

# Improved Budgeted Connected Domination and Budgeted Edge-Vertex Domination

Ioannis Lamprou

Ioannis Sigalas

Vassilis Zissimopoulos

Department of Informatics and Telecommunications,  
National and Kapodistrian University of Athens, Greece

IWOCA 2020, June 9<sup>th</sup>

DEPARTMENT OF INFORMATICS & TELECOMMUNICATIONS



HELLENIC REPUBLIC  
National and Kapodistrian  
University of Athens

EST. 1837



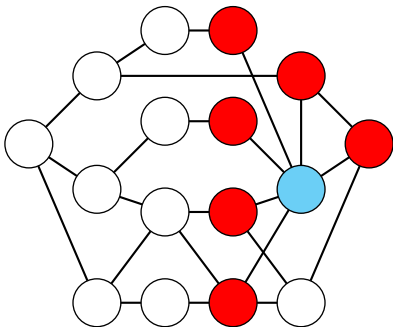
# Outline

- 1 Introduction
- 2 Budgeted Connected Domination
- 3 Budgeted Edge-Vertex Domination
- 4 Conclusions

# Motivation

## Dominating Sets' Applications

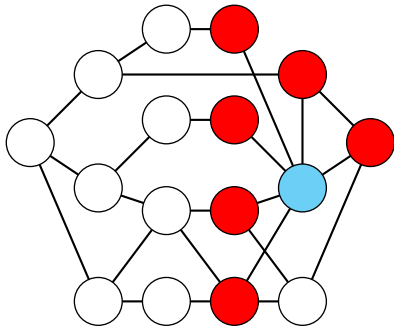
- Ad-hoc wireless (sensor) networks: coverage
- Social network: maximizing influence, friendships
- Electrical transmission lines: fault location



# Motivation

## Dominating Sets' Applications

- Ad-hoc wireless (sensor) networks: coverage
- Social network: maximizing influence, friendships
- Electrical transmission lines: fault location



## In this paper

- Budgeted Connected Domination
- Budgeted Edge-Vertex Domination

# Outline

- 1 Introduction
- 2 Budgeted Connected Domination**
- 3 Budgeted Edge-Vertex Domination
- 4 Conclusions

# Budgeted Connected Dominating Set

## Definition (Closed Neighborhood)

The **closed neighborhood**  $N[S]$  of a subset  $S \subseteq V$  contains all vertices in  $S$  and any vertex adjacent to at least one vertex in  $S$ .

## Definition (Dominating Set)

A (connected) dominating set is a (connected) subset of vertices  $S \subseteq V$  for which it holds  $N[S] = V$ .

# Budgeted Connected Dominating Set

## Definition (Closed Neighborhood)

The **closed neighborhood**  $N[S]$  of a subset  $S \subseteq V$  contains all vertices in  $S$  and any vertex adjacent to at least one vertex in  $S$ .

## Definition (Dominating Set)

A (connected) dominating set is a (connected) subset of vertices  $S \subseteq V$  for which it holds  $N[S] = V$ .

## Definition (BCDS)

Given a graph  $G = (V, E)$  and an integer  $k$ , select a subset  $S \subseteq V$  of size at most  $k$ , such that the subgraph induced by  $S$  is **connected** and  $|N[S]|$  is maximized.

# Background

## Budgeted Connected Dominating Set (BCDS)

- Origins: wireless sensor networks [Liu, Liang '05, Wang '11]
- $(1 - e^{-1})/13$  approximation [Khuller, Purohit, Sarpatwar, SODA '14]



# Background

## Budgeted Connected Dominating Set (BCDS)

- Origins: wireless sensor networks [Liu, Liang '05, Wang '11]
- $(1 - e^{-1})/13$  approximation [Khuller, Purohit, Sarpatwar, SODA '14]

## Our Results

- Improved  $(1 - e^{-7/8})/11$  approximation
  - Generalization of algorithm
  - Fine-tuning of analysis
- $(1 - e^{-1})$  inapproximability

# $(1 - e^{-1})/13$ approximation [Khuller, Purohit, Sarpatwar, SODA '14]

---

**Algorithm:** Greedy Profit Labeling Algorithm for BCDS

---

**Input** : A graph  $G = (V(G), E(G))$ ,  $k \in \mathbb{N}$

**Output:** A tree  $\tilde{T}$  on at most  $k$  vertices

- 1  $(D, p) \leftarrow GDS(G)$
  - 2  $T \leftarrow QST(G, (1 - e^{-1}) OPT, p)$
  - 3  $\tilde{T} \leftarrow Best_k(T, p)$
  - 4 **return**  $\tilde{T}$
- 

- 1 Find *greedy dominating set*  $D$  and profit function  $p$

# $(1 - e^{-1})/13$ approximation [Khuller, Purohit, Sarpatwar, SODA '14]

---

**Algorithm:** Greedy Profit Labeling Algorithm for BCDS

---

**Input** : A graph  $G = (V(G), E(G))$ ,  $k \in \mathbb{N}$

**Output:** A tree  $\tilde{T}$  on at most  $k$  vertices

- 1  $(D, p) \leftarrow GDS(G)$
  - 2  $T \leftarrow QST(G, (1 - e^{-1}) OPT, p)$
  - 3  $\tilde{T} \leftarrow Best_k(T, p)$
  - 4 **return**  $\tilde{T}$
- 

- 1 Find *greedy dominating set*  $D$  and profit function  $p$
- 2 Input  $p$  to a 2-approximation algorithm for *Quota Steiner Tree* (QST): Returns a “good”, but large-size tree

# $(1 - e^{-1})/13$ approximation [Khuller, Purohit, Sarpatwar, SODA '14]

---

**Algorithm:** Greedy Profit Labeling Algorithm for BCDS

---

**Input** : A graph  $G = (V(G), E(G))$ ,  $k \in \mathbb{N}$

**Output:** A tree  $\tilde{T}$  on at most  $k$  vertices

- 1  $(D, p) \leftarrow GDS(G)$
  - 2  $T \leftarrow QST(G, (1 - e^{-1}) OPT, p)$
  - 3  $\tilde{T} \leftarrow Best_k(T, p)$
  - 4 **return**  $\tilde{T}$
- 

- 1 Find *greedy dominating set*  $D$  and profit function  $p$
- 2 Input  $p$  to a 2-approximation algorithm for *Quota Steiner Tree* (QST): Returns a “good”, but large-size tree
- 3 Dynamic programming for best subtree of size most  $k$

# 1: Greedy Dominating Set [Khuller, Purohit, Sarpatwar, SODA '14]

---

## Algorithm: Greedy Dominating Set (GDS)

---

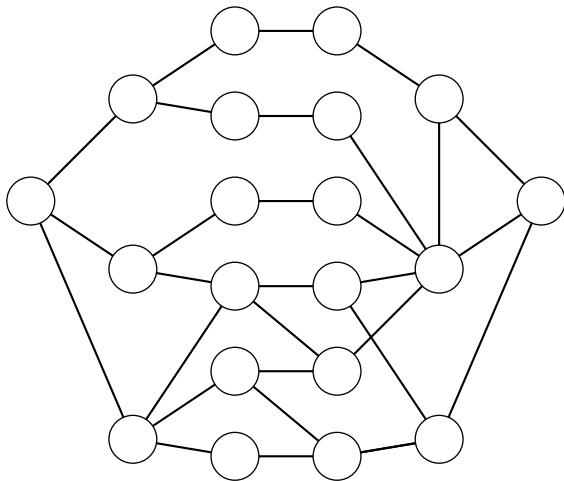
```

1  $D \leftarrow \emptyset$ 
2  $U \leftarrow V(G)$ 
3 foreach  $v \in V(G)$  do
4   |  $p(v) \leftarrow 0$ 
5 end
6 while  $U \neq \emptyset$  do
7   |  $w \leftarrow \arg \max_{v \in V(G) \setminus D} |N_U(v)|$  /*  $N_U(v) = N[v] \cap U$  */
8   |  $p(w) \leftarrow |N_U(w)|$ 
9   |  $U \leftarrow U \setminus N_U(w)$ 
10  |  $D \leftarrow D \cup \{w\}$ 
11 end
12 return  $(D, p)$ 

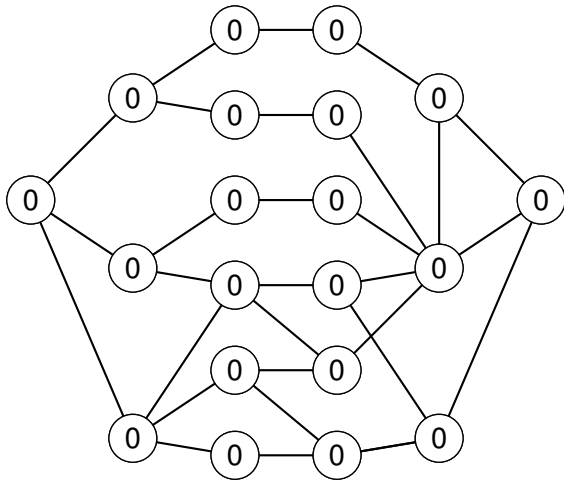
```

---

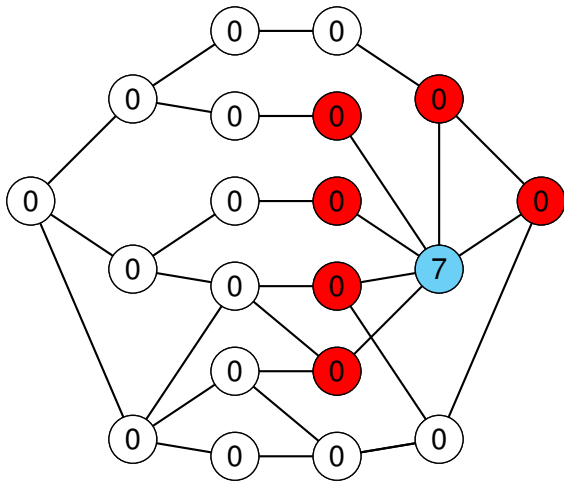
# 1: Greedy Dominating Set Example



# 1: Greedy Dominating Set Example

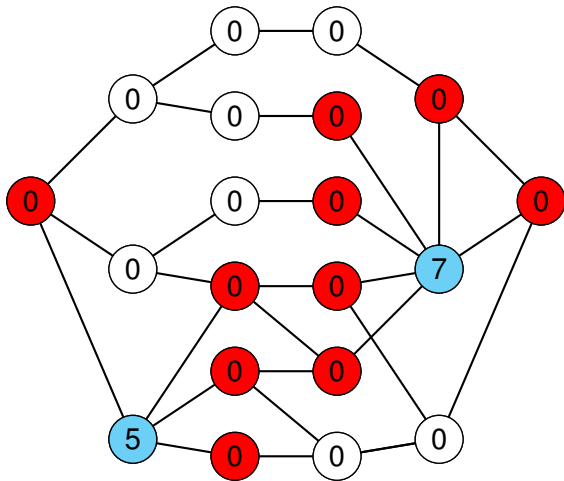


# 1: Greedy Dominating Set Example

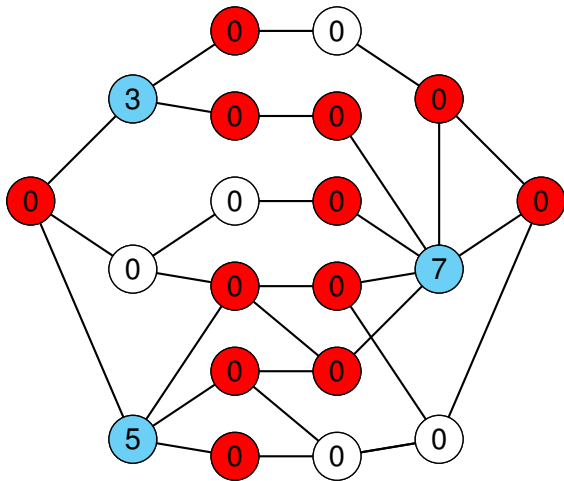




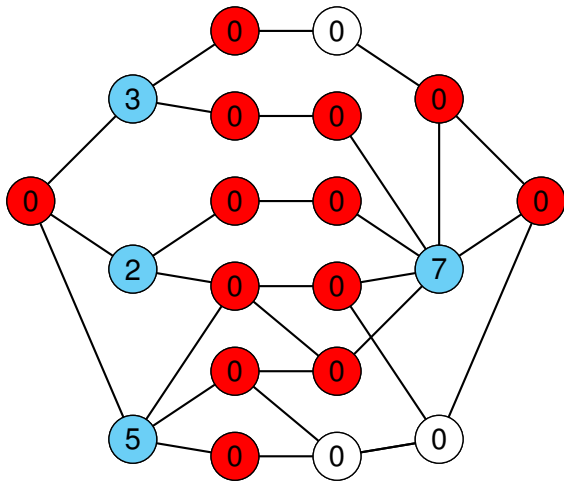
# 1: Greedy Dominating Set Example



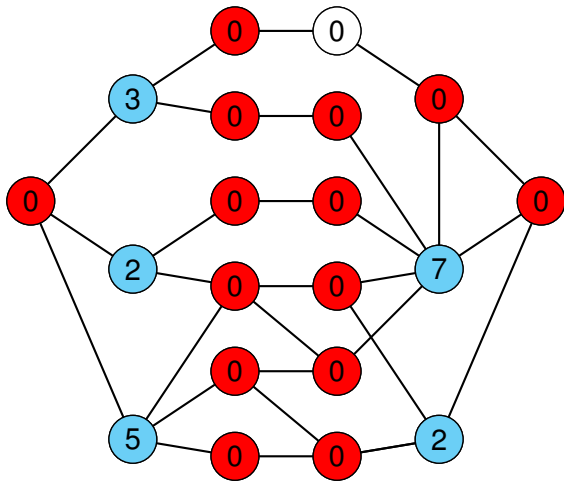
# 1: Greedy Dominating Set Example



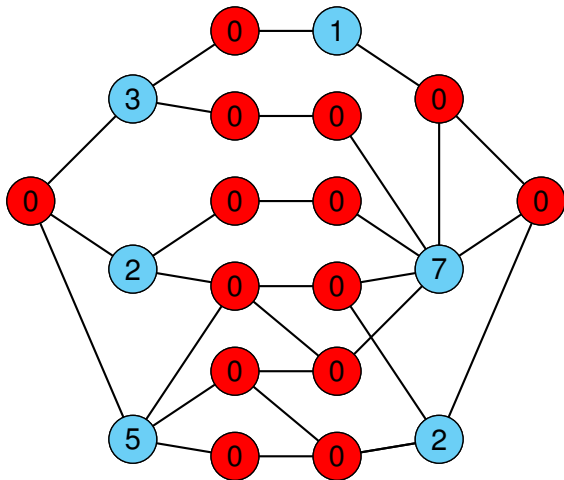
# 1: Greedy Dominating Set Example



# 1: Greedy Dominating Set Example



# 1: Greedy Dominating Set Example



## 2: Quota Steiner Tree [Khuller, Purohit, Sarpatwar, SODA '14]

### Definition (Quota Steiner Tree)

Given a graph  $G$ , a vertex profit function  $p$  and a quota  $q$ , find a minimum size subtree  $T$  subject to  $\sum_{v \in V(T)} p(v) \geq q$ .

### Lemma (Analysis)

There exists a set  $D' \subseteq D$  of size  $k$  with  $p(D') \geq (1 - e^{-1}) \text{OPT}$ , which can be connected using at most  $2k$  Steiner vertices.

## 2: Quota Steiner Tree [Khuller, Purohit, Sarpatwar, SODA '14]

### Definition (Quota Steiner Tree)

Given a graph  $G$ , a vertex profit function  $p$  and a quota  $q$ , find a minimum size subtree  $T$  subject to  $\sum_{v \in V(T)} p(v) \geq q$ .

### Lemma (Analysis)

There exists a set  $D' \subseteq D$  of size  $k$  with  $p(D') \geq (1 - e^{-1}) \text{OPT}$ , which can be connected using at most  $2k$  Steiner vertices.

- Call 2-approximation algorithm for QST [Garg '05, Johnson '00] with greedy profit function  $p$  and quota  $q = (1 - e^{-1}) \text{OPT}$ .
- By Lemma, the returned solution is a tree of size at most  $6k$  dominating at least  $(1 - e^{-1}) \text{OPT}$  vertices.

## 2: Quota Steiner Tree [Khuller, Purohit, Sarpatwar, SODA '14]

### Definition (Quota Steiner Tree)

Given a graph  $G$ , a vertex profit function  $p$  and a quota  $q$ , find a minimum size subtree  $T$  subject to  $\sum_{v \in V(T)} p(v) \geq q$ .

### Lemma (Analysis)

There exists a set  $D' \subseteq D$  of size  $k$  with  $p(D') \geq (1 - e^{-1}) \text{OPT}$ , which can be connected using at most  $2k$  Steiner vertices.

- Call **2-approximation** algorithm for QST [Garg '05, Johnson '00] with greedy profit function  $p$  and quota  $q = (1 - e^{-1}) \text{OPT}$ .
- By Lemma, the returned solution is a tree of size at most  **$6k$**  dominating at least  $(1 - e^{-1}) \text{OPT}$  vertices.



## 3: Dynamic Programming [Khuller, Purohit, Sarpatwar, SODA '14]

- Thus far,  $(6k, 1 - e^{-1})$  bicriteria approximation
- Last step: dynamic programming for best subtree of size  $k$
- Approximation analysis: tree decomposition scheme

### Lemma (Folklore)

Given any tree on  $n$  vertices, we can decompose it into two trees (by replicating a single vertex) such that the smaller tree has at most  $\lceil \frac{n}{2} \rceil$  vertices and the larger at most  $\lceil \frac{2n}{3} \rceil$  vertices.

- By recursive application of Lemma, a tree of size  $6k$  can be decomposed into **13** subtrees of size at most  $k$

# Our Approach

## Improvements

- A Generalization of quota analysis and algorithm modification
- B Use better tree decomposition technique

# The Modified Greedy Profit Labeling Algorithm

---

**Algorithm:** Modified Greedy Profit Labeling Algorithm for **BCDS(c)**

---

**Input** : A graph  $G = (V(G), E(G))$ ,  $k \in \mathbb{N}$

**Output:** A tree  $\tilde{T}$  on at most  $k$  vertices

- 1  $(D, p) \leftarrow GDS(G)$
  - 2  $T \leftarrow QST(G, (1 - e^{-c}) OPT, p)$
  - 3  $\tilde{T} \leftarrow Best_k(T, p)$
  - 4 **return**  $\tilde{T}$
- 

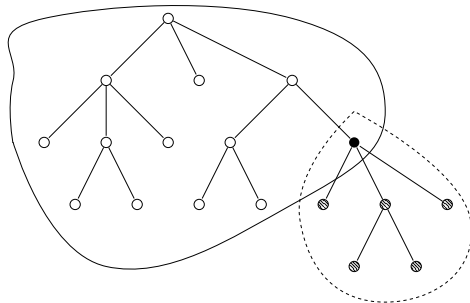
## Lemma (Improvement A)

There exists a set  $D' \subseteq D$ , where  $|D'| \leq \lceil ck \rceil$ , for a constant  $0 < c \leq 1$ , such that  $p(D') \geq (1 - e^{-c}) OPT$ . Further,  $D'$  can be connected using at most another  $k + \lceil ck \rceil$  Steiner vertices.

## Improvement B: Tree Decomposition I

### Definition

Given a tree  $T$ , an *eligible subtree*  $T'$  is a subtree of  $T$  rooted at some vertex  $i \in V_T$  such that, the forest obtained by deleting the edges with both endpoints in  $T'$  and then all the remaining vertices of degree 0 consists of a single tree.



## Improvement B: Tree Decomposition II

### Lemma ([Bermond et al. '16])

For each tree  $T$ , and for each  $k \in [1, |V_T|] \cap \mathbb{N}$ , there exists an eligible subtree  $T'$  of  $T$  such that  $k/2 \leq |V_{T'}| \leq k$ .

### Lemma

Given any tree  $T$  on  $ak$  vertices, where  $a \in \mathbb{N}$  is a constant, and  $k \geq 4a - 2$ , we can decompose it into  $2a$  subtrees each on at most  $k$  vertices.

## Overall: Improv. A, B, and 2-approx QST

### Lemma (Improvement A)

There exists a set  $D' \subseteq D$ , where  $|D'| \leq \lceil ck \rceil$ , for a constant  $0 < c \leq 1$ , such that  $p(D') \geq (1 - e^{-c}) \text{OPT}$ . Further,  $D'$  can be connected using at most another  $k + \lceil ck \rceil$  Steiner vertices.

### Lemma (Improvement B)

A tree on  $ak$  vertices, for a constant  $a \in \mathbb{N}$ , and  $k \geq 4a - 2$ , can be decomposed into  $2a$  subtrees each on at most  $k$  vertices.

## Overall: Improv. A, B, and 2-approx QST

### Lemma (Improvement A)

There exists a set  $D' \subseteq D$ , where  $|D'| \leq \lceil ck \rceil$ , for a constant  $0 < c \leq 1$ , such that  $p(D') \geq (1 - e^{-c}) \text{OPT}$ . Further,  $D'$  can be connected using at most another  $k + \lceil ck \rceil$  Steiner vertices.

### Lemma (Improvement B)

A tree on  $ak$  vertices, for a constant  $a \in \mathbb{N}$ , and  $k \geq 4a - 2$ , can be decomposed into  $2a$  subtrees each on at most  $k$  vertices.

### Theorem

There is a  $(1 - e^{-c}) / (\lceil 8c \rceil + 4)$  approximation for BCDS.

## Overall: Improv. A, B, and 2-approx QST

### Lemma (Improvement A)

There exists a set  $D' \subseteq D$ , where  $|D'| \leq \lceil ck \rceil$ , for a constant  $0 < c \leq 1$ , such that  $p(D') \geq (1 - e^{-c}) \text{OPT}$ . Further,  $D'$  can be connected using at most another  $k + \lceil ck \rceil$  Steiner vertices.

### Lemma (Improvement B)

A tree on  $ak$  vertices, for a constant  $a \in \mathbb{N}$ , and  $k \geq 4a - 2$ , can be decomposed into  $2a$  subtrees each on at most  $k$  vertices.

### Theorem

There is a  $(1 - e^{-c})/(\lceil 8c \rceil + 4)$  approximation for BCDS.

### Corollary

There is a  $(1 - e^{-7/8})/11$ -approximation for BCDS.



# Outline

- 1 Introduction
- 2 Budgeted Connected Domination
- 3 Budgeted Edge-Vertex Domination**
- 4 Conclusions

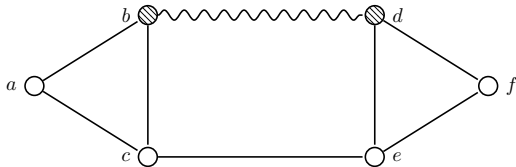
# Edge-Vertex Domination [Peters '86]

## Definition

For a set  $E' \subseteq E$ , we define

$$V(E') = \{v \in V \mid \exists u \in V \text{ such that } (v, u) \in E'\}$$

We define the *edge-set neighborhood* as  $N[E'] = N[V(E')]$ .



# Budgeted Edge-Vertex Domination

## Definition (BEVD)

Given a graph  $G = (V, E)$  and an integer  $k$ , select a subset  $E' \subseteq E$ , where  $|E'| \leq k$ , such that  $|N[E']|$  is maximized.

- **General case:** edges not necessarily connected  $\rightarrow$  forest
- **Connected case:** tree  $\rightarrow$  matches BCDS

# Budgeted Edge-Vertex Domination

## Definition (BEVD)

Given a graph  $G = (V, E)$  and an integer  $k$ , select a subset  $E' \subseteq E$ , where  $|E'| \leq k$ , such that  $|N[E']|$  is maximized.

- **General case:** edges not necessarily connected  $\rightarrow$  forest
- **Connected case:** tree  $\rightarrow$  matches BCDS

## Related Work [Miyano, Ono '11]

- Budgeted Vertex-Vertex  $[1 - e^{-1} + \epsilon, 1 - e^{-1}]$
- Budgeted Edge-Edge  $[1303/1304 + \epsilon, 1 - e^{-1}]$

# BEVD Approximation

## Definition (MAX- $k$ -COVER)

Given an integer  $k$  and a collection of sets  $S = \{S_1, S_2, \dots, S_m\}$ , find a set  $S' \subseteq S$ , where  $|S'| \leq k$ , which maximizes the number of covered elements  $|\bigcup_{S_i \in S'} S_i|$ .

## Theorem

There exists a  $(1 - e^{-1})$ -approximation algorithm for BEVD.

# BEVD Approximation

## Definition (MAX- $k$ -COVER)

Given an integer  $k$  and a collection of sets  $S = \{S_1, S_2, \dots, S_m\}$ , find a set  $S' \subseteq S$ , where  $|S'| \leq k$ , which maximizes the number of covered elements  $|\bigcup_{S_i \in S'} S_i|$ .

## Theorem

There exists a  $(1 - e^{-1})$ -approximation algorithm for BEVD.

## Proof.

Reduce to MAX- $k$ -COVER instance with  $S_i = N[e_i]$  and apply  $(1 - e^{-1})$ -approximation [Feige '98]. □

# BEVD Inapproximability

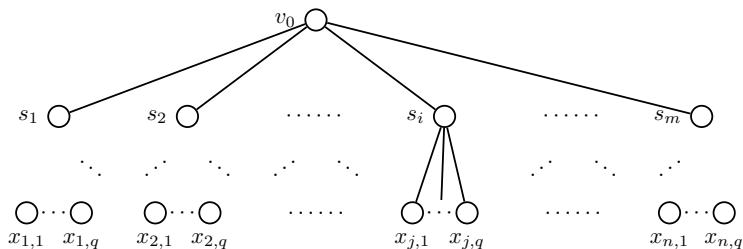


Fig.: The graph  $G$  constructed for the gap-preserving reduction. Vertex  $s_j$  is connected to vertices  $x_{j,1}, x_{j,2}, \dots, x_{j,q}$  in  $G$  if  $S_j \ni x_j$  in the MAX- $k$ -COVER instance.

## Theorem

For any  $\epsilon > 0$ , there is no polynomial time approximation algorithm for BEVD within a ratio of  $(1 - e^{-1} + \epsilon)$  unless  $P = NP$ .

# Outline

- 1 Introduction
- 2 Budgeted Connected Domination
- 3 Budgeted Edge-Vertex Domination
- 4 Conclusions**



# Conclusions

## Recap

- From  $(1 - e^{-1})/13$  to  $(1 - e^{-7/8})/11$  for BCDS  
(Independently:  $(1 - e^{-1})/12$  [Khuller, Purohit, Sarpatwar, SIAM '20])
- We introduced BEVD, and provided matching lower and upper bounds on its approximability

## Related Problems

- Partial Connected Dominating Set
- Budgeted/Partial Generalized Connected Domination
- Mixed Domination
- Expansion Ratio Variants

Thank you :)

# References

-  J.-C. Bermond, N. Cohen, D. Coudert, D. Letsios, I. Milis, S. Pérennes, V. Zissimopoulos, Bin Packing with Colocations, *WAOA '16*, LNCS, 10138:40–51.
-  U. Feige, A Threshold of  $\ln n$  for Approximating Set Cover, *Journal of the ACM*, vol. 45(4), pp. 634–652, 1998.
-  N. Garg, Saving an Epsilon: A 2-approximation for the k-MST Problem in Graphs, *Proceedings STOC '05*, pp. 396–402, 2005.
-  D.S. Johnson, M. Minkoff, and S. Phillips, The prize collecting Steiner tree problem: theory and practice, *ACM-SIAM SODA '00*, pp. 760–769, 2000.
-  S. Khuller, M. Purohit, K. K. Sarpatwar, Analyzing the optimal neighborhood: algorithms for budgeted and partial connected dominating set problems, *ACM-SIAM SODA '14*, pp. 1702-1713, 2014.
-  S. Khuller, M. Purohit, K. K. Sarpatwar, Analyzing the optimal neighborhood: algorithms for budgeted and partial connected dominating set problems, *SIAM J. DISCRETE MATH.*, vol. 34, no. 1, pp. 251–270, 2020.
-  Y. Liu, and W. Liang, Approximate Coverage in Wireless Sensor Networks, *IEEE Conference on Local Computer Networks 30th Anniversary*, pp. 68–75, 2005.
-  E. Miyano, and H. Ono, Maximum Domination Problem, *Proceedings of the Seventeenth Computing: the Australasian Theory Symposium (CATS)*, vol. 119, pp. 55–62, 2011.
-  K.W. Peters, Theoretical and Algorithmic Results on Domination and Connectivity, *PhD thesis, Clemson University*, Clemson, SC, 1986.
-  B. Wang, Coverage Problems in Sensor Networks, *ACM Computing Surveys*, vol. 43(4):32, 2011.