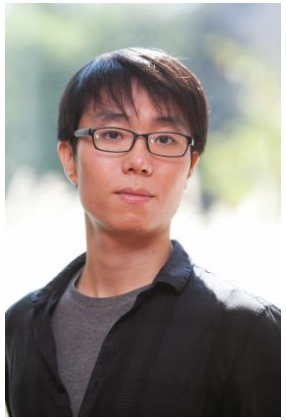
Finding Best LP Relaxations for Directed Cut Problems

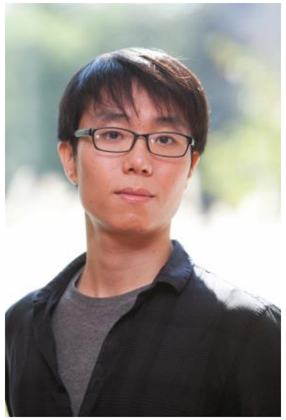
Euiwoong Lee

CMU → *Simons*

People & Papers



People & Papers



Multicut(H)

People & Papers

Multicut,
Interdiction,
Firefighter

Multicut(H)

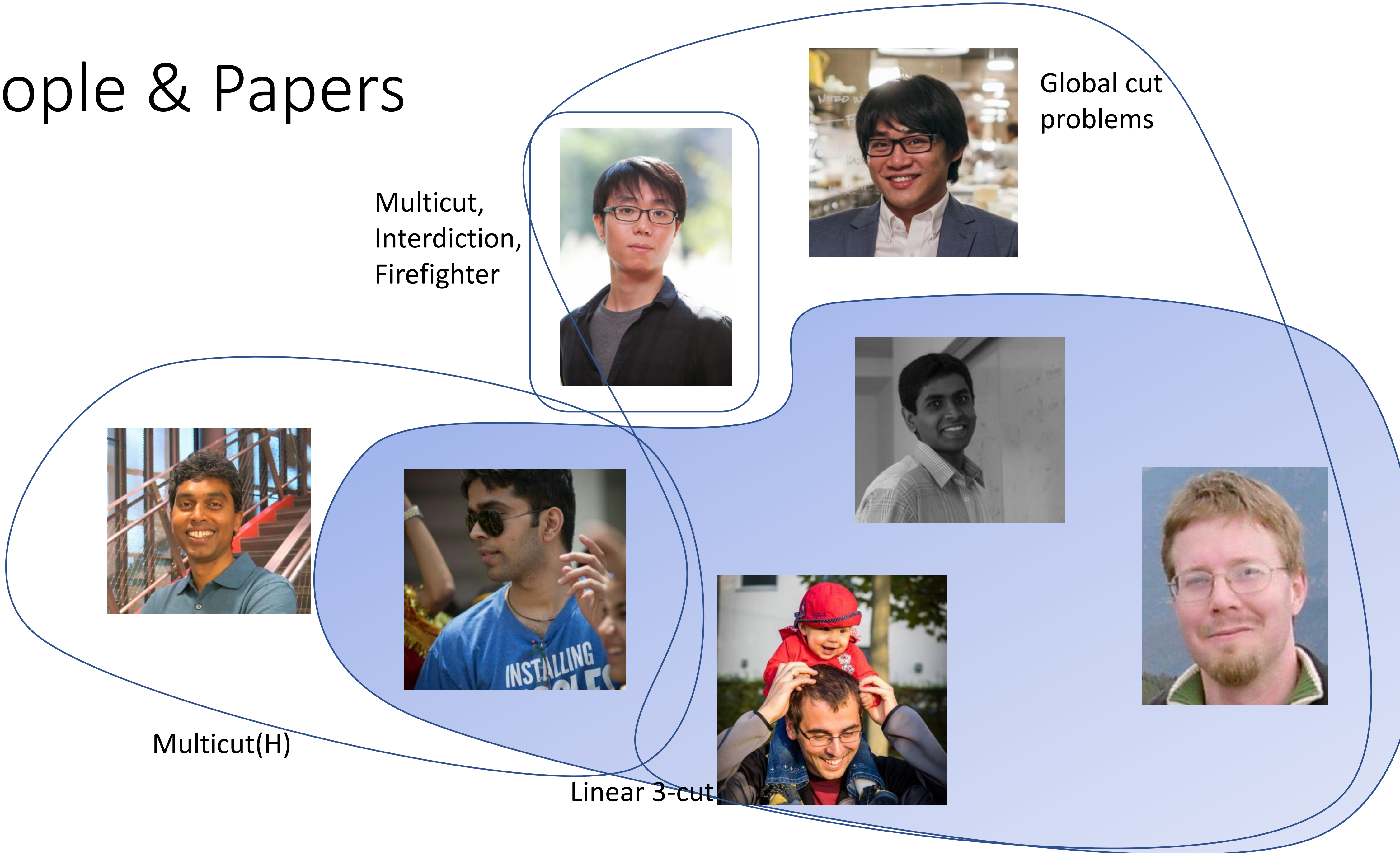
People & Papers

Global cut problems

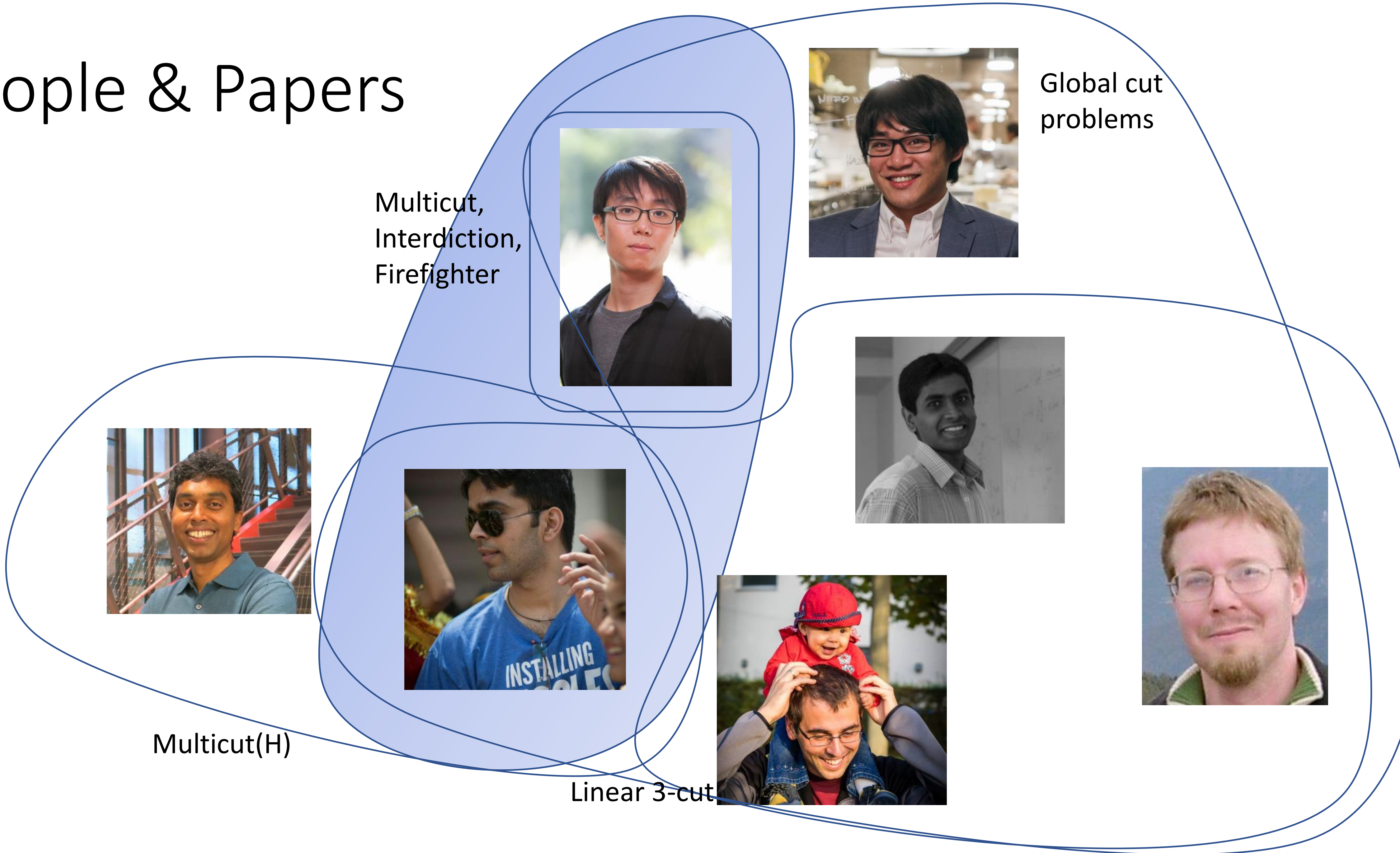
Multicut, Interdiction, Firefighter

Multicut(H)

People & Papers

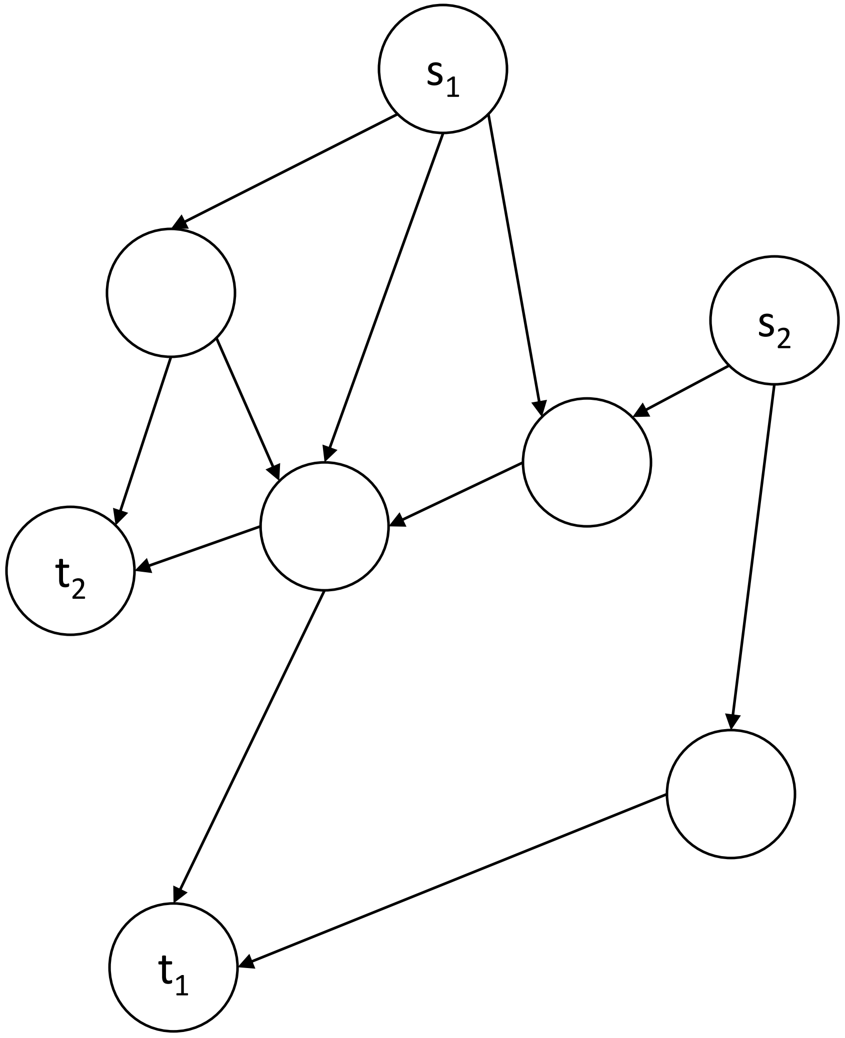


People & Papers



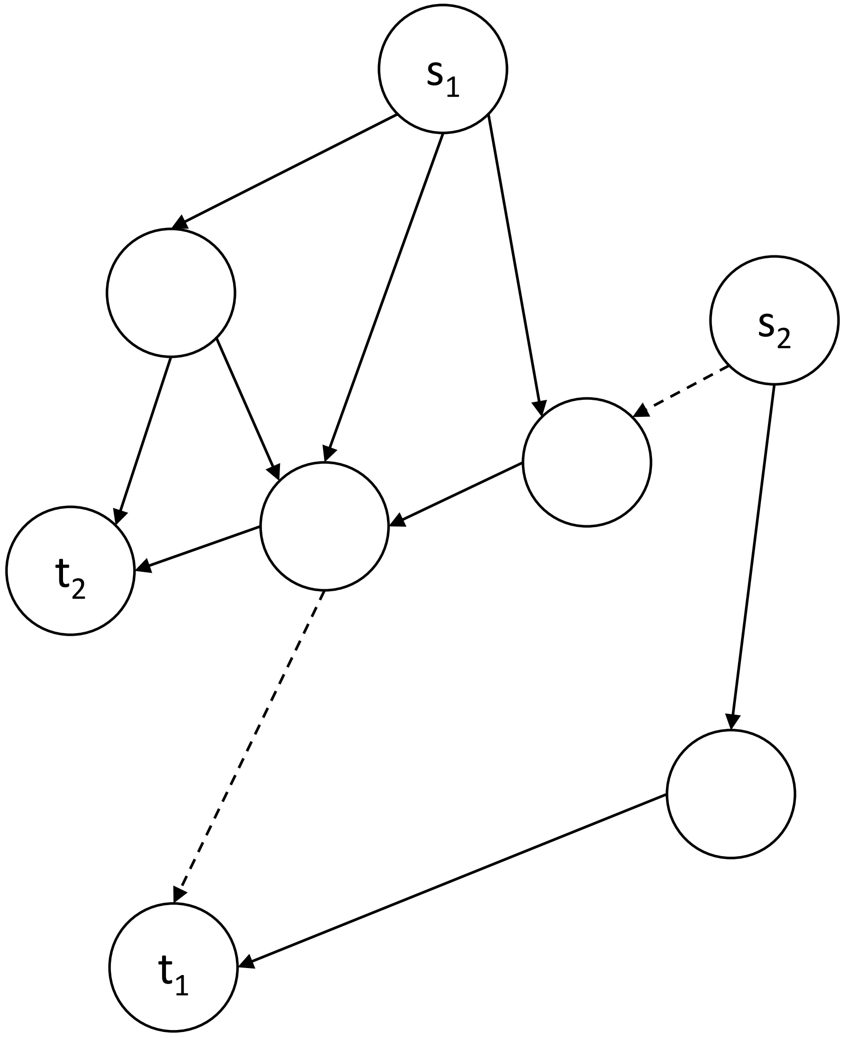
Directed Multicut

- Input
  - Directed Graph G=(V, E), k pairs $(s_1, t_1), \ldots, (s_k, t_k)$
- Goal
  - Remove minimum # of edges to cut all s_i-t_i path.



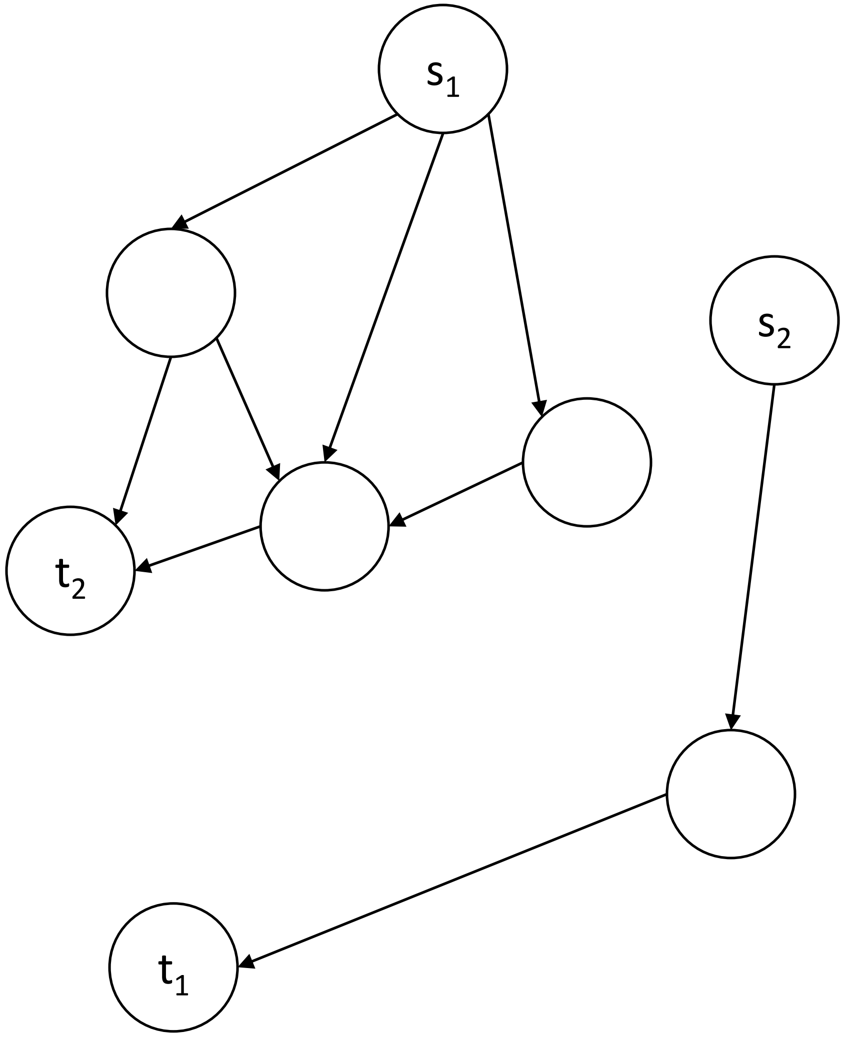
Directed Multicut

- Input
  - Directed Graph G=(V, E), k pairs $(s_1, t_1), \ldots, (s_k, t_k)$
- Goal
  - Remove minimum # of edges to cut all s_i-t_i path.



Directed Multicut

- Input
  - Directed Graph G=(V, E), k pairs $(s_1, t_1), \ldots, (s_k, t_k)$
- Goal
  - Remove minimum # of edges to cut all s_i-t_i path.



Directed Multicut (before 2017)

- In terms of n,

- [CKR 01] $\tilde{O}(n^{1/2})$-approx.
- [Gupta 03] $O(n^{1/2})$-approx.
- [AAC 07] $\tilde{O}(n^{11/23})$-approx.
- [CK 07]
  - $\widetilde{\Omega}(n^{1/7})$ flow-cut gap.
  - $2^{\Omega(\log^{1-\epsilon} n)}$-(NP) hard.

- In terms of k,

- Easy k-approx.
- [SSZ 00] $k = O(\log n / \log\log n)$,
  - Flow-cut gap is $k - o(1)$.
- [CM 16, EVW 13] 1.5-(UG) hard when $k = 2$.
  - From Undir. Node Multiway Cut
  - Best for any constant k?

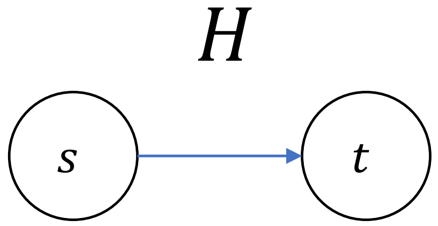
In 2017

- [CM 17, L 17]
  - Directed Multicut with k pairs is k-(UG) hard.
- [CM 17]
  - Reduction from CSP [EVW 13].
  - Interesting connections between different LP relaxations.
- [L 17]
  - Direct reduction from UG.
  - Easy(?) to adapt to other cut problems.

Only in [CM 17]

- Let $H = (V_H, E_H)$ be a fixed demand graph.
- Multicut(H)
  - Input: Supply graph $G = (V_G, E_G)$ and injective map $\pi: V_H \to V_G$.
  - Goal: Remove min # edges from G such that
    - $\forall (u, v) \in E_H$, there is no path from $\pi(u)$ to $\pi(v)$ in G.

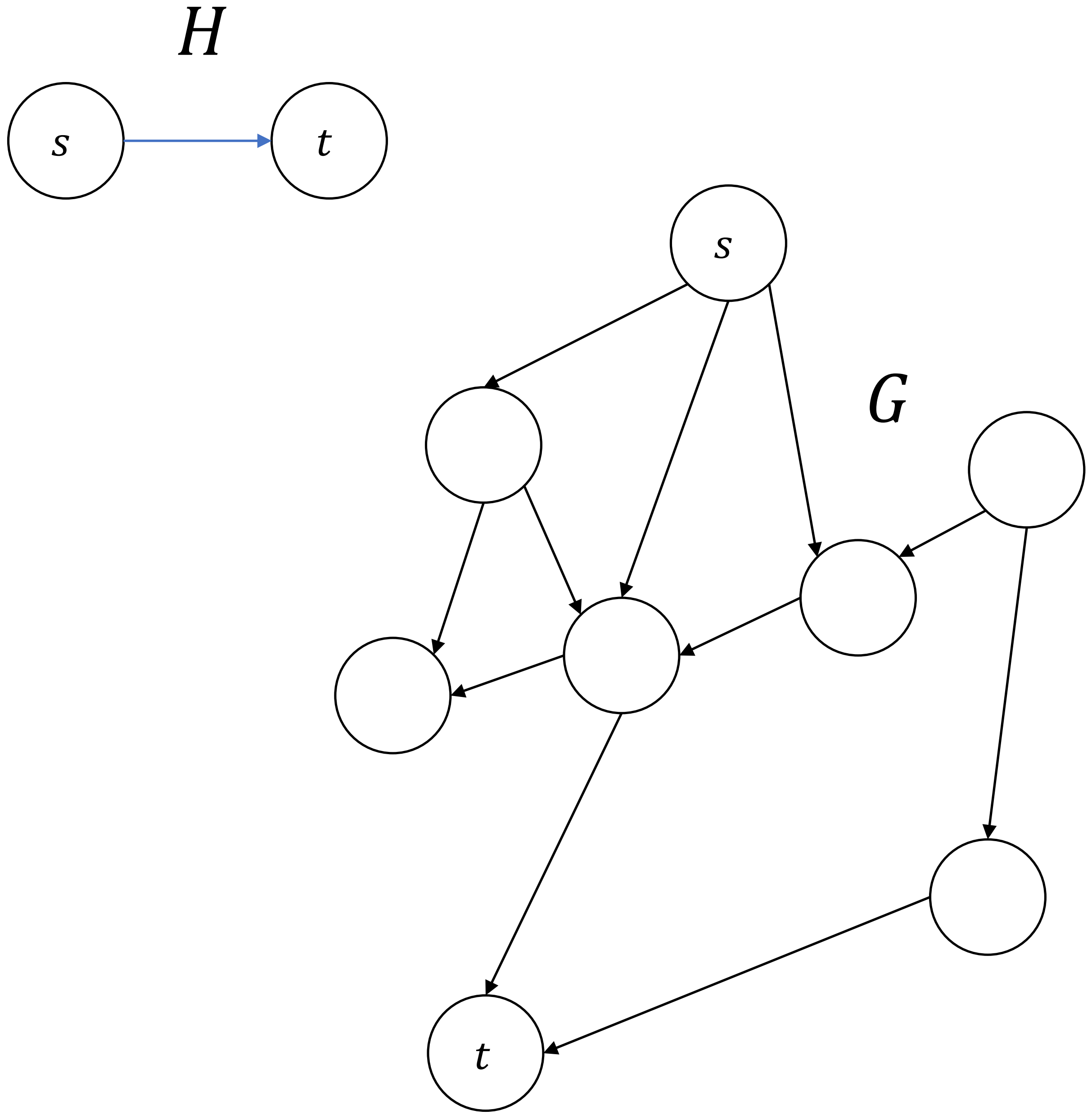
Only in [CM 17]



- Let $H = (V_H, E_H)$ be a fixed demand graph.
- Multicut(H)
  - Input: Supply graph $G = (V_G, E_G)$ and injective map $\pi: V_H \to V_G$.
  - Goal: Remove min # edges from G such that
    - $\forall (u, v) \in E_H$, there is no path from $\pi(u)$ to $\pi(v)$ in G.

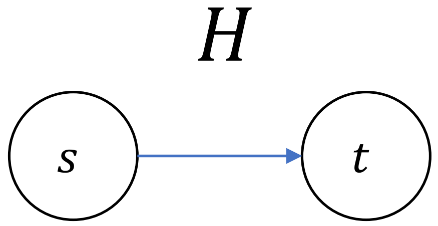
Only in [CM 17]

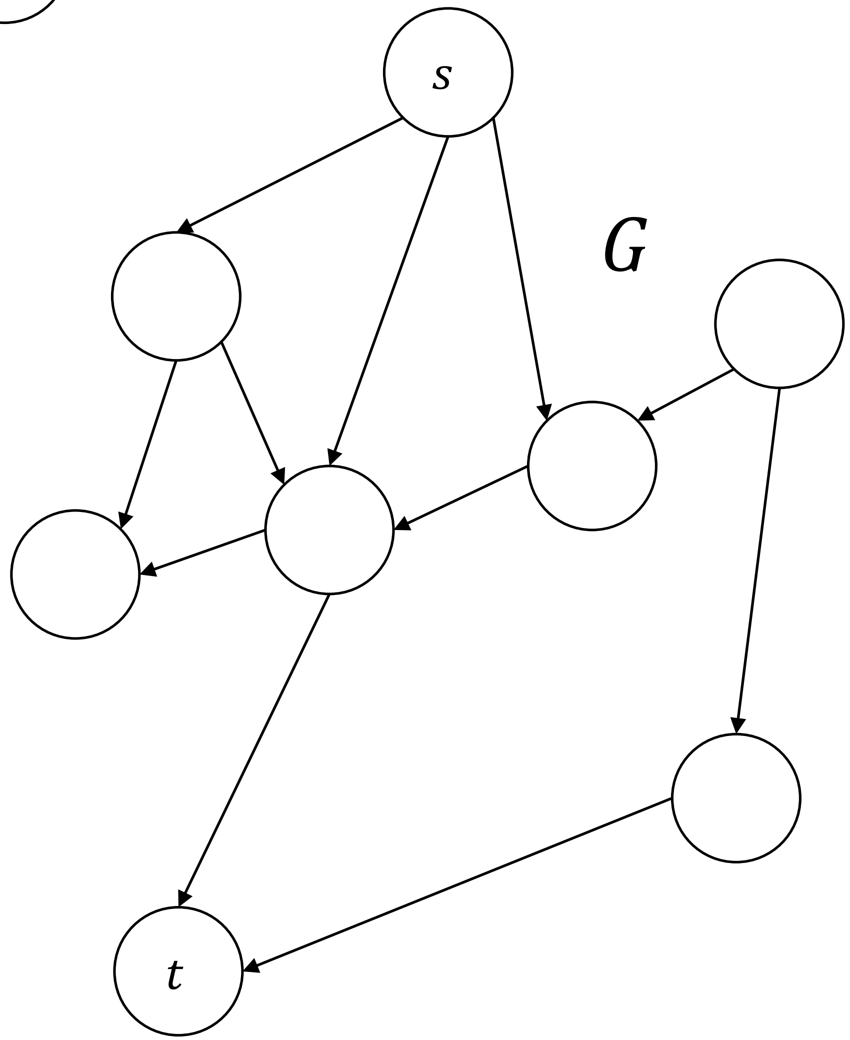
- Let $H = (V_H, E_H)$ be a fixed demand graph.
- Multicut(H)
  - Input: Supply graph $G = (V_G, E_G)$ and injective map $\pi: V_H \to V_G$.
  - Goal: Remove min # edges from G such that
    - $\forall (u, v) \in E_H$, there is no path from $\pi(u)$ to $\pi(v)$ in G.



Only in [CM 17]

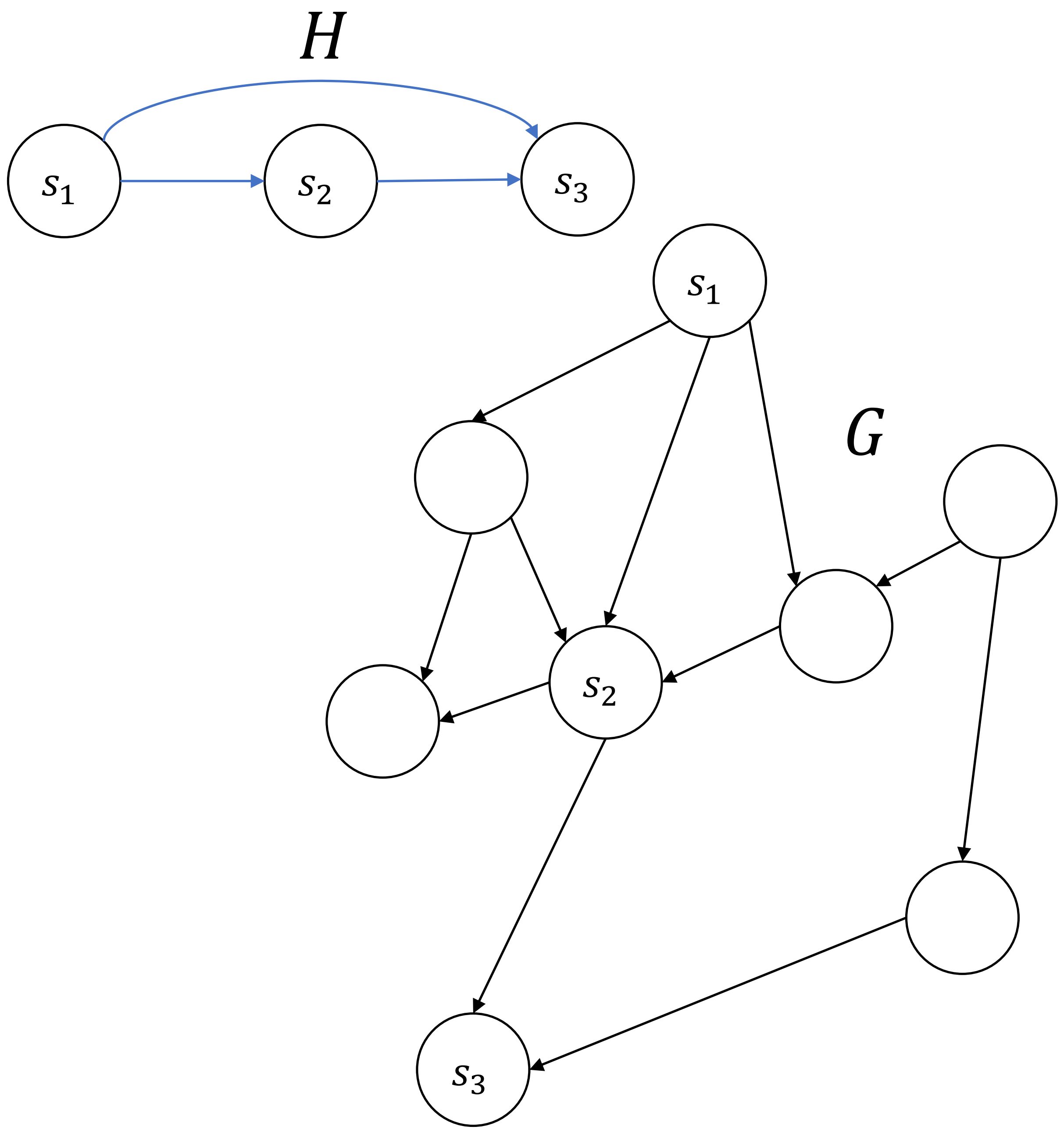
- Let $H = (V_H, E_H)$ be a fixed demand graph.
- Multicut(H)
  - Input: Supply graph $G = (V_G, E_G)$ and injective map $\pi: V_H \to V_G$.
  - Goal: Remove min # edges from G such that
    - $\forall (u, v) \in E_H$, there is no path from $\pi(u)$ to $\pi(v)$ in G.
- Multicut(1 edge) = Min s-t cut!





Only in [CM 17]

- Let $H = (V_H, E_H)$ be a fixed demand graph.
- Multicut(H)
  - Input: Supply graph $G = (V_G, E_G)$ and injective map $\pi: V_H \to V_G$.
  - Goal: Remove min # edges from G such that
    - $\forall (u, v) \in E_H$, there is no path from $\pi(u)$ to $\pi(v)$ in G.
- Multicut(complete DAG) = Linear k-cut ($k = |\mathrm{V_H}|$).



Multicut(H)

- Multicut(H):
  - Easy $|E_H|$-approximation.
  - Tight when H has k disjoint edges.

- Directed Multiway Cut (H = Complete Bidirected Graph)
  - [NZ97, CM16] 2-approx.

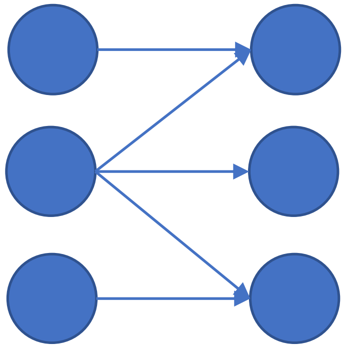
- k-Linear Cut (H = Complete DAG)
  - $O(\log k)$-approx. (Flow-cut gap open)
  - [BCKM 18?] 3-Linear Cut: $\sqrt{2}$-approx. (Matches flow-cut gap)

Multicut(H)

- Much better approximation ratio for some H!
  - All algorithms use flow-cut LP.

- Question] For some fixed H, will there a better relaxation?

Multicut(H)



- [CM 17] When H is a directed bipartite, Multicut(H) is UG-hard to approximate better than the worst flow-cut gap.
- What about general H?
- It is still open whether flow-cut gap is the best.
- [LM ??] There exists another LP relaxation (or estimation algorithm) such that it is UG-hard to do better.

Multicut(H)

- **[CM 17] When H is a directed bipartite, Multicut(H) is UG-hard to approximate better than the worst flow-cut gap.**
- What about general H?
- It is still open whether flow-cut gap is the best.
- **[LM ??] There exists another LP relaxation (or estimation algorithm) such that it is UG-hard to do better.**

Multicut(H)

- **[CM 17] When H is a directed bipartite, Multicut(H) is UG-hard to approximate better than the worst flow-cut gap.**
  - **Another proof based on [L 17]**
- What about general H?

- It is still open whether flow-cut gap is the best.

- [LM ??] There exists another LP relaxation (or estimation algorithm) such that it is UG-hard to do better.

Flow-Cut (Distance) LP

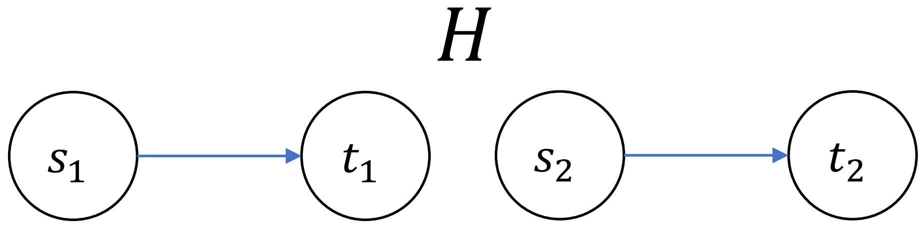
- Will consider vertex deletion version.
  - Cannot delete terminals ($T \coloneqq \pi(V_H)$).

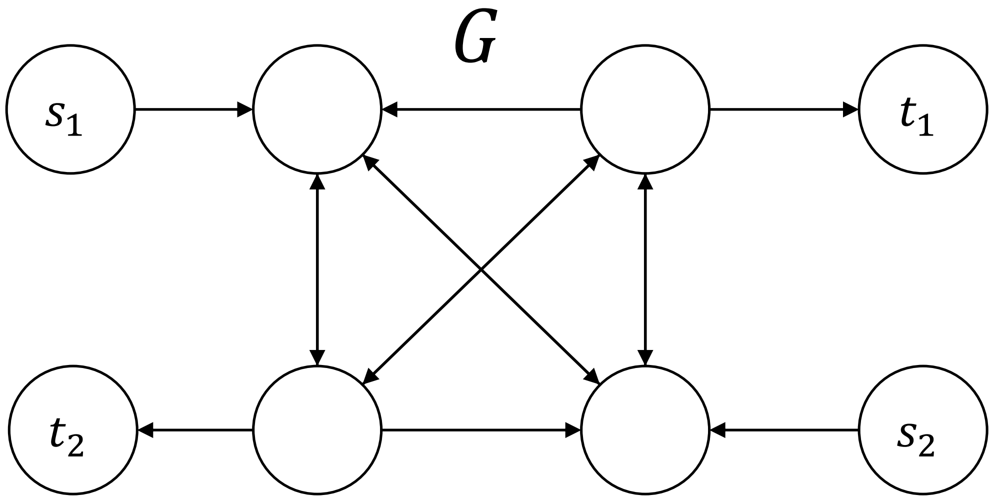
- Miniminze $\sum_{v \in V \setminus \mathrm{T}} x_v$
- Subject to $\sum_{v \in P \setminus \mathrm{T}} x_v \geq 1$ for $\forall (u, v) \in E_H$, and $\pi(u)$- $\pi(v)$ path P
- $x \geq 0$

LP Gap

- $OPT = 2$

- $LP = 4/3$
  - $x_v = 1/3$ for all $v \in V \setminus T$
  - Every s_1-t_1 or s_2- t_2 path involves 3 internal (non-terminal) vertices.

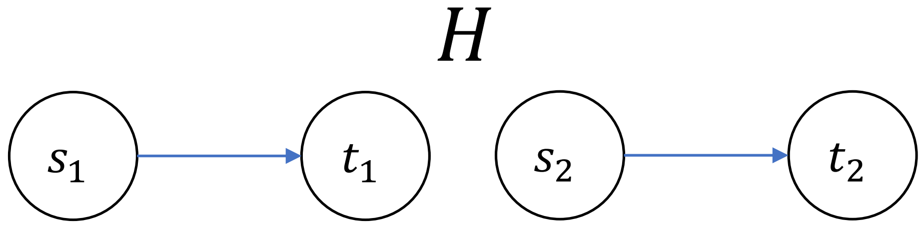


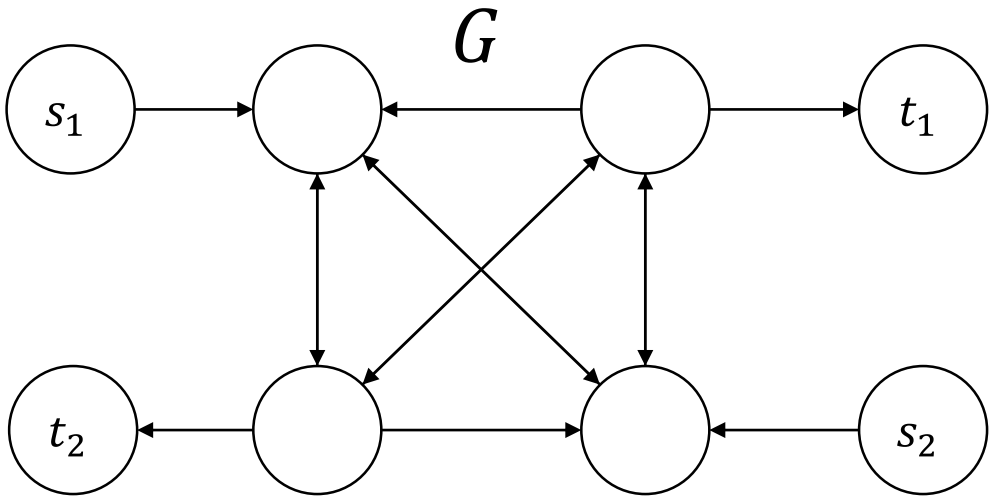


Dictator Test

- Just another instance of Multicut(H)

- Replace every internal vertex by a hypercube $[\ell]^R$

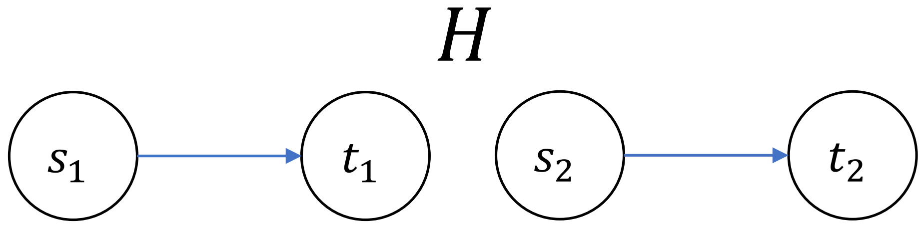


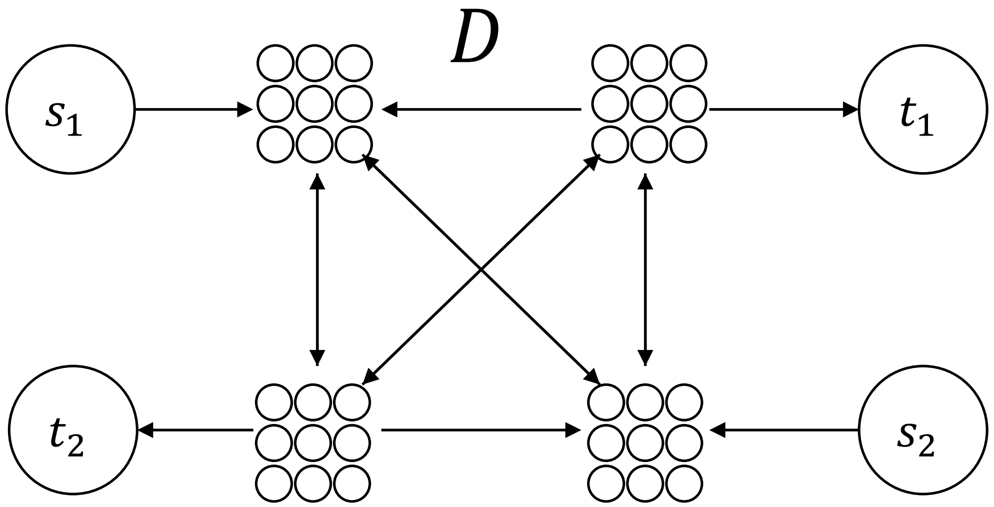


Dictator Test

- Just another instance of Multicut(H)

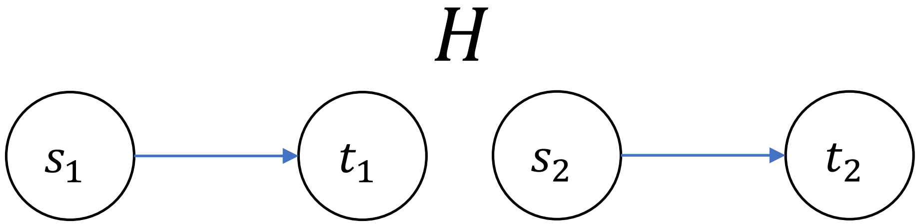
- Replace every internal vertex by a hypercube $[\ell]^R$

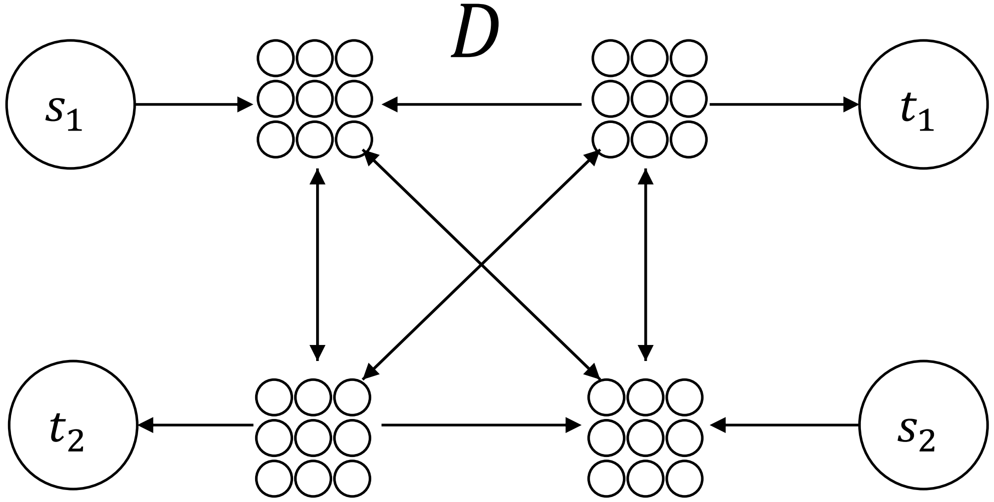




Dictator Test

- Just another instance of Multicut(H)
- Replace every internal vertex by a hypercube $[\ell]^R$
- Put edges
  - If $(u, v) \in E_G$, create some edges between corresponding hypercube "appropriately".

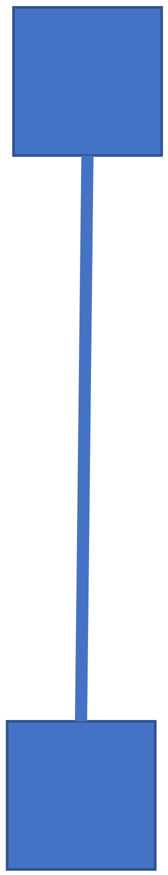




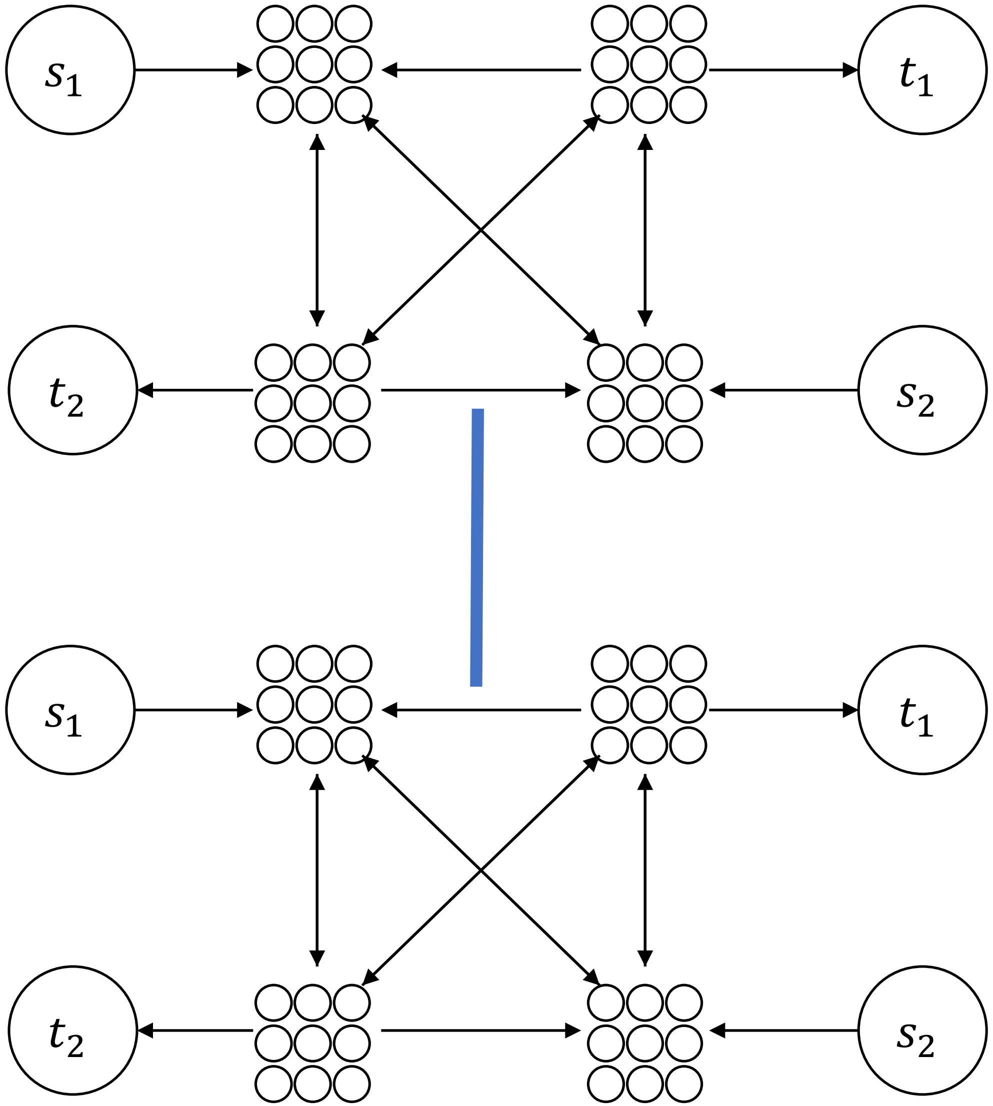
Reduction from UG

- Instance of Unique Games
  - A graph
  - Each edge is some constraint

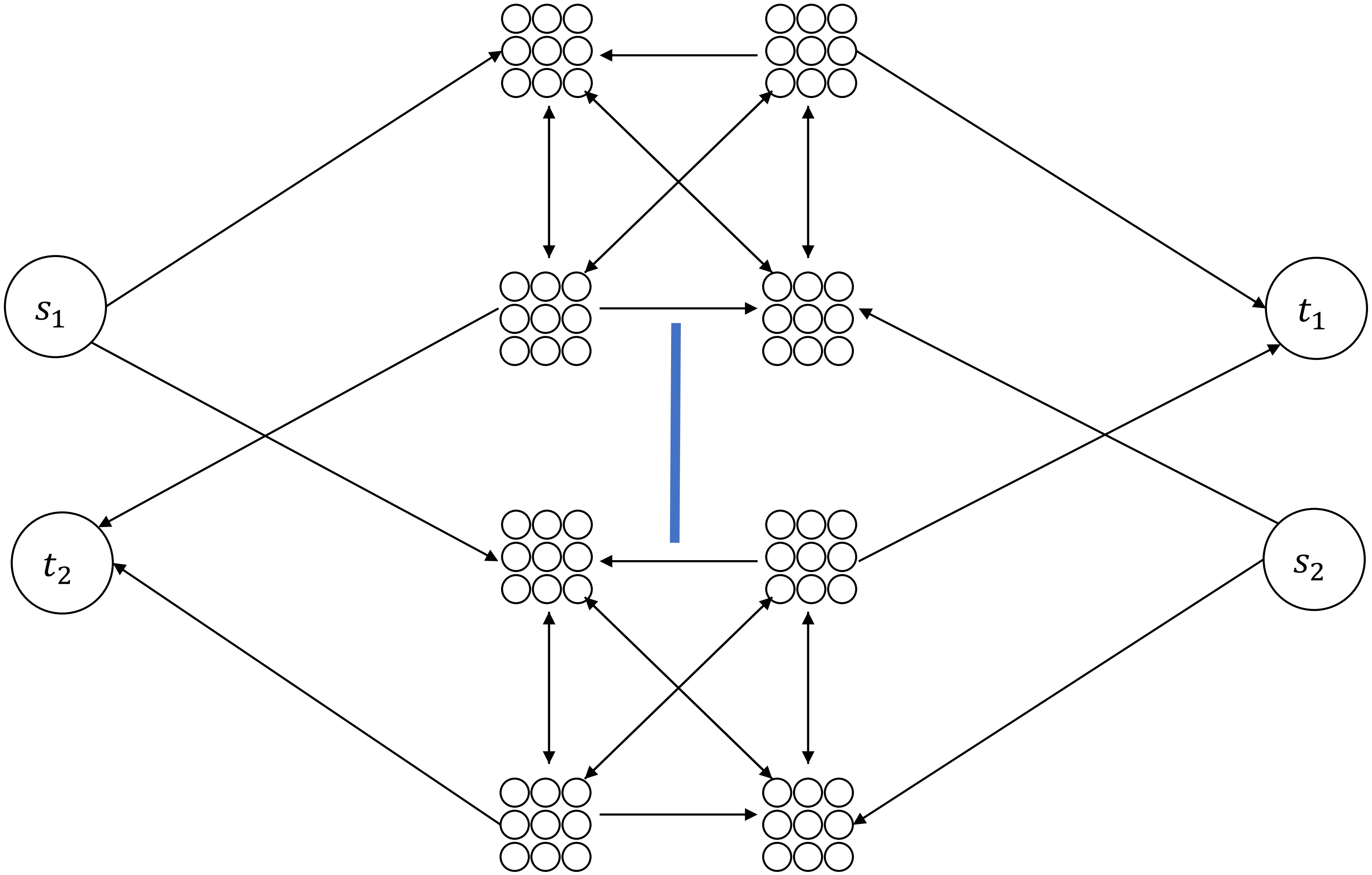
- Goal: Give a label to each vertex to
  - Maximize # of "satisfied" edges.



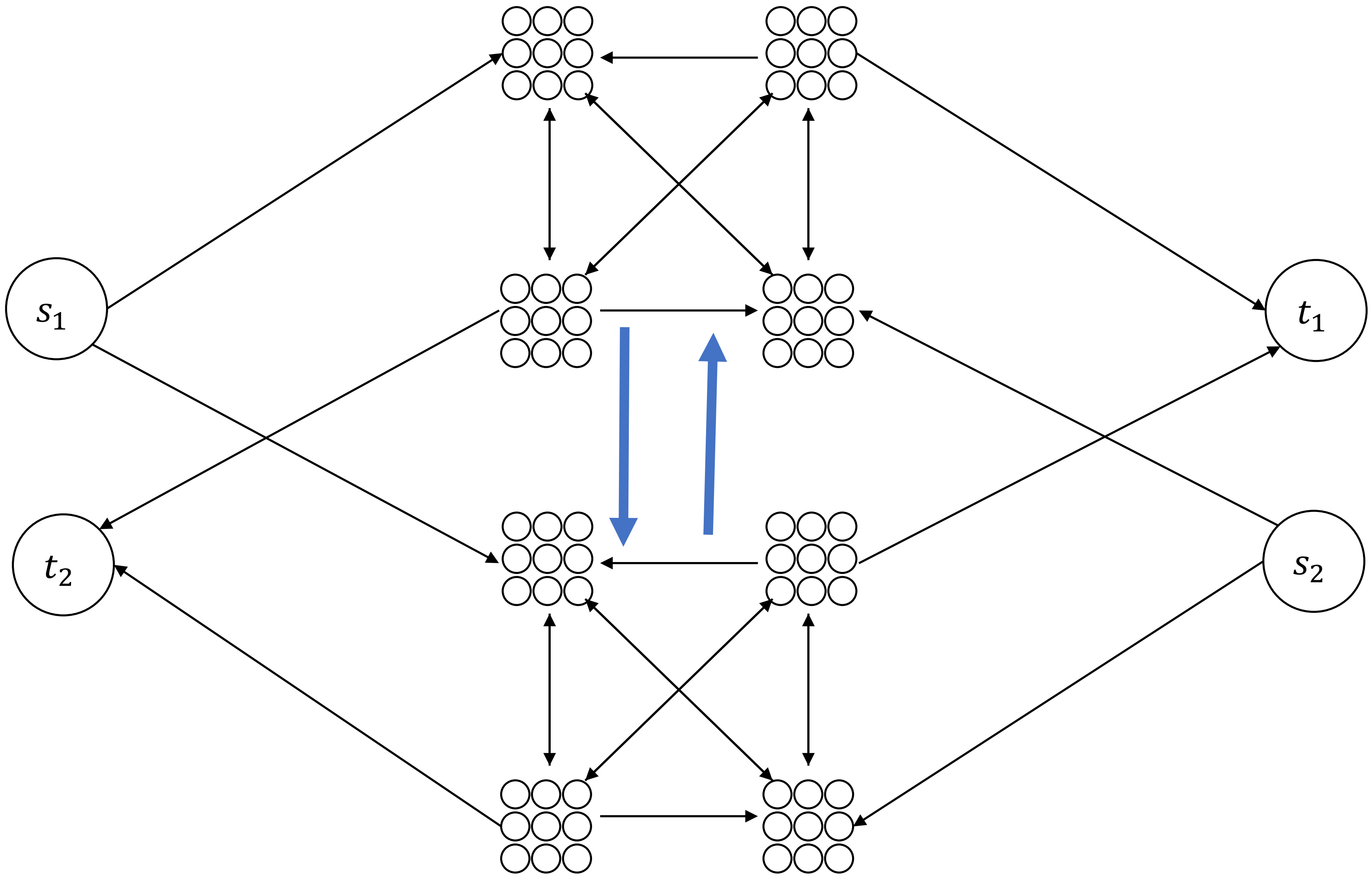
Reduction from UG



Reduction from UG

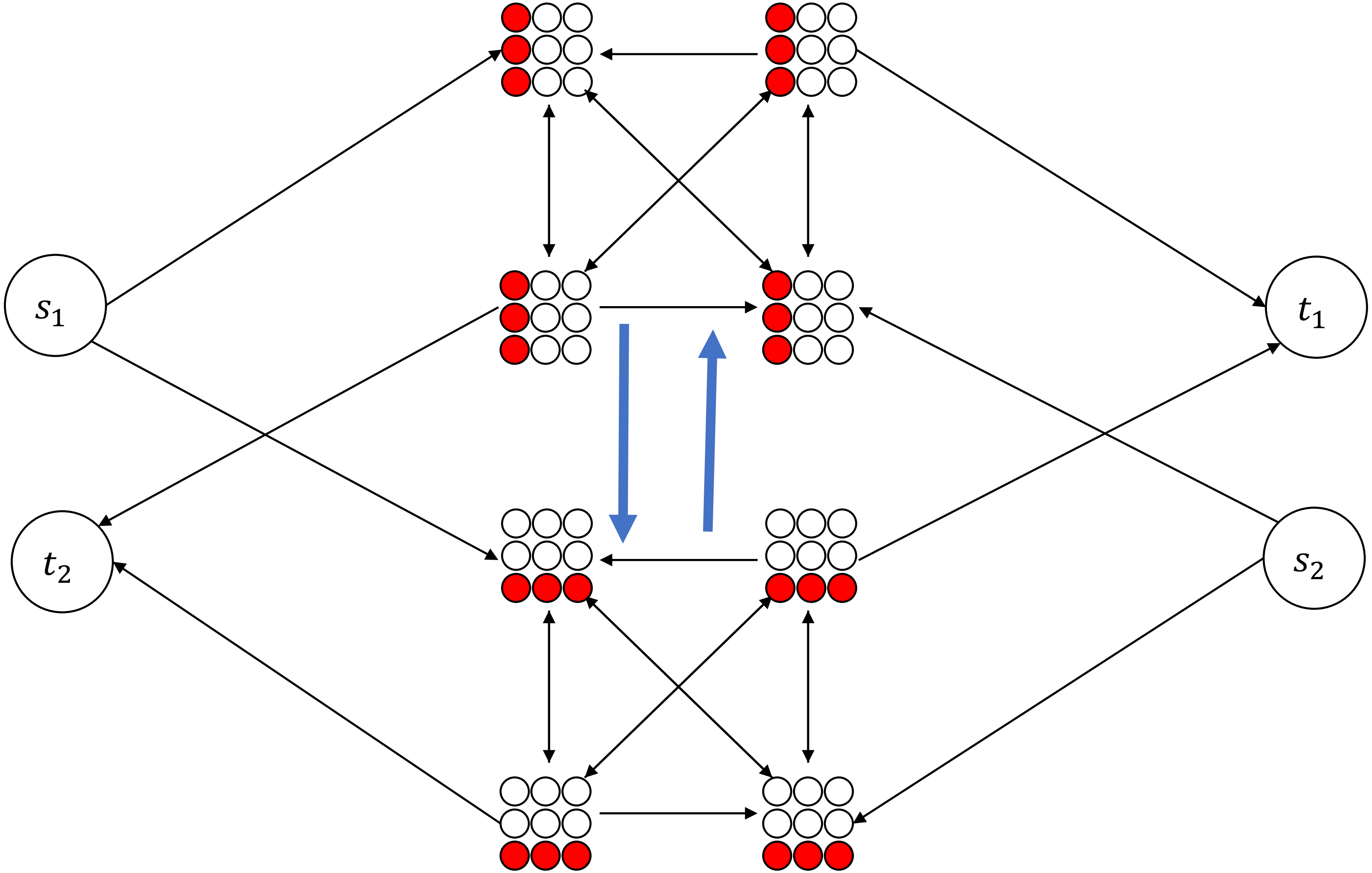


Reduction from UG



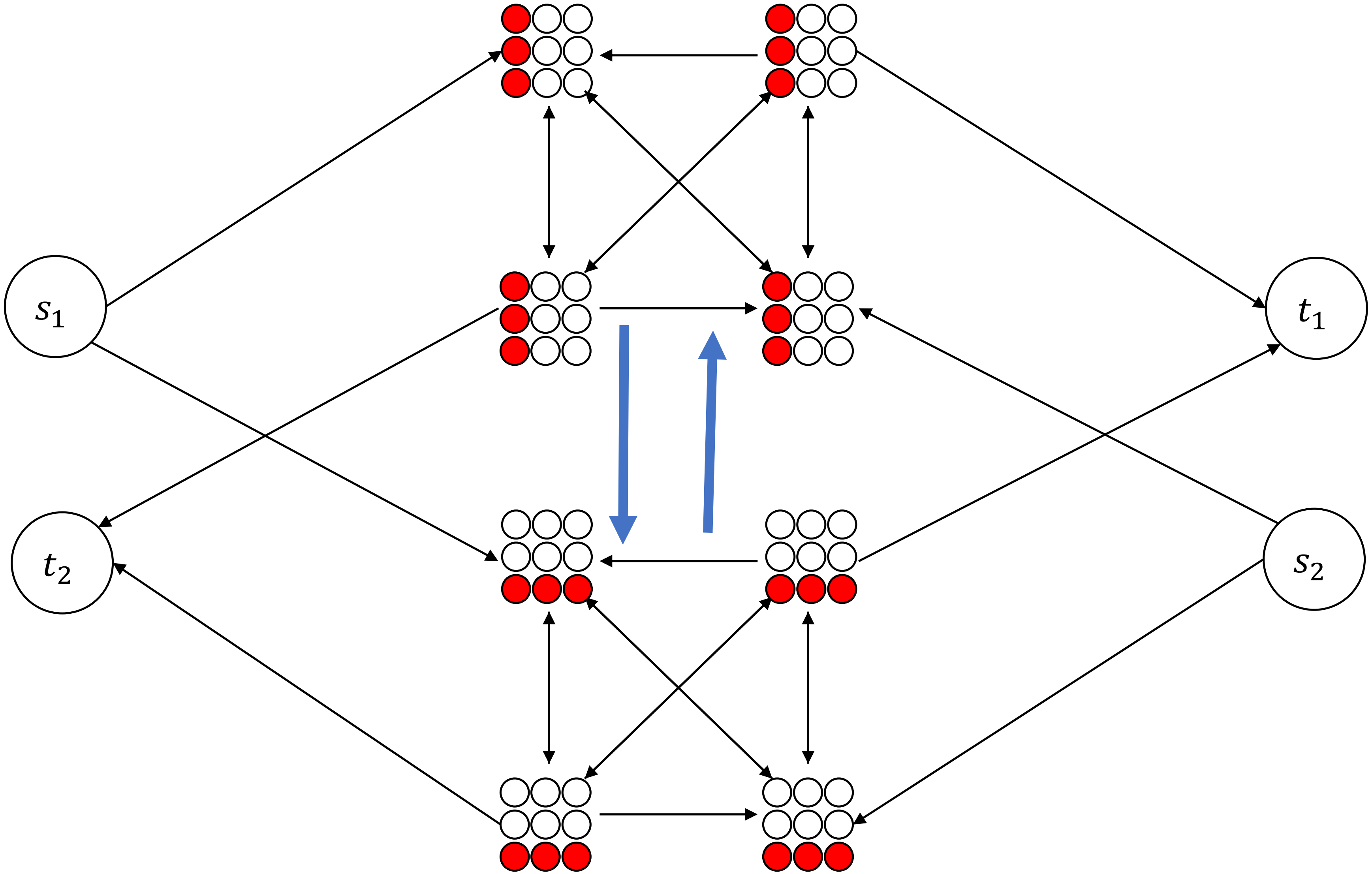
Reduction from UG

If UG instance **has a good labeling** that satisfies most constraints.



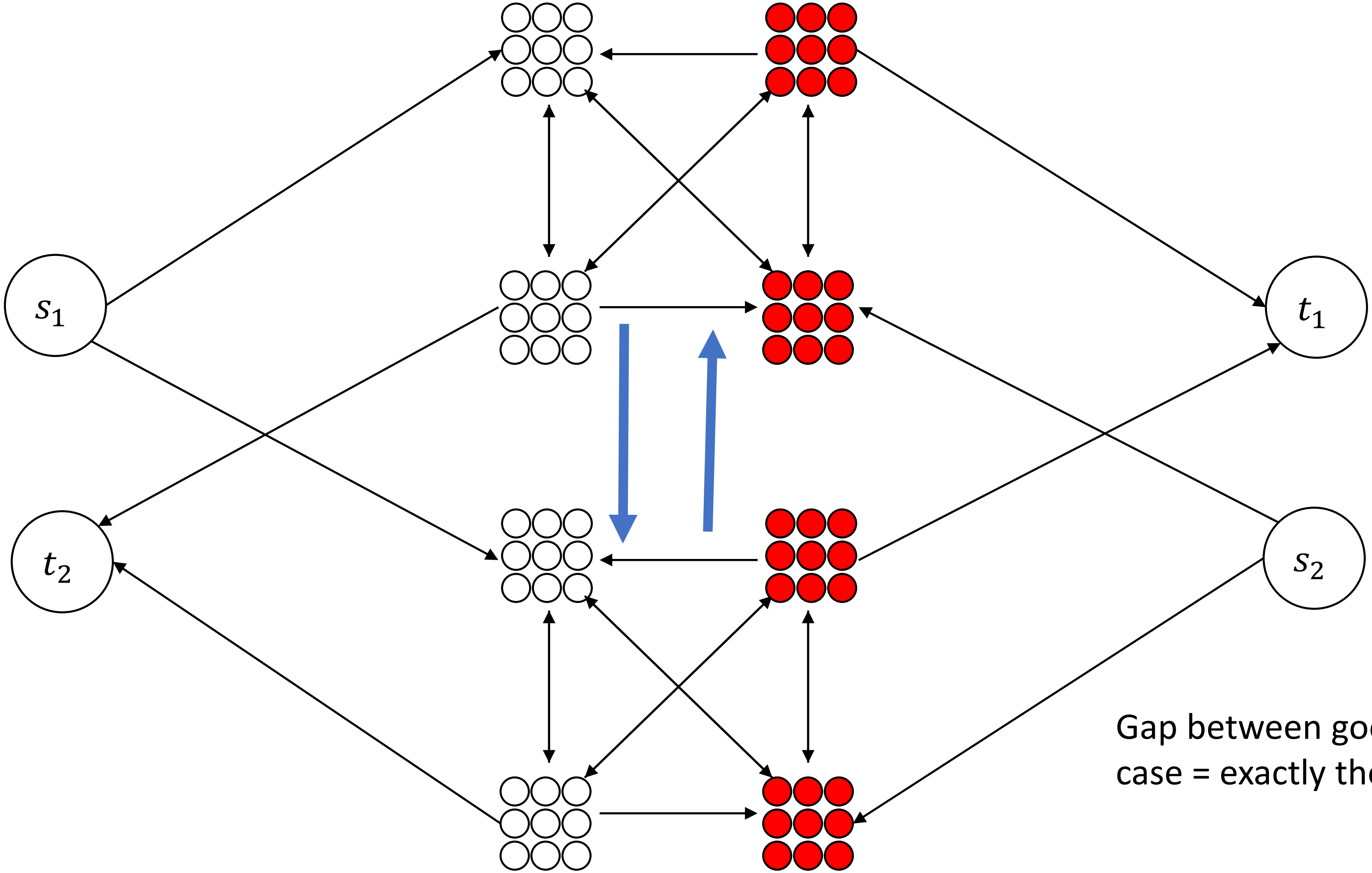
Reduction from UG

If UG instance **does not a good labeling**



Reduction from UG

If UG instance **does not a good labeling**



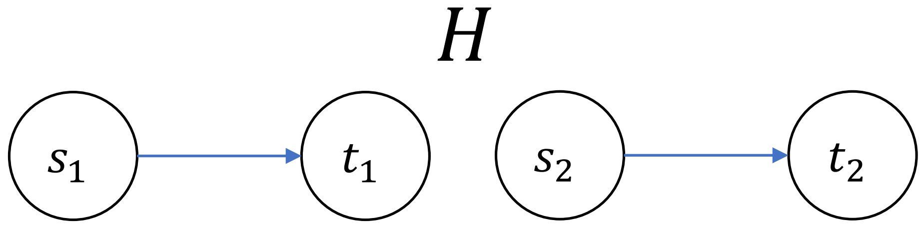
Gap between good case and bad case = exactly the LP gap.

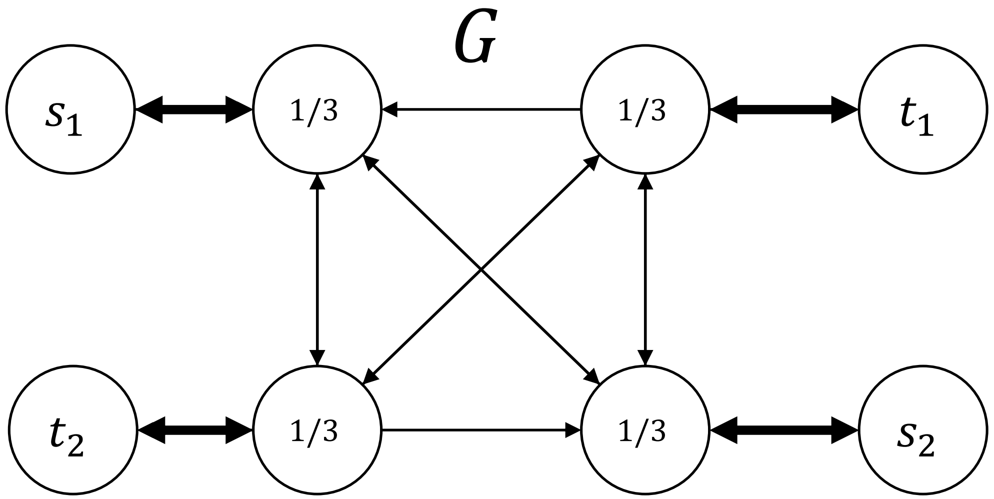
Multicut(H)

- **[CM 17] When H is a directed bipartite, Multicut(H) is UG-hard to approximate better than the worst flow-cut gap.**
- **What is wrong with general H?**

General H

- $x_v = 1/3$ for all $v \in V \setminus T$
  - Still feasible to LP.
  - Reduction does not work.
- Dist.(s_1-t_1) = 1, but
  - Dist.(s_1-s_2) = Dist.(s_2-t_1) = 2/3
- **Observation] In order to cut s_1 from t_1, we need to either**
  - **Cut s_1 from s_2 OR**
  - **Cut s_2 from t_1**






ALGO
RITHM
ALGORITHM

Best (estimation) Algorithm

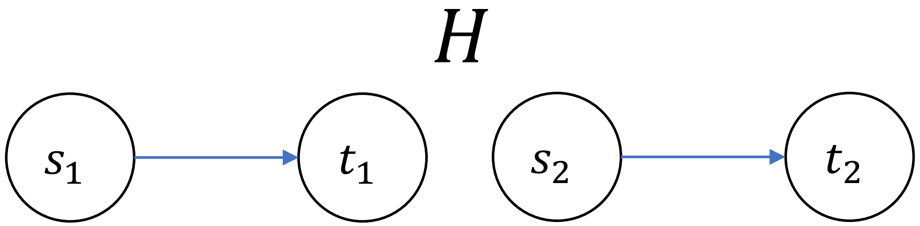
- Say F "unambiguous" if for every $(u, v) \in E_F$ and $w \in V_F$
  - Either $(u, w) \in E_F$ or $(w, v) \in E_F$
  - "If you cut (u, v), then you need to cut either (u, w) or (w, v)".
  - (Directed) complement of F is transitive.

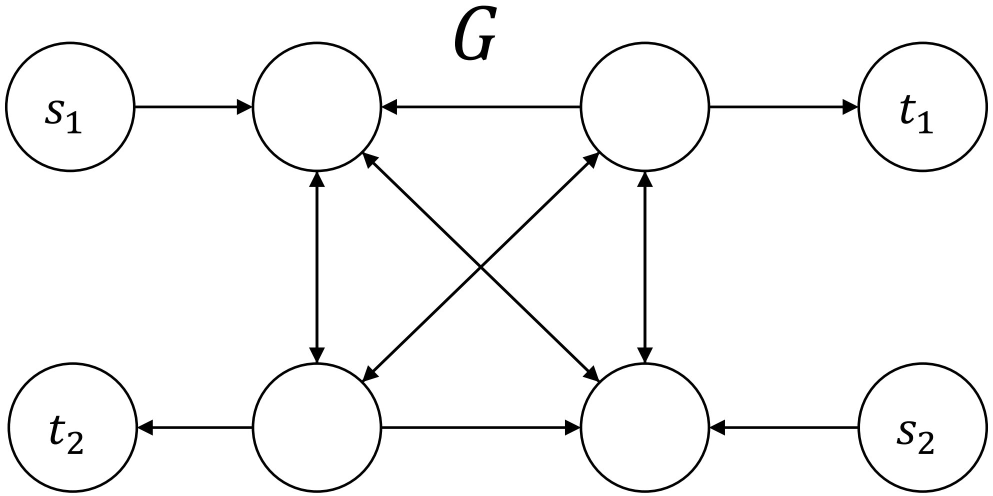
- Estimation algorithm for Directed Multicut(H).
  - Given a supply graph G,
  - Try every "unambiguous" $F = (V_H, E_F)$ s.t. $E_H \subseteq E_F$.
    - Compute Flow-cut relaxation value $LP(F, G)$.
  - Output the $\min_F LP(F, G)$.

Best (estimation) Algorithm

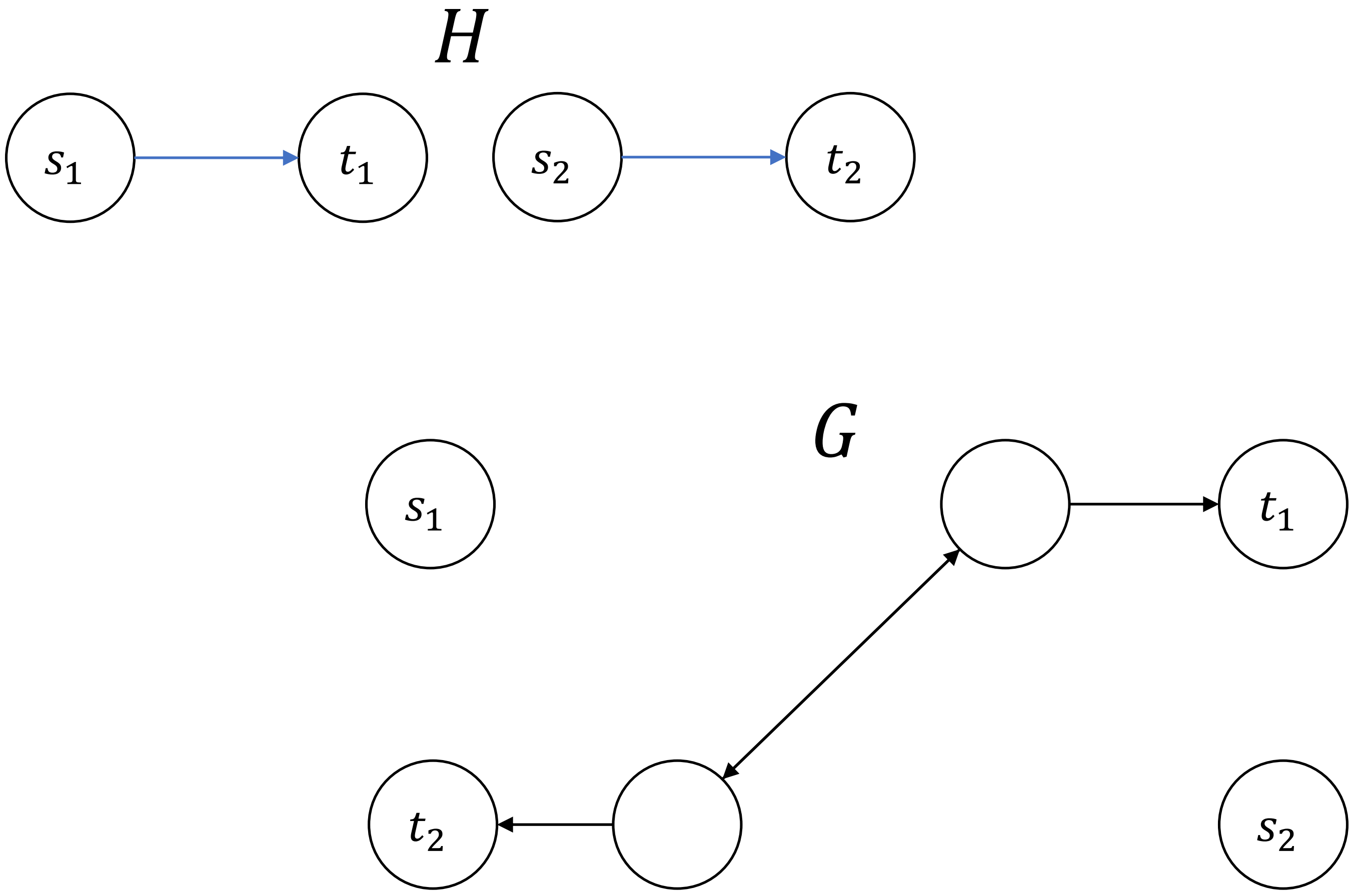
- Estimation algorithm for Directed Multicut(H).
  - Given a supply graph G,
  - Try every "unambiguous" $F = (V_H, E_F)$ s.t. $E_H \subseteq E_F$.
    - Compute Flow-cut relaxation value $LP(F, G)$.
  - Output the $\min_F LP(F, G)$.
- For every F, $LP(H, G) \leq LP(F, G)$.
- There exists F such that $LP(F, G) \leq OPT(F, G) = OPT(H, G)$

$$\exists F \text{ s.t. } LP(F,G) \leq OPT(F,G) = OPT(H,G)$$

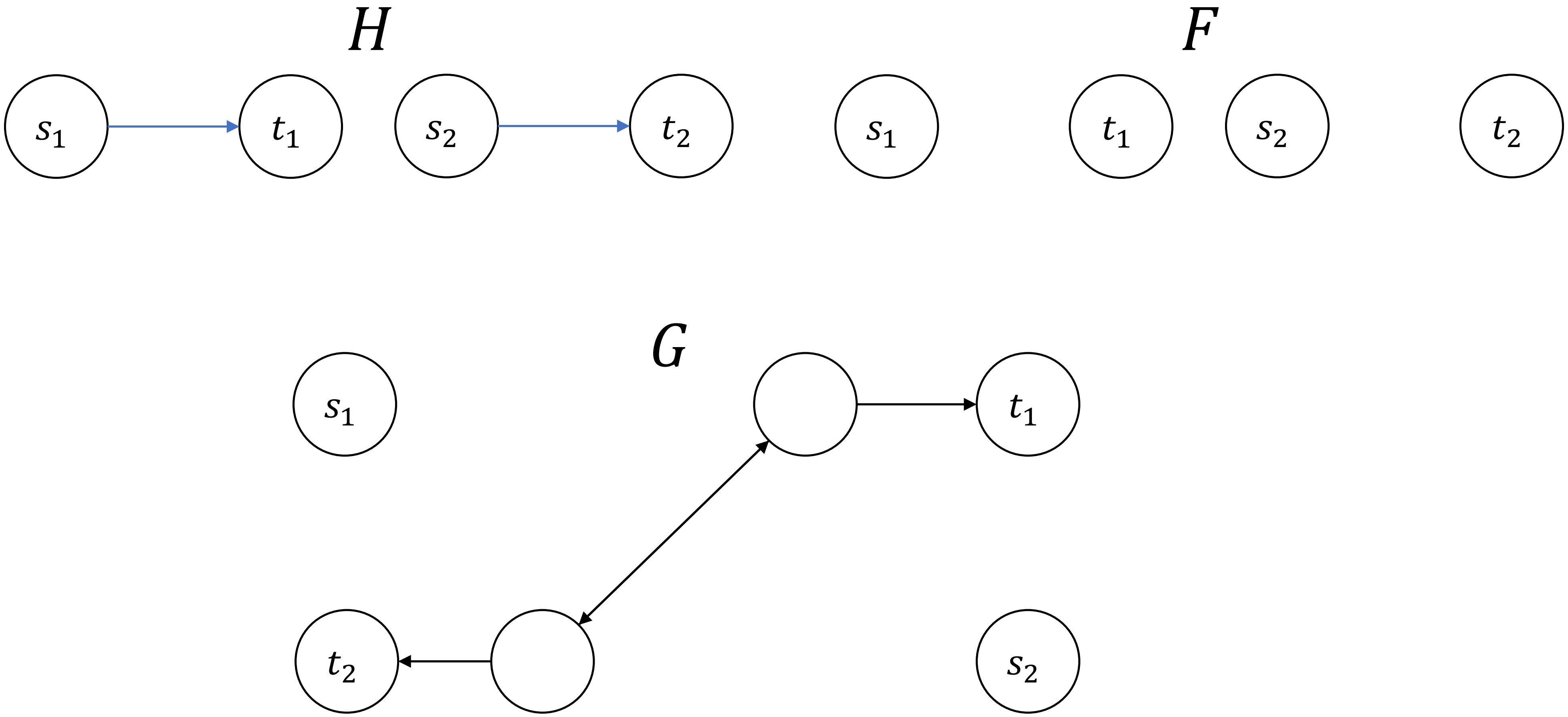




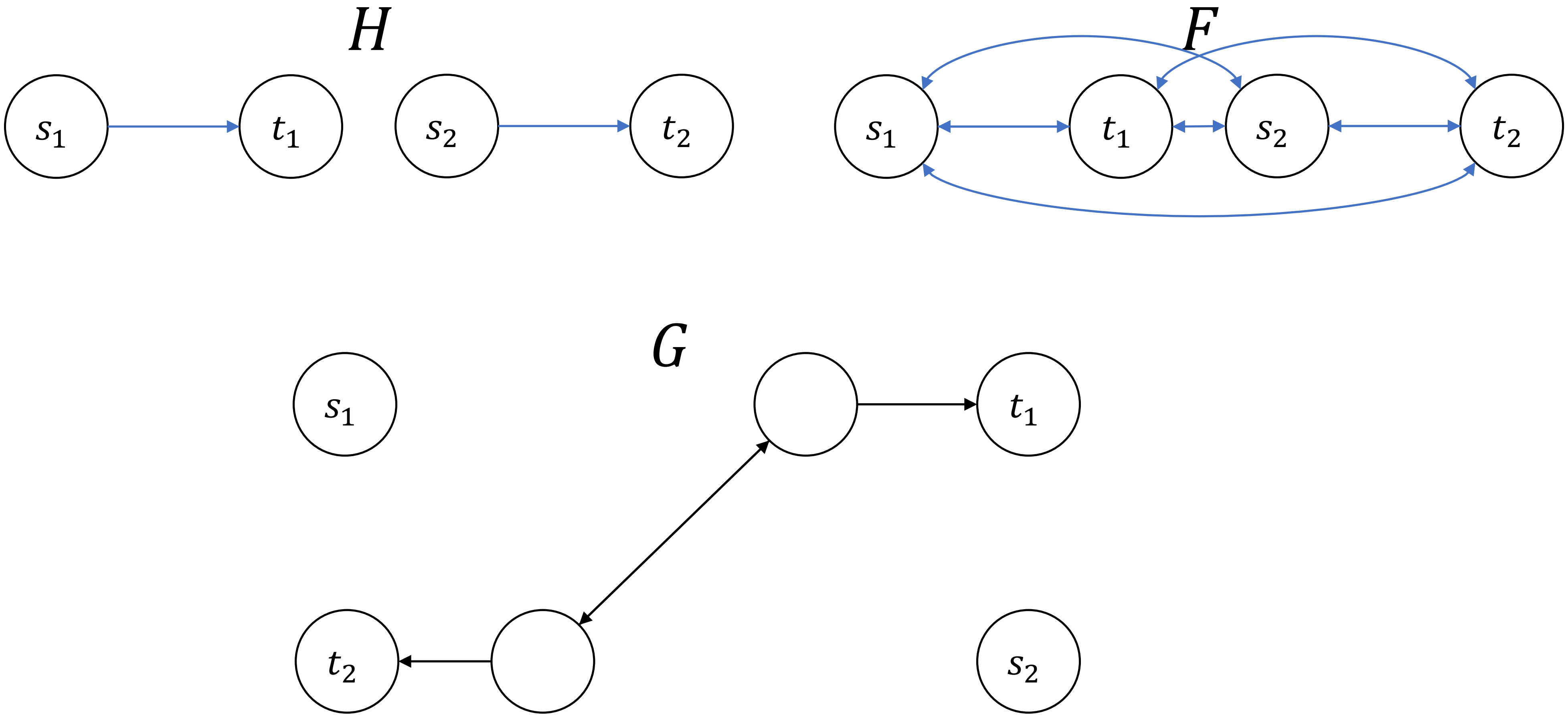
$$\exists F \text{ s.t. } LP(F,G) \le OPT(F,G) = OPT(H,G)$$



$$\exists F \text{ s.t. } LP(F,G) \leq OPT(F,G) = OPT(H,G)$$



$$\exists F \text{ s.t. } LP(F,G) \leq OPT(F,G) = OPT(H,G)$$



Best (estimation) Algorithm

- Estimation algorithm for Directed Multicut(H).
  - Given a supply graph G,
  - Try every "unambiguous" $F = (V_H, E_F)$ s.t. $E_H \subseteq E_F$.
    - Compute Flow-cut relaxation value $LP(F, G)$.
  - Output the $\min_F LP(F, G)$.
- For every F, $LP(H, G) \leq LP(F, G)$.
- There exists F such that $LP(F, G) \leq OPT(F, G) = OPT(H, G)$
- Therefore, $LP(H, G) \leq ALG(H, G) \leq OPT(H, G)$ for every G, H.
- Can be captured as a single LP (running a flow-cut LP for every F).

Best (estimation) Algorithm

- Estimation algorithm for Directed Multicut(H).
  - Given a supply graph G,
  - Try every "unambiguous" $F = (V_H, E_F)$ s.t. $E_H \subseteq E_F$.
    - Compute Flow-cut relaxation value $LP(F, G)$.
  - Output the $\min_F LP(F, G)$.
- What does a gap of this algorithm mean (for fixed H)?
  - An unambiguous $F \supseteq H$ and G s.t. $LP(F, G) \ll OPT(H, G)$.
- **[LM ??] For fixed H, a gap of this algorithm implies the matching UG-hardness.**

Undirected Analog

- Running time $2^{O(k^2)} n^{O(1)}$ when $k = |V_H|$.
- Undirected Multicut(H).
  - "Unambiguous" F: complete p-partite graph (complement = disjoint cliques).
  - Guess which terminals belong together, and run Multiway Cut
    - Already gives 1.3-approx. [SV13, BSW16] for every H in time $2^{O(k \log k)} n^{O(1)}$.
- Gap instance]
  - Unambiguous $F \supseteq H$ and G s.t. $EarthmoverLP(F, G) \ll OPT(H, G)$.
  - EarthmoverLP is already proved to be optimal for Multiway Cut [MNRS 08]
  - Their proof already proves that the above is best estimation algorithm for Undirected Multicut(H)?

Undirected Analog

- Running time $2^{O(k^2)}n^{O(1)}$ when $k = |V_H|$.
- Undirected Multicut(H).
  - "Unambiguous" F: complete p-partite graph (complement = disjoint cliques).
  - Guess which terminals belong together, and run Multiway Cut
    - Already gives 1.3-approx. [SV13, BSW16] for every H in time $2^{O(k \log k)}n^{O(1)}$.
- Gap instance]
  - Unambiguous $F \supseteq H$ and G s.t. $EarthmoverLP(F, G) \ll OPT(H, G)$.
  - EarthmoverLP is already proved to be optimal for Multiway Cut [MNRS 08]
  - Their proof already proves that the above is best estimation algorithm for Undirected Multicut(H)?

Global Cut Problems

- [BCKLX 17] Global versions
  - s-t Bicut: Given G **and** $\boldsymbol{s}, \boldsymbol{t}$, remove min # arcs s.t. $s \nrightarrow t$ and $t \nrightarrow s$.
  - Global Bicut: Given G, remove min # arcs s.t. $\exists \boldsymbol{s}, \boldsymbol{t}$ with $s \nrightarrow t$ and $t \nrightarrow s$.
- Undirected Analog
  - 3-way cut: Given G and $\boldsymbol{s}, \boldsymbol{t}, \boldsymbol{u}$, remove min # edges s.t. they are separated.
  - 3-cut : Given G, remove min # edges s.t. $\exists \boldsymbol{s}, \boldsymbol{t}, \boldsymbol{u}$ separated.
- 3-way cut: NP-hard. 3-cut: P
- s-t Bicut: 2-hard [CM 17, L 17]. Global Bicut: 1.998-approximation.

Hardness Framework

- [L 17] First $\omega(1)$-hardness for
  - Length-Bounded Cut
  - Shortest Path Interdiction
  - Firefighter (RMFC)
- Length-Control Dictatorship Test
  - Take (some) LP gap instances to UG-hardness.
- More cut problems?
  - General theorem that unifies current results?
  - How to formally unify various cut problems?

Open Problems

- Flow-Cut LP may be still optimal (save $2^{O(k^2)}$ time)!
  - $\exists G, H$ s.t. $LP(H, G) < ALG(H, G)$
  - But maybe $\max_G \frac{LP(H,G)}{OPT(H,G)} = \max_G \frac{ALG(H,G)}{OPT(H,G)}$??
- "Interesting H" where we can do much better than $|E_H|$-approx.?
  - Multiway Cut, Linear-k-Cut, ???
  - Using the new LP?
- Optimal rounding algorithms?
  - Undirected Multiway Cut [MNRS 08], Min CSP [EVW 13]

Thank you!