Hardness of Approximation meets Parameterized Complexity

Karthik C. S.

New York University

December 28, 2020

Global Outline

Day 1: The Setting

Day 2: Gap Creation

Day 3: Applications

Day 3 Outline

Part 1: Hardness of Approximating Set Cover

- Recap
- MinLabel
- Gap Translation to Set Cover

Day 3 Outline

Part 1: Hardness of Approximating Set Cover

- Recap
- MinLabel
- Gap Translation to Set Cover

Part 2: Hardness of Biclique

- Recap
- Hardness of Biclique

Day 3 Outline

Part 1: Hardness of Approximating Set Cover

- Recap
- MinLabel
- Gap Translation to Set Cover

Part 2: Hardness of Biclique

- Recap
- Hardness of Biclique

Part 3: Hardness of Approximating Clique

Day 3 Outline

Part 1: Hardness of Approximating Set Cover

- Recap
- MinLabel
- Gap Translation to Set Cover

Part 2: Hardness of Biclique

- Recap
- Hardness of Biclique

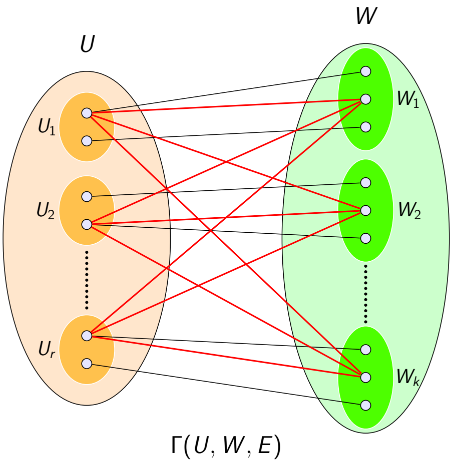
Part 3: Hardness of Approximating Clique

Part 4: Selected Open Problems

Part 1

Hardness of Approximating Set Cover

MaxCover: Recap



Each W_i is a Right Super Node
Each U_i is a Left Super Node

$S \subseteq W$ is a labeling of W if
$\forall i \in [k], |S \cap W_i| = 1$

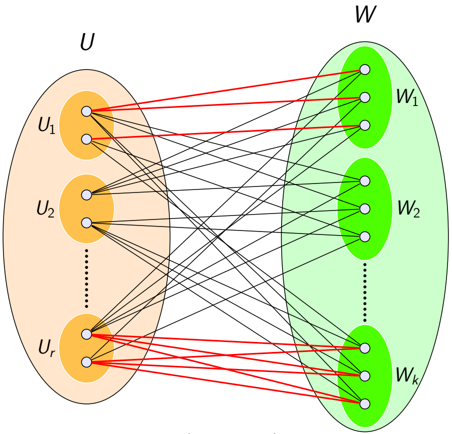
S covers U_i if
$\exists u \in U_i,\ \forall v \in S, (u, v) \in E$

MaxCover(Γ, S) = Fraction of U_i's covered by S

$$\text{MaxCover}(\Gamma) = \max_S \text{MaxCover}(\Gamma, S)$$

Determine if MaxCover$(\Gamma) = 1$ or MaxCover$(\Gamma) \leq s$

MaxCover: Projection Property



$\Gamma(U, W, E)$

Γ has projection property:

For every U_i and W_j,

Induced subgraph of (U_i, W_j) is:

- complete bipartite graph (i.e., irrelevant), or,
- $\forall w \in W_j$, deg(w)=1 (i.e., projection)

MaxCover with projection property is W[1]-Hard

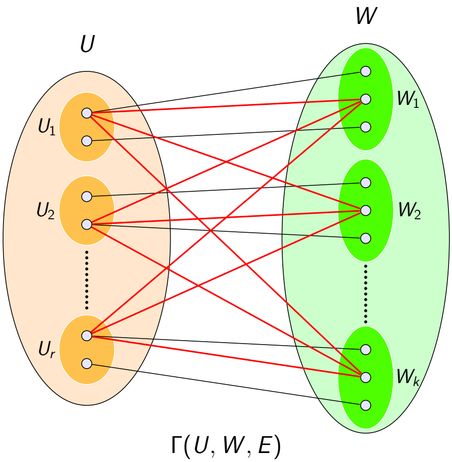
MaxCover: Gap Creation

Inapproximability of MaxCover using Reed Solomon Codes

There is a FPT reduction from MaxCover instance $\Gamma_0 = \left(U_0 = \bigcup_{j=1}^{r} U_j^0, W = \bigcup_{j=1}^{k} W_i, E_0\right)$ with projection property to a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{q} U_j, W = \bigcup_{j=1}^{k} W_i, E\right)$ such that

- If $\text{MaxCover}(\Gamma_0) = 1$ then $\text{MaxCover}(\Gamma) = 1$
- If $\text{MaxCover}(\Gamma_0) < 1$ then $\text{MaxCover}(\Gamma) \leq \frac{\log_q |U_0|}{q}$
- $|\Gamma| = \tilde{O}(q^r \cdot |W| \cdot \log |U_0|)$
- The reduction runs in time $q^r \cdot \text{poly}(|\Gamma_0|)$.

MinLabel [CCKLMNT'17]

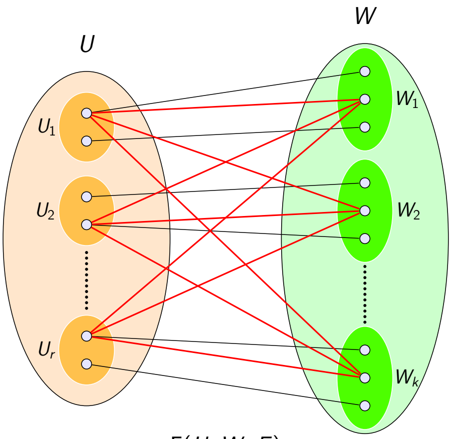


Each W_i is a Right Super Node
Each U_i is a Left Super Node

$S \subseteq W$ is a labeling of W if
$\forall i \in [k], |S \cap W_i| = 1$

S covers U_i if
$\exists u \in U_i,\ \forall v \in S, (u, v) \in E$

MinLabel [CCKLMNT'17]



$\Gamma(U, W, E)$

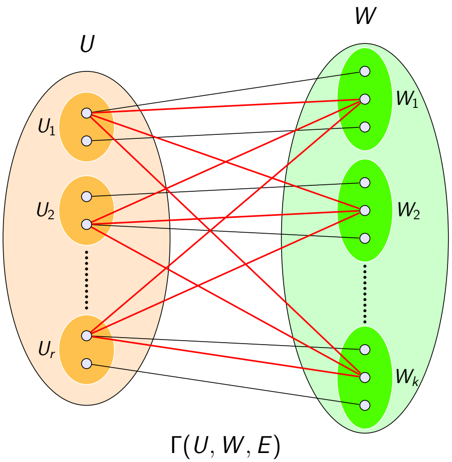
Each W_i is a Right Super Node
Each U_i is a Left Super Node

$S \subseteq W$ is a labeling of W if
$\forall i \in [k], |S \cap W_i| = 1$

S covers U_i if
$\exists u \in U_i,\ \forall v \in S, (u, v) \in E$

MinLabel(Γ) = smallest $X \subseteq W$:
$\forall i \in [r], \exists$labeling $S \subseteq X$,
S covers U_i

MinLabel [CCKLMNT'17]



Each W_i is a Right Super Node
Each U_i is a Left Super Node

$S \subseteq W$ is a labeling of W if
$\forall i \in [k], |S \cap W_i| = 1$

S covers U_i if
$\exists u \in U_i,\ \forall v \in S, (u, v) \in E$

MinLabel(Γ) = smallest $X \subseteq W$:
$\forall i \in [r], \exists$labeling $S \subseteq X$,
S covers U_i

Determine if MinLabel(Γ) $= k$
or MinLabel(Γ) $\geq s \cdot k$

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E \right)$,

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E \right)$,

- Completeness: If MaxCover(Γ) $= 1$, then MinLabel(Γ) $= k$

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E\right)$,

- Completeness: If $\text{MaxCover}(\Gamma) = 1$, then $\text{MinLabel}(\Gamma) = k$
- Soundness: If $\text{MaxCover}(\Gamma) \leq \varepsilon$, then $\text{MinLabel}(\Gamma) \geq (1/\varepsilon)^{1/k} \cdot k$

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E \right)$,

- Completeness: If $\text{MaxCover}(\Gamma) = 1$, then $\text{MinLabel}(\Gamma) = k$
- Soundness: If $\text{MaxCover}(\Gamma) \leq \varepsilon$, then $\text{MinLabel}(\Gamma) \geq (1/\varepsilon)^{1/k} \cdot k$

Completeness is obvious.

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E \right)$,

- Completeness: If MaxCover(Γ) $= 1$, then MinLabel(Γ) $= k$
- Soundness: If MaxCover(Γ) $\leq \varepsilon$, then MinLabel(Γ) $\geq (1/\varepsilon)^{1/k} \cdot k$

Completeness is obvious. In Soundness case, if $X \subseteq W$ is a MinLabel solution

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E \right)$,

- Completeness: If $\text{MaxCover}(\Gamma) = 1$, then $\text{MinLabel}(\Gamma) = k$
- Soundness: If $\text{MaxCover}(\Gamma) \leq \varepsilon$, then $\text{MinLabel}(\Gamma) \geq (1/\varepsilon)^{1/k} \cdot k$

Completeness is obvious. In Soundness case, if $X \subseteq W$ is a MinLabel solutionthen every labeling $S \subseteq X$ covers at most ε fraction of the left supernodes.

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E\right)$,

- Completeness: If $\text{MaxCover}(\Gamma) = 1$, then $\text{MinLabel}(\Gamma) = k$
- Soundness: If $\text{MaxCover}(\Gamma) \leq \varepsilon$, then $\text{MinLabel}(\Gamma) \geq (1/\varepsilon)^{1/k} \cdot k$

Completeness is obvious. In Soundness case, if $X \subseteq W$ is a MinLabel solutionthen every labeling $S \subseteq X$ covers at most ε fraction of the left supernodes. There are at most $\binom{|X|/k}{k}$ distinct labeling of W in X.

Maxcover to MinLabel

Reduction from MaxCover to MinLabel [CCKLMNT17]

Given a MaxCover instance $\Gamma = \left(U = \bigcup_{j=1}^{r} U_j, W = \bigcup_{j=1}^{k} W_i, E \right)$,

- Completeness: If $\text{MaxCover}(\Gamma) = 1$, then $\text{MinLabel}(\Gamma) = k$
- Soundness: If $\text{MaxCover}(\Gamma) \leq \varepsilon$, then $\text{MinLabel}(\Gamma) \geq (1/\varepsilon)^{1/k} \cdot k$

Completeness is obvious. In Soundness case, if $X \subseteq W$ is a MinLabel solutionthen every labeling $S \subseteq X$ covers at most ε fraction of the left supernodes. There are at most $\binom{|X|/k}{k}$ distinct labeling of W in X.

$$\binom{|X|/k}{k} \cdot \varepsilon \geq 1$$

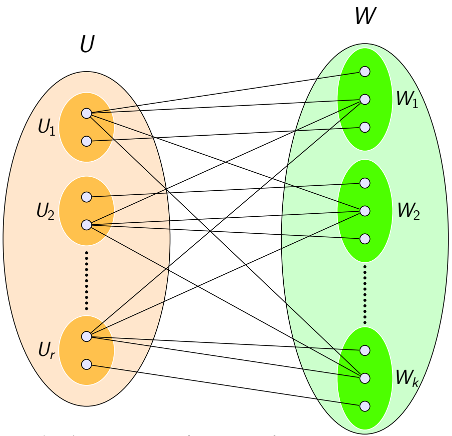
Improved Inapproximability of MinLabel

Inapproximability of MinLabel [K-LivniNavon'21]

There is a FPT reduction from MaxCover instance $\Gamma_0 = \left(U_0 = \bigcup_{j=1}^{r} U_j^0, W = \bigcup_{j=1}^{k} W_i, E_0\right)$ with projection property to a MinLabel instance $\Gamma = \left(U = \bigcup_{j=1}^{2^{O(q)}} U_j, W = \bigcup_{j=1}^{k} W_i, E\right)$ such that

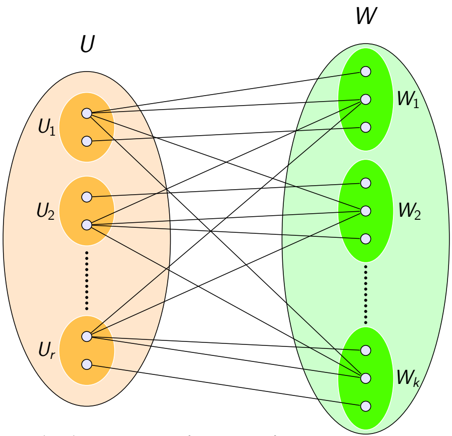
- If $\mathsf{MaxCover}(\Gamma_0) = 1$ then $\mathsf{MinLabel}(\Gamma) = k$
- If $\mathsf{MaxCover}(\Gamma_0) < 1$ then $\mathsf{MinLabel}(\Gamma) \geq \frac{q}{r}$
- $|\Gamma| = \tilde{O}(q^r \cdot |W| \cdot \log |U_0|)$
- The reduction runs in time $q^r \cdot \mathrm{poly}(|\Gamma_0|)$.

MinLabel to Set Cover [Feige'98]



$|U_i| = \ell$ $\Gamma(U, W, E)$

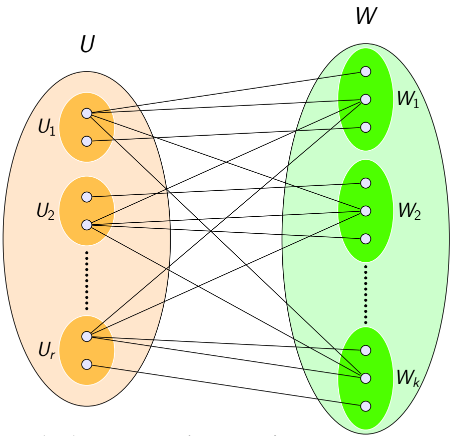
MinLabel to Set Cover [Feige'98]



$|U_i| = \ell$ $\Gamma(U, W, E)$

$$\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$$
$$\forall w \in W_j, S_w \in \mathcal{S}$$

MinLabel to Set Cover [Feige'98]

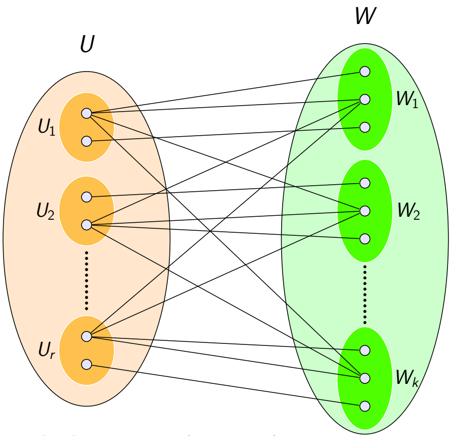


$|U_i| = \ell$ $\quad$ $\Gamma(U, W, E)$

$$\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$$
$$\forall w \in W_j, S_w \in \mathcal{S}$$

$$(i, f) \in S_w \Leftrightarrow \exists u \in U_i$$
$$(u, w) \in E \text{ and } f(u) = j$$

MinLabel to Set Cover [Feige'98]



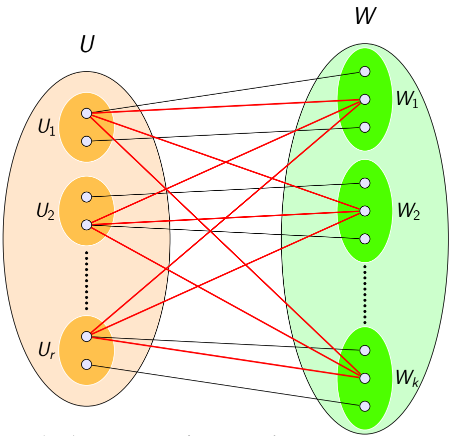
$|U_i| = \ell$ $\quad \Gamma(U, W, E)$

$$\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$$
$$\forall w \in W_j, S_w \in \mathcal{S}$$

$$(i, f) \in S_w \Leftrightarrow \exists u \in U_i$$
$$(u, w) \in E \text{ and } f(u) = j$$

$$|\mathcal{S}| = |W|, |\mathcal{U}| = r \cdot k^{\ell}$$

MinLabel to Set Cover [Feige'98]



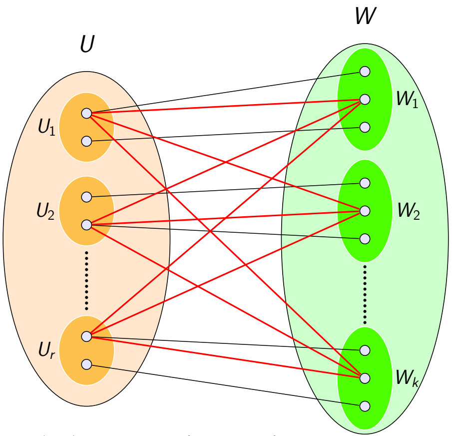
$|U_i| = \ell$ $\quad \Gamma(U, W, E)$

$$\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$$
$$\forall w \in W_j, S_w \in \mathcal{S}$$

$$(i, f) \in S_w \Leftrightarrow \exists u \in U_i$$
$$(u, w) \in E \text{ and } f(u) = j$$

$$|\mathcal{S}| = |W|, |\mathcal{U}| = r \cdot k^{\ell}$$

$(w_1, \ldots, w_k)$ is labeling
that covers every $U_i \Rightarrow$
$(S_{w_1}, \ldots, S_{w_k})$ covers $\mathcal{U}$



$|U_i| = \ell$ $\quad \Gamma(U, W, E)$

$$\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$$
$$\forall w \in W_j, S_w \in \mathcal{S}$$

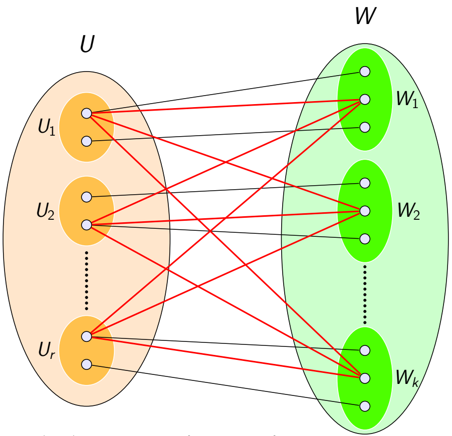
$(i, f) \in S_w \Leftrightarrow \exists u \in U_i$
$(u, w) \in E$ and $f(u) = j$

$$|\mathcal{S}| = |W|, |\mathcal{U}| = r \cdot k^{\ell}$$

$(w_1, \ldots, w_k)$ is labeling
that covers every $U_i \Rightarrow$
$(S_{w_1}, \ldots, S_{w_k})$ covers $\mathcal{U}$

$\forall (i, f) \in \mathcal{U},\ \exists u \in U_i,$
$(u, w_j) \in E (\forall j \in [k])$

MinLabel to Set Cover [Feige'98]



$|U_i| = \ell$ $\quad \Gamma(U, W, E)$

$$\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$$
$$\forall w \in W_j, S_w \in \mathcal{S}$$

$$(i, f) \in S_w \Leftrightarrow \exists u \in U_i$$
$$(u, w) \in E \text{ and } f(u) = j$$

$$|\mathcal{S}| = |W|, |\mathcal{U}| = r \cdot k^{\ell}$$

$(w_1, \ldots, w_k)$ is labeling that covers every $U_i \Rightarrow (S_{w_1}, \ldots, S_{w_k})$ covers $\mathcal{U}$

$$\forall (i, f) \in \mathcal{U},\ \exists u \in U_i,$$
$$(u, w_j) \in E (\forall j \in [k])$$

Determine if SetCover$(\mathcal{U}, \mathcal{S}) = k$
or SetCover$(\mathcal{U}, \mathcal{S}) \geq s \cdot k$ is hard!

MinLabel to Set Cover: Soundness Analysis

- $\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$
- $(i, f) \in S_w \Leftrightarrow \exists u \in U_i : (u, w) \in E$ and $f(u) = j$

MinLabel to Set Cover: Soundness Analysis

- $\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$
- $(i, f) \in S_w \Leftrightarrow \exists u \in U_i : (u, w) \in E$ and $f(u) = j$
- Suppose X is a set cover of size $sk - 1$

MinLabel to Set Cover: Soundness Analysis

- $\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$
- $(i, f) \in S_w \Leftrightarrow \exists u \in U_i : (u, w) \in E$ and $f(u) = j$
- Suppose X is a set cover of size $sk - 1$
- $\exists U_i$ not covered by any labeling in X

MinLabel to Set Cover: Soundness Analysis

- $\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$
- $(i, f) \in S_w \Leftrightarrow \exists u \in U_i : (u, w) \in E$ and $f(u) = j$
- Suppose X is a set cover of size $sk - 1$
- $\exists U_i$ not covered by any labeling in X
- For every $u \in U_i$ there is some $j \in [k]$ such that $W_j \cap X \cap N(u)$ is empty

MinLabel to Set Cover: Soundness Analysis

- $\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$
- $(i, f) \in S_w \Leftrightarrow \exists u \in U_i : (u, w) \in E$ and $f(u) = j$
- Suppose X is a set cover of size $sk - 1$
- $\exists U_i$ not covered by any labeling in X
- For every $u \in U_i$ there is some $j \in [k]$ such that $W_j \cap X \cap N(u)$ is empty
- Construct f using above u

MinLabel to Set Cover: Soundness Analysis

- $\mathcal{U} = \{(i, f) \mid i \in [r], f : [\ell] \to [k]\}$
- $(i, f) \in S_w \Leftrightarrow \exists u \in U_i : (u, w) \in E$ and $f(u) = j$
- Suppose X is a set cover of size $sk - 1$
- $\exists U_i$ not covered by any labeling in X
- For every $u \in U_i$ there is some $j \in [k]$ such that $W_j \cap X \cap N(u)$ is empty
- Construct f using above u
- (i, f) is not covered by X

Parameterized Inapproximability of Set Cover

Inapproximability of Set Cover [K-LivniNavon'21]

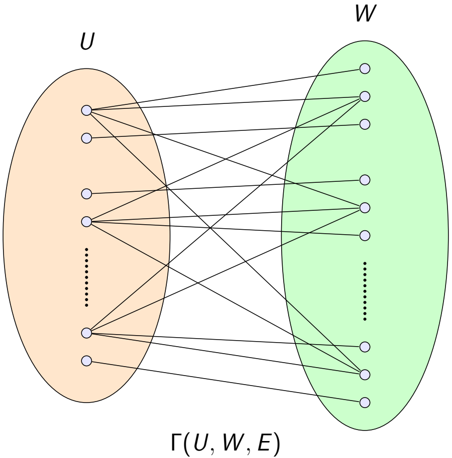
There is a FPT reduction from k-clique instance $G([n], E)$ to a Set Cover instance $(\mathcal{U}, \mathcal{S})$ such that

- If G has a k-clique then $\binom{k}{2}$ sets in $\mathcal{S}$ cover $\mathcal{U}$
- If G has no k-clique then $(\log n)^{1/k}$ sets in $\mathcal{S}$ are needed to cover $\mathcal{U}$
- $|\mathcal{U}|, |\mathcal{S}| \leq n$
- The reduction runs in time $2^{\text{poly}(k)} \cdot \text{poly}(n)$.

Part 2

Hardness of Biclique

One-Sided Biclique: Recap



Find k vertices in W
with most common neighbors

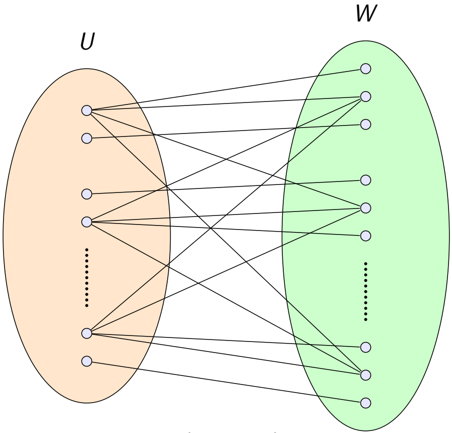
One-Sided Biclique: Gap Creation

Inapproximability of One-Sided Biclique (Lin'18)

There is a FPT reduction from k-Clique instance $G([n], E_0)$ to a One-Sided Biclique instance $\Gamma = (U, W, E)$ such that

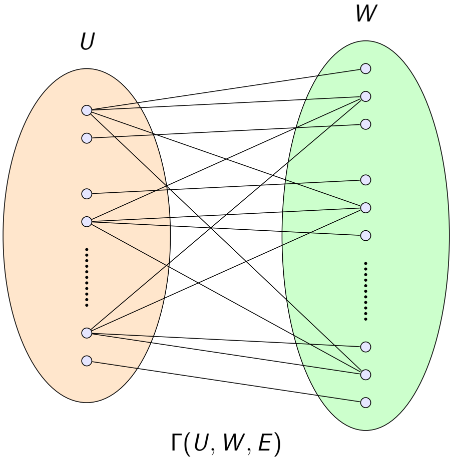
- If G has a k-clique then there are $\binom{k}{2}$ vertices in W which have $n^{1/k}$ common neighbors in U
- If G has no k-clique then for every $\binom{k}{2}$ vertices in W they have at most $(k+1)!$ common neighbors in U
- $|\Gamma| = n^3$
- The reduction runs in time poly(n)

Biclique: Definition



$\Gamma(U, W, E)$

Biclique: Definition



Find k vertices in W
with k common neighbors in U

Biclique: Hardness

Hardness of Biclique (Lin'18)

There is a FPT reduction from k-Clique instance $G([n], E_0)$ to a Biclique instance $\Gamma = (U, W, E)$ such that

- If G has a k-clique then there are $(k+1)! + 1$ vertices in W which have $(k+1)! + 1$ common neighbors in U
- If G has no k-clique then for every $(k+1)! + 1$ vertices in W they have at most $(k+1)!$ common neighbors in U
- $|\Gamma| = n^3$
- The reduction runs in time poly(n)

Part 3

Gap-ETH Hardness of Approximation of Clique

Gap-ETH

Gap-ETH

$\exists \varepsilon, \delta > 0$, no algorithm can solve $(1 \text{ vs. } 1-\delta)$-Gap 3-SAT on n variables in $2^{\varepsilon n}$ time.

Hardness of Approximating Clique

Inapproximability of Clique [Chalermsook et al. '17]

Assuming Gap-ETH, there is no FPT algorithm that can distinguish between the following two cases

Completeness: G has a k-clique

Soundness: G has no $(\log k)$-clique

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly.

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly. We will construct a graph G on independent disjoint sets $V_1, \ldots, V_k$.

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly. We will construct a graph G on independent disjoint sets $V_1, \ldots, V_k$.Each V_i contains a vertex for every satisfying partial assignment to clauses in C_i.

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly. We will construct a graph G on independent disjoint sets $V_1, \ldots, V_k$.Each V_i contains a vertex for every satisfying partial assignment to clauses in C_i. Insert an edge $(v_i, v_j) \in V_i \times V_j$ iff the partial assignments are consistent. Note $|V_i| \leq 2^{O(Dn/logk)}$.

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly. We will construct a graph G on independent disjoint sets $V_1, \ldots, V_k$.Each V_i contains a vertex for every satisfying partial assignment to clauses in C_i. Insert an edge $(v_i, v_j) \in V_i \times V_j$ iff the partial assignments are consistent. Note $|V_i| \leq 2^{O(Dn/logk)}$.

Completeness: If σ is a satisfying assignment to φ then we pick in V_i the restriction of σ to variables appearing in C_i.

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly. We will construct a graph G on independent disjoint sets $V_1, \ldots, V_k$.Each V_i contains a vertex for every satisfying partial assignment to clauses in C_i. Insert an edge $(v_i, v_j) \in V_i \times V_j$ iff the partial assignments are consistent. Note $|V_i| \leq 2^{O(Dn/logk)}$.

Completeness: If σ is a satisfying assignment to φ then we pick in V_i the restriction of σ to variables appearing in C_i.

Soundness: Any collection $C_{i_1}, \ldots, C_{i_{\log k}}$ contains $(1 - \delta/2)$ fraction of clauses of φ.

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly. We will construct a graph G on independent disjoint sets $V_1, \ldots, V_k$.Each V_i contains a vertex for every satisfying partial assignment to clauses in C_i. Insert an edge $(v_i, v_j) \in V_i \times V_j$ iff the partial assignments are consistent. Note $|V_i| \leq 2^{O(Dn/logk)}$.

Completeness: If σ is a satisfying assignment to φ then we pick in V_i the restriction of σ to variables appearing in C_i.

Soundness: Any collection $C_{i_1}, \ldots, C_{i_{\log k}}$ contains $(1 - \delta/2)$ fraction of clauses of φ. If $(v_{i_1}, \ldots, v_{i_{\log k}}) \in V_{i_1} \times \cdots \times V_{i_{\log k}}$ is a clique then we can find an assignment that satisfies $(1 - \delta/2)$ fraction of clauses.

Hardness of Approximating Clique: Proof

Given φ on n variables and cn clauses,for every $i \in [k]$, construct C_i by picking $Dn/\log k$ clauses randomly. We will construct a graph G on independent disjoint sets $V_1, \dots, V_k$.Each V_i contains a vertex for every satisfying partial assignment to clauses in C_i. Insert an edge $(v_i, v_j) \in V_i \times V_j$ iff the partial assignments are consistent. Note $|V_i| \leq 2^{O(Dn/logk)}$.

Completeness: If σ is a satisfying assignment to φ then we pick in V_i the restriction of σ to variables appearing in C_i.

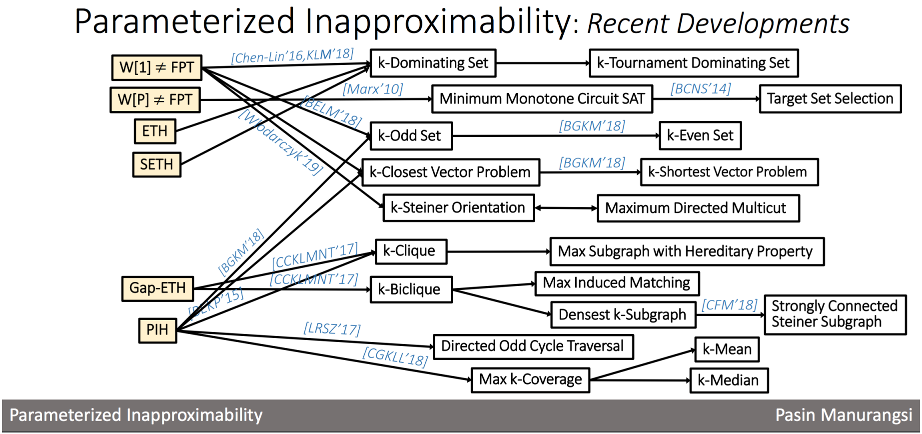
Soundness: Any collection $C_{i_1}, \dots, C_{i_{\log k}}$ contains $(1 - \delta/2)$ fraction of clauses of φ. If $(v_{i_1}, \dots, v_{i_{\log k}}) \in V_{i_1} \times \cdots \times V_{i_{\log k}}$ is a clique then we can find an assignment that satisfies $(1 - \delta/2)$ fraction of clauses.

Any $T(k) \cdot \text{poly}(|V|)$ algorithm for gap k-clique yields a $T(k) \cdot 2^{O(Dn/\log k)}$ algorithm for $(1 \text{ vs. } 1 - \delta)$-Gap 3-SAT.

Part 4

Open Problems from these Lectures

Parameterized Inapproximability: Partial Summary



Clique

Is it W[1]-Hard to approximate k-Clique to 1.01 factor?

Gap-ETH

ETH

$\exists \varepsilon > 0$, no algorithm can solve 3-SAT on n variables in $2^{\varepsilon n}$ time.

Gap-ETH

ETH

$\exists \varepsilon > 0$, no algorithm can solve 3-SAT on n variables in $2^{\varepsilon n}$ time.

Gap-ETH

$\exists \varepsilon, \delta > 0$, no algorithm can solve $(1 \text{ vs. } 1 - \delta)$-Gap 3-SAT on n variables in $2^{\varepsilon n}$ time.

Gap-ETH

ETH

$\exists \varepsilon > 0$, no algorithm can solve 3-SAT on n variables in $2^{\varepsilon n}$ time.

Gap-ETH

$\exists \varepsilon, \delta > 0$, no algorithm can solve $(1$ vs. $1 - \delta)$-Gap 3-SAT on n variables in $2^{\varepsilon n}$ time.

Does Gap-ETH follow from ETH?

PIH/Label Cover

PIH [Lokshtanov-Ramanujan-Saurabh-Zehavi'17]

Is it W[1]-Hard to approximate 2-CSP on k variables and n alphabet?

Set Cover

Is it W[2]-Hard to approximate k-Set Cover to 1.01 factor?

Set Cover

Is it W[2]-Hard to approximate k-Set Cover to 1.01 factor?

Is it W[1]-Hard to approximate k-Set Cover to $o(\log n)$ factor?

Set Cover

Is it W[2]-Hard to approximate k-Set Cover to 1.01 factor?

Is it W[1]-Hard to approximate k-Set Cover to $o(\log n)$ factor?

Is it W[1]-Hard to approximate k-MaxCoverage beyond $1 - 1/e$ factor?

Biclique

Is it W[1]-Hard to approximate k-Biclique to 1.01 factor?

MaxCover vs One-Sided Biclique

Is MaxCover equivalent to One-Sided Biclique?