Minimum Edge Blocker Dominating Set Problem

Foad Mahdavi Pajouh,

Department of Management Science and Information Systems

University of Massachusetts Boston

Foad.Mahdavi@umb.edu

Jose L. Walteros,

Department of Industrial and Systems Engineering

University at Buffalo

josewalt@buffalo.edu

Vladimir Boginski,

Department of Industrial and Systems Engineering

University of Florida

vb@ufl.edu

Eduardo L. Pasiliao,

Munitions Directorate,

Air Force Research Laboratory

pasiliao@eglin.af.mil

Abstract

This paper introduces and studies the minimum edge blocker dominating set problem (EBDP), which is formulated as follows. Given a vertex-weighted undirected graph and $r > 0$, remove a minimum number of edges so that the weight of any dominating set in the remaining graph is at least r. Dominating sets are used in a wide variety of graph-based applications such as the analysis of wireless and social networks. We show that the decision version of EBDP is NP-hard for any fixed $r > 0$. We present an analytical lower bound for the value of an optimal solution to EBDP and formulate this problem as a linear 0–1 program with a large number of constraints. We also study the convex hull of feasible solutions to EBDP and identify facet-inducing inequalities for this polytope. Furthermore, we develop the first exact algorithm for solving EBDP, which solves the proposed formulation by a branch-and-cut approach where nontrivial constraints are applied in a lazy fashion. Finally, we also provide the computational results obtained by using our approach on a test-bed of randomly generated instances and real-life power-law graphs.

Keywords: network interdiction, minimum weighted dominating set, NP-hardness, branch-and-cut algorithm, critical elements detection

1 Introduction

In a real-world system modeled as a graph, one often needs to identify a set of vertices or edges whose existence is critical for preserving a particular property of the system. Detecting a set of critical vertices or edges with respect to a particular property of a graph can also lead to a cost-efficient interdiction of that property of the underlying system. Detection of critical vertices or edges may be

done within the following two different problem settings. In the first problem setting, we are interested in identifying a subset of vertices or edges of smallest cardinality whose removal bounds the considered graph property by some user-defined parameter. The second problem setting is to find a subset of vertices or edges of a given size whose deletion results in the largest change to the considered graph property.

In the related literature, the first problem is referred to as *minimum vertex/edge blocker problem* (Bazgan et al., 2011; Ries et al., 2010; Zenklusen et al., 2009) and the second one is called *most vital vertices/edges problem* (Bar-Noy et al., 1995; Bazgan et al., 2010a, 2011). In this paper, we introduce the problem of detecting a smallest subset of edges in a weighted undirected graph whose removal bounds the minimum weight of a dominating set in the remaining graph. Adopting the terminology used in the literature for similar problems (Bazgan et al., 2011; Ries et al., 2010; Zenklusen et al., 2009), this problem will be referred to as the *minimum edge blocker dominating set problem (EBDP)*. To avoid excessively long acronyms, we use EBDP rather than MEBDP, since the only non-trivial setup of the considered problem is minimization.

1.1   Problem description and motivation

Given a simple nonempty undirected graph $G = (V, E)$ with weights on its vertices, let $V = \{1, 2, ..., n\}$, $|E| = m$, and vector $\mathbf{w} = [w_i \in \mathbb{Q}]$ $(i = 1, ..., n)$ represent the weights of all vertices $i \in V$. Here, the set of edges in graph G is defined as $E = \{(i, j) : i, j \in V, i < j, \text{ and there exists an edge between } i \text{ and } j\}$. Note that graph G is undirected, hence notation (i, j) does not represent any order for vertices or direction for edges and condition $i < j$ is simply used to avoid including the same edge twice in set E.

For a vertex $i \in V$, let $N_G(i)$ denote the set of vertices adjacent to i in G (neighborhood of i in G). Given a set $D \subseteq V$, let $W(D) = \sum_{i \in D} w_i$ and $\mathcal{E}_G(D) = E \cap [(D \times (V \setminus D)) \cup ((V \setminus D) \times D)]$. Note that $W(\emptyset) = 0$. A dominating set in G is a set $D \subseteq V$ such that for any vertex $i \in V \setminus D$, $N_G(i) \cap D \neq \emptyset$. A minimum weighted dominating set in G with respect to the weight vector $\mathbf{w}$ is a dominating set $D \subseteq V$ such that $W(D)$ is minimum. The weight of a minimum weighted dominating set in G with respect to $\mathbf{w}$ is denoted by $\gamma^{\mathbf{w}}(G)$. If the weight of each vertex in V is one (unweighted graphs), the superscript $\mathbf{w}$ in $\gamma^{\mathbf{w}}(G)$ is omitted for simplicity. In this case, $\gamma(G)$ is also referred to as the domination number of G. The minimum weighted dominating set problem is to find a dominating set of minimum weight in G. For the special case of unweighted graphs, this problem is referred to as the minimum dominating set problem, which is a well-known NP-complete problem (Garey and Johnson, 1979). Finally, given a set $S \subseteq E$, let $G\langle S \rangle$ denote the subgraph with vertex set V and edge set S (equivalently, $G\langle S \rangle = (V, S)$).

The minimum edge blocker dominating set problem (EBDP) is defined as follows: Given $r > 0$, find $S \subseteq E$ such that $\gamma^{\mathbf{w}}(G\langle E \setminus S \rangle) \geq r$ and $|S|$ is minimum. Note that vertices with negative weights are always included in any minimum weighted dominating set. Similarly, vertices with zero weights are either in a minimum weighted dominating set or can be added to one such set resulting in an alternate minimum weighted dominating set. Additionally, if there is an edge $(i, j) \in E$ such that $\min\{w_i, w_j\} + W(V \setminus \{i, j\}) < r$, then any set $S \subseteq E$ with $\gamma^{\mathbf{w}}(G\langle E \setminus S \rangle) \geq r$ should contain edge (i, j). Finally, if $W(V) < r$, then EBDP is infeasible. To avoid these trivialities, the assumption in the remainder of this paper is that $w_i > 0$ for all $i \in V$, $\min\{w_i, w_j\} + W(V \setminus \{i, j\}) \geq r$ for all $(i, j) \in E$, and $W(V) \geq r$. Note that for $r \leq 0$, $S = \emptyset$ is optimal and this problem is again trivial.

The importance of EBDP can be viewed from both attack- and defense-related perspectives. In the case of network interdiction in these settings, the attacker's objective is to ensure that a minimum-cardinality set of edges is deleted (disabled) in order to increase the weight of the minimum weighted dominating set in the remaining network. The weight of the minimum weighted dominating set is an important characteristic of real-world networks in various contexts, such as analysis of wireless (Cheng

et al., 2003; Ding et al., 2011; Li et al., 2005; Wu et al., 2010) and social (Zhu et al., 2010; Zou et al., 2009) networks. The weight assigned to each vertex in a sensor or social network may be interpreted as the cost of including that vertex in the dominating set. Therefore, a network with a low-weight dominating set has more cost-efficient communication routines (e.g., low cost communication with all vertices in the network).

On the other hand, a defender wants to identify the set of edges whose deletion would substantially increase the weight of the minimum weighted dominating set in the remaining graph, so that these edges could be better protected. Identifying these critical edges is an important step in preserving the low-cost communication with all vertices in the network.

1.2 Previous work and our contributions

The minimum vertex/edge blocker and the most vital vertices/edges problems have been studied in literature with respect to different graph properties, such as connectivity (Addis et al., 2013; Arulselvan et al., 2009; Di Summa et al., 2011; Di Summa et al., 2012; Shen and Smith, 2012; Shen et al., 2012; Veremyev et al., 2014), shortest path (Bar-Noy et al., 1995; Israeli and Wood, 2002; Khachiyan et al., 2008), maximum flow (Wollmer, 1964; Ghare et al., 1971; Wood, 1993; Altner et al., 2010), spanning tree (Frederickson and Solis-Oba, 1996; Bazgan et al., 2012, 2013), assignment (Bazgan et al., 2010b), 1-median (Bazgan et al., 2010a), 1-center (Bazgan et al., 2010a), matching (Zenklusen et al., 2009; Ries et al., 2010; Zenklusen, 2010), independent sets (Bazgan et al., 2011), vertex covers (Bazgan et al., 2011), and cliques (Mahdavi Pajouh et al., 2014).

The computational complexity of the most vital edges shortest path problem has been studied in Bar-Noy et al. (1995) and it was shown to be NP-hard. Israeli and Wood (2002) formulated the most vital edges shortest path problem as a mixed-integer linear program. They also proposed two decomposition algorithms based on Benders and so-called covering decompositions to solve this problem. The most vital edges/vertices shortest path problems (and minimum vertex/edge shortest path blocker problems) have been proven to be hard to approximate within a factor of 2 (and within a factor of 1.36, respectively) if $P \neq NP$ (Khachiyan et al., 2008).

An algorithm for solving the most vital edges maximum flow problem, which works on a dual shortest path problem, was proposed by Wollmer (1964). A branch-and-bound algorithm for solving this problem was also developed by Ghare et al. (1971). The most vital edges maximum flow problem was later shown to be NP-hard (Wood, 1993). Wood (1993) also modeled and solved this problem and its variations by using integer programming techniques. Valid inequalities to tighten the linear programming relaxation of the proposed formulation were also derived in this work. Altner et al. (2010) presented two classes of polynomially separable valid inequalities for the most vital edges maximum flow problem. They proved that the integrality gap of the integer program introduced by Wood (1993) is not bounded by a constant, even when strengthened by these two classes of valid inequalities.

According to Frederickson and Solis-Oba (1996), the most vital edges spanning tree problem is NP-hard and $O(log\ k)$-approximable, where k is the maximum number of edges allowed to be deleted in this problem. The most vital edges assignment problem (and minimum edge assignment blocker problem) has been shown to be NP-hard, and hard to approximate within a factor of 2 (and within a factor of 1.36, respectively) if $P \neq NP$ (Bazgan et al., 2010b). Bazgan et al. (2010a) investigated the complexity of the most vital edges (vertices) 1-median (1-center) problem and the minimum edge (vertex) 1-median (1-center) blocker problem and showed that these problems are NP-hard to approximate within a factor c, for some $c > 1$.

The complexity of the minimum edge matching blocker problem was studied by Zenklusen et al. (2009) and it was shown that this problem is NP-hard and remains NP-hard even on unweighted bipartite graphs. This problem is polynomial time solvable on grids and trees (Ries et al., 2010). The most vital vertices matching problem is NP-hard on weighted bipartite graphs while it is polynomial

time solvable on unweighted graphs within this class (Zenklusen, 2010).

The most vital vertices independent set/vertex cover problems (and the minimum vertex independent set/vertex cover blocker problems) are NP-hard on weighted bipartite graphs but polynomial-time solvable on unweighted bipartite graphs (Bazgan et al., 2011). These problems are polynomial time solvable on weighted cographs and graphs of bounded treewidth (Bazgan et al., 2011).

Mahdavi Pajouh et al. (2014) showed that the minimum vertex clique blocker problem is NP-hard on unweighted graphs. An analytical lower bound for the value of an optimal solution to the weighted version of this problem was also proposed in this work. The authors formulated this problem as a linear 0–1 program with an exponential number of constraints and developed a row generation based exact algorithm for solving this formulation.

A closely related concept to EBDP on unweighted graphs is the *bondage number* of a graph. The bondage number of an unweighted graph G is the smallest number of edges whose removal from G results in a graph with domination number greater than the domination number of G. The concept of bondage number was originally defined by Bauer et al. (1983), where it was referred to as *domination line-stability*. Later, Fink et al. (1990) formally introduced the bondage number as a measure for evaluating robustness or vulnerability of any given graph in terms of preserving the size of the smallest dominating set under edge failures. Hu and Xu (2012) showed that the problem of determining the bondage number of a graph is NP-hard. Upper bounds and lower bounds on the bondage number of a graph along with different generalizations of this concept have also been proposed in literature (Xu, 2013).

Despite similarity of the bondage number's definition to EBDP on unweighted graphs, these two concepts have major differences. In EBDP on unweighted graphs, we are not only concerned with increasing the domination number but also increasing it to a value which is greater than or equal to a given $r > 0$. Furthermore, cumulative sum of recursively calculated bondage numbers needed to increase the domination number to the desired level ($\geq r$) may not be equal to the optimal solution of EBDP on the considered graph. We show this by the following example. Consider a graph $G = (V, E)$, where $V = \{1, 2, 3, 4, 5\}$ and $E = \{(1,2), (2,3), (2,4), (2,5), (3,5), (4,5)\}$, and let $r = 3$. Note that $\gamma(G) = 1$, bondage number of G is also 1 and $(2,4)$ is an edge whose removal increases the domination number to 2. The bondage number of $G\langle E \setminus \{(2,4)\}\rangle$ is 2 and one can remove edges $(1,2)$ and $(4,5)$ to increase the domination number to 3. On the other hand, the optimal solution to EBDP on G with $r = 3$ is 2 ($S^* = \{(1,2), (2,5)\}$), which is strictly smaller than the cumulative sum of the two previously found bondage numbers. It is also important to note that unlike the bondage number, EBDP is defined in the framework of weighted graphs in which increasing the weight of a minimum weighted dominating set to some specific level is desired.

The computational complexity of EBDP has not been addressed in the literature. Additionally, to the best of our knowledge, there are no exact algorithms for solving this problem on general weighted graphs. In Section 2, we study the computational complexity of EBDP and show that the decision version of this problem is NP-hard. This section also includes an analytical lower bound for the value of an optimal solution to EBDP. In Section 3, we characterize the set of feasible solutions to EBDP and formulate it as a linear 0–1 program, which involves a large number of constraints. We also study the convex hull of feasible solutions to EBDP in this section and identify facet inducing inequalities for this polytope. Section 4 presents the first exact algorithm for EBDP, which solves the proposed formulation by a branch-and-cut algorithm with lazy implementation of nontrivial constraints. Section 5 summarizes the computational results obtained by solving EBDP using the developed algorithm on a test-bed of randomly generated instances and real-life power-law networks. Finally, Section 6 provides concluding remarks and some directions for future research.

2 Computational complexity and analytical lower bound

According to Hu and Xu (2012), the problem of finding the bondage number of a graph is NP-hard. However, the complexity of EBDP for any fixed $r > 0$ has not been established in literature. In this section, we show that the decision version of EBDP is NP-hard.

Note that if r is part of the input, it is easy to show that the minimum dominating set problem is a special case of EBDP, which makes the latter NP-hard. Interestingly, a stronger result about the complexity of EBDP will be proven in this section: the problem remains NP-hard for any fixed $r > 0$.

For a fixed $r > 0$, the decision version of EBDP (denoted by $< G, \mathbf{w}, c >$) is given by a simple nonempty graph $G = (V, E)$ with positive weights $\mathbf{w}$ on its vertices and an integer $c \geq 0$, and we ask if there exists a set $S \subseteq E$ such that $|S| \leq c$ and $\gamma^{\mathbf{w}}(G\langle E \setminus S \rangle) \geq r$.

Theorem 1 *The decision version of EBDP is NP-hard for any fixed $r > 0$.*

For a fixed $r > 0$, the intractability of EBDP is proved by a polynomial-time reduction from minimum dominating set problem (Garey and Johnson, 1979). Given a simple nonempty graph $G = (V, E)$ and an integer $q > 0$, the decision version of the minimum dominating set problem (denoted by $< G, q >$) is defined as follows: is there a dominating set of size at most q in G? The minimum dominating set problem is known to be NP-complete (Garey and Johnson, 1979). For any such minimum dominating set decision problem $< G, q >$, we replicate graph G and assign a positive weight of $r/(q+1)$ to each vertex of this graph. Let us denote the vector containing all weights on vertices of G by $\mathbf{w} = [w_i]$, where $w_i = r/(q+1)$ for all $i \in \{1, ..., n\}$. This procedure constructs in polynomial time an instance $< G, \mathbf{w}, 0 >$ of EBDP such that there exists a dominating set of size at most q in G if and only if there does not exist a set $S \subseteq E$ with $|S| \leq 0$ such that $\gamma^{\mathbf{w}}(G\langle E \setminus S \rangle) \geq r$. The necessity and sufficiency of this statement can be easily verified due to the fact that by the construction, these two problems are equivalent. This establishes that the decision version of EBDP is NP-hard for any fixed $r > 0$ and completes the proof of Theorem 1.

Note that as shown in proof of Theorem 1, it is NP-hard to distinguish instances of EBDP with optimum equal to zero. This implies that EBDP cannot be approximated within any (even non-constant) factor, unless P = NP.

Considering the intractability of EBDP, it will be beneficial to find analytical bounds for the value of an optimal solution to this problem. A set $S \subseteq E$ is said to be a feasible solution to EBDP if and only if $\gamma^{\mathbf{w}}(G\langle E \setminus S \rangle) \geq r$. Here, finding a feasible solution of minimum cardinality is desired. In order to present our proposed analytical lower bound, we need to define notation $\kappa(D, r)$ for any set $D \subseteq V$ and $r > 0$ as follows.

Definition 1 *For any set $D \subseteq V$ and $r > 0$, the integer $\kappa(D, r)$ is the minimum cardinality of a set $D' \subseteq (V \setminus D)$ such that $W(D \cup D') \geq r$.*

For any given set $D \subseteq V$ and $r > 0$, $\kappa(D, r)$ can be calculated by using a greedy procedure that runs in $O(n \log n)$. This procedure adds vertices of set $V \setminus D$ to set D in decreasing order of their weights until the sum of the weights over all vertices in D becomes larger than or equal to r. At termination of this procedure, the number of vertices added to D is $\kappa(D, r)$. Note that sorting elements of $V \setminus D$ is the most expensive operation in this procedure, which can be done in $O(n \log n)$ number of principal operations (Knuth, 1998).

Using notation $\kappa(., r)$, Proposition 1 presents an analytical lower bound for the value of an optimal solution to EBDP.

Proposition 1 *Let D be a dominating set in graph G and z^* denote the cardinality of any optimal solution to EBDP in graph G. Then $z^* \geq \kappa(D, r)$.*

Suppose $z^* < \kappa(D, r)$ for some dominating set $D \subseteq V$ and let S^* denote an optimal solution to EBDP. Let $D' = \{i \in V \setminus D : N_{G\langle E \setminus S^* \rangle}(i) \cap D = \emptyset\}$. Note that $|D'| \leq z^*$, because removing each element of S^* from E adds at most one element to D'. Since $z^* < \kappa(D, r)$ and $|D'| \leq z^*$, then $|D'| < \kappa(D, r)$. Now, $D \cup D'$ is a dominating set in $G\langle E \setminus S^* \rangle$. Additionally, since $|D'| < \kappa(D, r)$, then $W(D \cup D') < r$. This contradicts with feasibility of S^*.

3 Feasible solutions characterization and linear 0–1 programming formulation

To formulate EBDP, we first provide a characterization of the set of feasible solutions to this problem. Then, we further improve this characterization by removing a set of redundant inequalities and use it to develop a linear 0–1 programming formulation for EBDP with a large number of constraints.

3.1 Characterization of the set of feasible solutions

In order to present the initial characterization of the set of feasible solutions to EBDP, we need to define the following notations. Let Ω_G denote the set containing all dominating sets in graph G. For a dominating set $D \in \Omega_G$, define a whip of D in graph G as follows.

Definition 2 *Given a dominating set $D \subseteq V$, set $H \subseteq E$ is called a whip of D in graph G if and only if $H \subseteq \mathcal{E}_G(D)$ and $|N_{G\langle H \rangle}(i)| = 1$ for all $i \in V \setminus D$. The set containing all whips of D in graph G is denoted by Φ_G^D.*

Note that a dominating set might have more than one whip and a whip might be associated with more than one dominating set. The initial characterization of the set of feasible solutions to EBDP is given by the following proposition.

Proposition 2 *A set $S \subseteq E$ is a feasible solution to EBDP if and only if*

$$\sum_{e \in H} z_e \geq \kappa(D, r) \quad \forall D \in \Omega_G \ \forall H \in \Phi_G^D, \tag{1}$$

where $\boldsymbol{z} = [z_1, ..., z_m] \in \{0, 1\}^m$ is the incidence vector of S.

To show necessity, suppose that $S \subseteq E$ is a feasible solution to EBDP, but there exists a dominating set $\bar{D} \in \Omega_G$ that has a whip $\bar{H} \in \Phi_G^{\bar{D}}$ for which $\sum_{e \in \bar{H}} z_e < \kappa(\bar{D}, r)$. This means that $|S \cap \bar{H}| < \kappa(\bar{D}, r)$. Let $\hat{D}$ denote the set containing all isolated vertices in $G\langle \bar{H} \setminus S \rangle$ that belong to $V \setminus \bar{D}$. Note that $|\hat{D}| = |S \cap \bar{H}| < \kappa(\bar{D}, r)$ and hence, $W(\bar{D} \cup \hat{D}) < r$. Since $\bar{D} \cup \hat{D}$ is a dominating set in $G\langle E \setminus S \rangle$, this contradicts with feasibility of S.

To show sufficiency, suppose $\sum_{e \in H} z_e \geq \kappa(D, r)$ for all $D \in \Omega_G$ and $H \in \Phi_G^D$ but S is not a feasible solution to EBDP. So, there exists a dominating set $\bar{D}$ in $G\langle E \setminus S \rangle$ with $W(\bar{D}) < r$. Consider an arbitrary whip $\bar{H} \in \Phi_{G\langle E \setminus S \rangle}^{\bar{D}}$. We know that $\sum_{e \in \bar{H}} z_e = 0$ (since $\bar{H} \subseteq (E \setminus S)$ and $z_e = 0 \ \forall e \in (E \setminus S)$) and $\kappa(\bar{D}, r) \geq 1$, so $\sum_{e \in \bar{H}} z_e < \kappa(\bar{D}, r)$. Since $\bar{D} \in \Omega_G$ and $\bar{H} \in \Phi_G^{\bar{D}}$, this contradicts with $\sum_{e \in H} z_e \geq \kappa(D, r)$ for all $D \in \Omega_G$ and $H \in \Phi_G^D$.

Remark 1 *An instance of inequality (1) written for a dominating set $D^0 \in \Omega_G$ and a whip $H^0 \in \Phi_G^{D^0}$ is referred to as "whip inequality of dominating set D^0 and its whip H^0" and is denoted by "(D^0, H^0)-inequality" in the remainder of this article.*

It turns out that the result of Proposition 2 can be improved by removing a set of redundant whip inequalities from this characterization. To specify this set of redundant whip inequalities, we need to provide the following definitions.

Definition 3 *Given $r > 0$, let $\Omega_G(r)$ denote the set containing all dominating sets D in graph G such that $\kappa(D, r) > \kappa(D \cup \{d\}, r)$, where d is a vertex with minimum weight in $V \setminus D$. Note that $V \notin \Omega_G(r)$.*

Definition 4 *Given a dominating set $D \subseteq V$ and $r > 0$, let $\Phi_G^D(r)$ denote the set containing all whips H of D in graph G that satisfy the following conditions:*

(i) *For any vertex $i \in D \setminus (\cup_{j \in V \setminus D} N_{G\langle H \rangle}(j))$, either $(D \setminus \{i\}) \notin \Omega_G$ or $\kappa(D \setminus \{i\}, r) = \kappa(D, r)$.*

(ii) *For any edge $e \in \mathcal{E}_G(D) \setminus H$, either $|N_{G\langle H \rangle}(a^e)| + |N_{G\langle H \rangle}(c^e)| \geq 2$ or $\kappa((D \setminus \{a^e, c^e\}) \cup \{b^e\}, r) \leq \kappa(D, r)$, where a^e is the endpoint of e in D, b^e is the endpoint of e in $V \setminus D$, and c^e is the unique neighbor of b^e in $G\langle H \rangle$.*

Note that in Condition (i) of Definition 4, statement $i \in D \setminus (\cup_{j \in V \setminus D} N_{G\langle H \rangle}(j))$ is equivalent to requiring that i is an isolated vertex in $G\langle H \rangle$, and condition $(D \setminus \{i\}) \notin \Omega_G$ is equivalent to $N_G(i) \cap (D \setminus \{i\}) = \emptyset$. Using Definitions 3 and 4, the following theorem identifies a set of redundant whip inequalities in the characterization presented in Proposition 2.

Theorem 2 *Given a whip $\tilde{H} \in (\cup_{D \in \Omega_G} \Phi_G^D) \setminus (\cup_{D \in \Omega_G(r)} \Phi_G^D(r))$ and a dominating set $\tilde{D} \in \Omega_G$ such that $\tilde{H} \in \Phi_G^{\tilde{D}}$, the $(\tilde{D}, \tilde{H})$-inequality used in Proposition 2 to characterize the set of feasible solutions to EBDP is redundant in presence of all (D, H)-inequalities in which $D \in \Omega_G(r)$ and $H \in \Phi_G^D(r)$, and may be removed from this characterization.*

Note that if $\kappa(\tilde{D}, r) = 0$, then the $(\tilde{D}, \tilde{H})$-inequality used in Proposition 2 is redundant because of the fact that $\mathbf{z} \in \{0, 1\}^m$. Now, suppose $\kappa(\tilde{D}, r) > 0$. Here, we show that there exist $D \in \Omega_G(r)$ and $H \in \Phi_G^D(r)$ such that the whip inequality of dominating set D and its whip H dominates $(\tilde{D}, \tilde{H})$-inequality. To this aim, we first need to devise a procedure that transforms a dominating set $\tilde{D} \in \Omega_G$ with $\kappa(\tilde{D}, r) > 0$ and a whip $\tilde{H} \in \Phi_G^{\tilde{D}}$ to a dominating set $D \in \Omega_G(r)$ and a whip $H \in \Phi_G^D(r)$. One such procedure is presented by Algorithm 1 (which will also be referred to as $\text{PROC}(G(V, E), \mathbf{w}, r, \tilde{D}, \tilde{H})$). This algorithm starts by letting $D = \tilde{D}$ and $H = \tilde{H}$, and then recursively adding vertices with minimum weight in set $V \setminus D$ to set D while removing the edges incident to them from whip H until the condition of Definition 3 holds for D. Then, all vertices in $D \setminus (\cup_{j \in V \setminus D} N_{G\langle H \rangle}(j))$ for which Condition (i) of Definition 4 is violated are recursively removed from D while arbitrary edges in $\mathcal{E}_G(D)$ that are incident to them (one edge for each vertex being removed) are added to whip H. Finally, each edge e in $\mathcal{E}_G(D) \setminus H$ that violates Condition (ii) of Definition 4 is added to whip H while a^e and c^e are removed from D and b^e is added to this set (a^e, b^e and c^e are the vertices defined in Definition 4). This whole process is repeated until $D \in \Omega_G(r)$ and $H \in \Phi_G^D(r)$. Claim 1 shows that Algorithm 1 terminates in polynomial time. The proof of Claim 1 is provided in Appendix.

Claim 1 *The worst-case running time for Algorithm 1 is $O(n^4 + m^2 n^2)$.*

Given a whip $\tilde{H} \in (\cup_{D \in \Omega_G} \Phi_G^D) \setminus (\cup_{D \in \Omega_G(r)} \Phi_G^D(r))$ and a dominating set $\tilde{D} \in \Omega_G$ such that $\kappa(\tilde{D}, r) > 0$ and $\tilde{H} \in \Phi_G^{\tilde{D}}$, the (D, H)-inequality associated with the dominating set D and whip H returned at termination of Algorithm 1 dominates $(\tilde{D}, \tilde{H})$-inequality. This can be shown as follows. Let us refer to while loops in Lines 4-26, 5-13, 14-19 and 20-26 as *main loop*, *loop 1*, *loop 2* and *loop 3*, respectively. After each iteration of loop 1, the left-hand side of the (D, H)-inequality has one less variable compared to the left-hand side of the whip inequality associated with these sets before that

Algorithm 1 A procedure for transforming a dominating set $\tilde{D} \in \Omega_G$ with $\kappa(\tilde{D}, r) > 0$ and a whip $\tilde{H} \in \Phi_G^{\tilde{D}}$ to a dominating set $D \in \Omega_G(r)$ and a whip $H \in \Phi_G^D(r)$.

Input: Graph $G = (V, E)$ with weights $\mathbf{w} \in \mathbb{Q}_{>0}^n$ on its vertices, $r > 0$, a dominating set $\tilde{D} \in \Omega_G$ with $\kappa(\tilde{D}, r) > 0$, and a whip $\tilde{H} \in \Phi_G^{\tilde{D}}$.
Output: A dominating set $D \in \Omega_G(r)$ and a whip $H \in \Phi_G^D(r)$.
1: **procedure** PROC($G(V, E), \mathbf{w}, r, \tilde{D}, \tilde{H}$)
2:      $D \leftarrow \tilde{D}$ and $H \leftarrow \tilde{H}$.
3:      $D' \leftarrow D \setminus (\cup_{j \in V \setminus D} N_{G\langle H\rangle}(j))$.
4:      **while** $D \notin \Omega_G(r)$ or $H \notin \Phi_G^D(r)$ **do**
5:         **while** $D \notin \Omega_G(r)$ **do**   ▷ Checking the condition of Definition 3.
6:             Let d denote a vertex with minimum weight in $V \setminus D$.
7:             $D \leftarrow (D \cup \{d\})$.
8:             Let (d', d) denote the unique edge in H that is incident to d.
9:             $H \leftarrow (H \setminus \{(d', d)\})$.
10:            **if** $N_{G\langle H\rangle}(d') = \emptyset$ **then**
11:                $D' \leftarrow (D' \cup \{d', d\})$.
12:            **else**
13:                $D' \leftarrow (D' \cup \{d\})$.
14:         **while** $\exists i \in D'$: Condition (i) in Definition 4 is violated for i **do**
15:            $D \leftarrow (D \setminus \{i\})$, $D' \leftarrow (D' \setminus \{i\})$.
16:            Let e_i denote an arbitrary edge in $\mathcal{E}_G(D)$ that is incident to i.
17:            $H \leftarrow (H \cup \{e_i\})$.
18:            Let j denote the vertex in D that is incident to e_i.
19:            $D' \leftarrow (D' \setminus \{j\})$
20:         **while** $\exists e \in \mathcal{E}_G(D) \setminus H$: Condition (ii) in Definition 4 is violated for e **do**
21:            Let a^e denote the endpoint of e in D.
22:            Let b^e denote the endpoint of e in $V \setminus D$.
23:            Let c^e denote the unique neighbor of b^e in $G\langle H\rangle$.
24:            $D \leftarrow (D \setminus \{a^e, c^e\}) \cup \{b^e\}$.
25:            $D' \leftarrow (D' \setminus \{a^e\})$
26:            $H \leftarrow (H \cup \{e\})$.
27:      **return** D and H.

---

iteration, while the right-hand side of both inequalities are the same. Thus, the (D, H)-inequality obtained after each iteration of loop 1 dominates (or coincides with) the whip inequality associated with these sets before that iteration. Similarly, after each iteration of loop 2 (or loop 3), the left-hand side of the (D, H)-inequality has one more variable compared to the left-hand side of the whip inequality associated with these sets before that iteration, while its right-hand side is also larger by at least one unit. This again means that the (D, H)-inequality obtained after each iteration of loop 2 (or loop 3) dominates (or coincides with) the whip inequality associated with these sets before that iteration. This shows that the (D, H)-inequality associated with the dominating set D and whip H returned at termination of Algorithm 1 dominates $(\tilde{D}, \tilde{H})$-inequality. So, $(\tilde{D}, \tilde{H})$-inequality may be removed from characterization of Proposition 2.

As a result of Theorem 2, Corollary 1 presents a more compact characterization of the set of feasible solutions to EBDP.

Corollary 1 *A set $S \subseteq E$ is a feasible solution to EBDP if and only if*

$$\sum_{e \in H} z_e \geq \kappa(D, r) \quad \forall D \in \Omega_G(r) \ \forall H \in \Phi_G^D(r), \tag{2}$$

where $\mathbf{z} = [z_1, ..., z_m] \in \{0, 1\}^m$ is the incidence vector of S.

Remark 2 *Whip inequalities of type (2) are referred to as "r-critical-whip inequalities" in the remainder of this article.*

3.2   Convex hull of the incidence vectors of feasible solutions

In this section, we study the convex hull of the incidence vectors of all feasible solutions to EBDP in order to discover important properties of this polytope. These properties may be beneficial in the development of integer programming techniques to solve EBDP. Let $\mathcal{P}_G(r)$ denote the convex hull of the incidence vectors of all feasible solutions to EBDP on G for a given $r > 0$. Here, we show that $\mathcal{P}_G(r)$ is a full-dimensional polytope and the upper-bound constraint on each decision variable ($z_e \leq 1$, for each edge $e \in E$) induces a facet of $\mathcal{P}_G(r)$. Additionally, for the case of unweighted graphs ($\mathbf{w} = \mathbf{1}$), we develop necessary and sufficient conditions under which r-critical-whip inequalities and non-negativity constraints are facet inducing for $\mathcal{P}_G(r)$. These results are presented in the following theorem.

Theorem 3 *Given a simple nonempty graph $G = (V, E)$ with weights $\mathbf{w}$ on its vertices and $r > 0$, the following statements are true.*

(a) *$\mathcal{P}_G(r)$ is a full-dimensional polytope.*

(b) *Given an edge $e \in E$, inequality $z_e \leq 1$ induces a facet of $\mathcal{P}_G(r)$.*

(c) *For the case of unweighted graphs ($\mathbf{w} = \mathbf{1}$), the (D, H)-inequality associated with a dominating set $D \in \Omega_G(r)$ and a whip $H \in \Phi_G^D(r)$ induces a facet of $\mathcal{P}_G(r)$ if and only if*

   (c.1) *$H = E$, or*

   (c.2) *$|V| \geq r + 2$.*

(d) *For the case of unweighted graphs ($\mathbf{w} = \mathbf{1}$), inequality $z_e \geq 0$ for an edge $e \in E$ induces a facet of $\mathcal{P}_G(r)$ if and only if*

   (d.1) *$E = \{e\}$, or*

   (d.2) *$|V| \geq r + 2$.*

(a) To show that $\mathcal{P}_G(r)$ is a full-dimensional polytope, it is enough to show that there are $m + 1$ affinely independent points in $\mathcal{P}_G(r)$. Since $\min\{w_i, w_j\} + W(V \setminus \{i, j\}) \geq r$ for all $(i, j) \in E$ and $W(V) \geq r$, then sets $E \setminus \{e\}$ for all $e \in E$ along with set E are feasible solutions to the corresponding EBDP. It can be easily verified that the incidence vectors of these feasible solutions, which belong to $\mathcal{P}_G(r)$, are affinely independent. Hence, $\mathcal{P}_G(r)$ is a full-dimensional polytope.

   (b) Given an edge $e \in E$, it is enough to show that the dimension of the face of $\mathcal{P}_G(r)$ induced by $z_e \leq 1$ is $m - 1$. The face of $\mathcal{P}_G(r)$ induced by $z_e \leq 1$ is

$$\mathcal{P}_G^e(r) = \{\mathbf{z} \in \mathcal{P}_G(r) : z_e = 1\}.$$

Note that the dimension of $\mathcal{P}_G^e(r)$ is at most $m - 1$. Hence, it is enough to show that there are m affinely independent points in $\mathcal{P}_G^e(r)$. Since $\min\{w_i, w_j\} + W(V \setminus \{i, j\}) \geq r$ for all $(i, j) \in E$ and $W(V) \geq r$, then sets $E \setminus \{e'\}$ for all $e' \in E \setminus \{e\}$ along with set E are feasible solutions to the corresponding EBDP. The incidence vectors of these sets are affinely independent and belong to $\mathcal{P}_G^e(r)$. Thus, the dimension of $\mathcal{P}_G^e(r)$ is $m - 1$.

   (c) To show necessity, suppose the (D, H)-inequality associated with a dominating set $D \in \Omega_G(r)$ and a whip $H \in \Phi_G^D(r)$ in an unweighted graph induces a facet of $\mathcal{P}_G(r)$ but $E \setminus H \neq \emptyset$ and $|V| < r + 2$. Since $|V| \geq r + 1$ ($\min\{w_i, w_j\} + W(V \setminus \{i, j\}) \geq r$ for all $(i, j) \in E$), then $|V| = \lceil r \rceil + 1$ and

9

$|H| = \kappa(D,r) + 1$. Additionally, one can easily verify that a set $S \subseteq E$ is feasible to this EBDP if and only if $|S| \geq m - 1$.

The face of $\mathcal{P}_G(r)$ induced by the (D,H)-inequality is

$$\mathcal{P}_G^{D,H}(r) = \{\mathbf{z} \in \mathcal{P}_G(r) : \sum_{e \in H} z_e = \kappa(D,r)\}.$$

Here, since $\mathcal{P}_G^{D,H}(r)$ is a facet of $\mathcal{P}_G(r)$, then it is of dimension $m - 1$. Additionally, since $\mathcal{P}_G(r)$ is bounded, then $\mathcal{P}_G^{D,H}(r)$ is a bounded polyhedron of dimension $m - 1$. Let $n(\mathcal{P}_G^{D,H}(r))$ denote the number of extreme points of $\mathcal{P}_G^{D,H}(r)$. Dimension of $\mathcal{P}_G^{D,H}(r)$ is the dimension of its affine hull (by definition). Since $\mathcal{P}_G^{D,H}(r)$ is bounded, then its affine hull is the affine hull of its extreme points (as all other points of $\mathcal{P}_G^{D,H}(r)$ are convex combinations and thus affine combinations of its extreme points). The dimension of this affine hull cannot exceed $n(\mathcal{P}_G^{D,H}(r)) - 1$ (as any affine space spanned by k vectors has dimension at most $k - 1$). Thus, $n(\mathcal{P}_G^{D,H}(r)) \geq m$. Note that all extreme points of $\mathcal{P}_G^{D,H}(r)$ are the incidence vectors of some feasible solutions to EBDP. Therefore, there are at least m incidence vectors of feasible solutions to EBDP that belong to $\mathcal{P}_G^{D,H}(r)$.

Let $\mathbf{z}^0$ denote the incidence vector of a feasible solution to EBDP that belongs to $\mathcal{P}_G^{D,H}(r)$. Since $|H| = \kappa(D,r) + 1$, $\sum_{e \in H} z_e^0 = \kappa(D,r)$, and $\sum_{e \in E} z_e^0 \geq m - 1$, then $\sum_{e \in E \setminus H} z_e^0 = |E \setminus H|$. Therefore, the number of incidence vectors of feasible solutions to EBDP that belong to $\mathcal{P}_G^{D,H}(r)$ is $|H|$. Since $|H| < m$, this contradicts with existence of at least m incidence vectors of feasible solutions to EBDP that belong to $\mathcal{P}_G^{D,H}(r)$.

To show sufficiency, it is enough to show that given an unweighted graph G, $r > 0$, a dominating set $D \in \Omega_G(r)$ and a whip $H \in \Phi_G^D(r)$ that satisfy at least one of conditions (c.1) or (c.2), the dimension of $\mathcal{P}_G^{D,H}(r)$ is $m - 1$. Since the dimension of $\mathcal{P}_G^{D,H}(r)$ is at most $m - 1$, it is enough to show that there are m affinely independent points in $\mathcal{P}_G^{D,H}(r)$. Without loss of generality, assume the edges in H are numbered from 1 to $|H|$. For each edge $\varepsilon \in E$, construct a set I^ε as follows:

If $\varepsilon \in H$ and $1 \leq \varepsilon \leq \kappa(D,r) - 1$, then let

$$I^\varepsilon = E \setminus (\{\kappa(D,r) + 2, ..., |H|\} \cup \{\varepsilon\}).$$

If $\varepsilon \in H$ and $\kappa(D,r) \leq \varepsilon \leq |H|$, then let

$$I^\varepsilon = \{\varepsilon\} \cup (E \setminus \{\kappa(D,r), ..., |H|\}).$$

If $\varepsilon \in E \setminus H$, then first select a set $S^\varepsilon \subseteq H$ using the following approach. If $\varepsilon \in E \setminus H$ and both endpoints of ε (denoted by a^ε and b^ε) belong to D, then $|N_{G\langle H \rangle}(a^\varepsilon)| \geq 1$ and $|N_{G\langle H \rangle}(b^\varepsilon)| \geq 1$ because otherwise, Condition (i) of Definition 4 is not valid for D. In this case, let e_1 and e_2 denote two arbitrary edges in H that are incident to a^ε and b^ε, respectively. Select an arbitrary set $S^\varepsilon \subseteq (H \setminus \{e_1, e_2\})$ such that $|S^\varepsilon| = \kappa(D,r)$ (note that since condition (c.2) is valid, such a set S^ε exists). If $\varepsilon \in E \setminus H$ and only one endpoint of ε belongs to D (suppose $a^\varepsilon \in D$ and $b^\varepsilon \in V \setminus D$), then let d be an arbitrary vertex in $(N_{G\langle H \rangle}(a^\varepsilon) \cup N_{G\langle H \rangle}(c^\varepsilon)) \setminus \{b^\varepsilon\}$, where c^ε is the unique neighbor of b^ε in $G\langle H \rangle$. Note that vertex d exists because $\varepsilon \in \mathcal{E}_G(D) \setminus H$ and by Condition (ii) of Definition 4, $|N_{G\langle H \rangle}(a^\varepsilon)| + |N_{G\langle H \rangle}(c^\varepsilon)| \geq 2$. If $d \in N_{G\langle H \rangle}(a^\varepsilon)$, then select an arbitrary set $S^\varepsilon \subseteq (H \setminus \{(a^\varepsilon, d), (b^\varepsilon, c^\varepsilon)\})$ such that $|S^\varepsilon| = \kappa(D,r)$; otherwise, select an arbitrary set $S^\varepsilon \subseteq (H \setminus \{(c^\varepsilon, d), (b^\varepsilon, c^\varepsilon)\})$ with $|S^\varepsilon| = \kappa(D,r)$. Note that since condition (c.2) is valid, such a set S^ε exists in both cases. Finally, if $\varepsilon \in E \setminus H$ and both endpoints of ε (denoted by a^ε and b^ε) belong to $V \setminus D$, then let e_1 and e_2 denote the edges in H that are incident to a^ε and b^ε, respectively. Select an arbitrary set $S^\varepsilon \subseteq (H \setminus \{e_1, e_2\})$ such that $|S^\varepsilon| = \kappa(D,r)$ (note that since condition (c.2) is valid, such a set S^ε exists). After selecting set S^ε using the approach mentioned for any of these three cases, let

$$I^\varepsilon = S^\varepsilon \cup (E \setminus (H \cup \{\varepsilon\})).$$

10

Here, we show that for any edge $\varepsilon \in E$, set I^ε is a feasible solution to EBDP. To this aim, let $\hat{D}$ denote the set containing all isolated vertices in $G\langle E \setminus I^\varepsilon \rangle$ that belong to set $V \setminus D$. If $\varepsilon \in H$, then in $G\langle E \setminus I^\varepsilon \rangle$, there does not exist any pair of vertices in D that belong to the same connected component. Additionally, we have $|\hat{D}| = \kappa(D, r)$. Hence, $\gamma(G\langle E \setminus I^\varepsilon \rangle) \geq |D| + \kappa(D, r) \geq r$ and set I^ε is a feasible solution to EBDP. If $\varepsilon \in E \setminus H$ and at least one of its endpoints belongs to D, then in $G\langle E \setminus I^\varepsilon \rangle$, there is exactly one pair of vertices in D that belong to the same connected component (denoted by $\tilde{G}$). It can be easily verified that $\gamma(\tilde{G}) \geq 2$. Since $|\hat{D}| = \kappa(D, r)$, then $\gamma(G\langle E \setminus I^\varepsilon \rangle) \geq (|D| - 2) + 2 + \kappa(D, r) \geq r$ and set I^ε is a feasible solution to EBDP. If $\varepsilon \in E \setminus H$ and both endpoints of ε belong to $V \setminus D$, then either there does not exist any pair of vertices in D that belong to the same connected component or there is exactly one pair of vertices in D that belong to the same connected component (denoted by $\tilde{G}$). For both cases, $|\hat{D}| = \kappa(D, r)$. For the first case, $\gamma(G\langle E \setminus I^\varepsilon \rangle) \geq |D| + \kappa(D, r) \geq r$. In the second case, it can be verified that $\gamma(\tilde{G}) \geq 2$ and hence, $\gamma(G\langle E \setminus I^\varepsilon \rangle) \geq (|D| - 2) + 2 + \kappa(D, r) \geq r$. Therefore, set I^ε is again a feasible solution to EBDP. This means for each $\varepsilon \in E$, the incidence vector of I^ε (denoted by $\mathbf{z}^\varepsilon$) belongs to $\mathcal{P}_G(r)$. Additionally, $\sum_{e \in H} z_e^\varepsilon = \kappa(D, r)$ and hence, $\mathbf{z}^\varepsilon \in \mathcal{P}_G^{D,H}(r)$. It can be easily verified that vectors $\mathbf{z}^\varepsilon$ for all $\varepsilon \in E$ are affinely independent. So, the dimension of $\mathcal{P}_G^{D,H}(r)$ is $m - 1$.

(d) To show necessity, suppose inequality $z_e \geq 0$ for an edge $e \in E$ in an unweighted graph induces a facet of $\mathcal{P}_G(r)$ but $E \setminus \{e\} \neq \emptyset$ and $|V| < r + 2$. Since $|V| \geq r + 1$ ($\min\{w_i, w_j\} + W(V \setminus \{i, j\}) \geq r$ for all $(i, j) \in E$), then $|V| = \lceil r \rceil + 1$.

The face of $\mathcal{P}_G(r)$ induced by $z_e \geq 0$ is

$$\mathcal{P}_G^e(r) = \{\mathbf{z} \in \mathcal{P}_G(r) : z_e = 0\}.$$

Here, since $\mathcal{P}_G^e(r)$ is a facet of $\mathcal{P}_G(r)$, then it is of dimension $m - 1$. Additionally, since $\mathcal{P}_G(r)$ is bounded, then $\mathcal{P}_G^e(r)$ is a bounded polyhedron of dimension $m - 1$. As shown in proof of part (c), we have $n(\mathcal{P}_G^e(r)) \geq m$. Note that all extreme points of $\mathcal{P}_G^e(r)$ are the incidence vectors of some feasible solutions to EBDP. Therefore, there are at least m incidence vectors of feasible solutions to EBDP that belong to $\mathcal{P}_G^e(r)$.

Let $\mathbf{z}^1$ denote the incidence vector of a feasible solution to EBDP that belongs to $\mathcal{P}_G^e(r)$. Since $\mathbf{w} = \mathbf{1}$, $|V| = \lceil r \rceil + 1$ and $z_e^1 = 0$, then it can be easily verified that $\sum_{\varepsilon \in E \setminus \{e\}} z_\varepsilon^1 = m - 1$. Therefore, the number of incidence vectors of feasible solutions to EBDP that belong to $\mathcal{P}_G^e(r)$ is one. Since $1 < m$, this contradicts with existence of at least m incidence vectors of feasible solutions to EBDP that belong to $\mathcal{P}_G^e(r)$.

To show sufficiency, it is enough to show that given an unweighted graph G, $r > 0$ and an edge $e \in E$ that satisfy at least one of conditions (d.1) or (d.2), the dimension of $\mathcal{P}_G^e(r)$ is $m - 1$. Since the dimension of $\mathcal{P}_G^e(r)$ is at most $m - 1$, it is enough to show that there are m affinely independent points in $\mathcal{P}_G^e(r)$. For each edge $\varepsilon \in E$, construct a set I^ε as follows:

$$I^\varepsilon = E \setminus (\{e\} \cup \{\varepsilon\}).$$

Here, we show that for any edge $\varepsilon \in E$, set I^ε is a feasible solution to EBDP. Since $|V| - 1 \geq r$ ($\min\{w_i, w_j\} + W(V \setminus \{i, j\}) \geq r$ for all $(i, j) \in E$), then I^e is a feasible solution to EBDP. If $\varepsilon \in E \setminus \{e\}$, then $|V| \geq r + 2$ (by condition (d.2)) and it can be easily verified that I^ε is also a feasible solution to EBDP. This means for each $\varepsilon \in E$, the incidence vector of I^ε (denoted by $\mathbf{z}^\varepsilon$) belongs to $\mathcal{P}_G(r)$. Additionally, for each $\varepsilon \in E$, we have $z_e^\varepsilon = 0$ and hence, $\mathbf{z}^\varepsilon \in \mathcal{P}_G^e(r)$. It can be easily verified that vectors $\mathbf{z}^\varepsilon$ for all $\varepsilon \in E$ are affinely independent. So, the dimension of $\mathcal{P}_G^e(r)$ is $m - 1$.

3.3   Linear 0–1 programming formulation

Using the notations and results mentioned above, we can now formulate EBDP as a linear 0-1 program. Specifically, as it follows from Corollary 1, EBDP can be formulated as follows:

11

$$\min \sum_{e \in E} z_e \tag{3}$$

subject to

$$\sum_{e \in H} z_e \geq \kappa(D, r) \quad \forall D \in \Omega_G(r), H \in \Phi_G^D(r) \tag{4}$$

$$z_e \in \{0, 1\} \quad \forall e \in E, \tag{5}$$

where $\mathbf{z} = [z_e]$ in $\{0, 1\}^m$ is the incidence vector of a set $S \subseteq E$, $\kappa(., r)$ is the notation defined in Section 2, and finally, $\Omega_G(r)$ and $\Phi_G^D(r)$ are the sets defined in Section 3.1.

It should be mentioned that adding the whip inequality associated with a whip $\tilde{H} \in (\cup_{D \in \Omega_G} \Phi_G^D) \setminus (\cup_{D \in \Omega_G(r)} \Phi_G^D(r))$ and a dominating set $\tilde{D} \in \Omega_G$ such that $\tilde{H} \in \Phi_G^{\tilde{D}}$ to Formulation (3)-(5) will not improve the quality of the lower bound obtained by solving the linear programming relaxation of this formulation. As a result of Theorem 2, it can be verified that the corresponding $(\tilde{D}, \tilde{H})$-inequality is valid for any feasible solution to the linear programming relaxation of Formulation (3)-(5).

The number of r-critical-whip inequalities in a graph can be very large. As a result, Formulation (3)-(5) may involve a large number of constraints, and dealing with this formulation may require extensive computational effort. In the next section, we develop a branch-and-cut approach that applies the r-critical-whip inequalities in a lazy fashion. This allows one to reduce this computational effort and solve EBDP on moderate-size graph instances.

4 Row generation based exact algorithm

In this section, we develop the first exact algorithm for solving EBDP, which solves Formulation (3)-(5) by a branch-and-cut approach with lazy implementation of r-critical-whip inequalities (constraints of type (4)). Before presenting the details of this branch-and-cut algorithm, let us introduce the following notations. Let F^T denote the formulation associated with node T of the search tree and $\mathbf{z}^T$ denote an optimal solution to this formulation. Additionally, for an integral $\mathbf{z}^T$, let $S(\mathbf{z}^T)$ denote the set whose incidence vector is $\mathbf{z}^T$.

At the beginning of the branch-and-cut algorithm, the formulation associated with the root node of the search tree is the linear programming relaxation of Formulation (3)-(5), from which all r-critical-whip inequalities have been removed. The incumbent solution is also initialized to set E (since $W(V) \geq r$, then E is a feasible solution to EBDP). When node T of the search tree is selected for processing, we first solve formulation F^T. If F^T is infeasible, then node T is fathomed by infeasibility. If F^T is feasible, then $\sum_{e \in E} \mathbf{z}_e^T$ is a valid lower bound for the cardinality of the best solution found at the subtree rooted at node T. If $\sum_{e \in E} \mathbf{z}_e^T$ is not smaller than the cardinality of the incumbent solution, then node T is fathomed by bound. Otherwise, if $\mathbf{z}^T$ is not integral, then we branch by variable dichotomy and two children nodes are added to the set of unprocessed tree nodes. If $\mathbf{z}^T$ is integral, then feasibility of $S(\mathbf{z}^T)$ to EBDP is checked by solving the minimum weighted dominating set problem on $G\langle E \setminus S(\mathbf{z}^T)\rangle$. If $S(\mathbf{z}^T)$ is feasible to EBDP ($\gamma^{\mathbf{w}}(G\langle E \setminus S(\mathbf{z}^T)\rangle) \geq r$), then the present tree node is fathomed by feasibility and the incumbent solution is updated to $S(\mathbf{z}^T)$. Otherwise, a violated r-critical-whip inequality is detected and added to formulation F^T as a lazy constraint, and node T is processed again. This algorithm terminates after processing all unprocessed nodes in the search tree and the cardinality of the incumbent solution will be the optimal objective to EBDP on graph G.

For an integral $\mathbf{z}^T$, the minimum weighted dominating set problem on $G\langle E \setminus S(\mathbf{z}^T)\rangle$ used in our

exact algorithm to verify the feasibility of set $S(\mathbf{z}^T)$ is formulated as

$$\min \sum_{i \in V} w_i x_i \tag{6}$$

subject to

$$x_i + \sum_{j \in N_{G'}(i)} x_j \geq 1 \quad \forall i \in V \tag{7}$$

$$x_i \in \{0,1\} \quad \forall i \in V \tag{8}$$

where $G' = G\langle E \setminus S(\mathbf{z}^T)\rangle$. In case set $S(\mathbf{z}^T)$ is infeasible to EBDP ($\gamma^{\mathbf{w}}(G\langle E \setminus S(\mathbf{z}^T)\rangle) < r$), we detect an r-critical-whip inequality that is violated by $\mathbf{z}^T$ as follows. Let $\tilde{D}$ denote an optimal solution to the minimum weighted dominating set problem on $G\langle E \setminus S(\mathbf{z}^T)\rangle$. Consider an arbitrary whip $\tilde{H} \in \Phi^{\tilde{D}}_{G\langle E \setminus S(\mathbf{z}^T)\rangle}$. Note that $\tilde{H} \in \Phi^{\tilde{D}}_G$, $\kappa(\tilde{D}, r) > 0$ and $(\tilde{D}, \tilde{H})$-inequality is violated by $\mathbf{z}^T$ (since the left-hand side of this inequality equals zero using $\mathbf{z}^T$ while the right-hand side is positive). Using Algorithm 1 with sets $\tilde{D}$ and $\tilde{H}$ as its inputs, we can detect a dominating set $D \in \Omega_G(r)$ and a whip $H \in \Phi^D_G(r)$ such that (D, H)-inequality is violated by $\mathbf{z}^T$. Note that (D, H)-inequality dominates $(\tilde{D}, \tilde{H})$-inequality (as shown in proof of Theorem 2) and hence, (D, H)-inequality is also violated by $\mathbf{z}^T$.

5 Numerical results

This section presents the computational results obtained by solving EBDP on a test-bed of instances that includes Uniform Random Graphs (URG) and real-life power-law instances. We first used the exact algorithm proposed in Section 4 to solve these problem instances and then, we utilized a modified version of this algorithm to study the effect of strengthening detected violated whip inequalities by Algorithm 1 in performance of our proposed exact algorithm. In this modified version, for each integral $\mathbf{z}^T$ with an infeasible $S(\mathbf{z}^T)$, the violated whip inequality associated with a minimum weighted dominating set in $G\langle E \setminus S(\mathbf{z}^T)\rangle$ and one of its whips in this graph is directly added to formulation F^T without being improved (transformed to an r-critical-whip inequality) by Algorithm 1. In other words, the modified algorithm employs inequalities of type (1) to characterize the set of feasible solutions to EBDP instead of using inequalities of type (2). Employing violated inequalities of type (1) requires less computational effort but these inequalities are weaker compared to inequalities of type (2). One of the objectives of our numerical experiments in this section is to study the effect of this trade-off on performance of the exact algorithm for solving EBDP.

The computational experiments were performed on a server with two AMD Opteron™ 6128 Eight-Core CPUs and 12 gigabytes of RAM, running Linux x86_64, CentOS 5.9. All algorithms were implemented in C++ and Gurobi® Optimizer 5.6.2 was used to implement the proposed exact algorithm and its modified version. Each instance of Formulation (6)-(8) (used to verify the feasibility of an integral solution in both versions) was also solved by this solver with default settings.

The test-bed of instances consisted of uniform random graphs with 25 vertices and a group of real-life power-law networks. For the URG instances, the probability for existence of an edge between any pair of vertices (expected edge density) was chosen to be $p \in \{0.1, 0.2, 0.5, 0.7, 0.9\}$. We generated both unweighted and weighted URG instances. For the weighted instances, the weight assigned to each vertex is a uniformly distributed integer between 1 and 10. For each expected edge density p, we generated 10 unweighted and 10 weighted graph instances, resulting in a total of 100 instances. The value of parameter r was chosen from $\{8, 12, 16, 20\}$ for unweighted instances, and from $\{40, 60, 80, 100\}$ for weighted ones. The real-life network instances used in our experiments were collected from the University of Florida Sparse Matrix Collection database (Davis and Hu, 2011). The

graphs considered are `karate_34` (a social network from a karate club with 34 members at a U.S. university (Zachary, 1977)), `chesapeake_39` (a 39-vertex graph from the seasonal dynamics of the Chesapeake Bay ecosystem (Baird and Ulanowicz, 1989)), `dolphins_62` (a social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand (Lusseau et al., 2003)), and `terror_kreb_62` (a 62-member terrorist network from the hijackers responsible for the 9/11 attacks, compiled by Krebs (2002)). We also developed the weighted version of each one of these instances by assigning a uniformly distributed integer between 1 and 10 to each vertex of that instance as its weight. The value of parameter r for real-life network instances was chosen from {10, 15, 20, 25, 30, 40, 45, 50} for unweighted instances, and from {60, 90, 120, 150, 180, 240, 270, 300} for weighted ones. The running time limit for each instance of EBDP was set at 3600 seconds.

5.1 Results for the Uniform Random Graph (URG) instances

Table 1 shows percentage of instances solved to optimality (*Solved (%)*), average running time (*Time (s)*), average best solution found (*BSF*), and average optimality gap (*Gap(%)*) obtained by using the proposed exact algorithm on the test-bed of URG instances within 3600 seconds time limit. Table 2 shows the values of the aforementioned attributes acquired by employing the modified version of the proposed exact algorithm on the same test-bed within the same time limit.

Table 1: Percentage of instances solved to optimality (*Solved (%)*), average running time (*Time (s)*), average best solution found (*BSF*), and average optimality gap (*Gap(%)*) obtained by using the proposed exact algorithm on the test-bed of URG instances within 3600 seconds time limit. For a given edge density p and r, if there is at least one instance (out of 10) that was not solved to optimality within this time limit, then the corresponding number reported for *Time (s)* is a valid lower bound and for *BSF* is a valid upper bound on these criteria.

	Density	r (Unweighted)				r (Weighted)			
	(p)	8	12	16	20	40	60	80	100
	0.1	100	100	100	100	100	100	100	100
	0.2	100	90	40	20	100	80	10	40
Solved (%)	0.5	0	0	0	0	0	0	0	0
	0.7	0	0	0	0	0	0	0	0
	0.9	0	0	0	0	0	0	0	0
	0.1	0.0	0.2	2.3	4.2	0.0	2.8	5.1	9.0
	0.2	7.0	428.2	2509.6	3049.3	41.5	1071.4	3242.6	2567.8
Time (s)	0.5	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	0.7	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	0.9	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	0.1	0.2	4.7	11.7	20.5	1.2	5.0	9.8	16.2
	0.2	4.6	17.0	31.4	45.0	5.2	13.5	24.7	33.4
BSF	0.5	59.1	103.6	130.6	140.4	54.9	79.0	102.3	135.4
	0.7	106.6	155.5	185.5	199.2	85.8	127.9	156.9	179.9
	0.9	151.7	212.0	246.6	258.7	108.0	152.8	193.1	223.8
	0.1	0	0	0	0	0	0	0	0
	0.2	0	1	4	5	0	2	15	7
Gap (%)	0.5	41	40	33	18	53	45	38	35
	0.7	50	42	32	19	55	50	40	27
	0.9	52	44	34	19	54	48	41	32

For a given edge density p, the problem instances corresponding to very small or very large values of r are expected to be simpler compared to the instances associated with moderate values of this parameter. This intuition is justified by the fact that the number of binary vectors satisfying the constraint associated with a given dominating set D and one of its whips H (constraint of type (2) for Table 1 and constraint of type (1) for Table 2) is greater than or equal to $\binom{|H|}{\kappa(D,r)}$. For a given D

Table 2: Percentage of instances solved to optimality (*Solved (%)*), average running time (*Time (s)*), average best solution found (*BSF*), and average optimality gap (*Gap(%)*) obtained by using the modified version of the proposed exact algorithm on the test-bed of URG instances within 3600 seconds time limit. In this modified version, for each integral $\mathbf{z}^T$ with an infeasible $S(\mathbf{z}^T)$, the violated whip inequality associated with a minimum weighted dominating set in $G\langle E \setminus S(\mathbf{z}^T)\rangle$ and one of its whips in this graph is directly added to formulation F^T without being improved (transformed to an r-critical-whip inequality) by Algorithm 1. For a given edge density p and r, if there is at least one instance (out of 10) that was not solved to optimality within this time limit, then the corresponding number reported for *Time (s)* is a valid lower bound and for *BSF* is a valid upper bound on these criteria.

	Density	*r (Unweighted)*				*r (Weighted)*			
	(p)	*8*	*12*	*16*	*20*	*40*	*60*	*80*	*100*
	0.1	100	100	100	90	100	100	100	100
	0.2	100	60	0	0	100	70	0	0
Solved (%)	*0.5*	0	0	0	0	0	0	0	0
	0.7	0	0	0	0	0	0	0	0
	0.9	0	0	0	0	0	0	0	0
	0.1	0.0	0.4	12.9	392.7	0.1	3.5	16.9	302.1
	0.2	4.4	1481.6	3600.0	3600.0	59.5	1248.2	3600.0	3600.0
Time (s)	*0.5*	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	0.7	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	0.9	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	0.1	0.2	4.7	11.7	20.5	1.2	5.0	9.8	16.2
	0.2	4.6	17.1	40.1	47.5	5.2	17.7	33.3	47.0
BSF	*0.5*	75.7	126.0	136.4	140.9	74.0	112.0	150.5	145.1
	0.7	106.6	155.5	185.7	200.7	86.5	127.9	156.9	178.7
	0.9	151.7	212.0	246.6	259.0	108.0	152.8	193.1	224.6
	0.1	0	0	0	1	0	0	0	0
	0.2	0	3	25	15	0	9	32	37
Gap (%)	*0.5*	47	50	36	19	63	60	60	42
	0.7	50	42	33	20	61	54	43	28
	0.9	53	44	34	19	58	52	44	35

and H, as r increases, $\kappa(D, r)$ increases and the quantity $\binom{|H|}{\kappa(D,r)}$ increases, reaches a maximum and then decreases. This suggests that the size of the feasible region for the corresponding EBDP also increases, reaches a maximum and then decreases. The results presented in Tables 1 and 2 do not contradict this intuition. Specifically, this trend is observed for the optimality gap for the weighted instances of density $p = 0.2$ in Table 1, and unweighted instances of densities $p = 0.2$ and $p = 0.5$ in Table 2. It is important to note that as the edge density increases, the maximum optimality gap over all values of parameter r appears to happen at a smaller value of this parameter. This shift in the value of parameter r with the maximum optimality gap is the reason behind the fact that we do not observe this peak in optimality gap across all values of chosen densities. Similar behavior is also expected and justified for *Solved (%)* and *Time (s)* attributes in these tables.

With regard to the *BSF* attribute in Tables 1 and 2, for a given edge density p, as r increases, the average size of the best solution found also increases. Similarly, for a given r, as edge density p increases, the *BSF* attribute also increases. This behavior is very intuitive because as parameter r (or edge density p) increases, the number of constraints of type (2) for Table 1 and constraints of type (1) for Table 2 increases. This means the size of the feasible region for the corresponding EBDP decreases and the optimal objective value increases.

Comparing the results shown in Table 1 with the ones presented in Table 2, it can be concluded that the proposed exact algorithm outperforms the modified version of this algorithm on URG instances in all cases. This means improving the violated whip inequalities by transforming them into violated r-critical-whip inequalities by Algorithm 1 significantly improves the performance of the branch-and-cut algorithm on these instances.

5.2   Results for the power-law instances

Tables 3 and 4 show the running time (*Time (s)*), best solution found (*BSF*), and optimality gap (*Gap(%)*) obtained by using the proposed exact algorithm and its modified version on the test-bed of unweighted power-law instances within 3600 seconds time limit, respectively. Tables 5 and 6 show the values of the aforementioned attributes acquired by employing the proposed exact algorithm and its modified version on the test-bed of weighted power-law instances within the same time limit, respectively.

Table 3: Running time (*Time (s)*), best solution found (*BSF*), and optimality gap (*Gap(%)*) obtained by using the proposed exact algorithm on the test-bed of unweighted power-law instances within 3600 seconds time limit. For a given instance and r, if the problem was not solved to optimality within this time limit, then the corresponding number reported for *Time (s)* is a valid lower bound and for *BSF* is a valid upper bound on these criteria.

	Instance	r (Unweighted)							
	Instance	*10*	*15*	*20*	*25*	*30*	*40*	*45*	*50*
	karate_34	1.2	5.5	35.5	866.0	76.8	-	-	-
Time (s)	chesapeake_39	420.8	3600.0	3600.0	3600.0	3600.0	-	-	-
	dolphins_62	0.0	0.0	1.9	1219.2	3600.0	3600.0	3600.0	3600.0
	terror_kreb_62	0.0	0.0	0.1	116.0	371.2	3600.0	3600.0	3600.0
	karate_34	7	17	31	49	66	-	-	-
BSF	chesapeake_39	22	44	94	114	134	-	-	-
	dolphins_62	0	1	7	20	59	78	124	117
	terror_kreb_62	0	5	10	20	33	64	82	106
	karate_34	0	0	0	0	0	-	-	-
Gap (%)	chesapeake_39	0	9	31	19	11	-	-	-
	dolphins_62	0	0	0	0	42	12	29	7
	terror_kreb_62	0	0	0	0	0	5	4	6

Table 4: Running time (*Time (s)*), best solution found (*BSF*), and optimality gap (*Gap(%)*) obtained by using the modified version of the proposed exact algorithm on the test-bed of unweighted power-law instances within 3600 seconds time limit. In this modified version, for each integral $\mathbf{z}^T$ with an infeasible $S(\mathbf{z}^T)$, the violated whip inequality associated with a minimum weighted dominating set in $G\langle E \setminus S(\mathbf{z}^T)\rangle$ and one of its whips in this graph is directly added to formulation F^T without being improved (transformed to an r-critical-whip inequality) by Algorithm 1. For a given instance and r, if the problem was not solved to optimality within this time limit, then the corresponding number reported for *Time (s)* is a valid lower bound and for *BSF* is a valid upper bound on these criteria.

					r (Unweighted)				
	Instance	10	15	20	25	30	40	45	50
	karate_34	16.9	5.1	3600.0	3600.0	3600.0	-	-	-
Time (s)	chesapeake_39	3600.0	3600.0	3600.0	3600.0	3600.0	-	-	-
	dolphins_62	0.0	0.0	4.1	3600.0	3600.0	3600.0	3600.0	3600.0
	terror_kreb_62	0.0	0.0	0.1	3600.0	3600.0	3600.0	3600.0	3600.0
	karate_34	7	17	32	49	66	-	-	-
BSF	chesapeake_39	23	44	94	114	134	-	-	-
	dolphins_62	0	1	7	22	59	105	124	135
	terror_kreb_62	0	5	10	20	42	65	83	115
	karate_34	0	0	3	6	5	-	-	-
Gap (%)	chesapeake_39	9	9	33	21	13	-	-	-
	dolphins_62	0	0	0	18	47	38	32	23
	terror_kreb_62	0	0	0	5	29	9	8	16

Table 5: Running time (*Time (s)*), best solution found (*BSF*), and optimality gap (*Gap(%)*) obtained by using the proposed exact algorithm on the test-bed of weighted power-law instances within 3600 seconds time limit. For a given instance and r, if the problem was not solved to optimality within this time limit, then the corresponding number reported for *Time (s)* is a valid lower bound and for *BSF* is a valid upper bound on these criteria.

					r (Weighted)				
	Instance	60	90	120	150	180	240	270	300
	karate_34	185.1	3600.0	3600.0	3600.0	3600.0	-	-	-
Time (s)	chesapeake_39	3600.0	3600.0	3600.0	3600.0	3600.0	-	-	-
	dolphins_62	0.0	1.7	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	terror_kreb_62	0.0	46.5	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	karate_34	9	25	46	58	67	-	-	-
BSF	chesapeake_39	41	62	91	119	143	-	-	-
	dolphins_62	0	4	11	30	49	83	110	125
	terror_kreb_62	1	6	22	34	42	78	92	111
	karate_34	0	36	41	28	6	-	-	-
Gap (%)	chesapeake_39	54	47	44	37	26	-	-	-
	dolphins_62	0	25	18	50	49	35	34	25
	terror_kreb_62	0	0	50	47	36	32	27	20

Table 6: Running time ($Time$ (s)), best solution found (BSF), and optimality gap ($Gap(\%)$) obtained by using the modified version of the proposed exact algorithm on the test-bed of weighted power-law instances within 3600 seconds time limit. In this modified version, for each integral $\mathbf{z}^T$ with an infeasible $S(\mathbf{z}^T)$, the violated whip inequality associated with a minimum weighted dominating set in $G\langle E \setminus S(\mathbf{z}^T)\rangle$ and one of its whips in this graph is directly added to formulation F^T without being improved (transformed to an r-critical-whip inequality) by Algorithm 1. For a given instance and r, if the problem was not solved to optimality within this time limit, then the corresponding number reported for $Time$ (s) is a valid lower bound and for BSF is a valid upper bound on these criteria.

		r (Weighted)							
	Instance	60	90	120	150	180	240	270	300
	karate_34	317.2	3600.0	3600.0	3600.0	3600.0	-	-	-
Time (s)	chesapeake_39	3600.0	3600.0	3600.0	3600.0	3600.0	-	-	-
	dolphins_62	0.0	0.5	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	terror_kreb_62	0.0	52.6	3600.0	3600.0	3600.0	3600.0	3600.0	3600.0
	karate_34	9	31	39	58	71	-	-	-
BSF	chesapeake_39	41	62	91	119	143	-	-	-
	dolphins_62	0	3	11	30	49	83	110	125
	terror_kreb_62	1	6	22	34	42	78	92	111
	karate_34	0	48	33	31	21	-	-	-
Gap (%)	chesapeake_39	56	48	45	42	30	-	-	-
	dolphins_62	0	0	18	50	49	39	36	28
	terror_kreb_62	0	0	50	47	36	35	28	24

According to these tables, similar to the observation made for the URG instances (with some exceptions), for a given instance, as r increases, the optimality gap increases, reaches a maximum, and then decreases. Therefore, the problem instances corresponding to moderate values of r are more challenging compared to the ones associated with very smaller or very larger values of this parameter. This behavior of the optimality gap for a given instance can be explained by the same logic used for the URG instances. The value of parameter r at which the optimality gap reaches its maximum depends on the instance and varies from one instance to another. Similar behavior is observed for $Time$ (s) attribute in these tables. Considering BSF attribute, for each instance, as parameter r increases, the size of the best solution found also increases. This observation is again similar to the one made for URG instances and can be justified using a similar discussion.

Comparing the results shown in Table 3 with the ones presented by Table 4 (and the results shown in Table 5 with the ones presented by Table 6), it is observed that the proposed exact algorithm performs better than the modified version of this algorithm on power-law instances as well. This again indicates that improving the violated whip inequalities by Algorithm 1 and transforming them into violated r-critical-whip inequalities improves the performance of the proposed branch-and-cut approach.

6   Conclusion

In this article, we studied the minimum edge blocker dominating set problem (EBDP), which is the problem of removing a subset of edges of minimum cardinality such that the minimum weight of dominating sets in the remaining graph is bounded below by a given $r > 0$. The motivation of EBDP from a network interdictor's perspective is to optimally choose a set of edges to delete (disable) in order to ensure that the remaining network does not have a dominating set of a total weight less than r.

We showed that the decision version of EBDP is NP-hard for any fixed $r > 0$. An analytical lower bound on the value of an optimal solution to EBDP was obtained. Also, EBDP was formulated as

a linear 0–1 program with a large number of constraints. Moreover, facet-inducing inequalities for the convex hull of feasible solutions to EBDP were identified. The proposed formulation was then solved by a branch-and-cut approach where nontrivial constraints are imposed in a lazy fashion. This algorithm is the first known exact algorithm for EBDP. The performance of this exact algorithm was also studied on a test-bed of uniform random instances and some real-life power-law networks. It turned out that EBDP is rather computationally challenging for moderate values of r, whereas it can be solved in a reasonable time for very small or very large values of this parameter. It was also observed that strengthening the violated constraints detected at each node of the search tree by the approach presented in this article (Algorithm 1) improves the performance of the proposed branch-and-cut approach. For potential future studies, it would be beneficial to consider EBDP under the assumption that there is a deletion cost associated with each edge. Studying EBDP on graphs subject to probabilistic vertex/edge failures is another interesting challenge.

7   Acknowledgements

References

Addis, B., Summa, M. D., Grosso, A., 2013. Identifying critical nodes in undirected graphs: Complexity results and polynomial algorithms for the case of bounded treewidth. Discrete Applied Mathematics 161 (1617), 2349 – 2360.

Altner, D. S., Ergun, O., Uhan, N. A., 2010. The maximum flow network interdiction problem: Valid inequalities, integrality gaps, and approximability. Operations Research Letters 38 (1), 33–38.

Arulselvan, A., Commander, C. W., Elefteriadou, L., Pardalos, P. M., 2009. Detecting critical nodes in sparse graphs. Computers and Operations Research 36 (7), 2193–2200.

Baird, D., Ulanowicz, R. E., 1989. The seasonal dynamics of the Chesapeake bay ecosystem. Ecological Monographs 59 (4), 329–364.

Bar-Noy, A., Khuller, S., Schieber, B., 1995. The complexity of finding most vital arcs and nodes. Tech. Rep. CS-TR-3539, Department of Computer Science, University of Maryland.

Bauer, D., Harary, F., Nieminen, J., Suffel, C. L., 1983. Domination alteration sets in graphs. Discrete Mathematics 47 (2–3), 153–161.

Bazgan, C., Toubaline, S., Tuza, Z., 2011. The most vital nodes with respect to independent set and vertex cover. Discrete Applied Mathematics 159 (17), 1933–1946.

Bazgan, C., Toubaline, S., Vanderpooten, D., 2010a. Complexity of determining the most vital elements for the 1-median and 1-center location problems. In: Proceedings of the 4th Annual International Conference on Combinatorial Optimization and Applications. Vol. 6508. COCOA, pp. 237–251.

Bazgan, C., Toubaline, S., Vanderpooten, D., 2010b. Détermination des éléments les plus vitaux pour le problème d'affectation. In: Actes de la 10ème Conférence de la Société Franaise de Recherche Opérationelle et d'Aide à la Décision. ROADEF, pp. 1–2.

Bazgan, C., Toubaline, S., Vanderpooten, D., 2012. Efficient determination of the k most vital edges for the minimum spanning tree problem. Computers & Operations Research 39 (11), 2888 – 2898.

Bazgan, C., Toubaline, S., Vanderpooten, D., 2013. Critical edges/nodes for the minimum spanning tree problem: complexity and approximation. Journal of Combinatorial Optimization 26 (1), 178–189.

Cheng, X., Huang, X., Li, D., Wu, W., Du, D. Z., 2003. A polynomial-time approximation scheme for the minimum-connected dominating set in ad hoc wireless networks. Networks 42 (4), 202–208.

Davis, T. A., Hu, Y., 2011. The University of Florida sparse matrix collection. ACM Transactions on Mathematical Software 38 (1), 1–25.

Di Summa, M., Grosso, A., Locatelli, M., 2011. Complexity of the critical node problem over trees. Computers and Operations Research 38 (12), 1766–1774.

Di Summa, M., Grosso, A., Locatelli, M., 2012. Branch and cut algorithms for detecting critical nodes in undirected graphs. Computational Optimization and Applications 53 (3), 649–680.

Ding, L., Gao, X., Wu, W., Lee, W., Zhu, X., Du, D. Z., 2011. An exact algorithm for minimum CDS with shortest path constraint in wireless networks. Optimization Letters 5 (2), 297–306.

Fink, J. F., Jacobson, M. S., Kinch, L. F., Roberts, J., 1990. The bondage number of a graph. Discrete Mathematics 86 (1–3), 47–57.

Frederickson, G. N., Solis-Oba, R., 1996. Increasing the weight of minimum spanning trees. In: Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms. SODA, pp. 539–546.

Garey, M. R., Johnson, D., 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Company, New York.

Ghare, P. M., Montgomery, D. C., Turner, W. C., 1971. Optimal interdiction policy for a flow network. Naval Research Logistics Quarterly 18 (1), 37–45.

Hu, F.-T., Xu, J.-M., 2012. On the complexity of the bondage and reinforcement problems. Journal of Complexity 28 (2), 192–201.

Israeli, E., Wood, R. K., 2002. Shortest-path network interdiction. Networks 40 (2), 97–111.

Khachiyan, L., Boros, E., Borys, K., Elbassioni, K., Gurvich, V., Rudolf, G., Zhao, J., 2008. On short paths interdiction problems: Total and node-wise limited interdiction. Theory of Computing Systems 43 (2), 204–233.

Knuth, D., 1998. The Art of Computer Programming, Volume 3: (2Nd Ed.) Sorting and Searching. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA.

Krebs, V. E., 2002. Uncloaking terrorist networks. First Monday 7 (4).

Li, Y., Thai, M. T., Wang, F., Yi, C. W., Wan, P. J., Du, D. Z., 2005. On greedy construction of connected dominating sets in wireless networks. Wireless Communications and Mobile Computing 5, 927–932.

Lusseau, D., Schneider, K., Boisseau, O. J., Haase, P., Slooten, E., Dawson, S. M., 2003. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behavioral Ecology and Sociobiology 54 (4), 396–405.

Mahdavi Pajouh, F., Boginski, V., Pasiliao, E. L., 2014. Minimum vertex blocker clique problem. Networks 64, 48–64.

Ries, B., Bentz, C., Picouleau, C., de Werra, D., Costa, M., Zenklusen, R., 2010. Blockers and transversals in some subclasses of bipartite graphs: When caterpillars are dancing on a grid. Discrete Mathematics 310 (11), 132–146.

Shen, S., Smith, J. C., 2012. Polynomial-time algorithms for solving a class of critical node problems on trees and series-parallel graphs. Networks 60 (2), 103–119.

Shen, S., Smith, J. C., Goli, R., 2012. Exact interdiction models and algorithms for disconnecting networks via node deletions. Discrete Optimization 9 (3), 172 – 188.

Veremyev, A., Prokopyev, O. A., Pasiliao, E. L., 2014. An integer programming framework for critical elements detection in graphs. Journal of Combinatorial Optimization 28, 233 – 273.

Wollmer, R., 1964. Removing arcs from a network. Operations Research 12 (6), 934–940.

Wood, R. K., 1993. Deterministic network interdiction. Mathematical and Computer Modeling 17 (2), 1–18.

Wu, W., Gao, X., Pardalos, P. M., Du, D. Z., 2010. Wireless networking, dominating and packing. Optimization Letters 4 (3), 347–358.

Xu, J.-M., 2013. On bondage numbers of graphs: A survey with some comments. International Journal of Combinatorics 2013, 1–34.

Zachary, W. W., 1977. An information flow model for conflict and fission in small groups. Journal of anthropological research 33 (4), 452–473.

Zenklusen, R., 2010. Matching interdiction. Discrete Applied Mathematics 158 (15), 1676–1690.

Zenklusen, R., Ries, B., Picouleau, C., de Werra, D., Costa, M., Bentz, C., 2009. Blockers and transversals. Discrete Mathematics 309 (13), 4306–4314.

Zhu, X., Yu, J., Lee, W., Kim, D., Shan, S., Du, D. Z., 2010. New dominating sets in social networks. Journal of Global Optimization 48 (4), 633–642.

Zou, F., Zhang, Z., Wu, W., 2009. Latency-bounded minimum influential node selection in social networks. In: Wireless Algorithms, Systems, and Applications. Vol. 5682 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 519–526.

Appendix

Proof of Claim 1

Claim 1 *The worst-case running time for Algorithm 1 is $O(n^4 + m^2 n^2)$.*

Let us refer to while loops in Lines 4-26, 5-13, 14-19 and 20-26 as *main loop, loop 1, loop 2* and *loop 3*, respectively. Before executing Line 3, we also store D and H as two boolean arrays of sizes n and m, respectively. By using such arrays, checking membership of a vertex in D or an edge in H can be done in constant time. Constructing these arrays takes $O(n) + O(m)$. Moreover, we sort the elements of $V \setminus D$ in decreasing order of their weights and calculate $\kappa(D, r)$, which can be done in $O(n \log n)$.

During the execution of Line 3, we generate D' and $G\langle H\rangle$ by traversing H. Since $|H|$ is $O(n)$, then these can also be done in $O(n)$. We store D' as a boolean array as well. The following shows the worst-case running time for each iteration of the main loop.

While executing Line 4 of Algorithm 1, condition of Definition 3 can be checked in $O(1)$, Condition (i) of Definition 4 can be verified in $O(n(n+1)) = O(n^2)$ (for each $i \in D'$, finding if $(D \setminus \{i\}) \in \Omega_G$ takes $O(n)$ by checking if i has a neighbor in D; placing i in $V \setminus D$ while keeping this set sorted in decreasing order of its elements weights can be done in $O(n)$; and verifying if $\kappa(D \setminus \{i\}, r) = \kappa(D, r)$ takes $O(1)$), and Condition (ii) of Definition 4 can be checked in $O(m(n+1)) = O(mn)$ (for each edge e, finding if $e \in \mathcal{E}_G(D) \setminus H$ can be done in O(1); getting a^e, b^e, and c^e takes $O(1)$; placing a^e and c^e in $V \setminus D$ while keeping this set sorted in decreasing order of its elements weights can be done in $O(n)$; and verifying if $\kappa((D \setminus \{a^e, c^e\}) \cup \{b^e\}, r) = \kappa(D, r)$ takes $O(1)$). Loop 1 terminates in $O(n)$, Loop 2 will terminate in $O(n^3)$, and the worst-case running time for loop 3 is $O(m^2n)$. So, the worst-case running time for each iteration of the main loop is $O(n^3 + m^2n)$.

If in an iteration of the main loop, none of loops 2 and 3 are executed, then we refer to that iteration as type A; otherwise, that iteration is referred to as type B. Note that after an iteration of type A, Algorithm 1 terminates. Hence, Algorithm 1 terminates either after a series of iterations of type B, or after a series of iterations of type B and one iteration of type A (this case contains the scenario in which this algorithm terminates after one iteration that is of type A). Let n_B denote the total number of iterations of type B of the main loop throughout Algorithm 1. Note that set D obtained after an iteration of type B has larger $\kappa(.,r)$ compared to the set D used at the beginning of that iteration. Since $\kappa(.,r)$ is $O(n)$, then n_B is $O(n)$. Therefore, the total number of iterations of the main loop is $O(n+1) = O(n)$. Finally, the worst-case running time for Algorithm 1 is $O(n(n^3 + m^2n)) = O(n^4 + m^2n^2)$.