Dominating an s-t-Cut in a Network

Ralf Rothenberger[1], Sascha Grau[2], and Michael Rossberg[2]

[1] Friedrich-Schiller-Universität Jena
ralf.rothenberger@uni-jena.de
[2] Technische Universität Ilmenau
first.lastname@tu-ilmenau.de

Abstract. We study an optimization problem with applications in design and analysis of resilient communication networks: given two vertices s, t in a graph $G = (V, E)$, find a vertex set $X \subset V$ of minimum cardinality, such that X and its neighborhood constitute an s-t vertex separator. Although the problem naturally combines notions of graph connectivity and domination, its computational properties significantly differ from these relatives.

In particular, we show that on general graphs the problem cannot be approximated to within a factor of $2^{\log^{1-\delta} n}$, with $\delta = 1/\log\log^c n$ and arbitrary $c < 1/2$ (if P $\neq$ NP). This inapproximability result even applies if the subgraph induced by a solution set has the additional constraint of being connected. Furthermore, we give a $2\sqrt{n}$-approximation algorithm and study the problem on graphs with bounded node degree. With Δ being the maximum degree of nodes $V \setminus \{s, t\}$, we identify a $(\Delta + 1)$ approximation algorithm.

Keywords: graph theory, approximation algorithms, inapproximability

1 Introduction

In recent years, the development of secure overlay networks has strongly advanced (e.g. [1,2,3]). As a consequence, we are approaching a situation, where the effort an attacker needs to spend on identifying worthwile targets may exceed the costs of mounting the actual attack. This is especially true, since huge botnets, which are able to conduct massive denial-of-service attacks, can be cheaply rent on the internet. In contrast, while actively observing a network node will reliably reveal its communication partners, it might be connected to a risk of detection, a risk of failure or a considerable amount of resources necessary to obtain the involved nodes' addresses.

Motivated by these facts, we study a problem that we term as Cut Domination: given a graph G and a pair of particular nodes s and t, we

Published at SOFSEM2015. The original publication is available at www.springerlink.com (http://dx.doi.org/10.1007/978-3-662-46078-8_33).
Ralf Rothenberger was supported by the German Federal Ministry of Education and Research.

seek to select a node set X of minimum cardinality such that the nodes in X and their neighborhood constitute an s-t-vertex-separator.

This problem is posed to an attacker possessing knowledge about the network topology, but not about actual addresses of the participants (needed to mount the attack). Examples for such settings are virtual private networks with dynamic routing [1], where the data itself is encrypted and authenticated, but denial-of-service attacks may still cause a severe thread.

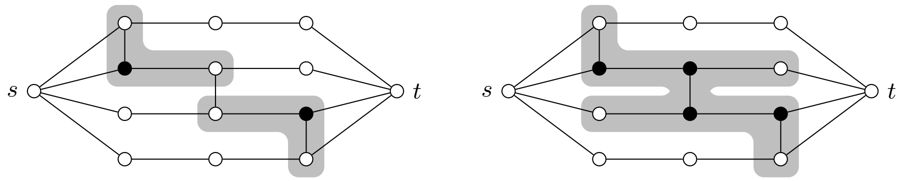
A related theoretical problem was first described in [4,5], where, given graph G and a budget $x \leq n$, the number of node pairs rendered unreachable by removing x nodes and their respective neighborhood was to be maximized. The derived decision problem was shown to be NP-complete, but needs to be solved in order to build more resilient networks.

Cut Domination differs from this problem as only paths between a specific node pair s, t are to be disrupted. One can imagine s and t as important and well-equipped communication partners, leaving the intermediate network nodes as easier targets for an attack. Although the related network formation problem is easy to solve (connect s and t by as many isolated, parallel paths as possible), the properties of Cut Domination are interesting in their own respect and might provide insight into the practically-motivated problem described in [4,5].

Our contributions are the following: After introducing a problem formalization in Sec. 2, we show in Sec. 3 that generally the Cut Domination problem cannot be approximated to within a factor of $2^{\log^{1-\delta} n}$, with $\delta = 1/\log\log^c n$ and arbitrary $c < 1/2$ (if P $\neq$ NP). This result also holds if the observed node set has to be connected (Connected Cut Domination) and for the respective weighted variant (Weighted Cut Domination). In Sec. 4 we give a $2\sqrt{n}$-approximation algorithm for Weighted Cut Domination, which can also be used to approximate Connected Cut Domination to within a ratio of $n^{2/3}$ of an optimal solution. Finally, in Sec. 5, we show a $(\Delta+1)$-approximation algorithm for Weighted Cut Domination, with Δ denoting the maximum degree of nodes $V \setminus \{s,t\}$. Since Cut Domination is a special case of Weighted Cut Domination, all upper bound results for the weighted variant also apply to the unweighted version.

2 Problem Definition & Notation

To formalize the studied optimization problem, we first introduce necessary notation: For an undirected graph $G = (V, E)$ and a node $u \in V$, let the *inclusive neighborhood of* u be $N^+(u) = \{u\} \cup \{v \in V \mid \{u, v\} \in E\}$.



(a) Smallest s-t-cut dominator (black) and its dominated s-t-cut (grey).

(b) Smallest *connected* s-t-cut dominator (black) and its dominated s-t-cut (grey).

Fig. 1. Variants of cut domination.

Analogously, for a set $U \subseteq V$, let $N^+(U) = \bigcup_{u \in U} N^+(u)$ be the *inclusive neighborhood of* U.

Furthermore, for an undirected graph $G = (V, E)$ and non-adjacent nodes $s, t \in V$, an *s-t-vertex-separator* is a node set $U \subseteq V \setminus \{s, t\}$ with the property that the removal of U from G disconnects s and t. It is a well-known result, that a minimum s-t-vertex-separator can be found in polynomial time [6]. Sometimes, such a set is also called an *s-t vertex cut*. In the same context, we define an *s-t-cut dominator* to be a set $U \subseteq V \setminus \{s, t\}$, so that $N^+(U) \setminus \{s, t\}$ is an s-t-vertex-separator of G. In other words, U dominates an s-t-vertex-separator.

Given a simple undirected graph $G = (V, E)$ and two non-adjacent nodes $s, t \in V$, the Cut Domination problem consists of finding a minimum s-t-cut dominator. Furthermore, we define the Connected Cut Domination problem of finding a *connected* s-t-cut dominator of minimum cardinality. Examples for typical solutions are given in Fig. 1.

Both problems admit a natural generalization by adding a weight function $w : V \to \mathbb{R}^+$ that assigns positive weights to the nodes of G. Trying to find a (connected) set $U \subseteq V \setminus \{s, t\}$ of *minimum weight* $w(U) = \sum_{v \in U} w(v)$ dominating an s-t-separator is called Weighted (Connected) Cut Domination.

In the rest of the paper let the problem size $n = |V \setminus \{s, t\}|$, the number of nodes excluding s and t.

3 Inapproximability of Cut Domination

We show an approximation-preserving polynomial-time reduction from Red-Blue Set Cover to Cut Domination.

The Red-Blue Set Cover problem is a generalization of the Set Cover problem, where the universe U is partitioned into two subsets, a set R of red elements and a set B of blue elements. We are given a collection of sets $\mathcal{S} = \{S_1, S_2, \ldots, S_n\}$ over the universe U and have to find a subcollection $C \subseteq \mathcal{S}$ containing all blue elements while also containing a minimum number of red elements. Let $R(C) = \bigcup_{S_i \in C} S_i \cap R$ denote the set of red elements covered by the subcollection C.

Carr et al. showed in [7] that Red-Blue Set Cover is $\mathcal{O}(2^{\log^{1-\delta} n})$-inapproximable with $\delta = 1/\log\log^c n$ for every constant $c < 1/2$, unless P=NP. This result even holds for Red-Blue Set Cover with the additional constraint that every set $S_i \in \mathcal{S}$ only contains one blue and two red elements.

Theorem 1. Cut Domination *is* $\mathcal{O}(2^{\log^{1-\delta} n})$*-inapproximable for every constant* $c < 1/2$*, with* $\delta = 1/\log\log^c n$*, if* P≠NP*.*

Proof. We are given an instance $I = (\mathcal{S}, R, B)$ of Red-Blue Set Cover with the constraint, that every set contains one blue and two red elements. W.l.o.g. we can assume, that every red element and every blue element is contained in at least one set $S \in \mathcal{S}$. Furthermore, we assume an arbitrary ordering of the sets in $\mathcal{S}$. We now build an instance $I' = (G = (V, E), s, t)$ of Cut Domination with the following properties:

(1) Every feasible solution $C \subseteq \mathcal{S}$ for I corresponds to a feasible solution $U \subseteq V \setminus \{s, t\}$ of size $|R(C)|$ for I'.
(2) For every feasible solution $U \subseteq V \setminus \{s, t\}$ for I', we can find a solution $C \subseteq \mathcal{S}$ for I with $|R(C)| \leq |U|$.

Starting from an empty graph, we first create nodes s and t. Then we add a complete subgraph of 'red' nodes $V_R = \{v_r \mid r \in R\}$, each of them corresponding to one of the red elements in R. Afterwards we construct two s-t-pathways, so called b-*connectors*, for each blue element $b \in B$ and connect them to some of the 'red' nodes. This is done in such a way, that all of the b-connectors have to be cut to disconnect s and t, while cutting them can be done by selecting pairs of 'red' nodes whose corresponding elements are in a set together with b. The construction of the b-connectors will be explained in greater detail now, for an arbitrary, but fixed blue element $b \in B$.

For every set $S_i \in \mathcal{S}$ that contains b, we do the following. First, we create a pair of nodes u_l^i, u_k^i corresponding to the red elements r_l, r_k in S_i. Second, we add edges $\{u_l^i, v_l\}$ and $\{u_k^i, v_k\}$. Third, we connect both u_l^i and u_k^i to each node of the previously created pair for b. If there is no previously created pair for b, we connect u_l^i and u_k^i to s. After examining all sets, we connect both nodes of the lastly created pair to t. This gives us the first b-connector. An example of such a b-connector can be seen in

Fig. 2. By repeating this procedure and creating node pairs w_l^i, w_k^i instead of u_l^i, u_k^i, we obtain the second b-connector.

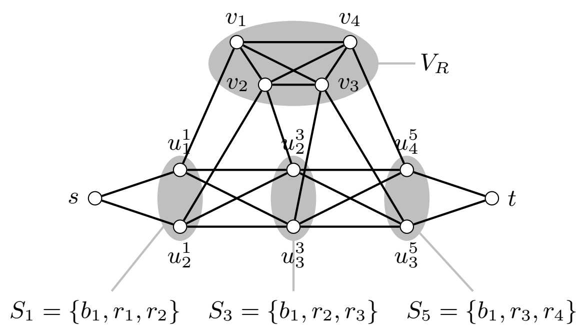


Fig. 2. The first b_1-connector.

We do this for all blue elements $b \in B$ to get the graph G. The construction can obviously be performed in polynomial time and creates a graph G with exactly $4|\mathcal{S}| + |R|$ nodes, excluding s and t. Since every red element appears in at least one set of $\mathcal{S}$ and every set contains exactly two red elements, there can be at most $2|\mathcal{S}|$ red elements. Thus, it holds that $|V(G) \setminus \{s,t\}| \leq 6|\mathcal{S}|$.

The only thing left for us to show is, how to transform feasible solutions of the Red-Blue Set Cover instance I to feasible solutions of the Cut Domination instance I' and vice versa while containing their costs.

Given a solution $C \subseteq \mathcal{S}$ of I we determine the set $R(C)$ of red elements covered by C and take the nodes $U = \{v_r \in V_R \mid r \in R(C)\}$ as an s-t-cut dominating set in G. Note that U only consists of nodes from V_R and does not contain any node of a b-connector. By definition, it holds that $|R(C)| = |U|$. Choosing at least one of the $v_r \in V_R$, all nodes of V_R are dominated since V_R induces a complete subgraph. Therefore the only s-t paths left may lead over the b-connectors. Consider an arbitrary blue element $b \in B$. Since C is a solution to I it has to cover b. Consequently, there has to be a set $S_i \in C$ containing b. Furthermore, both the red elements r_k and r_l from S_i have to be contained in $R(C)$. This means, that v_{r_k} and v_{r_l} are in U. According to the construction, there are node pairs u_k^i, u_l^i and w_k^i, w_l^i in the b-connectors and edges $\{v_{r_k},\ u_k^i\}$,$\{v_{r_l},\ u_l^i\}$,$\{v_{r_k},\ w_k^i\}$ and $\{v_{r_l},\ w_l^i\}$. Due to the fact, that v_{r_k} and v_{r_l} are chosen, u_k^i, u_l^i, w_k^i and w_l^i are dominated, therefore cutting both b-connectors. Since this holds for every $b \in B$, all b-connectors are cut, thus separating s and t.

Consider now a set $U \subseteq V(G) \setminus \{s, t\}$ that dominates an s-t-separator of G. First, we show that we can choose a set $U' \subseteq V_R$ with $|U'| \leq |U|$ that also dominates an s-t-separator of G. To do so, we consider the b-connectors. Due to the construction there are only two ways to dominate a cut of a b-connector: either by choosing a node of the connector or by choosing two nodes $v_{r_k}, v_{r_l} \in V_R$, so that $\{b, r_k, r_l\} \in \mathcal{S}$. In the second case, both b-connectors are cut and we are done. Otherwise, there has to be at least one node from U on each of the two b-connectors. Instead of taking these two nodes, we can arbitrarily choose a set $S_j \in \mathcal{S}$ with $b \in S_j$ and take the nodes $v_{r_k}, v_{r_l} \in V_R$ corresponding to the red elements $r_k, r_l \in S_j$. By doing so we still cut both b-connectors and additionally dominate all nodes from V_R, if they were not already. We can do this for all $b \in B$ while still containing the size of the solution. This first step ensures that all b-connectors are cut by nodes in V_R. Afterwards, we eliminate all nodes from $V \setminus V_R$ from the solution to obtain U'.

We now define $R(U')$ to be the set of all red elements whose corresponding nodes are in U'. We can choose C as the collection of all sets $S \in \mathcal{S}$ that contain only red elements from $R(U')$. Since $U' \subseteq V_R$ and $R(C) \subseteq R(U')$ it now holds that $|R(C)| \leq |R(U')| = |U'|$. Our transformation ensures that for every blue element $b \in B$ there are nodes $v_{r_k}, v_{r_l} \in U'$ with $\{b, r_k, r_l\} \in \mathcal{S}$. Especially $r_k, r_l \in R(U')$ and therefore $\{b, r_k, r_l\} \in C$. Since this holds for every $b \in B$, C has to cover all blue elements. □

Weighted Cut Domination has to be at least as difficult to approximate as the unweighted case. Hence, the inapproximability result also holds for the weighted variant of the problem. The s-t-cut dominating set that can be constructed from a Red-Blue Set-Cover solution consists only of nodes from V_R, therefore it is connected. Furthermore, we transform a feasible solution of the constructed graph G to a solution of the same size that consists only of nodes from V_R. This is possible for every s-t-cut dominating set of G. Especially, it is possible for every *connected* s-t-cut dominating set of G. Consequently, Theorem 1 also holds for Connected Cut Domination.

4 A $2\sqrt{n}$-Approximation in General Graphs

We give a $2\sqrt{n}$-approximation algorithm for Weighted Cut Domination. Since the unweighted problem is a special case of the weighted variant, the approximability result holds for both.
Algorithm 1 proceeds as follows: the weights appearing in the graph are, one by one, considered as maximum weight of a vertex from an optimal

Algorithm 1: Weighted Cut Domination Approximation (G, s, t, w)

foreach $w_i \in \{w(v) \mid v \in V \setminus \{s,t\}\}$ **do**
    $U_1 \leftarrow \emptyset$;
    while $\exists\, v \in V \setminus (U_1 \cup \{s,t\})\colon \frac{|N^+(v) \setminus (\{s,t\} \cup N^+(U_1))|}{w(v)} \geq \frac{\sqrt{n}}{w_i}$ **do**
        choose such a node v arbitrarily;
        $U_1 \leftarrow U_1 \cup \{v\}$;
    $G' \leftarrow G \setminus (N^+(U_1) \setminus \{s,t\})$;
    if *s and t are in the same connected component H of G'* **then**
        foreach $v \in V \setminus \{s,t\}$ **do**
            $w'(v) \leftarrow \min\{w(u) \mid u \in N^+(v) \setminus \{s,t\}\}$;
        $C \leftarrow$ min-vertex-cut(H, s, t, w');
        $C_i \leftarrow \emptyset$;
        foreach $v \in C$ **do**
            $v' \leftarrow \mathrm{argmin}\{w(u) \mid u \in N^+(v) \setminus \{s,t\}\}$;
            $C_i \leftarrow C_i \cup \{v'\}$;
        $C_i \leftarrow C_i \cup U_1$;
    else
        $C_i \leftarrow U_1$;
return $\mathrm{argmin}_{X \in \{C_i \mid 1 \leq i \leq n\}}\{w(X)\}$;

solution. For every weight w_i, we compute a candidate solution C_i as follows.

First, we greedily choose nodes v, which dominate at least $w(v)\sqrt{n}/w_i$ currently undominated nodes, including themselves, and consider their inclusive neighborhoods as dominated. This is repeated until either an s-t-cut is dominated or no appropriate node is left. The result of this greedy selection is a set U_1. Second, we consider the induced subgraph G' of currently undominated nodes. If s and t are not connected in G', the candidate solution for w_i is the set $C_i = U_1$ and the algorithm continues with w_{i+1}. Otherwise we consider the connected component H of G' which contains s and t. The nodes of $V \setminus \{s,t\}$ are then assigned new weights w', so that $w'(v) := \min\{w(u) \mid u \in N^+(v) \setminus \{s,t\}\}$. The new weights represent the cost to dominate these nodes. Now we compute a minimum s-t-cut C in (H, w'). Third, we choose a minimum-weight neighbor in G for each node $v \in C$ arbitrarily. This gives us a set C_i of weight at most $w'(C)$. The candidate solution for w_i is the set $C_i = U_1 \cup C_i$.

After calculating the sets C_i for $1 \leq i \leq n$, the set with minimum weight $w(C_i)$ is returned.

Theorem 2. *Algorithm 1 is a* $\sqrt{n} \cdot (1 + {w_{max}}/{OPT})$ *approximation algorithm, where* OPT *denotes the weight of an optimal solution and* w_{max} *denotes the maximum weight of a node from this optimal solution. Especially, this is at most* $2\sqrt{n}$*.*

Proof. Consider an optimal solution $V_{opt} \subseteq V \setminus \{s,t\}$. Now let

$$w_{max} = \max \{w(v) \mid v \in V_{opt}\}.$$

It follows that: (i) $w(v) \leq w_{max}$ for all $v \in V_{opt}$ and
(ii) $w_{max} \leq w(V_{opt})$.

Since Algorithm 1 does one round for each node's weight, there must be a round $1 \leq j \leq n$ with $w_j = w_{max}$. Consider the respective run of the algorithm's main loop.

To bound the weight of U_1, we take a look at the nodes $v^1, v^2, \ldots v^{|U_1|}$ of U_1 in the order in which they are included in U_1 by the algorithm. Let the set $U_1^k = \left\{v^1, \ldots, v^k\right\}$ be the set U_1 after the k-th round of the greedy selection and $U_1^0 = \emptyset$. Let the set of newly dominated nodes for v^k be $N^k := N^+(v^k) \setminus (\{s,t\} \cup N^+(U_1^{k-1}))$. These sets are pairwise disjoint.

Since every node $v^k \in U_1$ dominated at least $|N^k| \geq w(v^k)\sqrt{n}/w_{max}$ new nodes, it holds that

$$\begin{aligned} w(U_1) = \sum_{k=1}^{|U_1|} w(v^k) &\leq \sum_{k=1}^{|U_1|} |N^k| \; w_{max}/\sqrt{n} \\ &\leq \frac{w_{max}}{\sqrt{n}} n = \sqrt{n} \; w_{max}. \end{aligned}$$

After the greedy selection all $v \in V \setminus (U_1 \cup \{s,t\})$ fulfill

$$|N^+(v) \setminus (\{s,t\} \cup N^+(U_1))| < w(v)\sqrt{n}/w_{max}. \tag{1}$$

Now take a closer look at $G' = G \setminus (N^+(U_1) \setminus \{s,t\})$, the induced subgraph of currently undominated nodes. If s and t are cut in G', U_1 is an s-t-cut dominating set of weight at most $\sqrt{n}\; w(V_{opt})$ as desired. Let us now assume that this is not the case, i.e. there is a connected component H of G' which contains both s and t. We know that V_{opt} dominates an s-t-cut in G. Therefore, $V_{opt} \setminus U_1$ has to dominate an s-t-cut in H. It now holds that $(N^+(V_{opt} \setminus U_1) \setminus (N^+(U_1) \cup \{s,t\})) \cap V(H)$ is an s-t-cut in H. Therefore, the weight $w'(C)$ of the minimum w'-weight s-t-cut in H is at most $w'((N^+(V_{opt} \setminus U_1) \setminus (N^+(U_1) \cup \{s,t\})) \cap V(H))$. Furthermore, since for every node $v \in C$ there is a node $u \in N^+(v) \setminus \{s,t\}$ with $w(u) = w'(v)$, it holds that

$$w(C_j) \leq w'(C). \tag{2}$$

This leads to

$$\begin{aligned}
w(C_j) &\le w'(C) \\
&\le w'((N^+(V_{opt} \setminus U_1) \setminus (N^+(U_1) \cup \{s,t\})) \cap V(H)) \\
&\le \sum_{v \in V_{opt} \setminus U_1} \sum_{u \in N^+(v) \setminus (N^+(U_1) \cup \{s,t\})} w'(u) \\
&\le \sum_{v \in V_{opt} \setminus U_1} |N^+(v) \setminus (N^+(U_1) \cup \{s,t\})| w(v) \\
&\overset{(1)}{<} \sum_{v \in V_{opt} \setminus U_1} \sqrt{n}\; w(v) \frac{w(v)}{w_{max}} \\
&\overset{\text{(i)}}{\le} \sum_{v \in V_{opt} \setminus U_1} \sqrt{n}\; w(v) \le \sqrt{n}\; w(V_{opt}).
\end{aligned}$$

Hence, by uniting C_j and U_1, we obtain a set of nodes with weight at most $\sqrt{n}\; w(V_{opt}) + \sqrt{n}\; w_{max}$ in run j of the algorithm. Consequently, the algorithm returns a set of weight at most $\sqrt{n}(1 + {w_{max}}/{w(V_{opt})})\; w(V_{opt})$. $\square$

In the unweighted version all weights are 1. Therefore, the following simplifications of the algorithm can be applied. First, one run of the algorithm's main loop will be sufficient, since we know $w_{max} = 1$. Second, the greedy procedure only chooses nodes dominating at least $\sqrt{n}$ new nodes. Third, the minimum-weight-function is not necessary. It suffices to calculate a minimum cardinality s-t-vertex-cut of G'.
So, the algorithm will degenerate to greedily choosing nodes which dominate at least $\sqrt{n}$ new nodes and computing a minimum s-t-vertex-cut in the resulting graph of undominated nodes. The approximation ratio of this algorithm is $\sqrt{n}\ (1 + \frac{1}{OPT})$, where OPT denotes the *size* of the optimal solution. Since OPT is at least one the simplified algorithm would give a $2\sqrt{n}$-approximation in the worst case. We can improve this ratio by adding a preprocessing step that enumerates all subsets $U \subseteq V$ up to a constant size $k \in \mathbb{N}$ and checks whether they dominate an s-t-cut. The first subset to do is an optimal solution. If none of the subsets dominates an s-t-cut, the optimal solution has to be of size at least $k+1$. Therefore, the approximation ratio of the simplified algorithm with such a preprocessing step is at most $\sqrt{n}(1 + \frac{1}{k+1})$. The preprocessing needs $\mathcal{O}(kn^k(|V| + |E|))$ time, since for all $\sum_{i=1}^{k} \binom{n}{i}$ subsets it has to construct the graph G' of undominated nodes and test whether s and t are connected in G'. We state this observation in the following corollary.

Algorithm 2: Connected Cut Domination Approximation (G, s, t)

```
foreach c ∈ V \ {s,t} do
    foreach v ∈ V \ {s,t} do
        w(v) ← dist_{G\{s,t}}(c, v)
    foreach w_i ∈ {w(v) | v ∈ V \ {s,t}} do
        U_1 ← ∅;
        while ∃ v ∈ V \ (U_1 ∪ {s,t}): |N^+(v) \ ({s,t} ∪ N^+(U_1))| / w(v) ≥ √n / (w_i √(2w_i)) do
            choose such a node v arbitrarily;
            U_1 ← U_1 ∪ {v};
        G' ← G \ (N^+(U_1) \ {s,t});
        if s and t are in the same connected component H of G' then
            foreach v ∈ V \ {s,t} do
                w'(v) ← min {w(u) | u ∈ N^+(v) \ {s,t}};
            C ← min-vertex-cut(H, s, t, w');
            C_i ← ∅;
            foreach v ∈ C do
                v' ← argmin{w(u) | u ∈ N^+(v) \ {s,t}};
                C_i ← C_i ∪ {v'};
            C_i ← C_i ∪ U_1;
        else
            C_i ← U_1;
        C_{c,i} ← ∅;
        foreach v ∈ C_i do
            P(v,c) ← shortest v-c-path in G \ {s,t};
            C_{c,i} ← C_{c,i} ∪ V(P(v,c));
return argmin_{X ∈ {C_{c,i} | c ∈ V \ {s,t}, 1 ≤ i ≤ n}} {|X|};
```

Corollary 1. Cut Domination *can be approximated with ratio* $\sqrt{n} \cdot (1 + \frac{1}{k+1})$ *for every constant* $k \in \mathbb{N}$.

Algorithm 2 is variation of Algorithm 1 and approximates Connected Cut Domination. Now let $dist_{G\setminus\{s,t\}}(v, u)$ denote the minimum hop distance between nodes u and v in $G \setminus \{s, t\}$.

Theorem 3. *Algorithm 2 is a* $\sqrt{n}\sqrt{OPT}$ *approximation algorithm for* Connected Cut Domination, *where* OPT *denotes the size of an optimal solution. Especially, this is at most* $n^{2/3}$.

Proof. Now consider an optimal solution $V_{opt} \subseteq V \setminus \{s, t\}$ and let

$$d(v) = \max_{u \in V_{opt}} \left\{ dist_{G\setminus\{s,t\}}(v, u) \right\}$$

be the maximum distance of any node from V_{opt} to v. At some time the algorithm is bound to choose a *center node* $v_0 \in V_{opt}$ with the property that $d(v_0) \leq {}^{|V_{opt}|}/_{2}$. Then, weight $w(v)$ represents the number of nodes we have to add in the worst case to connect v to v_0. Considering run j with $w_j = w_{max} = d(v_0)$, it holds that

$$\text{(i) } w_v \leq w_{max} \text{ for all } v \in V_{opt} \qquad \text{and} \qquad \text{(ii) } w_{max} \leq {}^{|V_{opt}|}/_{2}.$$

The greedy phase now starts with $U_1 = \{v_0\}$ and chooses nodes v which dominate at least ${}^{w(v)\sqrt{n}}/_{w_{max}\sqrt{2w_{max}}}$ undominated nodes. Consequently, we obtain a set U_1 of weight at most

$$\begin{aligned} w(U_1) \;&\leq\; \sum_{k=1}^{|U_1|} |N_k|\; w_{max}\sqrt{2w_{max}}/\sqrt{n} \\ &\leq\; \frac{w_{max}\sqrt{2w_{max}}}{\sqrt{n}} n \\ &\overset{(ii)}{\leq}\; \frac{\sqrt{n}\sqrt{|V_{opt}|}\;|V_{opt}|}{2}. \end{aligned}$$

For the set C_j, it holds that

$$\begin{aligned} w(C_j) \;&\leq\; w'(N^+(V_{opt} \setminus U_1) \setminus (N^+(U_1) \cup \{s,t\})) \\ &\leq\; \sum_{v \in V_{opt} \setminus U_1} \; \sum_{u \in N^+(v) \setminus (N^+(U_1) \cup \{s,t\})} w'(u) \\ &\leq\; \sum_{v \in V_{opt} \setminus U_1} w(v) \frac{w(v)\sqrt{n}}{w_{max}\sqrt{2w_{max}}} \\ &\overset{(i)}{\leq}\; \frac{\sqrt{n}\sqrt{w_{max}}}{\sqrt{2}}(|V_{opt}| - 1) \\ &\overset{(ii)}{\leq}\; \frac{\sqrt{n}\sqrt{|V_{opt}|}}{2}|V_{opt}| - 1. \end{aligned}$$

To connect $U_1 \setminus \{v_0\}$ and C_j to v_0, we need at most $w(U_1) + w(C_j)$ nodes, including $U_1 \setminus \{v_0\}$ and C_j. Thus, it holds that

$$|X| \leq |C_{v_0,j}| \leq 1 + w(U_1) + w(C_j) \leq \sqrt{n}\sqrt{|V_{opt}|}|V_{opt}|.$$

$\square$

5 The Case of Bounded Vertex Degrees

We show that Weighted Cut Domination is $(\Delta + 1)$-approximable, if all but at most a logarithmic number of nodes are of degree Δ or less.

Theorem 4. *Let $W \subseteq V \setminus \{s,\ t\}$ with $|W| = \mathcal{O}(\log n)$. Then* Weighted Cut Domination *is $(\Delta_W + 1)$-approximable where Δ_W is the maximum degree of nodes from $V \setminus (\{s,t\} \cup W)$ in $G \setminus \{s,t\}$.*

Proof. Consider an algorithm that iterates all $2^{|W|}$ subsets $U \subseteq W$. For each $U \subseteq W$ the algorithm proceeds like one round of Algorithm 1, but with U taking the place of U_1, and calculates a candidate solution C_U. It then outputs the candidate solution of minimum weight.

Now we need to show, that this algorithm outputs a $(\Delta_W + 1)$-approximate solution. Let V_{opt} be a minimum weight s-t-cut dominating set of G. Furthermore, let $W_{opt} = V_{opt} \cap W$. Since $W_{opt} \subseteq W$, there is a round where $U = W_{opt}$. We know that every node from $V_{opt} \setminus W_{opt}$ is from $V \setminus (\{s,t\} \cup W)$ and therefore has a maximum degree of Δ_W in $G \setminus \{s,t\}$. It follows, that for every node $v \in V_{opt} \setminus W_{opt}$

$$\sum_{u \in N^+(v) \setminus \{s,t\}} w'(u) \leq (\Delta_W + 1) w(v). \tag{3}$$

Let us now consider the induced subgraph $G_{W_{opt}}$ of nodes which are not dominated by W_{opt}. We know that V_{opt} dominates an s-t-vertex-cut of G. Therefore, $V_{opt} \setminus W_{opt}$ has to dominate an s-t-vertex-cut of $G_{W_{opt}}$. It now holds, that $N^+(V_{opt} \setminus W_{opt}) \setminus (N^+(W_{opt}) \cup \{s,t\})$ is an s-t-cut of $G_{W_{opt}}$. Since C is a minimum s-t-cut of $G_{W_{opt}}$ according to w', it is also true that

$$w'(C) \leq w'(N^+(V_{opt} \setminus W_{opt}) \setminus (N^+(W_{opt}) \cup \{s,t\})). \tag{4}$$

Therefore, for the s-t-cut dominating set $C_{W_{opt}}$ constructed from C, it holds that

$$\begin{aligned} w(C_{W_{opt}}) &\overset{(2)}{\leq} w'(C) \\ &\overset{(4)}{\leq} w'(N^+(V_{opt} \setminus W_{opt}) \setminus (N^+(W_{opt}) \cup \{s,t\})) \\ &\leq \sum_{v \in V_{opt} \setminus W_{opt}} \sum_{u \in N^+(v) \setminus \{s,t\}} w'(u) \\ &\overset{(3)}{\leq} (\Delta_W + 1) w(V_{opt} \setminus W_{opt}). \end{aligned}$$

Hence, it holds that $w(W_{opt} \cup C_{W_{opt}}) \leq w(W_{opt}) + (\Delta_W + 1) w(V_{opt} \setminus W_{opt}) = (\Delta_W + 1) w(V_{opt}) - \Delta_W w(W_{opt})$. We obtain a $(\Delta_W + 1)$-approximation in the worst case and an upper bound for the size of the algorithm's solution. Since the algorithm computes $2^{|W|}$ induced subgraphs and minimum weight s-t-vertex-cuts, its running time is $\mathcal{O}(2^{|W|}(|V| + |E| + \sqrt{|V|}|E|))$, which is polynomial in n if and only if $|W| = \mathcal{O}(\log n)$. $\square$

Theorem 4 is especially relevant in practical applications, since communication overlay networks usually are of constant or logarithmic degree for scalability reasons.

As the minimum s-t-vertex-cut provides an upper bound for the minimum s-t-cut-dominating set, Cut Domination can be solved in polynomial time for all graphs with a minimum s-t-vertex-cut of constant size. In particular, this includes all graphs, where s and t have degrees bounded by a constant.

6 Conclusion

Although a minimum s-t-vertex-separator can be found in polynomial time, we showed that it is much more complex to efficiently *dominate* any s-t-vertex separator. In particular, we proved that the Cut Domination problem is not approximable to within a factor of $2^{\log^{1-\delta} n}$, with $\delta = 1/\log\log^c n$ and arbitrary $c < 1/2$ (if P $\neq$ NP) by reducing from Red-Blue Set Cover. Thus, its inapproximability is higher than that of Dominating Set, the problem of finding a smallest set of nodes dominating *all* nodes of a graph [8] (again, if P $\neq$ NP).

On the positive side, we were able to show that Weighted Cut Domination is $2\sqrt{n}$-approximable in general graphs and $(\Delta + 1)$-approximable in graphs with maximum degree Δ. In practice, the case of bounded node degrees is of special interest, since common overlay networks feature at most logarithmic degrees. The obtained (in-)approximability results are similar to the best known results for Red-Blue Set Cover [7,9], which is believed to be a canonical representative from the class of optimization problems with superpolylogarithmic but potentially subpolynomial approximability. Closing the gap between approximability and inapproximability of Weighted Cut Domination by showing stronger inapproximability or approximability results, as well as investigating inapproximability for graphs of bounded node degree, remains for future research.

Connected Cut Domination is also of special interest, as in computer networks attacks sometimes spread from one node to another. We showed that the inapproximability result carries on to Connected Cut Domination. An approximation algorithm similar to Algorithm 1 achieves an approximation ratio of $\sqrt{n}\sqrt{OPT}$. It remains open, whether the approximation ratio can be lowered to the same ratio as for Cut Domination or which ratio is achievable for Weighted Connected Cut Domination.

Another matter of interest is the relation of Cut Domination to the original problem described in [4,5]. It is a goal for future work to show similar approximability bounds for that problem.

References

1. Bollapragada, V., Khalid, M., Wainner, S.: IPSec VPN Design. Cisco Press (2005)
2. Vasserman, E., Jansen, R., Tyra, J., Hopper, N., Kim, Y.: Membership-concealing overlay networks. In: 16th ACM CCS, ACM (2009) 390–399
3. Clarke, I., Sandberg, O., Toseland, M., Verendel, V.: Private communication through a network of trusted connections: The dark freenet. Network (2010)
4. Rossberg, M., Girlich, F., Schaefer, G.: Analyzing and Improving the Resistance of Overlays against Bandwidth Exhaustion Attacks. In: International Workshop on Reliable Networks Design and Modeling (RNDM). (2012)
5. Girlich, F., Rossberg, M., Schaefer, G.: On the Resistance of Overlay Networks against Bandwidth Exhaustion Attacks. Accepted for Telecommunication Systems Journal (Special Issue) (2014)
6. Ahuja, R.K., Magnanti, T.L., Orlin, J.B.: Network Flows: Theory, Algorithms, and Applications. Prentice-Hall, Inc., Upper Saddle River, NJ, USA (1993)
7. Carr, R.D., Doddi, S., Konjevod, G., Marathe, M.: On the red-blue set cover problem. In: ACM-SIAM Symposium on Discrete Algorithms. (2000) 345–353
8. Alon, N., Moshkovitz, D., Safra, S.: Algorithmic construction of sets for k-restrictions. ACM Trans. Algorithms **2**(2) (April 2006) 153–177
9. Peleg, D.: Approximation algorithms for the label-covermax and red-blue set cover problems. Journal of Discrete Algorithms **5**(1) (2007) 55 – 64