program in Computer Science

Advisor: Dr. rer. nat. Yoshiko Wakabayashi

São Paulo, June 2017

The k-hop connected dominating set problem: approximation algorithms and hardness results

Rafael Santos Coelho

São Paulo, junho de 2017

The k-hop connected dominating set problem: approximation algorithms and hardness results

Esta versão da tese contém as correções e alterações sugeridas pela Comissão Julgadora durante a defesa da versão original do trabalho, realizada em 13/06/2017. Uma cópia da versão original está disponível no Instituto de Matemática e Estatística da Universidade de São Paulo.

Comissão Julgadora:

- Prof ª. Dra. Yoshiko Wakabayashi (orientadora) - IME-USP
- Prof. Dr. Aritanan Borges Garcia Gruber - UFABC
- Prof. Dr. Mário Leston Rey - UFABC
- Prof. Dr. Orlando Lee - UNICAMP
- Prof. Dr. Manoel Bezerra Campêlo Neto - UFC

Agradecimentos

ii

Resumo

COELHO, R. S. **The k-hop connected dominating set problem: approximation algorithms and hardness results**. 2017. 86 f. Tese (Doutorado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2017.

Seja G um grafo conexo e k um inteiro positivo. Um subconjunto D de vértices de G é um conjunto dominante conexo de k-saltos se o subgrafo de G induzido por D é conexo e se, para todo vértice v em G, existe um vértice u em D a uma distância não maior do que k de v. Estudamos neste trabalho o problema de se encontrar um conjunto dominante conexo de k-saltos com cardinalidade mínima (Min$k$-CDS).

Provamos que Min$k$-CDS é $\mathcal{NP}$-difícil em grafos planares bipartidos com grau máximo 4. Mostramos que Min$k$-CDS é $\mathcal{APX}$-completo em grafos bipartidos com grau máximo 4. Apresentamos limiares de inaproximabilidade para Min$k$-CDS para grafos bipartidos e $(1, 2)$-split, sendo que um desses é expresso em função de um parâmetro independente da ordem do grafo. Também discutimos a complexidade computacional do problema de se computar tal parâmetro.

No lado positivo, propomos um algoritmo de aproximação para Min$k$-CDS cuja razão de aproximação é melhor do que a que se conhecia para esse problema. Finalmente, quando $k = 1$, apresentamos dois novos algoritmos de aproximação para a versão do problema com pesos nos vértices, sendo que um deles restrito a classes de grafos com um número polinomial de separadores minimais. Além disso, discutimos uma formulação de programação linear inteira para essa versão do problema e provamos resultados poliédricos a respeito de algumas das desigualdades que constituem o politopo associado à formulação.

Palavras-chave: algoritmos de aproximação, complexidade computacional, conjunto dominante conexo de k-saltos, separador k-disruptivo minimal, limiar de inaproximabilidade, poliedro

iv

Abstract

COELHO, R. S. **The k-hop connected dominating set problem: approximation algorithms and hardness results**. 2017. 86 pp. Tese (Doutorado) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2017.

Let G be a connected graph and k be a positive integer. A vertex subset D of G is a k-hop connected dominating set if the subgraph of G induced by D is connected, and for every vertex v in G, there is a vertex u in D such that the distance between v and u in G is at most k. We study the problem of finding a minimum k-hop connected dominating set of a graph (Min$k$-CDS). We prove that Min$k$-CDS is $\mathcal{NP}$-hard on planar bipartite graphs of maximum degree 4. We also prove that Min$k$-CDS is $\mathcal{APX}$-complete on bipartite graphs of maximum degree 4. We present inapproximability thresholds for Min$k$-CDS on bipartite and on $(1, 2)$-split graphs. Interestingly, one of these thresholds is a parameter of the input graph which is not a function of its number of vertices. We also discuss the complexity of computing this graph parameter. On the positive side, we show an approximation algorithm for Min$k$-CDS. When $k = 1$, we present two new approximation algorithms for the weighted version of the problem, one of them restricted to graphs with a polynomially bounded number of minimal separators. Finally, also for the weighted variant of the problem where $k = 1$, we discuss an integer linear programming formulation and conduct a polyhedral study of its associated polytope.

Keywords: approximation algorithms, computational complexity, k-hop connected dominating set, k-disruptive separator, inapproximability threshold, polyhedra

Contents

List of Figures

Chapter 1

Introduction

A dominating set of a graph G is a vertex subset D such that, for every vertex v outside D, there is a vertex u in D adjacent to v. Graph domination, that is, the study of dominating sets, has become one of the most effervescent themes in graph theory. To get a sense of its explosive growth, in the past four years alone, Google Scholar registered thousands of publications on this topic; indeed, there have been even whole books devoted to it [HHS97, HHS98, DW12]. The success of the theory of graph domination is partly due to its widespread applicability. Throughout the years, dominating sets have been used to model problems arising in such diverse domains like social network theory, combinatorial game theory and bioinformatics [HHS98]. The history of graph domination stretches back to the early sixties, when the concept of a dominating set was first explicitly formulated by Ore [Ore62] and Berge [Ber62]. Since then, the subject of domination has branched out into countless variations such as independent domination, acyclic domination, total domination and paired domination just to name a few.

This thesis is centered around a particular type of domination, namely k-hop connected domination. Formally, a k-hop connected dominating set of a graph G, where k is a positive integer, is any vertex subset D that induces a connected subgraph of G and such that, for every vertex v of G, there is a vertex u in D within distance at most k from v, or, in other words, u is at most k edges or "hops" apart from v in G. From this point on, we

refer to 1-hop connected dominating sets shortly as connected dominating sets.

Introduced by Sampathkumar and Walikar [SW79], connected dominating sets received little attention in the beginning. At the "applied" end of the academic research spectrum, this situation started to change around the late nineties with the popularization of wireless networking. Unlike classical networks with a wired rigid infrastructure, wireless networks, by their very nature, lack a physical backbone to enforce some measure of topology control. To circumvent this problem, experts have come up with the idea of a **virtual backbone**, that is, a cluster of nodes in the network that work cooperatively in order to reduce overall routing costs and perform secondary tasks such as attenuating signal interference and monitoring battery power consumption (which is crucial, for example, in the case of sensor networks deployed in remote regions).

Broadly speaking, virtual backbones must satisfy three conditions: (i) every node *not* in the virtual backbone should reach some node in the virtual backbone through a short sequence of relay nodes; (ii) for every pair of nodes in the virtual backbone, there should be a path entirely contained *within* the virtual backbone linking them together, that is, the virtual backbone has to induce a connected "subnetwork" of the original network and (iii) virtual backbones should be small to minimize overhead. Unsurprisingly, k-hop connected dominating sets turned out to be an appropriate graph-theoretical abstraction for representing virtual backbones combinatorially. This conceptual bond between virtual backbones and k-hop connected dominating sets has spawned (and continues to do so) a vast body of application-oriented works, most of which thoroughly discussed in the surveys [BDTC05, YWWY13, YYSS15].

Shifting now the narrative towards the theoretical side of the story, connected dominating sets, as mathematical objects interesting in their own right, have also resonated within the combinatorial optimization community. When one phrases, in the language of graph theory, the challenge of locating virtual backbones in wireless networks, one is faced with the *minimum weight k-hop connected dominating set problem* (MINk-WCDS): given a vertex-weighted graph G, that is, a graph whose vertices have positive integers

(or weights) assigned to them, the objective is to find a minimum weight k-hop connected dominating set of G, that is, a k-hop connected dominating set D of G such that the sum of the weights of the vertices in D is minimum. We write MINk-CDS for the unweighted variant of the problem, and, when $k = 1$, we write simply MIN(W)CDS. Throughout the text, whenever the acronym MINk-(W)CDS is used in this manner, that is, without specific mention to the value of k, then k should be read as *any fixed* positive integer.

After the release of the first edition of Garey and Jonhson's seminal "Computers and Intractability" book [GJ79], there was a surge of papers [PLH83, BK85, WFP85, MB87, DM88, CCJ90, CS90, AR92, Kei93] dedicated to the investigation of the computational tractability and $\mathcal{NP}$-hardness of MINCDS on special classes of graphs such as bipartite graphs, co-comparability graphs, permutation graphs, planar graphs and so on. Subsequent to these developments, thanks to the consolidation of the field of approximation algorithms over the past decades, the approximability of MINCDS has been, time and time again, the target of numerous works [GK98, GK99, RDJ$^+$04, DH05, CC08, DGP$^+$08, GWZ$^+$10, CACdVK$^+$16]. Finally, in the last six years, some authors [SdCL11, FW12, GLdCS14] have also carried out computational experiments with integer linear programming (ILP) formulations for MINCDS.

A panoramic view, however, of the literature on MINk-CDS uncovers some discrepancies. Whereas MINk-CDS has been fertile ground for research in applied areas, there is a limited amount of papers that address MINk-CDS, when $k \geq 2$, from the approximability (especially in the general case) and computational hardness perspectives, as opposed to MINCDS. It is precisely this contrast that has motivated us to work on this problem. Specifically, we were driven mainly by the following questions:

1. Can we extend to MINk-CDS some of the insights that were proven fruitful in the design of approximation algorithms for MINCDS?

2. Does MINk-CDS become easier (in regard to its approximability) when restricted to graphs with small diameter (say $k + 1$)? (Note that MINk-CDS is trivial on graphs

with diameter at most k since, in such graphs, every single vertex is an optimal solution.)

3. Does the inapproximability threshold for MINk-CDS, under a reasonable complexity assumption, depend on k? To put it differently: does the approximation hardness of MINk-CDS vary when k grows?

4. Is MINk-CDS, on general graphs, amenable to a polyhedral study? In other words, can we, for instance, characterize all graphs G for which some natural associated polyhedron (say the convex hull of the incidence vectors of k-hop connected dominating sets of G) is full-dimensional? Can we find strong valid inequalities for this polyhedron and, furthermore, can we prove sufficient and necessary conditions for these inequalities to be facet-defining?

In what follows we present high-level answers to the aforementioned questions:

1. It appears that some of the techniques that have yielded approximation algorithms for MINCDS are not very easily adaptable to MINk-CDS when $k \geq 2$. Actually, even when these techniques do apply, the approximation algorithms that they produce seem to have poorer performance guarantees (in comparison with the approximations for MINCDS). In Chapter 5, we attempt to argue the reasons for that. On a more positive note, also in Chapter 5, we prove a sort of "meta-approximation" result which says that every approximation for MINCDS can be converted into an approximation for MINk-CDS, for every $k \geq 2$, at the expense of a worsened performance guarantee.

2. We answer this question for $k = 1$ in an asymptotic sense, that is, we prove, in Chapter 6, that the inapproximability threshold for MINCDS on graphs with diameter 2 is, asymptotically speaking, the same as the one for MINCDS on general graphs.

3. We show, in Chapter 6, that MINCDS and MINk-CDS for all $k \geq 2$ share the same inapproximability threshold, and it is independent of k.

4. We answer these questions partially for $k = 1$ in Chapter 4.

During the development of this work, we presented some of our results in conferences [CMW15, CW16] and, more recently, we had a paper [CMW17] accepted in the *Journal of Combinatorial Optimization*, already available online.

1.1   Organization of the thesis

The remainder of this thesis is organized as follows. In Chapter 2, we introduce most of the notation and terminology used throughout our work. In Chapter 3, we give an overview of the literature on MINk-CDS and we also list our contributions and discuss how they relate to some of the known results for the problem. We conduct a polyhedral study for MINCDS in Chapter 4. After that, in Chapter 5, we present approximation algorithms for MINk-CDS and, in Chapter 6, we compare them to some of our own results concerning approximation lower bounds. Lastly, in Chapter 7, we present concluding remarks and suggestions of avenues for future research.

Chapter 2

Preliminaries

In this chapter, we introduce basic notation and terminology used throughout the thesis. We gather them in separate sections according to the subject under discussion.

2.1 General notation and terminology

We assume that the reader is familiar with asymptotic notation and with some core concepts in computational complexity theory such as decision problems, polynomial-time reductions, polynomial-time computable functions, polynomial-time decidability of sets and well-known complexity classes. When in doubt, the reader may look into [GJ79, AB09].

We use mostly standard notation for set theory. Henceforth, the word set means any collection of distinct (that is, no repetitions allowed) elements. We denote by $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$ the sets of integer, rational and real numbers, respectively. Moreover, for every $A \subseteq \mathbb{R}$, we define $A_{>0} = A \setminus \{x \in A : x \leq 0\}$. We also define $A_{\geq 0} = A_{>0} \cup \{0\}$ and $\mathbb{B} = \{0, 1\}$. For every set W, $\mathcal{P}(W)$ denotes the **power set** of W, that is, the set of all subsets of W. For every real-valued function f defined on a set W and every non-empty finite subset $X \subseteq W$, we write $f(X)$ as a shorthand for $\sum_{a \in X} f(a)$. Finally, for every $l \in \mathbb{Z}_{>0}$, $\mathcal{H}(l)$ denotes the lth harmonic number, that is, $\mathcal{H}(l) = \sum_{i=1}^{l}(1/i)$.

From now on, the word minimal or maximal (with respect to a certain property) for a

set means minimal or maximal by inclusion. For every set W and every subset $C \subseteq \mathcal{P}(W)$, a **tranversal** of C is a subset $T \subseteq W$ such that T has non-empty intersection with every element of C.

2.2   Graph theory

In the course of this work, a **graph** G is an ordered pair (V, E), where V, a finite set, is called the **vertex set** of G and E, a set of two-element subsets of V, is called the **edge set** of G. Sometimes we also denote the vertex and edge sets of G by $V(G)$ and $E(G)$, respectively. The graph whose vertex set is empty is called the **null graph**. An edge between two vertices v and u is denoted by vu and we say that the edge vu is **incident** with v and u or that v and u are **adjacent**. We also say that v and u are the **endpoints** of vu. A graph H is said to be a **subgraph** of G, denoted by $H \subseteq G$, if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. In this case, we also say that G is a **supergraph** of H. We write $H = G$ if $H \subseteq G$ and $G \subseteq H$.

For any $X, Y \subseteq V(G)$, we define $E_G(X, Y) = \{vu \in E(G) : v \in X, u \in Y\}$. We also define $\delta_G(X) = E_G(X, V(G) \setminus X)$ and $E_G(X) = E_G(X, X)$. If the graph G is clear from context, we omit the subscript G when using $\delta_G(\cdot), E_G(\cdot, \cdot)$ and $E_G(\cdot)$. Furthermore, we write $G[X]$ to denote the **subgraph of G induced by** X, namely the graph $(X, E(X))$. We also say that X **induces** $G[X]$. We write $G - X$ to denote the graph produced by the removal of X from G, that is, $(V(G) \setminus X, E(G) \setminus (\delta(X) \cup E(X)))$. Similarly, for $F \subseteq E(G)$, $G - F$ denotes the graph $(V(G), E(G) \setminus F)$.

A **path** in a graph G is a sequence of distinct vertices such that every two consecutive vertices in the sequence are adjacent. A (v, u)-**path** is a path that starts at vertex v and ends at vertex u. The set of **internal vertices** of a (v, u)-path is the set that contains all the vertices in the path except v and u. We say that two or more paths are vertex-disjoint if they do not have vertices in common (except, possibly, the first and last vertices). A **cycle** is a closed path, that is, a path whose first and last vertices coincide. A **chord** in

a cycle is an edge joining two non-consecutive vertices in the cycle. A **chordless cycle** is a cycle without chords. An **induced cycle** is a chordless cycle that is induced in G by some vertex subset. The **length** of a path is equal to the length of its sequence minus 1.

A graph G is said to be **connected** if there is a path from v to u in G for any $v, u \in V(G)$. A graph is said to be **disconnected** if it is not connected. A **component** of G is a connected subgraph $H \subseteq G$ such that, for every connected $H' \subseteq G$, $H \subseteq H'$ implies $H = H'$. Note that a graph is connected if and only if it has exactly one component. Throughout this thesis, the null graph is considered to be disconnected, since it has zero components.

A **shortest path** between two vertices v and u in the same component of a graph G is a path of minimum length over all paths between v and u in G. For any two vertices v and u in G, we denote by $\mathrm{dist}_G(v, u)$ the length of a shortest path from v to u in G when there is one. If v and u belong to different components, then we define $\mathrm{dist}_G(v, u) = \infty$. When no confusion arises, we omit the subscript G in $\mathrm{dist}_G(\cdot, \cdot)$. The **diameter** of G, denoted by $\mathrm{diam}(G)$, is defined as $\max_{v, u \in V(G)} \mathrm{dist}(v, u)$.

Let $k \in \mathbb{Z}_{>0}$. We denote by G^k the **kth power** of G, which is defined as the graph with vertex set $V(G)$ where two vertices are adjacent if and only if they are within distance at most k in G, that is, $G^k = (V(G), \{vu : \mathrm{dist}_G(v, u) \leq k\})$. For every $v \in V(G)$, the **open k-neighborhood** (or **open neighborhood** when $k = 1$) of v in G, denoted by $N_G^k(v)$ (or $N_G(v)$ when $k = 1$), is defined as $\{u \in V(G) : \mathrm{dist}(v, u) \leq k\} \setminus \{v\}$. The **closed k-neighborhood** (or **closed neighborhood** when $k = 1$) of v in G, denoted by $N_G^k[v]$, is defined as $N_G^k(v) \cup \{v\}$. We extend these concepts to arbitrary non-empty subsets of vertices. For every non-empty $X \subseteq V(G)$, we define $N_G^k(X) = (\cup_{v \in X} N_G^k(v)) \setminus X$ and also $N_G^k[X] = N_G^k(X) \cup X$. For any $v, u \in V(G)$, if $v \in N_G^k(u)$, then we say that v is a **k-neighbor** (or **neighbor** when $k = 1$) of u. We say that a vertex v is **k-universal** (or simply **universal** if $k = 1$) in G if $N_G^k[v] = V(G)$. As usual, we drop the subscript G when the graph G is apparent.

A **k-hop dominating set** (k-DS for short or DS if $k = 1$) of a graph G is a set

$D \subseteq V(G)$ such that $N^k[D] = V(G)$. If D is a k-DS of G and $G[D]$ is connected, then we say that D is a k-***hop connected dominating set*** (k-CDS for short or CDS if $k = 1$) of G. From this point onwards, k denotes any fixed positive integer. For simplicity, if D is a (C)DS of G, then we call D a (connected) dominating set. For any undefined terminology and notation, we refer the reader to [Die06]. Along the text, various names of classes of graphs are mentioned and, in order to keep the presentation streamlined, we have decided to omit most of their definitions. If necessary, the reader should consult [BLS99, GYZ13] or the references cited in the following chapters.

2.3   Polyhedral theory

Unless otherwise stated, all vectors mentioned in this thesis are column real vectors. If v is a vector, then we denote by v^T the ***transpose*** of v. Sometimes we omit the superscript T that indicates the transpose of a vector to avoid cluttering the notation. Let $n \in \mathbb{Z}_{>0}$. Let us define $\mathbf{0}_n$ and $\mathbf{1}_n$ to be the vectors all of whose n entries are zeroes and ones, respectively. Sometimes we write $\mathbf{0}$ and $\mathbf{1}$ when n is implicit. We say that vectors $v^1, \ldots, v^l \in \mathbb{R}^n$ are ***affine-independent*** if the system of linear equations $\sum_{i=1}^{l} \lambda_i v^i = \mathbf{0}$ and $\sum_{i=1}^{l} \lambda_i = 0$ over $\mathbb{R}$, where $\lambda_1, \ldots, \lambda_l$ are the unknowns, has exactly one solution, namely $\lambda_i = 0$ for $i \in \{1, \ldots, l\}$. For every $v \in \mathbb{R}^n$ and $i \in \{1, \ldots, n\}$, we denote by v_i the ith entry of v. A ***convex combination*** of the vectors $v^1, \ldots, v^l$ is any vector expressed as $\sum_{i=1}^{l} \gamma_i v^i$ where $\gamma_1, \ldots, \gamma_l \in \mathbb{R}_{\geq 0}$ and $\sum_{i=1}^{l} \gamma_i = 1$.

A ***halfspace*** in $\mathbb{R}^n$ is a set of the form $\{x \in \mathbb{R}^n : a^T x \geq b\}$, where $a \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ and $b \in \mathbb{R}$. A set of the form $\{x \in \mathbb{R}^n : a^T x = b\}$ is called a ***hyperplane*** in $\mathbb{R}^n$. A ***polyhedron*** P in $\mathbb{R}^n$ is any set in the form $\{x \in \mathbb{R}^n : Dx \geq c\}$, where D is a matrix and c is a vector of appropriate dimensions. We say that a polyhedron P has ***dimension*** at least (at most) ℓ if there exists at least (at most) $\ell + 1$ affine-independent vectors in P. We denote the dimension of P by $\dim(P)$.

The ***convex hull*** of a finite set of vectors $W \subseteq \mathbb{R}^n$, denoted by $\mathrm{conv}(W)$, is the set

of all vectors in $\mathbb{R}^n$ that can be written as convex combinations of vectors in X. Notice that $\operatorname{conv}(W)$, for any finite set $W \subseteq \mathbb{R}^n$, is a polyhedron. Furthermore, we say that a polyhedron P is a **polytope** if there exists a finite set $W \subseteq \mathbb{R}^n$ such that $P = \operatorname{conv}(W)$.

Consider $\pi \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, $\pi_0 \in \mathbb{R}$ and a polyhedron P. An inequality $\pi^T x \geq \pi_0$ is said to be **valid** for P if $P \subseteq \{x \in \mathbb{R}^n : \pi^T x \geq \pi_0\}$. A vector $x^* \in \mathbb{R}^n$ **violates** the inequality $\pi^T x \geq \pi_0$ if $\pi^T x^* < \pi_0$. If $\pi^T x \geq \pi_0$ is valid for P, then a set of the form $\{x \in P : \pi^T x = \pi_0\}$ is said to be a **face** of P. In this case, we also say that $\pi^T x \geq \pi_0$ **induces a face** of P. Observe that every face of a polyhedron is also a polyhedron. A face F of P is said to be a **proper** face if $\emptyset \neq F \neq P$. Furthermore, F is called a **facet** if $\dim(F) = \dim(P) - 1$. In other words, a facet is a maximal proper face of P. Finally, if $\{x \in P : \pi^T x = \pi_0\}$ is a facet of P, then we say that the inequality $\pi^T x \geq \pi_0$ **defines** (or **induces**) a facet of P. For further details concerning polyhedral theory, the reader is referred to [Sch86, CCZ14].

2.4 Optimization problems

In this work, we address mainly optimization problems. An **optimization problem** Π is defined as a quadruple $(\mathcal{I}_\Pi, \mathcal{SOL}_\Pi, m_\Pi, \operatorname{goal}_\Pi)$, where:

1. $\mathcal{I}_\Pi$ is the set of **instances** of Π.

2. $\mathcal{SOL}_\Pi$ is the **feasible solution function** that associates to every $x \in \mathcal{I}_\Pi$ the set of feasible solutions $\mathcal{SOL}_\Pi(x)$ of x.

3. m_Π is the **measure function**, defined for pairs (x, y) such that $x \in \mathcal{I}_\Pi$ and $y \in \mathcal{SOL}_\Pi(x)$. For every such pair, $m_\Pi(x, y)$ is a positive integer which is the **value** of the feasible solution y of the instance x of Π.

4. $\operatorname{goal}_\Pi \in \{\max, \min\}$ determines if Π is a maximization or a minimization problem.

For every $x \in \mathcal{I}_\Pi$, we define $\operatorname{OPT}_\Pi(x) = \operatorname{goal}_\Pi\{m_\Pi(x, y) : y \in \mathcal{SOL}_\Pi(x)\}$. Throughout this thesis, we assume that m_Π is computable in polynomial time. Moreover, we also

assume that $\mathcal{I}_\Pi$ and $\mathcal{SOL}_\Pi(x)$, for every $x \in \mathcal{I}_\Pi$, are polynomial-time decidable sets.

To briefly illustrate how optimization problems can be formalized within the framework we have just described above, let us consider the *maximum weight clique problem* (MaxWClique). In what follows, we define the set of instances, the feasible solution function, the measure function and the goal of MaxWClique:

(i) $\mathcal{I}_{\text{MaxWClique}}$ is the set of all ordered pairs (G, w), where G is a graph and w is function that assigns positive integers to the vertices of G, that is, $w : V(G) \to \mathbb{Z}_{>0}$.

(ii) For every $(G, w) \in \mathcal{I}_{\text{MaxWClique}}$, $\mathcal{SOL}_{\text{MaxWClique}}(G, w)$ is the set of all *cliques* of G, that is, vertex subsets C of G such that $vu \in E(G)$ for every distinct $v, u \in C$.

(iii) For every $(G, w) \in \mathcal{I}_{\text{MaxWClique}}$ and every $C \in \mathcal{SOL}_{\text{MaxWClique}}(G, w)$, $m_{\text{MaxWClique}}((G, w), C)$ is defined as $w(C)$, that is, the sum of the weights of the vertices in C. Recall that $w(C) = \sum_{v \in C} w(v)$;

(iv) $\text{goal}_{\text{MaxWClique}} = \max$.

Henceforth we give all the definitions of optimization problems in a more straightforward fashion. Finally, it is important to notice that any optimization problem Π has an associated decision problem, say Π_D, which we call the ***decision version*** of Π. If Π is a minimization problem, then Π_D asks, given $\ell \in \mathbb{Z}_{>0}$ and $x \in \mathcal{I}_\Pi$, if there is $y \in \mathcal{SOL}_\Pi(x)$ such that $m_\Pi(x, y) \leq \ell$. Analogously, if Π is a maximization problem, then Π_D is the decision problem that asks, given $\ell \in \mathbb{Z}_{>0}$ and $x \in \mathcal{I}_\Pi$, if there is $y \in \mathcal{SOL}_\Pi(x)$ such that $m_\Pi(x, y) \geq \ell$.

2.5   Approximation algorithms

Let Π be an optimization problem. For every instance x of Π and every feasible solution y of x, the ***approximation factor*** (or ***performance guarantee***) of y, denoted by $R_\Pi(x, y)$, is defined as the maximum between $m_\Pi(x, y)/\text{OPT}_\Pi(x)$ and $\text{OPT}_\Pi(x)/m_\Pi(x, y)$.

We say that y is an r-approximate solution for x if $R_\Pi(x, y) \leq r$. An algorithm $\mathcal{A}$ is said to be an r-**approximation algorithm** (or an r-**approximation** for short) for Π if, for every input instance x of Π, $\mathcal{A}$ outputs an r-approximate solution $\mathcal{A}(x)$ of x in polynomial time (in the size of x). We should emphasize that r does not need to be a constant; actually, r can be a function of the size (or of any parameter) of the input instance. If r is indeed a constant, then we say that $\mathcal{A}$ is a constant approximation for Π.

A **polynomial-time approximation scheme** (PTAS for short) for a problem Π is a family of algorithms $\{\mathcal{A}_\varepsilon\}_{\varepsilon \in \mathbb{Q}_{>0}}$ such that for every *fixed* rational $\varepsilon > 0$, A_ε is a $(1 + \varepsilon)$-approximation for Π. We denote by $\mathcal{APX}$ the class of optimization problems (or the **approximation class**) that have a constant approximation.

We say that an optimization problem Π is $\mathcal{APX}$-**complete under L-reductions** (or simply that Π is $\mathcal{APX}$-**complete**) if Π belongs to $\mathcal{APX}$ and there is an L-reduction from every problem in $\mathcal{APX}$ to Π. An L-**reduction** from an optimization problem Π_A to another optimization problem Π_B is a quadruple (f, g, α, β), where f and g are polynomial-time computable functions (that is polynomial-time algorithms), and α and β are positive constants such that the following conditions hold:

(i) If I is an instance of Π_A, then $f(I)$ is an instance of Π_B.

(ii) $\mathrm{OPT}_{\Pi_B}(f(I)) \leq \alpha \, \mathrm{OPT}_{\Pi_A}(I)$ for every instance I of Π_A.

(iii) For every instance I of Π_A and every feasible solution S to $f(I)$ with objective value $m_{\Pi_B}(f(I), S)$, the algorithm g returns a solution $g(S)$ to I with objective value $m_{\Pi_A}(I, g(S))$ such that

$$|\mathrm{OPT}_{\Pi_A}(I) - m_{\Pi_A}(I, g(S))| \leq \beta \, |\mathrm{OPT}_{\Pi_B}(f(I)) - m_{\Pi_B}(f(I), S)|.$$

For more information regarding other approximation measures (besides the performance guarantee), other approximation classes and approximation-preserving reducibilities (besides L-reductions), we direct the reader to [Cre97, APMS$^+$99, CKST99].

Chapter 3

Literature review and contributions

In this chapter, we highlight some of the known results in the literature on MINk-CDS and mention the contributions of our work. To make the literature review part clearer, in the first three sections, we discuss positive results (by that we generally mean algorithms for polynomial-time solvable cases as well as approximation algorithms) for MINk-CDS, then we move on to negative (that is, $\mathcal{NP}$-hardness and inapproximability) results for the problem and, finally, we touch on works that deal with computational experiments to assess the strength and practicality of ILP formulations for MINk-CDS. Hereafter, let G be a (possibly vertex-weighted) n-vertex connected graph (that is, a connected graph with n vertices).

3.1 Polynomial cases and approximation algorithms

Many authors have devised efficient algorithms for MinCDS on restricted classes of graphs, such as distance-hereditary [DM88], permutation [CS90], doubly chordal [Mos93], strongly chordal [WFP85] and trapezoid graphs [Lia95]. MinWCDS, that is, the weighted counterpart of MinCDS, is known to be solvable in polynomial time on series-parallel [WFP85], interval [RR88], distance-hereditary [HGC98] and permutation graphs [AR92]. Furthermore, MINk-CDS can be solved efficiently on distance-hereditary graphs [BD98], HT-graphs [Dra93] and graphs with bounded treewidth [BL15]. The gist of these algo-

rithms usually boils down to a clever combination of structural insights, problem-tailored data structures and a direct application of general algorithmic design techniques, mainly dynamic programming or divide and conquer.

On the side of approximation algorithms for MIN(W)CDS and MINk-CDS, we discuss the following works, placing more emphasis on approximations for the general case. Starting with MINCDS, the first contributions, to the best of our knowledge, were given by Guha and Khuller [GK98]. Their ideas ultimately led to the development of two categories of approximation algorithms for MINCDS: local algorithms, that is, algorithms that expand a connected partial solution (according to some greedy criteria) until it becomes a dominating set of the input graph; and global algorithms, that is, algorithms that begin by finding a "good" dominating and possibly disconnected partial solution that is further augmented to induce a connected subgraph. Usually, local algorithms operate only "within the confines" of the open neighborhood (or sometimes of larger neighborhoods, say, the open 2-neighborhood) of the partial solution under construction. Due to this "shortsightedness" trait, local approximation algorithms tend to perform worse than global algorithms in terms of solution quality. Guha and Khuller [GK98] presented a local $2(\mathcal{H}(\Delta(G)) + 1)$-approximation and a global $(\ln \Delta(G) + 3)$-approximation for MINCDS.

Six years later, Ruan et al. [RDJ$^+$04] introduced a $(\ln \Delta(G) + 2)$-approximation for MINCDS. Ruan et al.'s work helped shape a new strand of approximation algorithms for the problem, which we refer to as potential-guided algorithms. Such algorithms gradually build a partial solution for the input graph by enlarging the empty set one vertex per iteration while seeking to greedily optimize a real-valued function defined for every vertex subset of the input graph, called the potential function. This function encodes how close to being a CDS a partial solution is in such a way that a partial solution corresponds to a CDS if and only if it is a "local optimum" (for some precise notion of local optimum) of the potential function. In each iteration, a vertex is chosen so as to optimize its marginal value (under the potential function) with respect to the partial solution built up to that point. The potential function must be carefully selected in order to fulfill a list of technical

side requirements. In essence, these requirements play two important roles. Firstly, they ensure solution feasibility when the algorithms halts (that is, when the algorithm finds a local optimum) and, secondly, they endow the potential function with a "submodular-like flavor", which comes in handy in the proof of the approximation factor.

Building on some of the ideas in Ruan et al.'s paper, Du et al. [DGP$^+$08] proposed another potential-guided approximation algorithm with a performance guarantee of $(1 + \varepsilon)(1 + \ln(\Delta(G) - 1))$, where $0 < \varepsilon \leq 1$ is a previously fixed constant. The main difference between Ruan et al. [RDJ$^+$04] and Du et al. [DGP$^+$08] algorithms is that while the former "grows" a partial solution one vertex at a time, the latter iteratively adds to the partial solution a non-empty vertex subset of the input graph with size at most $1/\varepsilon$ and with optimal weighted marginal value (under the potential function) relative to the current partial solution. In the context of Du et al.'s work, "weighted marginal value" simply means the total marginal value of the added vertex subset divided by its size. Finally, Du et al. [DGP$^+$08] prove the already mentioned approximation factor by using a much more involved and refined analysis.

After staying dormant for nearly 20 years, local approximation algorithms for MɪɴCDS have been recently revived by Khuller and Yang [KY16]. The work of these two authors was essentially motivated by the following question: how can the performance of local approximations for MɪɴCDS be enhanced if one allows them to use larger neighborhoods (that is, open t-neighborhoods for $t \geq 2$) and more sophisticated greedy criteria to expand the partial solution iteration after iteration? Improving on Guha and Khuller's paper [GK98], they managed to derive two new local approximations for the problem, one with a performance guarantee of $\mathcal{H}(2\Delta(G)+1)+1$ and another with a performance guarantee of $\mathcal{H}(\Delta(G)) + \sqrt{\mathcal{H}(\Delta(G))} + 1$. Lastly, we would also like to mention that Bonsma and Zickfield [BZ08] gave a 4/3-approximation for MɪɴCDS on cubic graphs.

For MɪɴWCDS, that is, when we allow vertices to have arbitrary positive integer weights, Guha and Khuller [GK98] proposed a $3\ln n$-approximation for general n-vertex graphs, which was then improved to a $(1.35+\varepsilon)\ln n$-approximation, for every fixed $\varepsilon > 0$,

by the same authors in [GK99].

To the extent of our knowledge, no approximation algorithms for MINk-CDS on general graphs have been proposed so far. Ren and Zhao [RZ11], however, presented an approximation algorithm for a generalization of MINk-CDS. In the *minimum connected set cover problem* (MINCSC), we are given a universe (a non-empty finite set) U, a subset $\mathcal{C} \subseteq \mathcal{P}(U)$, a graph $\mathcal{G}$ with vertex set $\mathcal{C}$, and the objective is to find $K \subseteq \mathcal{C}$ with minimum cardinality such that $\mathcal{G}[K]$ is connected and K covers U (that is, every element of U belongs to some element of K). Note that every instance of MINk-CDS can be efficiently reduced to an instance of MINCSC. For every graph G, set $U = V(G)$, $\mathcal{C} = \{N_G^k[v] : v \in V(G)\}$, and, finally, let $\mathcal{G}$ be the graph such that $V(\mathcal{G}) = \mathcal{C}$ and, for every distinct $u, v \in V(G)$, the vertices in $\mathcal{G}$ that correspond to $N_G^k[u]$ and $N_G^k[v]$ are adjacent if and only if $uv \in E(G)$. Note that G and $\mathcal{G}$ are isomorphic graphs; indeed, the only thing that differentiates them is how their respective vertex sets are labelled. Ren and Zhao's [RZ11] algorithm for MINCSC outputs a $D_c(\mathcal{G})(\mathcal{H}(\max_{X \in \mathcal{C}} |X| - 1) + 1)$-approximate solution, where $D_c(\mathcal{G})$ is the maximum distance in $\mathcal{G}$ taken over all pairs of vertices $X, Y \in \mathcal{C}$ such that $X \cap Y \neq \emptyset$.

Notice that Ren and Zhao's algorithm translates into a $2k(\mathcal{H}(\Delta(G^k)) + 1)$-approximation for MIN$k$-CDS. We should stress that Ren and Zhao's paper [RZ11] addresses MINCSC (not MINCDS), even though in its last section they argue that MINCSC generalizes MINCDS. (Actually, as mentioned earlier, MINCSC can be seen as a generalization of MINk-CDS for all k.)

Before closing this section, we should also point out that there is a substantial pool of works dealing with polynomial-time approximation schemes as well as distributed approximations for MIN(W)CDS and MINk-CDS on special classes of graphs. For a selection of these papers, we recommend the reader to go to [WAF02, CHL+03, DMP+05, DH05, NH06b, ZGWD09, GWZ+10, CACdVK+16, AdMRS17, JPD17] and other references therein.

3.2 $\mathcal{NP}$-hardness and inapproximability results

As for $\mathcal{NP}$-hardness results, MinCDS has been proven to be $\mathcal{NP}$-hard, for example, for split [WFP85], chordal bipartite [MB87], and planar bipartite graphs [WFP85]. Moreover, Nguyen and Huynh [NH06a] showed that Min$k$-CDS is $\mathcal{NP}$-hard on planar unit disk graphs of maximum degree 4 and Lokshtanov et al. [LMP$^+$13] proved that Min$k$-CDS is $\mathcal{NP}$-hard on graphs with diameter $k+1$. These are the strongest $\mathcal{NP}$-hardness results for Min$k$-CDS as far as we know. For further details concerning computational complexity results for MinCDS, we direct the reader to the excellent (but now somewhat outdated) book by Haynes, Hedetniemi and Slater [HHS98].

When it comes to approximation hardness results, Chlebík and Chlebíková [CC04] showed that, for every fixed $\varepsilon > 0$, there is no $(1 - \varepsilon) \ln n$-approximation algorithm for MinCDS on n-vertex bipartite and split graphs, unless $\mathcal{NP} \subseteq \text{DTIME}(n^{O(\log \log n)})$. Four years later, the same authors [CC08] proved that there exist constants $C > 0$ and $B_0 \geq 3$ such that, for every $B \geq B_0$, it is $\mathcal{NP}$-hard to approximate MinCDS to within a factor of $\ln B - C \ln \ln B$ on bipartite graphs with maximum degree at most B. Moreover, Bonsma [Bon12] proved that MinCDS is $\mathcal{APX}$-complete on cubic graphs. As far as we know, there have been no inapproximability results for Min$k$-CDS (that is, for general k) prior to this thesis.

3.3 Polyhedral results and computational experiments with ILP formulations

Although experimental papers related to ILP formulations for Min$k$-CDS fall out of the scope of this thesis, in the sequence, we very briefly mention some works in this line of research. In 2011, Simonetti, da Cunha and Lucena [SdCL11] proposed and tested an ILP formulation for MinWCDS with an exponential number of constraints. In 2012, compact formulations were introduced and empirically evaluated by Fan and Watson [FW12].

Three years ago, a hydrib approach combining a Benders' decomposition algorithm with a branch-and-cut method was developed by Gendron, Lucena, da Cunha and Simonetti [GLdCS14]. Gendron et al.'s experimental results, to the best of our knowledge, still stand out as the best ones overall (in terms of running time, integrality gaps and sizes of instances solved to optimality).

In regard to polyhedral results, we are not aware of any papers that directly address Min$k$-CDS (for general k or even for any particular value of k). Nevertheless, we must cite the work of Fujie [Fuj04], who conducted a polyhedral study for a problem computationally equivalent to MinCDS (in the sense of polynomial-time reductions) known as the *weighted maximum leaf spanning tree problem* (MaxWLST): given a vertex-weighted connected graph G, find a spanning tree of G such that the sum of the weights of its leaves is maximum. Notice that an n-vertex graph has a CDS with at most l vertices if and only if it has a spanning tree with at least $n - l$ leaves (and this can easily be generalized to a weighted setting). For a sample of papers concerning computational tests with ILP formulations for MaxWLST, we refer the reader to [Fuj03, LMS10, RLU15, GS17] and references therein.

3.4   Contributions

Our contributions advance the state of the art of the literature on Min$k$-CDS in three directions: approximation algorithms, inapproximability and polyhedral results for Min$k$-CDS and MinCDS.

Firstly, on the side of polyhedral results, we present, in Chapter 4, a polyhedral study related to a natural ILP formulation (with vertex-based decision variables only) for MinWCDS. We characterize the graphs for which the associated polytope (called the connected dominating set polytope) is full-dimensional, we discuss valid inequalities for this polytope and show necessary and sufficient conditions under which these inequalities are facet-inducing.

As for approximation algorithms, in Chapter 5, we prove a type of meta-approximation theorem which says that, for every graph G, an $f(G)$-approximation for MinCDS on G can be turned into a $kf(G^k)$-approximation for Min$k$-CDS on G. As a consequence, we derive an algorithm that finds, for every fixed $0 < \varepsilon \le 1$, a $k(1+\varepsilon)(1 + \ln(\Delta(G^k) - 1))$-approximation for Min$k$-CDS on G, which is an improvement (asymptotically by a factor of 2) on Ren and Zhao's $2k(\mathcal{H}(\Delta(G^k)) + 1)$-approximation [RZ11]. To the best of our knowledge, when $k \ge 2$, this approximation algorithm for Min$k$-WCDS on general graphs is the first to appear in the literature.

We also propose two approximation algorithms for MinWCDS, one of them restricted to special classes of graphs (to be formally defined in Chapter 5), namely graphs with a polynomial number of minimal separators (which are vertex subsets whose removal disconnects the graph). The first algorithm has an approximation factor which is logarithmic in the number of minimal separators of the input graph. Thus, we deem it more suitable for graphs with "few" separators (possibly graphs without induced long cycles, as will be discussed in greater length in Chapter 5).

The second approximation algorithm works on general graphs and it has a performance guarantee that depends on a parameter of the input graph, namely the cardinality of its largest minimal separator, which is independent of its order (in the sense that this parameter does not necessarily grow with the number of vertices of the graph). Therefore, this second algorithm seems to be more appropriate for graphs whose minimal separators contain a small number of vertices (say bounded by a constant). Interestingly, we show that, for certain classes of graphs, the approximation factor of this algorithm is close to the best one can hope for, assuming $\mathcal{P} \ne \mathcal{NP}$.

In regard to hardness results, we show, in Chapter 6, that MinCDS is $\Omega(\log n)$-hard to approximate even on n-vertex split graphs with diameter 2 if $\mathcal{NP} \not\subseteq \mathrm{DTIME}(n^{O(\log \log n)})$. Asymptotically, this threshold is the same as the one proved by Chlebík and Chlebíková [CC04] for MinCDS but it holds for the smaller class of split graphs of diameter 2 (surely the smallest value of the diameter for which the problem is non-trivial). Also in

Chapter 6, we prove that $\text{MIN}k\text{-CDS}$ is $\mathcal{NP}$-hard on planar bipartite graphs of maximum degree 4, thus extending and improving on a theorem of White, Farber and Pulleyblank [WFP85] that says that MINCDS is $\mathcal{NP}$-hard on planar bipartite graphs. Moreover, we present inapproximability thresholds for $\text{MIN}k\text{-CDS}$, generalizing the ones already known for MINCDS, on bipartite graphs and on a superclass of split graphs called $(1,2)$-split graphs. We also show that $\text{MIN}k\text{-CDS}$ is $\mathcal{APX}$-complete on bipartite graphs of maximum degree 4; this can be considered a step towards a generalization of Bonsma's [Bon12] result (who proved that MINCDS is $\mathcal{APX}$-complete on cubic graphs) for all k. We would like to emphasize that these results are shown to hold for every fixed k.

Chapter 4

Polyhedral results

In this chapter, we study MINWCDS from a polyhedral viewpoint. First, in Section 4.1, we present an ILP formulation for the problem, and, next, in Section 4.2, we single out the class of graphs for which the associated polyhedron (which we call the CDS polytope) is full-dimensional. Also in Section 4.2 we carry out an in-depth analysis of some valid inequalities for this polytope by showing necessary and sufficient conditions that make them facet-defining. Then, we discuss the separation problem for a class of valid inequalities, namely the separator inequalities. Throughout this chapter, if nothing is stated, G denotes a connected graph.

4.1 A vertex-based ILP formulation for MINWCDS

We first present some definitions. A ***separator*** of G is defined to be a vertex subset $S \subseteq V(G)$ such that $G - S$ is disconnected. A vertex $v \in V(G)$ is ***cut vertex*** if $\{v\}$ is a separator of G; if v is not a cut vertex, then v is called a ***non-cut vertex***. We denote by $\mathcal{S}(G), \mathcal{C}(G)$ and $\overline{\mathcal{C}}(G)$ the sets of all minimal separators, cut vertices and non-cut vertices of G, respectively. For every non-empty subset $A \subseteq \mathbb{R}$, we denote by $A^{V(G)}$ the set of vectors all of whose entries are indexed by the vertices of G and belong to A.

We now state a theorem proved by Kanté, Limouzy, Mary and Nourine [KLMN11], to be referenced many times, that characterizes CDSs of a graph G in terms of tranversals

of the set of all separators of G.

Theorem 4.1 (Kanté et al. [KLMN11]). *A vertex subset D of a connected graph G is a CDS if and only if $D \cap \Gamma \neq \emptyset$ for every separator Γ of G.*

Consider an instance (G, w) of MINWCDS. In what follows, we present an ILP formulation for MINWCDS (which we refer to as formulation $\mathcal{F}$) based on Kanté et al.'s result. For every vertex $v \in V(G)$, we assign a 0-1 decision variable x_v to v in such a way that $x_v = 1$ if and only if v is included in the solution.

$$(\mathcal{F}) \qquad \text{minimize} \quad \sum_{v \in V(G)} w(v) x_v$$

$$\text{subject to} \quad \sum_{v \in \Gamma} x_v \geq 1 \qquad \text{for every separator } \Gamma \text{ of } G \qquad (4.1)$$

$$x \in \mathbb{B}^{V(G)}$$

The inequalities (4.1) are called ***separator inequalities***. For every separator Γ of G, the separator inequality associated with Γ ensures that the solution contains at least one of the vertices in Γ. Therefore, by Theorem 4.1, we conclude that $\mathcal{F}$ is a valid ILP formulation for MINWCDS.

4.2 Studying the CDS polytope

For every $D \subseteq V(G)$, we define χ^D as the ***incidence vector*** of D, that is, the 0-1 vector in $\mathbb{B}^{V(G)}$ such that, for every $v \in V(G)$, $\chi_v^D = 1$ if and only if $v \in D$.

The ***connected dominating set polytope*** (or ***CDS polytope*** for short) associated with G, denoted by $P_{\text{CDS}}(G)$, is defined as

$$P_{\text{CDS}}(G) = \text{conv}(\{x \in \mathbb{B}^{V(G)} : x \text{ is the incidence vector of a CDS of } G\}).$$

First, we determine the dimension of $P_{\text{CDS}}(G)$.

Proposition 4.2. *Let G be an n-vertex connected graph. Then $dim(P_{\mathrm{CDS}}(G)) = n - |\mathcal{C}(G)|$. Therefore, $P_{\mathrm{CDS}}(G)$ is full-dimensional if and only if G is 2-connected.*

Proof. First, let us prove that $\dim(P_{\mathrm{CDS}}(G)) \leq n - |\mathcal{C}(G)|$. Note that if $v \in \mathcal{C}(G)$, then, by Theorem 4.1, we know that v belongs to every CDS of G, which implies that $\dim(P_{\mathrm{CDS}}(G)) \leq n - |\mathcal{C}(G)|$. Moreover, if $v \in \overline{\mathcal{C}}(G)$, then $V(G) \setminus \{v\}$ induces a connected subgraph of G and, therefore, $\chi^{V(G)\setminus\{v\}}$ is a CDS of G. That said, it is not too difficult to show that all the vectors in the set $\{\chi^{V(G)}\} \cup \{\chi^{V(G)\setminus\{v\}} : v \in \overline{\mathcal{C}}(G)\}$ are affine-independent and belong to $P_{\mathrm{CDS}}(G)$. Consequently, $\dim(P_{\mathrm{CDS}}(G)) \geq |\overline{\mathcal{C}}(G)| = n - |\mathcal{C}(G)|$.

Now let us show the second claim holds, namely that $P_{\mathrm{CDS}}(G)$ is full-dimensional if and only if G is 2-connected. If G is 2-connected, then $\mathcal{C}(G) = \emptyset$ and so $\dim(P_{\mathrm{CDS}}(G)) = n$. For the converse, if $P_{\mathrm{CDS}}(G)$ is full-dimensional, then G has no cut vertices. Hence, by definition, G is 2-connected. $\qquad\square$

From this point forward, in order to ease some of the proofs, we assume that we are dealing only with 2-connected graphs. In what follows, we present an argument to justify this assumption. In essence, we prove that MINWCDS restricted to 2-connected graphs is computationally equivalent to the same problem restricted to graphs with cut vertices.

Lemma 4.3. *If* MINWCDS *on 2-connected graphs is polynomial-time solvable, then* MINWCDS *is polynomial-time solvable on graphs with cut vertices.*

Proof. Suppose there is an efficient algorithm for solving MINWCDS on 2-connected graphs, say $\mathcal{A}$. We claim that there is also a polynomial-time algorithm that solves the same problem on graphs with cut vertices. Let (G, w) be an instance of MINWCDS and suppose that G has at least one cut vertex. Next we describe a reduction that takes (G, w) and produces another instance (H, w') of MINWCDS, where H is 2-connected, in such a way that any optimal solution for one of these instances can be efficiently turned into an optimal solution for the other. We define H as follows: $V(H) = V(G) \cup \{v^* : v \in V(G)\}$ and $E(H) = \{vv^* : v \in V(G)\} \cup \{vu^*, v^*u, vu, v^*u^* : vu \in E(G)\}$. We call the vertices of H that belong to $\{v^* : v \in V(G)\}$ star vertices. Lastly, $w'(v) = w'(v^*) = w(v)$ for every

$v \in V(G)$. It is easy to check that (H, w') can be constructed in polynomial time in the size of (G, w).

Now let us define the projection of a vertex in H onto a vertex in G. If $w \in V(H)$ is a star vertex of H, then, by construction, $w = v^*$ for some $v \in V(G)$ and, in this case, v is said to be the projection of w onto G. If w is not a star vertex of H, then w itself is its projection onto G. In the sequence, we argue that H is 2-connected. To do that, it suffices to show that $H - x$ is connected for every vertex $x \in V(H)$.

Claim 4.4. *H is 2-connected.*

Proof. Since, by the construction of H, $N_H(v) = N_H(v^*)$ for every $v \in V(G)$, then for any $x, a, b \in V(H)$, $a \neq x$ and $b \neq x$, if there is an (a, b)-path in H that goes through x, then there is an (a, b)-path in $H - x$. So H has no cut vertices. Therefore, H is 2-connected. □

Claim 4.5. *Let $D' \subseteq V(H)$ be a minimal CDS of H. Then there is a minimal CDS of H with exactly $|D'|$ vertices, none of which is a star vertex.*

Proof. Suppose that $v^* \in D'$ for some $v \in V(G)$. Note that $N_H(v) = N_H(v^*)$. Therefore, $H[D']$ is connected if and only if $H[(D' \setminus \{v^*\}) \cup \{v\}]$ is connected. Similarly, D' is a DS of H if and only if $(D' \setminus \{v^*\}) \cup \{v\}$ is a DS of H. Thus, $(D' \setminus \{v^*\}) \cup \{v\}$ is a minimal CDS of H with exactly $|D'|$ vertices and no star vertices. □

Note that any minimal CDS of G, by the construction of H, is also a minimal CDS of H. Conversely, since for every minimal CDS D of H, there is a minimal CDS of H with no star vertices and with size exactly $|D|$, it follows that $\text{OPT}_{\text{MINWCDS}}(G, w) = \text{OPT}_{\text{MINWCDS}}(H, w')$.

So, to conclude, if $\mathcal{A}$ is an efficient algorithm for solving MINWCDS on 2-connected graphs, then there is a polynomial-time algorithm $\mathcal{A}'$ for solving the same problem on graphs with cut vertices: given (G, w), run the reduction as described to obtain (H, w'), run $\mathcal{A}$ on (H, w') and then project the solution returned by $\mathcal{A}$ onto G, thus finding an

optimal CDS of (G, w). Note that $\mathcal{A}'$ always runs in polynomial-time and its correctness follows from all our previous claims and remarks. $\square$

We prove in the sequence the converse implication, that is, if there is an efficient way of solving MinWCDS on graphs with cut vertices, then there is also an efficient way of solving the same problem on 2-connected graphs. To accomplish that, we introduce a sort of intermediate problem, which is a variant of MinWCDS called the *minimum weight connected dominating set problem with a fixed vertex* (MinWCDSFV): given a vertex-weighted graph G and a vertex $u \in V(G)$, find a minimum weight CDS of G that contains u. The main idea of the proof is to show that MinWCDS on 2-connected graphs can be reduced to polynomially many instances of MinWCDSFV and, then, each of these instances of MinWCDSFV, in turn, can be reduced to an instance of MinWCDS on graphs with cut vertices.

Lemma 4.6. *If* MinWCDS *is polynomial-time solvable on graphs with cut vertices, then* MinWCDS *is polynomial-time solvable on* 2-*connected graphs.*

Proof. First we describe a reduction that maps an instance of MinWCDSFV, say (G, w, u), to an equivalent instance (H_u, w') of MinWCDS, where H_u is a graph with a cut vertex. Here we mean equivalent in the sense that any optimal solution to one of these instances yields, in polynomial time, an optimal solution to the other. Let H_u be the graph with vertex set $V(H_u) = V(G) \cup \{z_u\}$ and edge set $E(H_u) = E(G) \cup \{z_u u\}$. Now, we define $w'(v) = w(v)$ for every $v \in V(G)$ and $w'(z_u) = 0$. Observe that u is a cut vertex in H_u and that (H_u, w') can be built in polynomial time in the size of (G, w, u).

By Theorem 4.1, every CDS of H_u includes u. Additionally, it is easy to check that no minimal CDS of H_u contains z_u. Therefore, every minimal CDS of H_u is also a minimal CDS of G *that contains* u. Conversely, every minimal CDS of G that contains u is a minimal CDS of H_u. Consequently, $\text{OPT}_{\text{MinWCDS}}(H_u, w') = \text{OPT}_{\text{MinWCDSFV}}(G, w, u)$. Note that is also straightforward to check that, for every instance (G, w) of MinWCDS, $\text{OPT}_{\text{MinWCDS}}(G, w) = \min_{u \in V(G)} \text{OPT}_{\text{MinWCDSFV}}(G, w, u)$.

To finish the proof of the lemma, assume that there is a polynomial-time algorithm $\mathcal{A}$ that solves MinWCDS on graphs with cut vertices. Then there is a polynomial-time algorithm $\mathcal{A}'$ for solving MinWCDS on 2-connected graphs: given (G, w), for every vertex u in G, run the reduction as described to obtain (H_u, w'), then run $\mathcal{A}$ on (H_u, w') and return the solution with minimum weight *among all solutions found*. Note that $\mathcal{A}'$ always runs in polynomial time and its correctness follows from all our previous remarks. $\square$

Now we focus on some of the basic valid inequalities for the CDS polytope.

Proposition 4.7. *Let G be a 2-connected graph and v be a vertex in G. Then the following statements hold for $P_{\mathrm{CDS}}(G)$:*

(i) *The inequality $x_v \leq 1$ is facet-defining.*

(ii) *The inequality $x_v \geq 0$ is facet-defining if and only if there is no separator in G of size exactly 2 that contains v.*

Proof. Let G and v be as in the hypothesis.

(i) Let $\mathcal{F}_v^1 = \{x \in P_{\mathrm{CDS}}(G) : x_v = 1\}$. Clearly, $\dim(\mathcal{F}_v^1) \leq n - 1$. So now it suffices to show that there are n affine-independent vectors in $\mathcal{F}_v^1$. Indeed, since G is 2-connected, then $V(G) \setminus \{u\}$ is a CDS of G for every $u \in V(G) \setminus \{v\}$. So all the vectors in the set $\{\chi^{V(G)}\} \cup \{\chi^{V(G)\setminus\{u\}} : u \in V(G) \setminus \{v\}\}$ are CDSs of G, affine-independent and belong to $\mathcal{F}_v^1$. Therefore, $\dim(\mathcal{F}_v^1) = n - 1$.

(ii) Let $\mathcal{F}_v^0 = \{x \in P_{\mathrm{CDS}}(G) : x_v = 0\}$. Then, of course, $\dim(\mathcal{F}_v^0) \leq n-1$. Firstly, assume that there is no separator in G of size 2 containing v. In this case, it is not too difficult to check that all the vectors in the set $\{\chi^{V(G)\setminus\{v\}}\} \cup \{\chi^{V(G)\setminus\{v,u\}} : u \in V(G) \setminus \{v\}\}$ are CDSs of G, affine-independent and belong to $\mathcal{F}_v^0$ and, thus, $\dim(\mathcal{F}_v^0) \geq n - 1$. Conversely, suppose that the inequality $x_v \geq 0$ induces a facet of $P_{\mathrm{CDS}}(G)$. We claim that there can be no separators in G of size 2 containing v. In order to derive a contradiction, suppose not and let $\{u, v\}$ be a separator in G. Then $x_u + x_v \geq 1$ is a separator inequality for $P_{\mathrm{CDS}}(G)$. Therefore, since $x_u \leq 1$ is valid for the CDS

polytope of G, one may easily check that $\mathcal{F}_v^0 \subsetneq \{x \in P_{\mathrm{CDS}}(G) : x_u = 1\}$. Hence, $\mathcal{F}_v^0$ is not a facet of $P_{\mathrm{CDS}}(G)$, a contradiction.

$\square$

Before we present Theorem 4.11, the main result of this section, we first introduce some terminology. Let Γ be a separator of G. Then we say that a vertex $v \in \Gamma$ is a **_hanger_** in Γ if, for every component C of $G - \Gamma$, $N(v) \cap V(C) \neq \emptyset$. We call Γ is a (v, u)-**_separator_**, where $v, u \in V(G)$, if v and u belong to different components in $G - \Gamma$. A vertex $v \in V(G) \setminus \Gamma$ is Γ-**_critical_** if v is a cut vertex of C_v, where C_v denotes the component in $G - \Gamma$ that contains v. We say that Γ **_protects_** a Γ-critical vertex v if, for every component Q of $C_v - v$, there is a vertex $u \in \Gamma$ such that $N_G(u) \cap V(Q) \neq \emptyset$. Finally, a separator Γ is called **_safe_** if Γ protects every Γ-critical vertex in G.

In what follows, we state three auxiliary results which will turn out to be useful in the proof of Theorem 4.11, one of them is a folklore result that provides a sufficient condition for a valid inequality to be facet-inducing for a full-dimensional polyhedron.

Lemma 4.8. *Let G be a connected graph and Γ a separator of G. Then Γ is a minimal separator of G if and only if every vertex in Γ is a hanger.*

Proof. Let Γ be a minimal separator of a connected graph G and let C be one of the components in $G - \Gamma$. Consider $v \in \Gamma$ and $u \in V(C)$. Suppose, for the sake of contradiction, that $N(v) \cap V(C) = \emptyset$. Then $\Gamma \setminus \{v\}$ is a (u, v)-separator in G, which contradicts the minimality of Γ. Therefore, if Γ is minimal, every vertex in Γ is a hanger. Conversely, if every vertex in Γ is a hanger, then it is easy to check that no proper subset of Γ disconnects G and, thus, Γ is minimal.

$\square$

Lemma 4.9. *Let G be a connected graph, let $D \subseteq V(G)$ be a CDS of G, let Γ be an unsafe separator in G and let $v \in V(G) \setminus \Gamma$ be a Γ-critical vertex that is not protected by Γ. If $v \notin D$, then $|D \cap \Gamma| \geq 2$.*

Proof. Let G, Γ, D and v be as in the statement of the lemma. Recall that C_v denotes the component of $G - \Gamma$ that contains v. Since Γ is a separator of G, there is a vertex $u \in \Gamma \cap D$. Suppose, for the sake of contradiction, that $D \cap \Gamma = \{u\}$. Since, by hypothesis, Γ does not protect v, then there is a component Q of $C_v - \{v\}$ such that $N_G(u) \cap V(Q) = \emptyset$. We claim that $V(Q) \cap D = \emptyset$. Indeed, if that were not the case, since v is a cut vertex of C_v and $v \notin D$, there would be no path *entirely contained in* D connecting u to any vertex in $V(Q) \cap D$, a contradiction since $G[D]$ is connected. Thus, $V(Q) \cap D = \emptyset$. However, this implies that D is not a DS of G because none of the vertices in $V(Q)$ is dominated by D, again a contradiction. Therefore, we conclude that $|D \cap \Gamma| \geq 2$. □

Proposition 4.10. *Let P be a full-dimensional polyhedron and F be a face of P induced by the inequality $\pi x \geq \pi_0$, where $\pi \neq \mathbf{0}$. If, for every valid inequality $\pi' x \geq \pi'_0$ for P, where $\pi' \neq \mathbf{0}$, the inclusion $F \subseteq \{x \in P : \pi' x = \pi'_0\}$ implies that $\pi' = \eta\pi$ and $\pi'_0 = \eta\pi_0$ for some $\eta \in \mathbb{R}_{>0}$, then F is a facet of P.*

Theorem 4.11. *Let G be a 2-connected graph and Γ a separator of G. Then the separator inequality associated with Γ induces a facet of $P_{\text{CDS}}(G)$ if and only if the Γ is minimal and safe.*

Proof. Let G and Γ be as in the hypothesis. We prove first the necessity part of the theorem. So suppose that the separator inequality associated with Γ induces a facet of $P_{\text{CDS}}(G)$. Let $F_\Gamma = \{x \in P_{\text{CDS}}(G) : \chi^\Gamma x = 1\}$. Note that Γ must be minimal because, if not, then it is easy to check that $F_\Gamma \subsetneq \{x \in P_{\text{CDS}}(G) : \chi^{\Gamma'} x = 1\}$, where Γ' is a minimal separator of G strictly contained in Γ. But this contradicts the assumption that F_Γ is a facet of $P_{\text{CDS}}(G)$.

Now let us show that if F_Γ is a facet of $P_{\text{CDS}}(G)$, then Γ is a safe separator of G. In order to obtain a contradiction, suppose that that is not the case, and consider the following claim.

Claim 4.12. *There is a Γ-critical vertex $w \in V(G) \backslash \Gamma$ such that $\tilde{x}_w = 1$ for every extreme point $\tilde{x}$ of F_Γ.*

Proof. Since Γ is not safe, then there is a Γ-critical vertex $w \in V(G) \setminus \Gamma$ such that Γ does not protect w. Suppose the claim is false. Then there is an extreme point $\tilde{x}$ of F_Γ such that $\tilde{x}_w = 0$. Since $\tilde{x}$ is an extreme point of F_Γ and F_Γ is a facet of $P_{\mathrm{CDS}}(G)$, then $\tilde{x}$ is an extreme point of $P_{\mathrm{CDS}}(G)$. Thus there exists a CDS D of G such that $\tilde{x} = \chi^D$. Since w is a Γ-critical vertex of G that is not protected by Γ and $w \notin D$, then, by Lemma 4.9, we conclude that $|\Gamma \cap D| \geq 2$, which is a contradiction because $\tilde{x} = \chi^D \in F_\Gamma$. $\qquad\square$

As a consequence of Claim 4.12, we deduce $F_\Gamma \subsetneq \{x \in P_{\mathrm{CDS}}(G) : x_w = 1\}$ since $\chi^{V(G)} \in \{x \in P_{\mathrm{CDS}}(G) : x_w = 1\} \setminus F_\Gamma$ (recall that $|\Gamma| \geq 2$ because G is 2-connected by hypothesis) and, from our previous remarks, it is easy to conclude that $F_\Gamma \subseteq \{x \in P_{\mathrm{CDS}}(G) : x_w = 1\}$. Once again, this contradicts the assumption that F_Γ is a facet of $P_{\mathrm{CDS}}(G)$. Therefore, Γ is minimal and safe.

We have yet to prove the sufficiency part of the theorem, that is, that F_Γ is facet-defining for $P_{\mathrm{CDS}}(G)$ if Γ is a minimal and safe separator of G. So suppose that Γ is indeed minimal and safe. Consider a valid inequality $\pi'x \geq \pi'_0$ for $P_{\mathrm{CDS}}(G)$, where $\pi' \neq \mathbf{0}$. Let F' be the face of $P_{\mathrm{CDS}}(G)$ induced by $\pi'x \geq \pi'_0$ and suppose that $F_\Gamma \subseteq F'$. Our goal is to show that $\pi' = \eta\chi^\Gamma$ and $\pi'_0 = \eta$ for some $\eta \in \mathbb{R}_{>0}$. We begin by determining the entries of π'.

We claim that $\pi'_v = 0$ for every vertex $v \in V(G) \setminus \Gamma$. We split the proof of this claim in two cases, depending on whether v is or is not a Γ-critical vertex of G. First we address the Γ-non-critical vertices in $G - \Gamma$. Let us define $D_u = \{u\} \cup (V(G) \setminus \Gamma)$ for every $u \in \Gamma$. We know that Γ is minimal by hypothesis and so, for every Γ-non-critical vertex v in $G - \Gamma$, Lemma 4.8 ensures that D_u and $D_u \setminus \{v\}$, for any $u \in \Gamma$, are CDSs of G. Moreover, again for any $u \in \Gamma$ and any Γ-non-critical vertex v in $G - \Gamma$, both χ^{D_u} and $\chi^{D_u \setminus \{v\}}$ belong to F_Γ and also F', since F' contains F_Γ. Therefore, $\pi'_v = 0$, as claimed.

Now we prove that $\pi'_v = 0$ for every Γ-critical vertex v in $G - \Gamma$. Since Γ is safe by hypothesis, then, for every Γ-critical vertex v in $G - \Gamma$, there is a vertex $u \in \Gamma$, such that $\widehat{D}_v = \{u\} \cup (V(G) \setminus (\Gamma \cup \{v\}))$ is a CDS of G. So, for every Γ-critical vertex v in $G - \Gamma$, both $\chi^{\widehat{D}_v}$ and $\chi^{\widehat{D}_v \cup \{v\}}$ belong to F_Γ and F', since F' contains F_Γ. Thus, $\pi'_v = 0$.

Finally, we claim that $\pi'_v = \pi'_0$ for every $v \in \Gamma$. Indeed, since χ^{D_v} belongs to F_Γ and F' for any $v \in \Gamma$, it follows that $\pi'_v = \pi'_0$. Note that $\pi'_0 \neq 0$ since, by hypothesis, $\pi' \neq \mathbf{0}$. In fact, $\pi'_0 > 0$ given that $\pi' \chi^{V(G)} = |\Gamma| \pi'_0 \geq \pi'_0$ and $|\Gamma| \geq 2$. Therefore, $\pi' = \eta \chi^\Gamma$ and $\pi'_0 = \eta$ where $\eta = \pi'_0$. So, by Proposition 4.10, we conclude that F_Γ is a facet of $P_{\mathrm{CDS}}(G)$, and this ends the proof of the theorem.

$\square$

The next result guarantees that facet-defining separator inequalities for the CDS polytope are the only facet-defining inequalities with non-negative integer coefficients and right-hand side equal to 1.

Proposition 4.13. *Let G be a 2-connected graph and $\pi \in \mathbb{Z}_{\geq 0}^{V(G)}$. Suppose $\pi x \geq 1$ is a valid inequality for $P_{\mathrm{CDS}}(G)$. Then there exists a minimal separator Γ of G such that $\{x \in P_{\mathrm{CDS}}(G) : \pi x = 1\} \subseteq F_\Gamma$, where F_Γ is the face of $P_{\mathrm{CDS}}(G)$ induced by the inequality $\chi^\Gamma x \geq 1$.*

Proof. Define $S_\pi \subseteq V(G)$ as the vertex subset of G such that $v \in S_\pi$ if and only if $\pi_v > 0$. Of course, $S_\pi \neq \emptyset$. Moreover, note that S_π must be a separator of G because, if not, then by Theorem 4.1, $V(G) \setminus S_\pi$ would be a CDS of G and this would contradict the validity of the inequality $\pi x \geq 1$ since $\pi \chi^{V(G) \setminus S_\pi} = 0$. Now let Γ be a minimal separator of G such that $\Gamma \subseteq S_\pi$. It is not difficult to see that every point in $P_{\mathrm{CDS}}(G)$ that satisfies $\pi x \geq 1$ also satisfies the separator inequality associated with Γ. Therefore, $\{x \in P_{\mathrm{CDS}}(G) : \pi x = 1\} \subseteq F_\Gamma$, where F_Γ is the face of $P_{\mathrm{CDS}}(G)$ induced by Γ. $\square$

At this point, we would like to briefly discuss the computational complexity of *the separation problem associated with the separator inequalities*: given a graph G and a vector $\tilde{x} \in \mathbb{R}^{V(G)}$ such that $\mathbf{0} \leq \tilde{x} \leq \mathbf{1}$, decide if $\tilde{x}$ satisfies the separator inequalities for G or else find a separator of G, say Γ, such that $\chi^\Gamma \tilde{x} < 1$? By the celebrated result of Groetschel, Lovász and Schrijver [GLS12], that establishes the computational equivalence between separation and strong optimization problems over compact convex sets, it follows that the separation problem related to the separator inequalities reduces to the so-called

minimum weight separator problem (MINWS): given a graph with non-negative rational weights assigned to its vertices, find a minimum weight separator in it, that is, a separator such that the sum of the weights of its constituent vertices is minimum. In 2000, Henzinger, Rao and Gabow [HRG00] presented a strongly polynomial-time algorithm that solves MINWS in $O(\kappa_1 nm \log(n^2 m))$ time for any input vertex-weighted graph with n vertices, m edges and vertex-connectivity (that is, the size of a smallest separator in the graph) κ_1. Therefore, the next proposition follows immediately from our previous remarks.

Proposition 4.14. *The linear relaxation of formulation $\mathcal{F}$ can be solved in polynomial time*

Before concluding this section, we would like to make an important observation about some of the results proven here. Let $n, m \in \mathbb{Z}_>$. We say that two polyhedra, say $P \subseteq \mathbb{R}^n$ and $Q \subseteq \mathbb{R}^m$, are ***isomorphic*** if there are affine transformations $t_P : \mathbb{R}^n \to \mathbb{R}^m$ and $t_Q : \mathbb{R}^m \to \mathbb{R}^n$ such that $t_P(t_Q(x)) = x$ for every $x \in Q$ and $t_Q(t_P(y)) = y$ for every $y \in P$. If P and Q are isomorphic, it is possible to prove that there is a one-to-one correspondence between the extreme points of P and Q. As mentioned in Section 3.3 of Chapter 3, Fujie [Fuj04] published a paper (that has only come to our knowledge recently) presenting a polyhedral study for MAXWLST (the weighted maximum leaf spanning tree problem) wherein he gives a vertex-based ILP formulation for the problem and proves sufficient and necessary conditions under which some valid inequalities define facets of the associated polytope. By examining Fujie's work, it is straightforward to see that his ILP formulation with vertex-based decision variables for MAXWLST and formulation $\mathcal{F}$ for MINWCDS induce isomorphic polytopes.

Chapter 5

Approximation algorithms

In this chapter, we discuss approximation algorithms for $\textsc{Min}k$-CDS and $\textsc{Min}(\text{W})$CDS. In Section 5.1, we address a "meta-approximation" algorithm for $\textsc{Min}k$-CDS. Next, in Section 5.2, we discuss two approximations for $\textsc{Min}$WCDS, one of them restricted to special classes of graphs.

5.1 A "meta-approximation" for $\textsc{Min}k$-CDS

We show first a meta-approximation algorithm for $\textsc{Min}k$-CDS, namely algorithm $\textsc{ApproxMin}k$-CDS (see Algorithm 1). This algorithm uses (as a subroutine) an approximation algorithm, say $\mathcal{A}$, for $\textsc{Min}$CDS. Given an input graph G, $\textsc{ApproxMin}k$-CDS computes G^k, and then it runs $\mathcal{A}$ on G^k, thus finding a CDS, say D, of G^k. Finally, it connects the components of $G[D]$ (in case $G[D]$ is disconnected) by adding at most $(k-1)(|D|-1)$ extra vertices to D.

Theorem 5.1. *Let G be a graph. If there exists an algorithm for $\textsc{Min}$CDS with approximation factor $f(G)$, then, for every $k \in \mathbb{Z}_>$, there exists an approximation algorithm for $\textsc{Min}k$-CDS with approximation factor $kf(G^k)$.*

Proof. Let G be the input graph and let $\mathcal{A}$ be the polynomial-time $f(G)$-approximation algorithm for $\textsc{Min}$CDS (used in step 2). Clearly, $\textsc{ApproxMin}k$-CDS runs in polynomial

---

Algorithm 1 APPROXMINk-CDS

Input: A graph G

Subroutine: An approximation algorithm $\mathcal{A}$ for MINCDS

Output: An approximate k-CDS of G

1: Compute G^k ▷ the kth power of G
2: $D \leftarrow \mathcal{A}(G^k)$ ▷ $\mathcal{A}$ is an approximation algorithm for MINCDS
3: $S \leftarrow D$
4: **while** $G[S]$ is not connected **do**
5:     Take vertices v and u in different components of $G[S]$ such that $\mathrm{dist}_G(v, u)$ is minimum

6:     Compute a shortest (v, u)-path in G
7:     Let P be the set of internal vertices of the path obtained in line 6
8:     $S \leftarrow S \cup P$
9: **end while**
10: **return** S

---

time. Moreover, it produces a k-CDS of G.

Let $S \subseteq V(G)$ be the solution output by APPROXMINk-CDS. Since a k-CDS of G is a CDS of G^k, it follows that $\mathrm{OPT}_{\mathrm{MINCDS}}(G^k) \leq \mathrm{OPT}_{\mathrm{MIN}k\text{-}\mathrm{CDS}}(G)$. Thus, to complete the proof, it suffices to show that $|S| \leq k f(G^k) \mathrm{OPT}_{\mathrm{MINCDS}}(G^k)$. By hypothesis, we have that $|D| \leq f(G^k) \mathrm{OPT}_{\mathrm{MINCDS}}(G^k)$. If $G[D]$ is connected, then the proof is finished.

Assume now that $G[D]$ has at least two components. Let t be the number of iterations performed by the while loop in lines 4–9. Let $S_0 = D$ and, for every $j \in \{1, \ldots, t\}$, let S_j be the set S at the end of the jth iteration of the while loop. For every $j \in \{1, \ldots, t\}$, let u_j and v_j be the vertices chosen in the jth iteration in line 5, let C_{u_j} and C_{v_j} be the components of $G[S_{j-1}]$ that contain u_j and v_j, respectively, and let P_j be the set P chosen in line 7 in the jth iteration. Fix some $j \in \{1, \ldots, t\}$. Since $G^k[S_{j-1}]$ is connected, there is an edge of G^k with an endpoint, say w, in $V(C_{u_j})$ and an endpoint, say y, in some superset of $V(C_{v_j})$. Since y and w are adjacent in G^k, then y is within distance at most k from w in G and, thus, the same holds for u_j and v_j. Therefore, $|P_j| \leq k-1$ and $G[S_j]$ has at least one component less than $G[S_{j-1}]$. So we conclude that $t \leq |D| - 1$. Furthermore, we have $|S| \leq |D| + (k-1)(|D|-1) \leq k|D|$, and the result follows. $\square$

It is known that Du et al. [DGP$^+$08] proposed an approximation algorithm for MinCDS with ratio $(1 + \varepsilon)(1 + \ln(\Delta(G) - 1))$ for every fixed $0 < \varepsilon \leq 1$. The following is a consequence of Theorem 5.1 using Du et al.'s algorithm as a subroutine.

Corollary 5.2. *For every $k \in \mathbb{Z}_>$ and every fixed $0 < \varepsilon \leq 1$, there exists a polynomial-time algorithm for Min$k$-CDS with approximation ratio $k(1 + \varepsilon)(1 + \ln(\Delta(G^k) - 1))$, where G is the input graph.*

As mentioned in Chapter 3, Ren and Zhao [RZ11] presented an approximation algorithm for the minimum connected set cover problem (a generalization of Min$k$-CDS, as discussed earlier in Chapter 3) that, for every input graph G and every k, translates into a $2k(\mathcal{H}(\Delta(G^k)) + 1)$-approximation for Min$k$-CDS. Since $\ln(\ell - 1) < H(\ell)$ for every integer $\ell \geq 2$, Corollary 5.2 shows an improvement (asymptotically by a factor of 2) on Ren and Zhao's algorithm.

Observe that algorithm ApproxMin$k$-CDS indicates that Min$k$-CDS admits a constant approximation on bounded degree graphs (because if $\Delta(G)$ is bounded from above by a constant, then the same holds for $\Delta(G^k)$ for fixed k). The approximation factor of this algorithm follows from the previous corollary and can be expressed in terms of k and the degree bound.

Corollary 5.3. *For every $k \in \mathbb{Z}_>$, there is a constant approximation algorithm for Min$k$-CDS on bounded degree graphs.*

Before we go to the next section, we would like to offer some comments to address Question 1 that was posed at the end of Chapter 1. Recall that, in Section 3.1 of Chapter 3, we grouped all approximation algorithms that have been proposed so far for MinCDS in three classes, namely global, local and potential-guided algorithms. It is straightforward to see that ApproxMin$k$-CDS fits into the category of global approximations for Min$k$-CDS since it is an algorithm that begins by finding a good (but possibly disconnected) k-DS of the input graph and ends up adding more vertices into the solution to make it connected. We did succeed in proving a local approximation for Min$k$-CDS but

we chose not to include it in this work because its proven performance guarantee is much inferior to APPROXMINk-CDS (as is normally the case with local approximations when compared with global approximations). What we want to argue here is why we failed to design a potential-guided approximation for MINk-CDS.

Note that, for any graph G, if D is a CDS of G, then $G[D \cup S]$ is connected for *every* $S \subseteq V(G)$. This simple observation turns out to be instrumental in the proofs that appear in [RDJ+04] and [DGP+08], the only papers, as far as we know, that have proposed potential-guided algorithms for MINCDS. Unfortunately, this statement does not generalize when it comes to k-CDSs. To be more precise, it is *not* true that, for every k-CDS D of G and every $S \subseteq V(G)$, $G[D \cup S]$ is connected. Not only that, if $G[D \cup S]$ is indeed disconnected, it seems to be hard to (i) upper bound the number of components of $G[D \cup S]$ by some function of $|D|$ *or* (ii) to prove an upper bound strictly in terms of $|D|$ on the number of vertices that need to be added into the solution in order to make $G[D \cup S]$ a connected subgraph of G. These two technicalities are precisely what rendered all our struggles ineffective. Finally, it would be nice to see if one could either solve these issues (if not on general graphs, then perhaps on restricted classes of graphs) or if one could get passed them altogether by coming up with a totally different kind of algorithmic insight for an approximation for MINk-CDS.

5.2   Approximations for MIN(W)CDS

We turn now our attention to MINWCDS. Before we describe the approximation algorithms for MINWCDS, we define some concepts. Recall that a separator of a graph G is a subset $\Gamma \subseteq V(G)$ such that $G - \Gamma$ has more components than G. We say that a subset $\Gamma \subseteq V(G)$ is a k-***hop domination disruptive separator*** (or k-***disruptive separator*** for short) if Γ is a separator of G and, for every component C of $G - \Gamma$, $V(C)$ is not a k-CDS of G. In other words, Γ is a k-disruptive separator of G if and only if Γ intersects every minimal k-CDS of G. Note that a vertex subset of G is a 1-

disruptive separator of G if and only if it is a separator of G. We denote by $\mathcal{S}_k(G)$ (or simply $\mathcal{S}(G)$ when $k = 1$) the set of all minimal k-disruptive separators of a graph G. Let $\sigma_k(G) = \max_{\Gamma \in \mathcal{S}_k(G)} |\Gamma|$ and we write $\sigma(G)$ when $k = 1$.

We say that a class $\mathcal{C}$ of graphs *has a polynomial number of minimal k-disruptive separators* (or that $\mathcal{C}$ *has poly-k-separators* for short) if there exists a univariate polynomial p such that $|\mathcal{S}_k(G)| \leq p(n)$ for every n-vertex graph G belonging to $\mathcal{C}$. We also say that a graph G *has poly-k-separators* if G belongs to some class with poly-k-separators. For simplicity, when $k = 1$, we write poly-separators to refer to minimal poly-1-separators.

Many well-studied classes of graphs have poly-separators. Examples include chordal graphs [CG06], circular-arc graphs [Klo96], weakly chordal graphs [BT01], co-comparability graphs with bounded dimension [BLS99], $2K_2$-free graphs [DSM16] and P_4-sparse graphs [NP06] (for more examples, see [KK98]).

Now we discuss two approximation algorithms for MinWCDS. The first one (see Algorithm 2), called CoverApproxMinWCDS, works on classes of graphs with poly-separators. In order to explain how this algorithm functions, we have to define an auxiliary problem, namely the *minimum weight set cover problem* (MinWSC), which has the following description: given a universe U, a subset $\mathcal{C} \subseteq \mathcal{P}(U)$ and a weight function $w' : \mathcal{C} \to \mathbb{Z}_>$, the objetive is to find a minimum weight cover $K \subseteq \mathcal{C}$ of U.

In short, CoverApproxMinWCDS works as follows: firstly, it takes an instance (G, w) of MinWCDS on graphs with poly-separators and reduces it to an equivalent instance of MinWSC, in the sense that an optimal solution to one of these instances yields in polynomial time an optimal solution to the other; and, secondly, it runs on the resulting instance an approximation algorithm for MinWSC, proposed by Chvátal [Chv79], denoted here as MinWSC-Chvátal.

Before we delve into the details of CoverApproxMinWCDS, we need to make two remarks. First, we would like to mention that Berry, Bordat and Cogis [BBC99] designed an algorithm that, for every given n-vertex graph G, enumerates all minimal separators of G in $O(n^3)$ time per separator. In what fallows, we denote this algorithm by the acronym

BBC; it will be used as a subroutine in Algorithm 2. Second, we would like to, once more, draw attention to Theorem 4.1, stated in Chapter 4, which is a result, proven by Kanté et al. [KLMN11], that characterizes CDSs of graphs in terms of transversals of separators.

Algorithm 2 CoverApproxMinWCDS

Input: A graph G with poly-separators and a weight function $w : V(G) \to \mathbb{Z}_>$
Subroutine: MinWSC-Chvátal (an approximation for MinWSC) and algorithm BBC
Output: An approximate CDS of G
 1: Run algorithm BBC on G to compute $\mathcal{S}(G)$
 2: **for** all $v \in V(G)$ **do**
 3:     $\mathcal{F}_v \leftarrow \{\Gamma \in \mathcal{S}(G) : v \in \Gamma\}$
 4: **end for**
 5: $\mathcal{F} \leftarrow \{\mathcal{F}_v : v \in V(G)\}$
 6: Let $w' : \mathcal{F} \to \mathbb{Q}_>$ be the weight function s.t. $w'(\mathcal{F}_v) = w(v)$ for every $v \in V(G)$
 7: $K \leftarrow$ MinWSC-Chvátal$(\mathcal{S}(G), \mathcal{F}, w')$
 8: $S \leftarrow \{v \in V(G) \colon \mathcal{F}_v \in K\}$
 9: **return** S

Theorem 5.4. *Let (G, w) be an instance of* MinWCDS*, where G is a graph with poly-separators. Algorithm* CoverApproxMinWCDS *applied to (G, w) outputs an $\mathcal{H}(|\mathcal{S}(G)|)$-approximate solution in polynomial time.*

Proof. First, note that, since G has poly-separators by hypothesis, the algorithm always runs in polynomial time. Let $S \subseteq V(G)$ be the solution output by the algorithm. By construction, S is a transversal of $\mathcal{S}(G)$, and thus, by Theorem 4.1, S is a CDS of G. As proved in [Chv79], for any instance $(U, \mathcal{C}, w')$ of MinWSC, Chvátal's algorithm finds an $\mathcal{H}(\ell)$-approximate cover of U, where ℓ is the cardinality of the largest set in $\mathcal{C}$. Therefore, CoverApproxMinWCDS yields an $\mathcal{H}(|\mathcal{S}(G)|)$-approximate solution for MinWCDS on (G, w). □

Notice that most of the classes of graphs with poly-separators mentioned earlier (for example, chordal graphs, weakly chordal graphs, P_4-sparse graphs) exclude graphs with long induced cycles. That seems to be no coincidence. There have been a few papers

studying lower and upper bounds on the maximum number of minimal separators over all n-vertex graphs, a quantity usually denoted as $\text{sep}(n)$. Fomin, Kratsch, Todinca and Villanger [FKTV08] showed in 2008 that $\text{sep}(n) \in \Omega(1.4422^n) \cap O(1.7087^n)$ and, then, four years later, Fomin and Villanger [FV12] were able to improve the upper bound to $O(n1.6181^n)$. Recently, Gaspers and Mackenzie [GM15] found a simpler proof of this upper bound and they also managed to lift the lower bound to $\omega(1.4521^n)$. Interestingly, both of these lower bounds are achieved by artificially constructed families of n-vertex graphs either with *many* (usually $\Omega(n)$) "short" induced cycles (by short we mean containing exactly 4 vertices) or with few (usually $O(1)$) *very long* induced cycles (spanning a large fraction of the number of vertices). Therefore, we posit that CoverApproxMinWCDS may perform well, possibly better then Guha and Khuller's $(1.35 + \varepsilon) \ln n$-approximation for MinWCDS, on n-vertex graphs with relatively high edge density and small chordality (that is, graphs all of whose induced cycles contain at most some low constant number of vertices), like split graphs (which have fewer minimal separators than vertices).

Lastly, we discuss one more approximation algorithm for MinWCDS, which we call LPApproxMinWCDS (see Algorithm 3). Recall that, for every graph G, $\sigma_k(G)$ denotes the cardinality of the largest k-disruptive minimal separator of G. For simplicity, sometimes we write σ_k (and σ if $k = 1$) when the graph G is implicit or irrelevant. We prove that LPApproxMinWCDS is a σ-approximation for MinWCDS. Before we do that, we would like to stress one peculiar aspect of the approximation factor of this algorithm: not only is σ a parameter of the input graph independent of its size (because it does not necessarily grow as a function of the size of the graph; for instance, $\sigma(T) = 1$ for every tree T and $\sigma(C) = 2$ for every cycle C), but also, as will be shown next, computing σ (actually, computing σ_k for any fixed k) is an NP-hard problem.

Theorem 5.5. *For every $k \in \mathbb{Z}_>$, it is NP-hard to compute $\sigma_k(G)$ even for planar bipartite graphs G with maximum degree 4.*

Proof. The *maximum doubly connected cut problem* (MaxDCC) consists in finding in a connected graph G a ***doubly connected cut*** $(S, \overline{S})$ of maximum cardinality, that is, a

partition $(S, \overline{S})$ of $V(G)$ that maximizes the number of **crossing edges** (which are edges in $E(S, \overline{S})$) and such that $G[S]$ and $G[\overline{S}]$ are connected.

Haglin and Venkatesan [HSV91] showed that the decision version of MAXDCC is $\mathcal{NP}$-complete on 3-connected cubic planar graphs. We present a reduction from MAXDCC to the decision version of our problem, which asks, given a graph G and an integer q, if G has a minimal k-disruptive separator containing at least q vertices.

Let G be a 3-connected cubic planar graph. We build from G a graph G' as follows. First, we take G and subdivide each of its edges exactly once. For every $e \in E(G)$, we denote by w_e the vertex of the subdivision of edge e. For every v, we take a new path P_v (disjoint from the subdivision of G), with initial vertex u_v and with k vertices. Then, for every $v \in V(G)$, we add the edge $u_v v$ (that is, we append the path P_v to v). Therefore, $V(G') = V(G) \cup \{w_e \colon E(G)\} \cup (\cup_{v \in V(G)} V(P_v))$ and so G' has $|V(G)| + |E(G)| + k\,|V(G)|$ vertices. The construction of G' is depicted in Figure 5.1. Clearly, G' can be constructed in time polynomial in the size of G. Furthermore, G' is planar, bipartite and it has maximum degree 4.
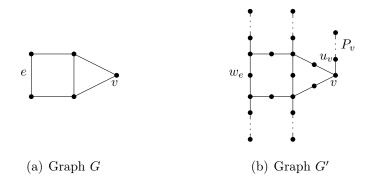

(a) Graph G (b) Graph G'

Figure 5.1: *Graph G' obtained from G with the construction described in the proof of Theorem 5.5.*

Observe that each vertex v in $V(G)$ is a cut vertex of G'. Thus every minimal k-disruptive separator in G' that contains any vertex of $V(G)$ has size exactly 1. Moreover, for every v in $V(G)$, it is clear that no minimal k-CDS of G' intersects $V(P_v)$. Consequently, every minimal k-disruptive separator of G' of size larger than 1 is contained in $\{w_e \colon e \in E(G)\}$ (remember that a vertex subset is a minimal k-disruptive separator if

and only if it is minimal tranversal of the set of all minimal k-CDSs). It is easy now to see that $(S, \overline{S})$ is a doubly connected cut in G if and only if $\{w_f \in V(G') : f \in E(S, \overline{S})\}$ is a minimal k-disruptive separator of G'. Therefore, G has a doubly connected cut with at least q crossing edges if and only if G' has a minimal k-disruptive separator of size at least q. $\square$

Now we can address LPApproxMinWCDS (shown below). In essence, Algorithm 3 is an LP-rounding procedure that works as follows: given an instance (G, w) of MinWCDS, LPApproxMinWCDS efficiently solves the linear relaxation of formulation $\mathcal{F}$ on (G, w) (discussed in Section 4.1 of Chapter 4) using a separation routine for the separator inequalities (namely, an algorithm by Henzinger, Rao and Gabow [HRG00], henceforth denoted as HRG, mentioned in Section 4.2 of Chapter 4).

Two key observations have to be made: (i) each time the separation oracle is called, only minimal violated cuts (that is, violated separator inequalities associated with minimal separators) are added into the cut pool and (ii) we use an auxiliary variable to store the size, say t, of the current largest violated cut added up to that point (that is, the size of the largest minimal separator returned by the separation oracle). Once an optimal solution, say $x^* \in \mathbb{R}^{V(G)}$, for the linear relaxation of $\mathcal{F}$ on (G, w) has been computed, the algorithm repeatedly tries to round up and down the fractional coordinates in x^* until a CDS of G is found and it does so by carefully increasing t (and, thus, decreasing $1/t$, known as the **rounding threshold**).

Theorem 5.6. *Let (G, w) be an instance of* MinWCDS. *Algorithm* LPApproxMin-WCDS *outputs, in polynomial time, a $\sigma(G)$-approximate solution for* MinWCDS *on* (G, w).

Proof. Consider an instance (G, w) of MinWCDS. Let $x^* \in \mathbb{R}^{V(G)}$ be the solution of the linear relaxation of $\mathcal{F}$ found in line 1 and let t be as indicated in line 2. Moreover, let us define, for every integer $q \in \mathbb{Z}_>$, the vertex subset $D_q = \{v \in V(G) : x_v^* \geq 1/q\}$. We say that $1/q$ is a **valid** rounding threshold for (G, w) and x^* if D_q is a CDS of G. First we

---

Algorithm 3 LPApproxMinWCDS

Input: A graph G and a weight function $w : V(G) \to \mathbb{Z}_>$

Subroutine: The HRG algorithm

Output: An approximate CDS of (G, w)

1: Let $x^* \in \mathbb{R}^{V(G)}$ be an optimal solution to the linear program

$$\min \left\{ \sum_{v \in V(G)} w(v) x_v : \sum_{v \in \Gamma} x_v \geq 1 \text{ for all separators } \Gamma \text{ in } G, \mathbf{0} \leq x \leq \mathbf{1} \right\}$$

found by using the HRG algorithm as the separation routine

2: Let t be the size of the largest violated cut added in line 1

3: $S \leftarrow \emptyset$

4: **repeat**   ▷ The algorithm tries to find a valid rounding threshold

5:     $S \leftarrow \{v \in V(G) : x_v^* \geq 1/t\}$

6:     $t \leftarrow t + 1$

7: **until** S is not a CDS of G

8: **return** S

---

prove that every valid rounding threshold yields an approximate solution for MinWCDS on (G, w).

Claim 5.7. *If $1/q$ is a valid rounding threshold for (G, w) and x^*, then D_q is a q-approximate solution for* MinWCDS *on (G, w).*

Proof. Suppose $1/q$ is a valid rounding threshold for (G, w) and x^*. Let OPT* be the objective value of x^*, that is, OPT$^* = \sum_{v \in V(G)} x_v^* w(v)$. By the definition of D_q and x^*, we have that $w(D_q) = \sum_{v \in D_q} w(v) \leq q \sum_{v \in D_q} x_v^* w(v) \leq q \sum_{v \in V(G)} x_v^* w(v) = q \, \mathrm{OPT}^*$. Hence, the claim follows since OPT$^* \leq \mathrm{OPT}_{\mathrm{MinWCDS}}(G, w)$.   □

Claim 5.8. *The value $1/\sigma(G)$ is a valid rounding threshold for (G, w) and x^*.*

Proof. As mentioned earlier, recall that only minimal violated cuts are added into the cut pool in line 1. Therefore, for every $\Gamma \in \mathcal{S}(G)$ (recall that $\mathcal{S}(G)$ is the set of all minimal separators of G), we have that $\chi^\Gamma x^* \geq 1$. Actually, for every $\Gamma \in \mathcal{S}(G)$, note that

$$\max\{x_v^* : v \in \Gamma\} \geq 1/|\Gamma| \geq 1/\sigma(G).$$

Thus, $D_{\sigma(G)}$ is a transversal of the set of all minimal separators (hence all separators) of

G. Then, by Theorem 4.1, we conclude that $D_{\sigma(G)}$ is a CDS of G. Consequently, $1/\sigma(G)$ is a valid rounding threshold for (G, w) and x^*. □

By the definition of variable t in line 2, we have that $t \leq \sigma(G)$. By Claim 5.8, the value of t never goes above $\sigma(G)$ during the execution of the repeat-until loop in lines 4–7, thus we conclude that the algorithm always runs in polynomial time and, furthermore, Claim 5.7 ensures that the returned solution S has an approximation factor at most $\sigma(G)$, that is, $w(S) \leq \sigma(G)\mathrm{OPT}_{\mathrm{MinWCDS}}(G, w)$.

□

We discuss now why we were not able to extend Theorem 5.6 to MINk-WCDS, that is, what prevented us from deriving, say, a σ_k-approximation for MINk-WCDS.

Taking a closer look at algorithm LPApproxMinWCDS, one can say that its most vital component is the polynomial-time solvability of the linear program in line 1, which is guaranteed by the fact that the separation problem associated with the separator inequalities admits an efficient separation procedure. A very natural generalization of this linear program would be as follows: given an instance (G, w) of MINk-WCDS, find a vector $x \in \mathbb{R}^n$, where $\mathbf{0} \leq x \leq \mathbf{1}$, that minimizes the objective function $\sum_{v \in V(G)} x_v w(v)$ subject to the constraints $\sum_{v \in \Gamma} x_v \geq 1$ imposed on all k-disruptive separator Γ in G. Let us call such constraints the ***k-disruptive separator inequalities***. Clearly, if the separation problem related to the k-disruptive separator inequalities were computationally tractable, then the polynomial-time solvability of the "generalized version" of the linear program in line 1 would easily follow.

So here we get to the crux of matter: perhaps it is not the case that k-disruptive separator inequalities *can* be separated efficiently and, of course, that comes down to the fact that computing minimum weight k-disruptive separators in general vertex-weighted graphs appears to be an $\mathcal{NP}$-hard problem. We tried to prove the $\mathcal{NP}$-hardness of the so-called *minimum weight k-disruptive separator problem* (MINk-WS), but we were not successful, and this prompts us to state the following conjecture:

Conjecture 5.9. *For every $k \in \mathbb{Z}_>$, if $k \geq 2$, then MINk-WS is an $\mathcal{NP}$-hard problem on general graphs.*

Another intriguing matter related to MINk-WS is the very combinatorial structure of minimal k-disruptive separators, which seems to be much more complex, we think, when compared to "regular" minimal separators (that is, 1-disruptive separators). At first, one is tempted to prove some kind of relation between the sets $\mathcal{S}_k(G)$, $\mathcal{S}(G)$ and $\mathcal{S}(G^k)$ that holds for every graph G. We attempted to show that $\mathcal{S}_k(G) \subseteq \mathcal{S}(G) \cup \mathcal{S}(G^k)$. However, as illustrated in Figure 5.2, already for $k = 2$, such inclusion is false. It is straightforward to check, in Figure 5.2, that the set of vertices $S = \{x, y, z\}$ is a minimal 2-disruptive separator and *not* a minimal separator because it properly contains a separator, for instance $S' = \{y, z\}$. Moreover, one can check that the vertices x, y, z do not form a separator in the square of the graph depicted in Figure 5.2. It would be interesting to find out, for general graphs or even for particular classes of graphs, how the set of all minimal k-disruptive separators, when $k \geq 2$, "interacts" with some other graph parameters.
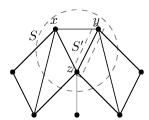

Figure 5.2: *The vertex subset $S = \{x, y, z\}$ is a minimal 2-disruptive separator but it is not a minimal separator.*

Chapter 6

Hardness results

In this chapter, we address hardness results for $\textsc{Min}k$-CDS. In Section 6.1, we begin by proving some inapproximability results for the "classic" variant of the problem where $k = 1$. Then, in Section 6.2, we focus on $\textsc{Min}k$-CDS.

6.1 Hardness results for $\textsc{MinCDS}$

In this section, we strengthen some known results in the literature on $\textsc{MinCDS}$ concerning hardness of approximation. Before we state and prove them, we introduce some notation and terminology. For any graph G, we denote by $\alpha(G)$ and $\omega(G)$ the ***stability number*** (the size of the largest stable set) and the ***clique number*** (the size of the largest clique) of G, respectively. We say that G is a ***split graph*** if $V(G)$ can be partitioned into two sets, say K and S, such that K is a clique and S is a stable set in G. Such a partition (K, S) is called a ***split partition*** of G. We point out that Heggernes and Kratsch [HK07] designed a $O(n)$ algorithm (henceforth called the HK algorithm) that finds a split partition for any given n-vertex split graph.

We now discuss the main results of this section. Firstly, we show that $\textsc{MinCDS}$ is $\Omega(\log n)$-hard to approximate even on n-vertex split graphs with diameter 2, assuming $\mathcal{NP} \not\subseteq \mathrm{DTIME}(n^{O(\log \log n)})$. Our proof is based on an $\mathcal{NP}$-hardness result given by Lokshtanov et al. [LMP$^+$13] for $\textsc{MinCDS}$ on split graphs with diameter 2. We show that

the polynomial-time reduction discussed in [LMP$^+$13] can actually be leveraged to prove hardness of approximation for MINCDS on split graphs with diameter 2. First, we present two theorems, for future reference, and a support lemma.

Theorem 6.1 (Chlebík and Chlebíková [CC04]). *For every fixed $\varepsilon > 0$, MINCDS cannot be approximated to within a factor of $(1 - \varepsilon) \ln n$ on n-vertex split graphs and on bipartite graphs, unless $\mathcal{NP} \subseteq \mathrm{DTIME}(n^{O(\log \log n)})$.*

Theorem 6.2 (Golumbic [Gol04], Theorem 6.2). *Let G be a split graph and let (K, S) be a split partition of G. If K is a maximum clique and S is a maximum stable set in G, then (K, S) is the unique split partition of G.*

Lemma 6.3. *For every fixed $\varepsilon > 0$, MINCDS cannot be approximated to within a factor of $(1 - \varepsilon) \ln n$ on n-vertex split graphs with a unique split partition having both sides of even cardinality, unless $\mathcal{NP} \subseteq \mathrm{DTIME}(n^{O(\log \log n)})$.*

Proof. In what follows, we describe a polynomial-time reduction that, for every given split graph G, produces a split graph G' with a unique split partition having both sides of even cardinality.

The reduction goes as follows. Let G be a split graph. We run the HK algorithm on G to obtain a split partition (K, S) of G. Let $p = |K|$ and $\ell = |S|$ and suppose that $K = \{c_1, \ldots, c_p\}$ and $S = \{s_1, \ldots, s_\ell\}$. It is straightforward to see that we may assume that S is a maximal stable set in G. Let G' be a disjoint copy of G. For each $v \in V(G)$, we denote by v' the copy of v in G'. Consider the natural split partition $K' = \{v' : v \in K\}$ and $S' = \{v' : v \in S\}$ of G'. We now define a graph H obtained from the union of G and G' by adding all possible edges between vertices in K and K'. Clearly, $(K \cup K', S \cup S')$ is a split partition of H, and $\alpha(H) = |S \cup S'|$ and $\omega(H) = |K \cup K'|$. Hence, by Theorem 6.2, H has a unique split partition. Moreover, this split partition has both sides of even cardinality.

It is easy to prove that if G has a CDS with at most q vertices, then H has a CDS with at most $2q$ vertices. The converse also holds. Indeed, suppose H has CDS $\hat{D}$ such that $|\hat{D}| \leq 2q$. We may assume that $\hat{D} \subseteq K \cup K'$. We can also suppose, without loss of

generality, that $|\hat{D} \cap K| \leq |\hat{D} \cap K'|$. Since $\hat{D}$ is a CDS of H, then, by the construction of H, we conclude that $\hat{D} \cap K$ is a CDS of G with at most q vertices.

Thus, we conclude that $\text{OPT}_{\text{MinCDS}}(H) \leq 2\text{OPT}_{\text{MinCDS}}(G)$. Now we are ready to prove the lemma. Assume $\mathcal{NP} \not\subseteq \text{DTIME}(n^{O(\log \log n)})$ and suppose, for a contradiction, that there exists an approximation algorithm with ratio $(1 - \varepsilon) \ln n$, where $\varepsilon < 1$ is a fixed positive constant, for MinCDS on n-vertex split graphs with a unique split partition having both sides of even cardinality. Let us call such algorithm $\mathcal{A}_\varepsilon$. Consider the following algorithm $\mathcal{A}'$ that, for every n-vertex split graph G, runs as follows.

Step 1. Check if $n^\varepsilon < 2$. If yes, then solve MinCDS on G by brute force and return an optimal solution. Otherwise, go to the next step;

Step 2. Run the reduction described previously on G to obtain H;

Step 3. Run $\mathcal{A}_\varepsilon$ on H to obtain $\hat{D}$;

Step 4. Compute D from $\hat{D}$ (as explained in the reduction) and return D.

One may easily check that $\mathcal{A}'$ is a polymonial-time algorithm. We claim that $\mathcal{A}'$ always returns a $(1 - \varepsilon^2) \ln n$-approximate CDS of G. Indeed, if $\mathcal{A}'$ halts on step 1, then, by construction, it returns an optimal CDS of G. Suppose that $\mathcal{A}'$ halts after step 4. By hypothesis, $|\hat{D}| \leq (1 - \varepsilon) \ln |V(H)| \text{OPT}_{\text{MinCDS}}(H)$. But now, since $|\hat{D}| \geq 2|D|$, $\text{OPT}_{\text{MinCDS}}(H) \leq 2\text{OPT}_{\text{MinCDS}}(G)$, $|V(H)| = 2n$ and $n^\varepsilon \geq 2$, we conclude that

$$|D| \leq ((1 - \varepsilon) \ln n^{1+\varepsilon}) \, \text{OPT}_{\text{MinCDS}}(G) = ((1 - \varepsilon^2) \ln n) \, \text{OPT}_{\text{MinCDS}}(G).$$
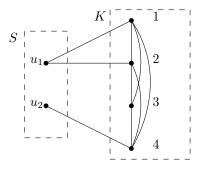
Therefore, the existence of $\mathcal{A}'$ contradicts Theorem 6.1, and the result follows.   □

The next result indicates that the inapproximability threshold for MinCDS in Theorem 6.1, proven by Chlebík and Chlebíková [CC04], remains unchanged, asymptotically speaking, even if we restrict the problem to graphs of diameter 2. We note that the reduction constructed in the proof of Chlebík and Chlebíková's result produces graphs with diameter greater than 2.
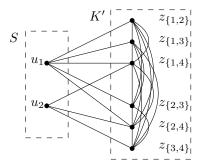
Theorem 6.4. *For any constant $c < 1/4$, MinCDS cannot be approximated to within a factor of $c \ln n$ even on n-vertex split graphs with diameter 2, unless $\mathsf{NP} \subseteq \mathrm{DTIME}(n^{O(\log \log n)})$.*

Proof. Let us first recall the reduction constructed by Lokshtanov et al. [LMP+13] to show that MinCDS is NP-hard on split graphs with diameter 2.

Given a split graph G with a unique split partition having both sides of even cardinality, first we run the HK algorithm to find the split partition (K, S) of $V(G)$. We define a graph G' with vertex set $V(G') = S \cup K'$, where $K' = \{z_e : e \in E(G[K])\}$. For every edge $e = uv \in E(G[K])$, G' has an edge $z_e w$ for every $w \in (N_G(u) \cup N_G(v)) \cap S$. Moreover, G' has all possible edges between vertices belonging to K', that is, K' induces a clique in G'. This construction is depicted in Figure 6.1. Clearly, G' is a split graph with diameter 2 and can be constructed in time polynomial in the size of G. Furthermore, observe that $|V(G')| \leq |V(G)|^2$.

(a) Graph G with split partition (K, S).

(b) Graph G' with diameter 2 and split partition (K', S).

Figure 6.1: *Reduction described in the proof of Theorem 6.4.*

We prove first that, for every CDS D of G with $|D| = q$, we can find in polynomial time a CDS D' of G' with $|D'| \leq (q+1)/2$. We may assume that $D \subseteq K$. Note that if $|D|$ is odd, then D is properly contained in K because, by hypothesis, $|K|$ is even. Consider $\widetilde{D} \subseteq V(G)$ defined as follows: let $\widetilde{D} = D$ if $|D|$ is even and let $\widetilde{D} = D \cup \{v\}$ otherwise, where $v \in K \setminus D$. Note that, by construction, $|\widetilde{D}|$ is even. Let $s = |\widetilde{D}|$ and suppose that $\widetilde{D} = \{v_1, \ldots, v_s\}$. Let $D' = \{z_{\{v_i, v_{i+s/2}\}} \in V(G') : i = 1, \ldots, s/2\}$. Observe that $|D'| \leq (q+1)/2$. Since D is a CDS of G, then, by the construction of G', it is

straightforward to check that D' is a CDS of G'. Thus, we conclude that, for every CDS of G with size q, we can find, in polynomial time, a CDS of G' with size at most $(q+1)/2$. Thus, we conclude that $\mathrm{OPT}_{\mathrm{MinCDS}}(G') \leq \mathrm{OPT}_{\mathrm{MinCDS}}(G)$.

Now we prove that, for every CDS D' of G' with $|D'| = q$, we can find in polynomial time a CDS D of G such that $|D| \leq 2q$. We may assume that $D' \subseteq K'$. In this case, the set $D = \{u, v \in V(G) : z_{\{u,v\}} \in D'\}$ has the desired properties.

Thus, given an $\alpha \ln |V(G')|$-approximate CDS of G', where α is some positive constant, we can find, in polynomial time, a $4\alpha \ln |V(G)|$-approximate CDS of G.

To conclude the proof of the theorem, assume that $\mathcal{NP} \not\subseteq \mathrm{DTIME}(n^{O(\log \log n)})$, take a constant $c < 1/4$, and suppose there exists an approximation algorithm, say $\mathcal{A}_c$, with ratio $c \ln n$, for MINCDS on n-vertex split graphs with diameter 2. Let $\mathcal{A}'$ be the algorithm for MINCDS on the class of graphs G described in Lemma 6.3 defined as follows: given G, it constructs G', as we mentioned previously, and runs $\mathcal{A}_c$ on G'. For $\varepsilon \leq 1 - 4c < 1$, note that algorithm $\mathcal{A}'$ obtains a $(1 - \varepsilon) \ln |V(G)|$-approximate CDS of G, a contradiction to Lemma 6.3. $\qquad\square$

6.2   Hardness results for MINk-CDS

In this section, we address complexity issues regarding MINk-CDS from the standpoint of finding exact or approximate solutions.

As mentioned in Chapter 3, Nguyen and Huynh [NH06a] showed that MINk-CDS is $\mathcal{NP}$-hard on planar unit disk graphs with maximum degree 4. Their proof, which is quite involved, clearly implies that MINk-CDS is $\mathcal{NP}$-hard on planar graphs with maximum degree 4. The next result, with a simpler proof, has the same implication. Furthermore, it strengthens and generalizes a theorem of White et al. [WFP85] who showed that MINCDS is $\mathcal{NP}$-hard on planar bipartite graphs.

Theorem 6.5. *For every $k \in \mathbb{Z}_>$, the decision version of MINk-CDS is $\mathcal{NP}$-complete on planar bipartite graphs of maximum degree 4.*

Proof. We say that a vertex subset K of a graph G is a *connected vertex cover* (CVC) of G if every edge in G has at least one endpoint in K and $G[K]$ is connected. The *minimum connected vertex cover problem* (MINCVC) seeks for a CVC of G of minimum cardinality.

Fernau and Manlove [FM09] proved that the decision version of MINCVC is NP-complete on planar bipartite graphs of maximum degree 4. We present a polynomial-time reduction from the decision version of MINCVC to the decision version of MINk-CDS, which is clearly a problem in NP.

Let G be a connected planar bipartite graph with maximum degree 4. For each edge $e \in E(G)$ with endpoints u and v, we remove e from G, take a disjoint path P_e with $k+1$ new vertices, and then add edges $w_e u$ and $w_e v$, where w_e is an endpoint of P_e. Let G' be the graph obtained from G with this procedure. Clearly, G' is planar bipartite, and has maximum degree 4. The reduction is depicted in Figure 6.2.
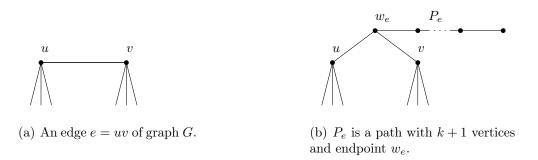

(a) An edge $e = uv$ of graph G.   (b) P_e is a path with $k+1$ vertices and endpoint w_e.

Figure 6.2: *Gadget described in the proof of Theorem 6.5.*

We claim that G has a CVC of size at most q if and only if G' has a k-CDS of size at most $|E(G)| + q$.

Let K be a CVC of G with $|K| \leq q$. Take $D = K \cup \{w_e \colon e \in E(G)\}$. Clearly, the size of D is at most $|E(G)| + q$. Since K induces a connected vertex cover of G, it follows that D induces a connected subgraph of G'. For every $e \in E(G)$, the distance between each vertex of P_e and its endpoint w_e is at most k. Consequently, D is a k-CDS of G' such that $|D| \leq |E(G)| + q$.

Let D' be a k-CDS of G' with $|D'| \leq |E(G)| + q$. Take $K = D' \cap V(G)$. Since, for every $e \in E(G)$, the path P_e in G' has $k+1$ vertices, we conclude that $w_e \in D'$, and thus

$|D' \cap (V(G') \setminus V(G))| \geq |E(G)|$. Consequently, $|K| \leq q$. Since $w_e \in D'$ for every $e \in E(G)$ and D' induces a connected subgraph of G', we conclude that K is a vertex cover of G and $G[K]$ is connected. Therefore, K is a CVC of G, and this completes the proof. $\square$
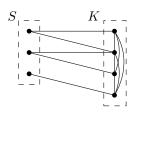
Let $p, q \in \mathbb{Z}_>$. We say that a graph G is (p,q)-*split* if there is a partition (K, S) of $V(G)$ such that $\alpha(G[K]) \leq p$ and $\omega(G[S]) \leq q$. We call such a partition a (p,q)-split partition. Note that every split graph belongs to the class of (p,q)-split graphs, since every split graph is a $(1, 1)$-split graph (see Gyárfás [Gya98]).

We present now an inapproximability result for MINk-CDS that can be seen as a generalization of Theorem 6.1. First, we observe that every split graph has a 2-universal vertex (just take any vertex in the clique side of a split partition). Thus, MINk-CDS becomes trivial on split graphs for every $k \geq 2$, and therefore one can only obtain a result like Theorem 6.6 for a superclass of split graphs. We were able to prove an inapproximabilty result for the class of $(1, 2)$-split graphs, as we show in what follows.
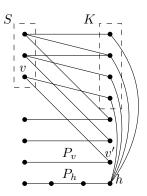
Theorem 6.6. *For every $k \in \mathbb{Z}_>$ and every fixed $\varepsilon > 0$, MINk-CDS cannot be approximated to within a factor of $(1 - \varepsilon) \ln n$ on n-vertex $(1, 2)$-split graphs and on bipartite graphs, unless $\mathcal{NP} \subseteq \mathrm{DTIME}(n^{O(\log \log n)})$.*

Proof. We first prove the claimed inapproximability threshold for bipartite graphs. We next show a reduction from MINCDS on split graphs to MINk-CDS on bipartite graphs. Naturally, we assume that $k \geq 2$ because the result is already proven for $k = 1$.

Let G be a split graph. Firstly, we run the HK algorithm on G and obtain a split partition (K, S) of G. Let G' be the graph obtained from G as follows. For every $v \in S$, take a disjoint path P_v with $k-1$ new vertices and endpoint v', and add an edge connecting v to v'. Additionally, take another disjoint path P_h with $k+1$ new vertices and endpoint h. Then, for each $v \in K$, add an edge connecting v to h. Finally, remove all edges with both endpoints in K. We depict the construction of G' from G in Figure 6.3. Clearly, G' is bipartite and can be constructed in time polynomial in the size of G. Furthermore, observe that $|V(G')| \leq k|V(G)| + k + 1$.

(a) Graph G with split partition (K, S).

(b) Bipartite graph G' obtained from G when $k = 3$.

Figure 6.3: *Reduction described in the proof of Theorem 6.6 (bipartite case).*

We claim that G has a CDS of size at most q if and only if G' has a k-CDS of size at most $q + 1$. Let D be a CDS of G such that $|D| \leq q$. We may assume, without loss of generality, that $D \subseteq K$. Let us define $D' = D \cup \{h\}$. Since D is a CDS in G, then D' is a k-CDS in G'. Hence, for every CDS of G with size at most q, we can find, in polynomial time, a k-CDS of G' with size at most $q + 1$.

Consider now a k-CDS D' of G' such that $|D'| \leq q + 1$. Since D' induces a connected subgraph of G' and $|V(P_h)| = k + 1$, we conclude that $h \in D'$. Let us define $D = D' \cap K$. Since D' is a k-CDS of G', one may easily verify that D is a CDS of G and $|D| \leq q$. Hence, for every k-CDS of G' with size at most $q+1$, we can find, in polynomial time, a CDS of G with size at most q. Therefore, we conclude that $\mathrm{OPT}_{\mathrm{MIN}k\text{-CDS}}(G') = \mathrm{OPT}_{\mathrm{MINCDS}}(G)+1$.

Assume $\mathcal{NP} \nsubseteq \mathrm{DTIME}(n^{O(\log \log n)})$ and suppose, to the contrary, that there exists an approximation algorithm for MINk-CDS on n-vertex bipartite graphs with ratio $(1 - \varepsilon) \ln n$, where $\varepsilon < 1$ is some fixed positive constant. Let us call such algorithm $\mathcal{A}_\varepsilon$. Consider the following algorithm $\mathcal{A}'$ that, for every input n-vertex split graph G runs as follows.

Step 1. Check if $n < k + 2$. If yes, then solve MINCDS on G by brute force. Otherwise, go to the next step;

Step 2. Check if $\mathrm{OPT}_{\mathrm{MINCDS}}(G) < 1/\varepsilon$ (by enumerating, via brute force, all possible

solutions, if any, with at most $\lfloor 1/\varepsilon \rfloor$ vertices; remember that ε is a fixed constant). If yes, then solve MINCDS on G by brute force. Otherwise, go to the next step;

Step 3. Check if $n^{\varepsilon^2} < (k+1)$. If yes, then solve MINCDS on G by brute force. Otherwise, go to the next step;

Step 4. Run the reduction described previously on G to obtain G';

Step 5. Run $\mathcal{A}_\varepsilon$ on G' to obtain D';

Step 6. Compute D from D' (as explained in the reduction) and return D.

It is immediate that $\mathcal{A}'$ is a polynomial time algorithm. Now we claim that $\mathcal{A}'$ always find a $(1 - \varepsilon^4) \ln n$-approximate CDS of G. If $\mathcal{A}'$ halts before step 4, then it returns an optimal CDS of G. Suppose now that $\mathcal{A}'$ halts after step 6. By hypothesis, we have that $|D'| \le (1 - \varepsilon) \ln |V(G')| \text{OPT}_{\text{MINCDS}}(G')$. But since $|D'| > |D|$, $n \ge k + 2$, $\text{OPT}_{\text{MINCDS}}(G) \ge 1/\varepsilon$ and $n^{\varepsilon^2} \ge (k + 1)$, we conclude that

$$|D| \le (1 - \varepsilon) \ln n^{1+\varepsilon^2} (1 + \varepsilon) \text{OPT}_{\text{MINCDS}}(G) = (1 - \varepsilon^4) \ln n \text{OPT}_{\text{MINCDS}}(G).$$

Therefore, the existence of $\mathcal{A}'$ contradicts Theorem 6.1, and the result follows.

Now we prove the result for $(1, 2)$-split graphs. Since the reduction for $(1, 2)$-split graphs is quite similar to the one we discussed for bipartite graphs, we present only a sketch of the proof. The idea is to show a reduction from MINCDS on split graphs to MINk-CDS on $(1, 2)$-split graphs. Consider a split graph G. As before, we begin by runnning the HK algorithm on G and we obtain a split partition (K, S) of G.

Let G' be the graph obtained from G as follows. For every vertex $v \in S$, we replace it by a path P_v with k vertices in such a way that vertex v is identified with an endpoint of P_v. We keep all other vertices and edges of G intact. In summary, $V(G') = K \cup (\cup_{v \in S} V(P_v))$ and $E(G') = E(G) \cup (\cup_{v \in S} E(P_v))$. Furthermore, note that $|V(G')| \le k|V(G)|$. Note also that $\alpha(G'[K]) = 1$ and $\omega(G'[V(G') \setminus K]) = 2$. Thus, G' is a $(1, 2)$-split graph. Moreover, it can be constructed in time polynomial in the size of G.

It is not hard to see that G has a CDS of size at most q if and only if G' has a k-CDS of size at most q. Therefore, $\text{OPT}_{\text{MIN}k\text{-CDS}}(G') = \text{OPT}_{\text{MINCDS}}(G)$.

Assume that there exists an approximation algorithm for MINk-CDS on n-vertex $(1,2)$-split graphs with ratio $(1 - \varepsilon)\ln n$, where $\varepsilon < 1$ is some fixed positive constant. Let us call such algorithm $\mathcal{A}_\varepsilon$. Consider the following algorithm $\mathcal{A}'$ that, for every input n-vertex split graph G runs as follows.

Step 1. Check if $n^\varepsilon < k$. If yes, then solve MINCDS on G by brute force. Otherwise, go to the next step;

Step 2. Run the reduction described previously on G to obtain G';

Step 3. Run $\mathcal{A}_\varepsilon$ on G' to obtain D';

Step 4. Compute D from D' (as explained in the reduction) and return D.

One may easily check that $\mathcal{A}'$ is a polynomial-time algorithm. We claim that $\mathcal{A}'$ always returns a $(1 - \varepsilon^2)$-approximate CDS of G. If $\mathcal{A}'$ halts in step 1, then it returns an optimal CDS of G. Suppose now that $\mathcal{A}'$ halts after step 4. By hypothesis, we have that $|D'| \leq (1 - \varepsilon)\ln|V(G')|\text{OPT}_{\text{MINCDS}}(G')$. But since $|D'| = |D|$, $|V(G')| \leq kn$, $n^\varepsilon \geq k$ and $\text{OPT}_{\text{MINCDS}}(G) = \text{OPT}_{\text{MIN}k\text{-CDS}}(G')$, we conclude that

$$|D| \leq (1 - \varepsilon)\ln n^{1+\varepsilon}\text{OPT}_{\text{MINCDS}}(G) = (1 - \varepsilon^2)\ln n\text{OPT}_{\text{MINCDS}}(G).$$

Therefore, under the hypothesis that $\mathcal{NP} \not\subseteq \text{DTIME}(n^{O(\log\log n)})$, the existence of $\mathcal{A}'$ contradicts Theorem 6.1, and so the result follows.

$\square$

As we mentioned in the literature review subsection, Bonsma [Bon12] proved that MINCDS is $\mathcal{APX}$-complete on cubic graphs. The next theorem can be seen as a move in the direction of extending Bonsma's result.

Theorem 6.7. *For every $k \in \mathbb{Z}_>$, MINk-CDS is $\mathcal{APX}$-complete on bipartite graphs of maximum degree* 4.

Proof. We present an L-reduction from MINCVC to MINk-CDS. In fact, we show that the reduction described in the proof of Theorem 6.5 is an L-reduction. Then, we use the fact that MINCVC is $\mathcal{APX}$-hard on bipartite graphs of maximum degree 4 [EGM10].

Let G be a bipartite graph with maximum degree 4, and let G' be the graph obtained from G using the reduction presented in the proof of Theorem 6.5. Observe that, for each CVC K of G, we can construct, in polynomial time, a k-CDS D of G' such that $|D| \leq |K| + |E(G)|$ (see the proof of Theorem 6.5). Taking K as a minimum CVC of G, we can conclude that $\mathrm{OPT}_{\mathrm{MIN}k\text{-CDS}}(G') \leq \mathrm{OPT}_{\mathrm{MINCVC}}(G) + |E(G)|$. (This inequality will be used next.)

Since every vertex in G has degree at most 4, for every CVC K of G, it follows that $|E(G)| \leq 4|K|$. Therefore, for each CVC K of G, we can find, in polynomial time, a k-CDS D of G' such that $|D| \leq 5|K|$. Thus, $\mathrm{OPT}_{\mathrm{MIN}k\text{-CDS}}(G') \leq 5\,\mathrm{OPT}_{\mathrm{MINCVC}}(G)$.

Conversely, given a k-CDS D of G', we know that, for each $e \in E(G)$, the set D contains w_e and at least one of the endpoints of e. Thus, if we take $K = D \cap V(G)$, we have that K is a CVC of G and $|K| \leq |D| - |E(G)|$. Therefore,

$$|K| - \mathrm{OPT}_{\mathrm{MINCVC}}(G) \leq |D| - |E(G)| - \mathrm{OPT}_{\mathrm{MINCVC}}(G) \leq |D| - \mathrm{OPT}_{\mathrm{MIN}k\text{-CDS}}(G').$$

This concludes the proof of the L-reduction. Consequently, MINk-CDS is $\mathcal{APX}$-hard on bipartite graphs of maximum degree 4. By Corollary 5.3, we know that the problem is in $\mathcal{APX}$, and therefore it is an $\mathcal{APX}$-complete problem. $\square$

In the proof of Theorem 5.6, given in Section 5.2 of Chapter 5, we showed that, for every graph G, there exists a $\sigma(G)$-approximation algorithm for MINWCDS, where $\sigma(G)$ is the cardinality of the largest minimal separator of G. We next prove that the approximation ratio of algorithm LPAPPROXMINWCDS, though simple as it is, is near-optimal. But before we do that, we need the following lemma, which can be considered a

step towards a generalization of Theorem 4.1 (which is the result of Kanté et al. [KLMN11] that shows that a vertex subset of a graph is a CDS if and only if it is a tranversal of the set of all separators of the graph).

Lemma 6.8. *Let G be a graph and $\mathcal{S}_k(G)$ be the set of all minimal k-disruptive separators of G. A set $D \subseteq V(G)$ is a minimal k-CDS of G if and only if D is a minimal transversal of $\mathcal{S}_k(G)$.*

Proof. Firstly, we prove that every k-CDS of G is a transversal of $\mathcal{S}_k(G)$. Let $D \subseteq V(G)$ be minimal a k-CDS of G and consider a k-disruptive separator Γ of G. We claim that $D \cap \Gamma \neq \emptyset$. Indeed, if $D \cap \Gamma = \emptyset$, then D is entirely contained in some component of $G - \Gamma$, a contradiction to the fact that Γ is a k-disruptive separator of G.

We next show that every minimal transversal of $\mathcal{S}_k(G)$ is a minimal k-CDS of G. Let $D \subseteq V(G)$ be a minimal transversal of $\mathcal{S}_k(G)$. Firstly, suppose to the contrary that $G[D]$ is not connected. Since D intersects every k-disruptive separator of G, there must be a component of $G[D]$, say C, such that $V(C)$ is a k-CDS of G; otherwise, $V(G) \setminus D$ would be a k-disruptive separator of G, contradicting the assumption on D. As we have shown before, $V(C)$ contains a transversal of $\mathcal{S}_k(G)$, but, since $V(C)$ is strictly contained in D, this contradicts the minimality of D. Thus, $G[D]$ is connected.

We now claim that D is a k-DS of G. Suppose to the contrary that there exists a vertex $v \in V(G)$ such that $v \notin N^k[D]$. Therefore, $N^k[v] \cap D = \emptyset$ and we conclude that v is not a k-universal vertex of G, that is, $\{v\}$ is not a k-CDS of G. Let us define $\Gamma = N^k(v)$. We claim that Γ is a k-disruptive separator of G. Clearly, $G - \Gamma$ is disconnected. Let C be a component of $G - \Gamma$. If $v \in V(C)$, then $V(C) = \{v\}$ and we already know that $\{v\}$ is not a k-CDS of G. If $v \notin V(C)$, then $v \notin N_k(V(C))$ and, once again, $V(C)$ is not a k-CDS of G. Hence, Γ is a k-disruptive separator of G, a contradiction to the fact that D is a transversal of $\mathcal{S}_k(G)$. This concludes the proof that D is a k-DS of G. Since $G[D]$ is also connected, D is a k-CDS of G. Finally, D is a minimal k-CDS of G because any k-CDS of G strictly contained in D would be a transversal of $\mathcal{S}_k(G)$, a contradiction the fact that D is a minimal transversal.

Let $D \subseteq V(G)$ be a minimal k-CDS of G. We have shown that D intersects every minimal k-disruptive separator of G. Now, we claim that D is a minimal transversal of $\mathcal{S}_k(G)$. Observe that any transversal of $\mathcal{S}_k(G)$ strictly contained in D would be a k-CDS of G, and this contradicts the fact that D is a minimal k-CDS of G. Therefore, D is a minimal transversal of $\mathcal{S}_k(G)$ and the result follows. $\square$

So now let us prove, assuming $\mathcal{P} \neq \mathcal{NP}$, that the performance guarantee of algorithm LPAPPROXMINWCDS is close to the best we can hope for.

Theorem 6.9. *For every $k \in \mathbb{Z}_>$ and a fixed integer $p \geq 2$, let $\mathcal{G}_p$ be the class of graphs G with $\sigma_k(G) = p$. For every $k \in \mathbb{Z}_>$ and every fixed $\varepsilon > 0$, if $\mathcal{P} \neq \mathcal{NP}$, then MIN$k$-CDS cannot be approximated to within a factor of*

$$\max \left\{ p - 1 - \varepsilon, \ 10\sqrt{5} - 21 \right\}$$

on the class $\mathcal{G}_p$. Moreover, this claim holds even when we restrict it to $(1,2)$-split graphs in $\mathcal{G}_p$.
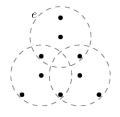
Proof. Let $\mathcal{H}$ be a hypergraph. We say that $\mathcal{H}$ is **simple** if none of its hyperedges is contained within another. Moreover, for every integer $p \geq 2$, we say that $\mathcal{H}$ is a p-uniform hypergraph if all of its hyperedges have cardinality exactly p. A subset $K \subseteq V(\mathcal{H})$ is said to be a vertex cover of $\mathcal{H}$ if K intersects each hyperedge of $\mathcal{H}$. The *minimum hypergraph vertex cover problem* (MINHVC) consists in finding a vertex cover of $\mathcal{H}$ of minimum cardinality.

We show an L-reduction from MINHVC on simple p-uniform hypergraphs (fixed $p \geq 2$) to MINk-CDS on the class $\mathcal{G}_p$. We consider two cases.

Case 1: $p \geq 3$.
Dinur, Guruswami, Khot and Regev [DGKR05] proved that, under $\mathcal{P} \neq \mathcal{NP}$, MINHVC has no $(p - 1 - \varepsilon)$-approximation on p-uniform simple hypergraphs, for every $p \geq 3$ and every fixed $\varepsilon > 0$.
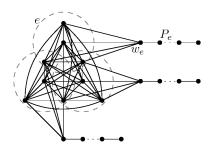
Let $\mathcal{H}$ be an n-vertex p-uniform simple hypergraph with $|E(\mathcal{H})| \geq 2$. From $\mathcal{H}$, we construct a graph G with $V(G) \supseteq V(\mathcal{H})$ as follows. For every $e \in E(\mathcal{H})$, we take a disjoint path P_e with k new vertices, and denote by w_e one of its end vertices. Then, we add an edge connecting w_e to every vertex in e. Finally, we make all vertices belonging to $V(\mathcal{H})$ pairwise adjacent in G, that is, $G[V(\mathcal{H})]$ is a clique. This construction is depicted in Figure 6.4. Note that G is a $(1,2)$-split graph that can be constructed in polynomial time in the size of $\mathcal{H}$.

(a) A 4-uniform hypergraph $\mathcal{H}$ with 3 hyperedges (dashed circles).

(b) Path P_e with k vertices and end vertex w_e.

(c) Graph G. The vertices of the hypergraph $\mathcal{H}$ induce a clique in G.

Figure 6.4: *Reduction from* MINHVC *to* MINk-CDS*: construction of G from the hypergraph $\mathcal{H}$.*

Recall that $\mathcal{S}_k(G)$ denotes the set of all minimal k-disruptive separators of G. We claim that $\mathcal{S}_k(G) = E(\mathcal{H})$. It is easy to see that $E(\mathcal{H}) \subseteq \mathcal{S}_k(G)$. Now we prove that $\mathcal{S}_k(G) \subseteq E(\mathcal{H})$. For that, it suffices to show that, for every minimal k-disruptive separator Γ of G, there exists $e \in E(\mathcal{H})$ such that $\Gamma = N(w_e) \cap V(\mathcal{H})$. Consider $\Gamma \in \mathcal{S}_k(G)$. Note that, every minimal k-CDS of G is contained in $V(\mathcal{H})$. Therefore, due to the minimality of Γ, since, by definition, Γ intersects every minimal k-CDS of G, one may easily check

that $\Gamma \subseteq V(\mathcal{H})$. Suppose, to the contrary, that $N(w_e) \cap V(\mathcal{H}) \neq \Gamma$ for all $e \in E(\mathcal{H})$. Since $N(w_e) \cap V(\mathcal{H}) \in \mathcal{S}_k(G)$ for every $e \in E(\mathcal{H})$, again, by the minimality of Γ, it follows that, for every $e \in E(\mathcal{H})$, $N(w_e) \cap V(\mathcal{H})$ is not strictly contained in Γ. Hence, $(N(w_e) \cap V(\mathcal{H})) \setminus \Gamma \neq \emptyset$ for every $e \in E(\mathcal{H})$. By the construction of G, we conclude that Γ is not a separator of G, a contradiction. Therefore, we have $\mathcal{S}_k(G) = E(\mathcal{H})$. Consequently, $\sigma_k(G) = p$, that is, $G \in \mathcal{G}_p$.

Now, we claim that G has a k-CDS of size q if and only if $\mathcal{H}$ has a vertex cover of size q. Consider a k-CDS D of G. We may assume that D is minimal and that $D \subseteq V(\mathcal{H})$. By Lemma 6.8, D is a transversal of $\mathcal{S}_k(G)$. Since $E(\mathcal{H}) = \mathcal{S}_k(G)$, we conclude that D is a vertex cover of $\mathcal{H}$.

Conversely, let K be a vertex cover of $\mathcal{H}$. We may assume that K is minimal. Since $E(\mathcal{H}) = \mathcal{S}_k(G)$, it follows that K is a transversal of $\mathcal{S}_k(G)$. By Lemma 6.8, K is a k-CDS of G. Hence, we conclude that $\mathrm{OPT}_{\mathrm{MIN}k\text{-}\mathrm{CDS}}(G) = \mathrm{OPT}_{\mathrm{MINHVC}}(\mathcal{H})$, and this finishes the proof of the L-reduction.

Thus, every α-approximation for MINk-CDS on $(1,2)$-split graphs in $\mathcal{G}_p$ yields an α-approximation for MINHVC on p-uniform simple hypergraphs. In view of the result shown by Dinur et al. [DGKR05], we conclude that, if $\mathcal{P} \neq \mathcal{NP}$, for every $\varepsilon > 0$, there is no $(p - 1 - \varepsilon)$-approximation algorithm for MINk-CDS on $(1,2)$-split graphs in the class $\mathcal{G}_p$.

Case 2: $p = 2$.

In this case, we refer to the *minimum vertex cover problem* (MINVC), which is simply the restriction of MINHVC to graphs (which are 2-uniform hypergraphs); and for this problem, Dinur and Safra [DS05] showed that MINVC has no $(10\sqrt{5} - 21)$-approximation if $\mathcal{P} \neq \mathcal{NP}$. In order to prove the result we claim, we use the same reduction discussed before, and from $\mathcal{H}$ (which is now a graph) we construct a $(1,2)$-split graph G with $\sigma_k(G) = 2$. The result follows analogously, this time using the inapproximability result for MINVC.

$\square$

Chapter 7

Conclusions

We studied, in this thesis, the minimum weight k-hop connected dominating set problem, a generalization of the well-known minimum connected dominating set problem. We proved that, for every k, the decision version of MINk-CDS is $\mathcal{NP}$-hard on planar bipartite graphs of maximum degree 4 (showing that the hardness of MINCDS carries over to MINk-CDS on the same subclass of graphs, since White et al. [WFP85] had proved that MINCDS is $\mathcal{NP}$-hard on planar bipartite graphs).

We also proved a number of results on the (in)approximability of MINk-CDS. We showed that MINk-CDS is $\mathcal{APX}$-complete on the class of bipartite graphs with maximum degree 4 (we recall that it has been proven by Bonsma [Bon12] that MINCDS on cubic graphs is $\mathcal{APX}$-complete).

We showed that the inapproximability threshold $((1 - \varepsilon) \ln n)$ of MINCDS that holds already for n-vertex split graphs (and also for bipartite graphs), proved by Chlebík and Chlebíková [CC04], can be generalized to MINk-CDS on $(1, 2)$-split graphs (and also for bipartite graphs). We note here that, for $k \geq 2$, the last result does not hold for split graphs (on which the problem is trivial). We also showed an inapproximability threshold close to that of Chlebík and Chlebíková [CC04] for MINCDS on the smaller class of split graphs of diameter 2 (the smallest value of the diameter for which the problem is non-trivial).

On the positive side, we presented a type of meta-approximation theorem which says that, for every graph G, an $f(G)$-approximation for MinCDS on G can be turned into a $kf(G^k)$-approximation for Min$k$-CDS on G. As a consequence, we obtained an algorithm for Min$k$-CDS that returns a $k(1+\epsilon)(\ln(\Delta(G^k)-1)+1)$-aproximate k-CDS for every graph G and every fixed $0 < \varepsilon \leq 1$. This result improves (asymptotically by a factor of 2) on the approximation originally proposed by Ren and Zhao [RZ11] for the minimum connected set cover problem (a generalization of Min$k$-CDS). Ren and Zhao's algorithm translates into a $2k(H(\Delta(G^k)) + 1)$-approximation for Min$k$-CDS. Furthermore, we showed two approximation algorithms for the weighted version of MinCDS, one of them restricted to graphs with polynomially many minimal separators, a class that includes, for instance, chordal graphs.

As future steps, we think it would be interesting to further investigate the graph parameter $\sigma_k(G)$, possibly finding classes of graphs G for which this parameter is bounded by a constant, or graphs for which $\mathcal{S}_k(G)$ is polynomially bounded. Likewise, it would be worthwhile to settle or refute Conjecture 5.9, which concerns the computational complexity of finding minimum weight k-disruptive separators in general vertex-weighted graphs. Another line of research would be the design of better approximation algorithms for Min$k$-CDS, when $k \geq 2$, for special classes of graphs, such as the cubic graphs.

Bibliography

[AB09]   S. Arora and B. Barak. *Computational Complexity: A Modern Approach.* Cambridge University Press, 2009. 7

[AdMRS17]   S. A. Amiri, P. O. de Mendez, R. Rabinovich and S. Siebertz. Distributed domination on graph classes of bounded expansion. *ArXiv pre-prints arXiv:1702.02848*, 2017. 18

[APMS+99]   G. Ausiello, M. Protasi, A. Marchetti-Spaccamela, G. Gambosi, P. Crescenzi and V. Kann. *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties.* Springer-Verlag, 1999. 13

[AR92]   K. Arvind and C. P. Regan. Connected domination and steiner set on weighted permutation graphs. *Information Processing Letters*, 41(4):215–220, 1992. 3, 15

[BBC99]   A. Berry, J. P. Bordat and O. Cogis. Generating all the minimal separators of a graph. In *Proceedings of the 25th International Workshop on Graph-Theoretic Concepts in Computer Science*, pages 167–172. Springer-Verlag, 1999. 39

[BD98]   A. Brandstädt and F. F. Dragan. A linear-time algorithm for connected r-domination and Steiner tree on distance-hereditary graphs. *Networks*, 31(3):177–182, 1998. 15

[BDTC05]   J. Blum, M. Ding, A. Thaeler and X. Cheng. Connected dominating set in sensor networks and MANETs. In D. Du and P. M. Pardalos, editors, *Handbook of Combinatorial Optimization*, pages 329–369. Springer, 2005. 2

[Ber62]   C. Berge. *The Theory of Graphs and Its Applications.* Methuen, 1962. 1

[BK85]   A. Brandstädt and D. Kratsch. On the restriction of some NP-complete graph problems to permutation graphs. In *International Conference on Fundamentals of Computation Theory*, pages 53–62. Springer, 1985. 3

[BL15]   G. Borradaile and H. Le. Optimal dynamic program for r-domination problems over tree decompositions. *CoRR*, abs/1502.00716, 2015. 15

[BLS99]   A. Brandstädt, V. Le and J. Spinrad. *Graph Classes: A Survey.* Society for Industrial and Applied Mathematics, 1999. 10, 39

[Bon12]   P. Bonsma. Max-leaves spanning tree is APX-hard for cubic graphs. *J. of Discrete Algorithms*, 12:14–23, 2012. 19, 22, 56, 63

[BT01]   V. Bouchitté and I. Todinca. Treewidth and minimum fill-in: Grouping the minimal separators. *SIAM Journal on Computing*, 31(1):212–232, 2001. 39

[BZ08]   P. Bonsma and F. Zickfeld. *A 3/2-Approximation Algorithm for Finding Spanning Trees with Many Leaves in Cubic Graphs*, pages 66–77. Springer Berlin Heidelberg, 2008. 17

[CACdVK+16]   V. Cohen-Addad, E. Colin de Verdière, P. N. Klein, C. Mathieu and D. Meierfrankenfeld. Approximating connectivity domination in weighted bounded-genus graphs. In *Proceedings of the 48th Annual ACM Symposium on Theory of Computing*, pages 584–597. ACM, 2016. 3, 18

[CC04]   M. Chlebík and J. Chlebíková. Approximation hardness of dominating set problems. In *European Symposium on Algorithms*, pages 192–203. Springer, 2004. 19, 21, 48, 49, 63

[CC08]   M. Chlebík and J. Chlebíková. Approximation hardness of dominating set problems in bounded degree graphs. *Information and Computation*, 206(11):1264–1275, 2008. 3, 19

[CCJ90]   B. N. Clark, C. J. Colbourn and D. S. Johnson. Unit disk graphs. *Discrete Mathematics*, 86(1):165–177, 1990. 3

[CCZ14]   M. Conforti, G. Cornuéjols and G. Zambelli. *Integer Programming*, volume 271. Springer, 2014. 11

[CG06]   L. S. Chandran and F. Grandoni. A linear time algorithm to list the minimal separators of chordal graphs. *Discrete Mathematics*, 306(3):351–358, 2006. 39

[CHL+03]   X. Cheng, X. Huang, D. Li, W. Wu and D. Du. A polynomial-time approximation scheme for the minimum connected dominating set in ad hoc wireless networks. *Networks*, 42:202–208, 2003. 18

[Chv79]   V. Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979. 39, 40

[CKST99]   P. Crescenzi, V. Kann, R. Silvestri and L. Trevisan. Structure in approximation classes. *SIAM Journal on Computing*, 28(5):1759–1782, 1999. 13

[CMW15]   R. S. Coelho, P. F. S. Moura and Y. Wakabayashi. The k-hop connected dominating set problem: hardness and polyhedra. *Electronic Notes in Discrete Mathematics*, 50:59–64, 2015. 5

[CMW17] R. S. Coelho, P. F. S. Moura and Y. Wakabayashi. The k-hop connected dominating set problem: approximation and hardness. *Journal of Combinatorial Optimization*, pages 1–24, 2017. 5

[Cre97] P. Crescenzi. A short guide to approximation preserving reductions. In *Proceedings of the 12th Annual IEEE Conference on Computational Complexity*, pages 262–273. IEEE, 1997. 13

[CS90] C. J. Colbourn and L. K. Stewart. Permutation graphs: Connected domination and steiner trees. *Discrete Mathematics*, 86(1):179–189, 1990. 3, 15

[CW16] R. S. Coelho and Y. Wakabayashi. Approximation algorithms for the k-hop connected dominating set problem. 4th International Symposium on Combinatorial Optimization (ISCO 2016), Vietri sul Mare, Italy, 2016. 5

[DGKR05] I. Dinur, V. Guruswami, S. Khot and O. Regev. A new multilayered PCP and the hardness of hypergraph vertex cover. *SIAM Journal of Computing*, 34(5):1129–1146, 2005. 59, 61

[DGP+08] D. Du, R. L. Graham, P. M. Pardalos, P. Wan, W. Wu and W. Zhao. Analysis of greedy approximations with nonsubmodular potential functions. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 167–175. Society for Industrial and Applied Mathematics, 2008. 3, 17, 37, 38

[DH05] E. D. Demaine and M. Hajiaghayi. Bidimensionality: new connections between FPT algorithms and PTASs. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 590–601. Society for Industrial and Applied Mathematics, 2005. 3, 18

[Die06] R. Diestel. *Graph Theory*. Electronic Library of Mathematics. Springer, 2006. 10

[DM88] A. D'Atri and M. Moscarini. Distance-hereditary graphs, Steiner trees, and connected domination. *SIAM Journal on Computing*, 17(3):521–538, 1988. 3, 15

[DMP+05] D. Dubhashi, A. Mei, A. Panconesi, J. Radhakrishnan and A. Srinivasan. Fast distributed algorithms for (weakly) connected dominating sets and linear-size skeletons. *Journal of Computer and System Sciences*, 71(4):467–479, 2005. 18

[Dra93] F. Dragan. HT-graphs: centers, connected r-domination and Steiner trees. *Computer Science*, 1(2):64–83, 1993. 15

[DS05] I. Dinur and S. Safra. On the hardness of approximating minimum vertex cover. *Annals of Mathematics*, 162(1):439–485, 2005. 61

[DSM16] S. Dhanalakshmi, N. Sadagopan and V. Manogna. On $2K_2$-free graphs - structural and combinatorial view. *ArXiv pre-prints arXiv:1602.03802*, 2016. 39

[DW12] D.Z. Du and P.J. Wan. *Connected Dominating Set: Theory and Applications*. Springer Optimization and Its Applications. Springer New York, 2012. 1

[EGM10] B. Escoffier, L. Gourvès and J. Monnot. Complexity and approximation results for the connected vertex cover problem in graphs and hypergraphs. *J. of Discrete Algorithms*, 8(1):36–49, 2010. 57

[FKTV08] F. V. Fomin, D. Kratsch, I. Todinca and Y. Villanger. Exact algorithms for treewidth and minimum fill-in. *SIAM Journal on Computing*, 38(3):1058–1079, 2008. 41

[FM09] H. Fernau and D. F. Manlove. Vertex and edge covers with clustering properties: Complexity and algorithms. *J. of Discrete Algorithms*, 7(2):149–167, 2009. 52

[Fuj03] T. Fujie. An exact algorithm for the maximum leaf spanning tree problem. *Comput. Oper. Res.*, 30(13):1931–1944, 2003. 20

[Fuj04] T. Fujie. The maximum-leaf spanning tree problem: Formulations and facets. *Networks*, 43(4):212–223, 2004. 20, 33

[FV12] F. V. Fomin and Y. Villanger. Treewidth computation and extremal combinatorics. *Combinatorica*, 32(3):289–308, 2012. 41

[FW12] N. Fan and J. Watson. Solving the connected dominating set problem and power dominating set problem by integer programming. In *International Conference on Combinatorial Optimization and Applications*, pages 371–383. Springer, 2012. 3, 19

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1979. 3, 7

[GK98] S. Guha and S. Khuller. Approximation algorithms for connected dominating sets. *Algorithmica*, 20(4):374–387, 1998. 3, 16, 17

[GK99] S. Guha and S. Khuller. Improved methods for approximating node weighted steiner trees and connected dominating sets. *Information and computation*, 150(1):57–74, 1999. 3, 18

[GLdCS14] B. Gendron, A. Lucena, A. S. da Cunha and L. Simonetti. Benders decomposition, branch-and-cut, and hybrid algorithms for the minimum connected dominating set problem. *INFORMS Journal on Computing*, 26(4):645–657, 2014. 3, 20

[GLS12]   M. Grötschel, L. Lovász and A. Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2. Springer Science & Business Media, 2012. 32

[GM15]   S. Gaspers and S. Mackenzie. On the number of minimal separators in graphs. In *International Workshop on Graph-Theoretic Concepts in Computer Science*, pages 116–121. Springer, 2015. 41

[Gol04]   M. C. Golumbic. *Algorithmic graph theory and perfect graphs*, volume 57. Elsevier, 2004. 48

[GS17]   L. Gouveia and L. Simonetti. Spanning trees with a constraint on the number of leaves. a new formulation. *Comput. Oper. Res.*, 81(C):257–268, 2017. 20

[GWZ+10]   X. Gao, W. Wang, Z. Zhang, S. Zhu and W. Wu. A ptas for minimum d-hop connected dominating set in growth-bounded graphs. *Optimization Letters*, 4(3):321–333, 2010. 3, 18

[Gya98]   A. Gyarfas. Generalized split graphs and Ramsey numbers. *J. Combinatorial Theory, A*, 81(2):255–261, 1998. 53

[GYZ13]   J. L. Gross, J. Yellen and P. Zhang. *Handbook of Graph Theory, Second Edition*. Chapman & Hall/CRC, 2nd edition, 2013. 10

[HGC98]   Y. Hong-Gwa and G. J. Chang. Weighted connected domination and Steiner trees in distance-hereditary graphs. *Discrete Applied Mathematics*, 87(1):245–253, 1998. 15

[HHS97]   T. W. Haynes, S. Hedetniemi and P. Slater. *Domination in graphs: advanced topics*. Marcel Dekker, 1997. 1

[HHS98]   T.W. Haynes, S. Hedetniemi and P. Slater. *Fundamentals of Domination in Graphs*. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, 1998. 1, 19

[HK07]   P. Heggernes and D. Kratsch. Linear-time certifying recognition algorithms and forbidden induced subgraphs. *Nordic J. of Computing*, 14(1):87–108, 2007. 47

[HRG00]   M. R. Henzinger, S. Rao and H. N. Gabow. Computing vertex connectivity: New bounds from old techniques. *Journal of Algorithms*, 34(2):222–250, 2000. 33, 43

[HSV91]   D. J. Haglin and M. S. Venkatesan. Approximation and intractability results for the maximum cut problem and its variants. *IEEE Transactions on Computers*, 40(1):110–113, 1991. 42

[JPD17]   R. K. Jallu, P. R. Prasad and G. K. Das. Distributed construction of connected dominating set in unit disk graphs. *Journal of Parallel and Distributed Computing*, 2017. 18

[Kei93]   J. M. Keil. The complexity of domination problems in circle graphs. *Discrete Applied Mathematics*, 42(1):51–63, 1993. 3

[KK98]   T. Kloks and D. Kratsch. Listing all minimal separators of a graph. *SIAM Journal on Computing*, 27(3):605–613, 1998. 39

[KLMN11] M. Kanté, V. Limouzy, A. Mary and L. Nourine. Enumeration of minimal dominating sets and variants. In *Lecture Notes in Computer Science*, volume 6914, pages 298–309. Springer, 2011. 23, 24, 40, 58

[Klo96]   T. Kloks. Treewidth of circle graphs. *International Journal of Foundations of Computer Science*, 7(02):111–120, 1996. 39

[KY16]   S. Khuller and S. Yang. Revisiting Connected Dominating Sets: An Optimal Local Algorithml. In K. Jansen, C. Mathieu, J. D. P. Rolim and C. Umans, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 60, pages 11:1–11:12. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2016. 17

[Lia95]   Y. D. Liang. Steiner set and connected domination in trapezoid graphs. *Information Processing Letters*, 56(2):101–108, 1995. 15

[LMP+13] D. Lokshtanov, N. Misra, G. Philip, M. S. Ramanujan and S. Saurabh. Hardness of r-dominating set on graphs of diameter $(r + 1)$. In *International Symposium on Parameterized and Exact Computation*, pages 255–267. Springer, 2013. 19, 47, 48, 50

[LMS10]   A. Lucena, N. Maculan and L. Simonetti. Reformulations and solution algorithms for the maximum leaf spanning tree problem. *Computational Management Science*, 7(3):289–311, 2010. 20

[MB87]   H. Müller and A. Brandstädt. The NP-completeness of steiner tree and dominating set for chordal bipartite graphs. *Theoretical Computer Science*, 53(2):257–265, 1987. 3, 19

[Mos93]   M. Moscarini. Doubly chordal graphs, Steiner trees, and connected domination. *Networks*, 23(1):59–69, 1993. 15

[NH06a]   T. N. Nguyen and D. T. Huynh. Connected d-hop dominating sets in mobile ad hoc networks. In *Proc. of the 4th International Symposium on Modeling and Optimization in Mobile Ad Hoc and Wireless Networks*, pages 1–8, 2006. 19, 51

[NH06b]   T. Nieberg and J. Hurink. *A PTAS for the Minimum Dominating Set Problem in Unit Disk Graphs*, pages 296–306. Springer Berlin Heidelberg, 2006. 18

[NP06]   S. D. Nikolopoulos and L. Palios. Minimal separators in P_4-sparse graphs. *Discrete mathematics*, 306(3):381–392, 2006. 39

[Ore62]   O. Ore. *Theory of graphs*. Colloquium Publications - American Mathematical Society. American Mathematical Society, 1962. 1

[PLH83]   J. Pfaff, R. Laskar and S. T. Hedetniemi. NP-completeness of total and connected domination and irredundance for bipartite graphs. In *Tech. Rept. 428*. Clemson University Clemson, SC, 1983. 3

[RDJ+04] L. Ruan, H. Du, X. Jia, W. Wu, Y. Li and K. Ko. A greedy approximation for minimum connected dominating sets. *Theoretical Computer Science*, 329(1):325–330, 2004. 3, 16, 17, 38

[RLU15]   M.F. Reis, O. Lee and F.L. Usberti. Flow-based formulation for the maximum leaf spanning tree problem. *Electronic Notes in Discrete Mathematics*, 50:205–210, 2015. 20

[RR88]   G. Ramalingam and C. P. Rangan. A unified approach to domination problems on interval graphs. *Information Processing Letters*, 27(5):271–274, 1988. 15

[RZ11]   W. Ren and Q. Zhao. A note on Algorithms for connected set cover problem and fault-tolerant connected set cover problem. *Theoretical Computer Science*, 412(45):6451–6454, 2011. 18, 21, 37, 64

[Sch86]   A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1986. 11

[SdCL11] L. Simonetti, A. S. da Cunha and A. Lucena. The minimum connected dominating set problem: Formulation, valid inequalities and a branch-and-cut algorithm. In *Network Optimization*, pages 162–169. Springer, 2011. 3, 19

[SW79]   E. Sampathkumar and H. B. Walikar. The connected domination number of a graph. *J. Math. Phys. Sci*, 13(6):607–613, 1979. 2

[WAF02]   P. Wan, K. M. Alzoubi and O. Frieder. Distributed construction of connected dominating set in wireless ad hoc networks. In *Proceedings 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1597–1604, 2002. 18

[WFP85]   K. White, M. Farber and W. Pulleyblank. Steiner trees, connected domination and strongly chordal graphs. *Networks*, 15(1):109–124, 1985. 3, 15, 19, 22, 51, 63

[YWWY13] J. Yu, N. Wang, G. Wang and D. Yu. Connected dominating sets in wireless ad hoc and sensor networks - A comprehensive survey. *Computer Communications*, 36(2):121–134, 2013. 2

[YYSS15] A. K. Yadav, R. S. Yadav, R. Singh and A. K. Singh. Connected dominating set for wireless ad hoc networks: a survey. *International Journal of Engineering Systems Modelling and Simulation*, 7(1):22–34, 2015. 2

[ZGWD09] Z. Zhang, X. Gao, W. Wu and D. Du. A PTAS for minimum connected dominating set in 3-dimensional wireless sensor networks. *Journal of Global Optimization*, 45(3):451–458, 2009. 18