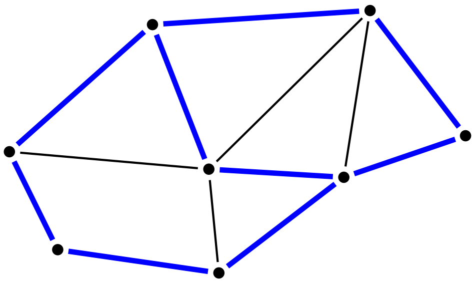
Better-Than-2 Approximations for Forest Augmentation and Weighted Tree Augmentation

Vera Traub

ETH Zürich

joint work with Fabrizio Grandoni, Afrouz Jabal Ameli, and Rico Zenklusen

Minimum weight 2-edge-connected spanning subgraphs

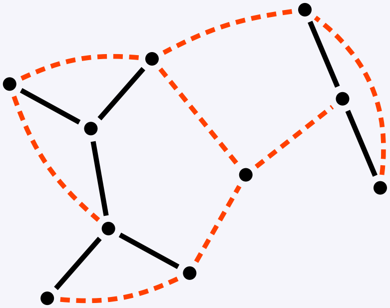


Min-Weight 2-ECSS

Instance: graph $G = (V, E)$, weights $w : E \to \mathbb{R}_{\geq 0}$

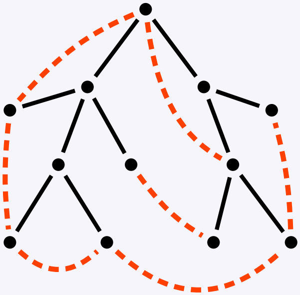
Task: Find a min-weight 2-edge-connected spanning subgraph.

Forest Augmentation

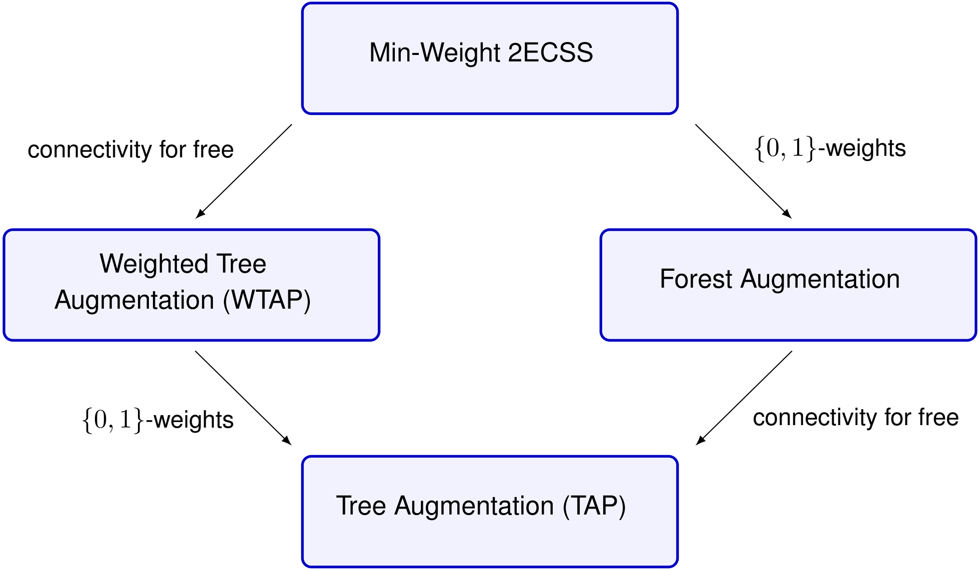


Links have weight/cost 1.
Forest edges have cost 0.

Weighted Tree Augmentation



Links have positive weight/cost.
Tree edges have cost 0.


Min-Weight 2ECSS
connectivity for free
$\{0,1\}$-weights
Weighted Tree Augmentation (WTAP)
Forest Augmentation
$\{0,1\}$-weights
connectivity for free
Tree Augmentation (TAP)

Hardness and classical approximation algorithms

- Kortsarz, Krauthgamer, and Lee [2004]:

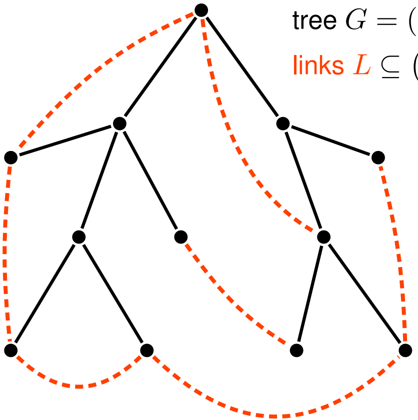
  TAP is APX-hard even on trees of diameter 5.

- Classical techniques, e.g. primal-dual algorithms, iterative rounding, provide a 2-approximation for a wide class of network design problems, including Min-Weight 2ECSS.

Can we get better-than-2 approximations?

Weighted Tree Augmentation

Weighted Tree Augmentation (WTAP)



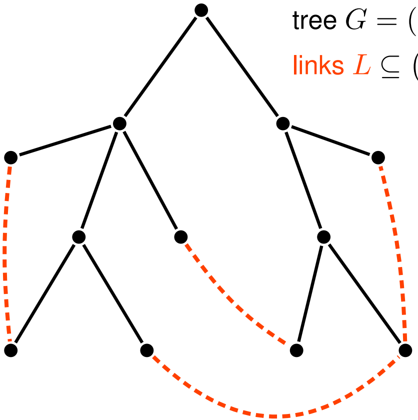
tree $G = (V, E)$

links $L \subseteq \binom{V}{2}$ with weights $w : L \to \mathbb{R}_{>0}$

WTAP

Find a min weight set $F \subseteq L$ of links s.t. G becomes 2-edge-connected when adding F.

Weighted Tree Augmentation (WTAP)



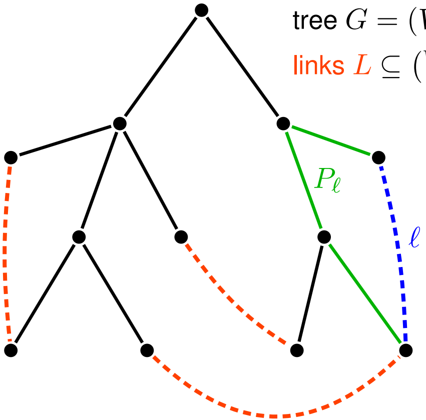
tree $G = (V, E)$

links $L \subseteq \binom{V}{2}$ with weights $w : L \to \mathbb{R}_{>0}$

WTAP

Find a min weight set $F \subseteq L$ of links s.t. G becomes 2-edge-connected when adding F.

Weighted Tree Augmentation (WTAP)



tree $G = (V, E)$

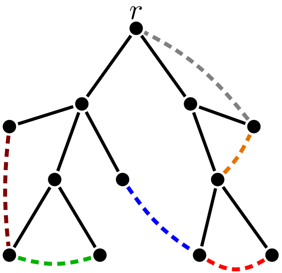
links $L \subseteq \binom{V}{2}$ with weights $w : L \to \mathbb{R}_{>0}$

WTAP

Find a min weight set $F \subseteq L$ of links s.t.
G becomes 2-edge-connected when adding F.

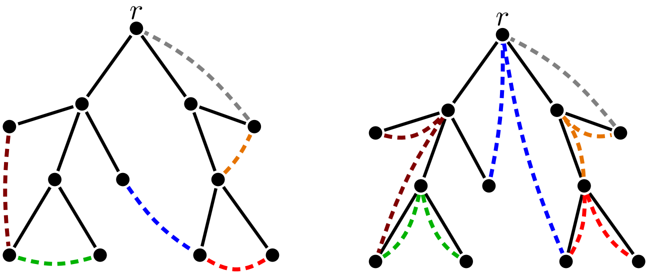
Equivalent:
Every edge $e \in E$ must be covered by a link $\ell \in F$,
i.e., $e \in P_\ell$ for some $\ell \in F$.

2-approximation algorithm



① Pick an arbitrary root $r \in V$.

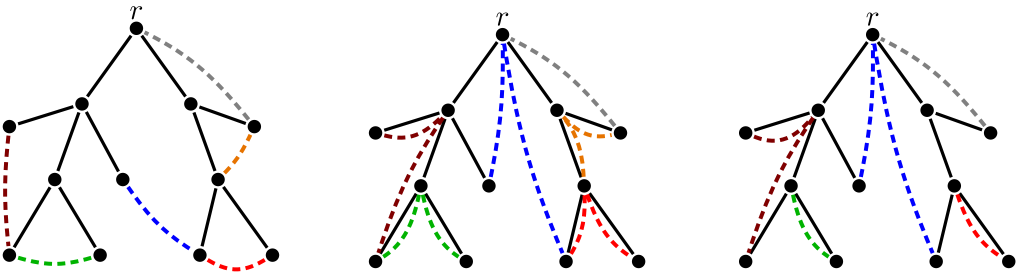
2-approximation algorithm



① Pick an arbitrary root $r \in V$.

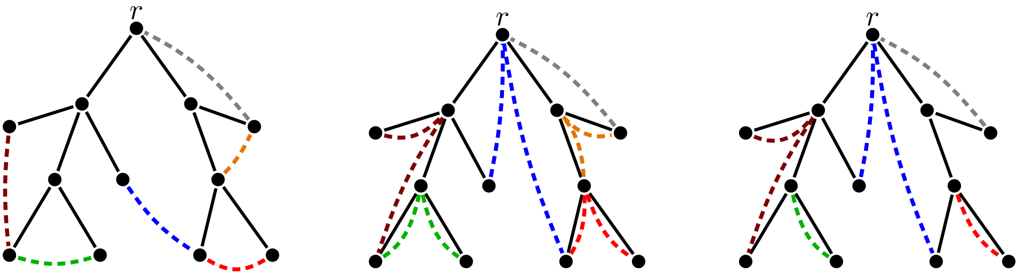
② “Split” every link ℓ into two up-links, each with weight $w(\ell)$.

2-approximation algorithm



1. Pick an arbitrary root $r \in V$.
2. "Split" every link ℓ into two up-links,
   each with weight $w(\ell)$.
3. Compute an optimal up-link solution.

2-approximation algorithm



1. Pick an arbitrary root $r \in V$.
2. "Split" every link ℓ into two up-links, each with weight $w(\ell)$.
3. Compute an optimal up-link solution.

solve natural LP (integral), or use dynamic programming

Better-than-2 approximations for special cases

- **unweighted tree augmentation (TAP):** 1.393-approximation [Cecchetto, T., Zenklusen, 2021]

  (improving on [Nagamochi, 2003], [Even, Feldmann, Kortsarz, Nutov, 2009], [Cheriyan, Gao, 2018], [Kortaz, Nutov, 2016], [Kortaz, Nutov, 2018], [Adjashvilli, 2018], [Nutov, 2017], [Fiorini, Groß, Könemann, Sanità, 2018], [Grandoni, Kalaitzis, Zenklusen, 2018])

- **constant-diameter trees:** $(1 + \ln 2)$-approximation [Cohen, Nutov, 2013]

- better-than-2 approximation if an opt. solution to natural LP has no small fractional values [Iglesias, Ravi, 2018]

Our result

Theorem [T., Zenklusen, 2021]

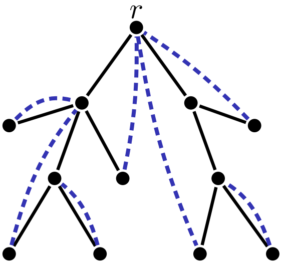
There is a $(1.5 + \varepsilon)$-approximation algorithm for Weighted Tree Augmentation (WTAP) for any fixed $\varepsilon > 0$.

Relative greedy algorithm: $(1 + \ln 2 + \varepsilon)$-approximation

1. Compute a structured 2-approximate solution.
2. Show that every structured suboptimal solution can be improved.

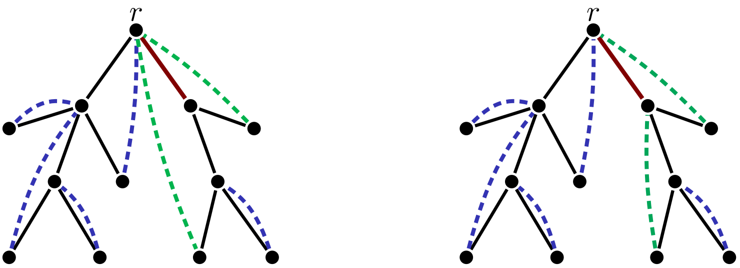
The Relative Greedy Algorithm

The starting solution for relative greedy



① Compute optimal up-link solution U (2-approximation).

The starting solution for relative greedy

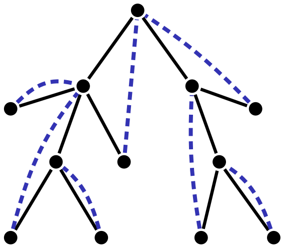


1. Compute optimal up-link solution U (2-approximation).
2. “Shorten” up-links s.t. P_u with $u \in U$ are disjoint, i.e., every edge is covered by exactly one link.

Relative greedy: improving structured solutions

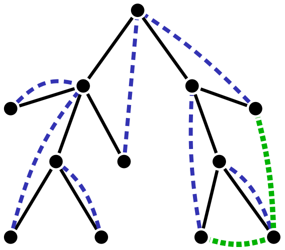
Invariant: $U \cup F$ is a WTAP solution.

1. U := 2-approximate up-link solution s.t. the paths P_u with $u \in U$ are disjoint.
   $F := \emptyset$



Relative greedy: improving structured solutions

Invariant: $U \cup F$ is a WTAP solution.



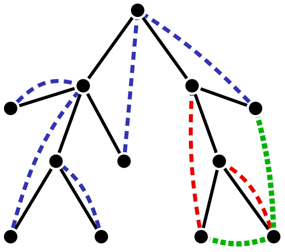
① U := 2-approximate up-link solution s.t. the paths P_u with $u \in U$ are disjoint.
$F := \emptyset$

② As long as $w(U \cup F)$ decreases:

- Select a component $C \subseteq L$.
- Add C to F.

Relative greedy: improving structured solutions

Invariant: $U \cup F$ is a WTAP solution.



① U := 2-approximate up-link solution s.t. the paths P_u with $u \in U$ are disjoint.
$F := \emptyset$

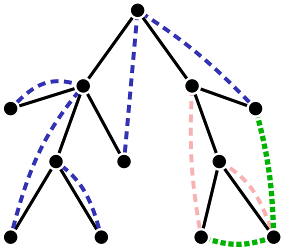
② As long as $w(U \cup F)$ decreases:

- Select a component $C \subseteq L$.
- Add C to F.
- Remove the following from U:

$$\mathrm{Drop}_U(C) := \left\{ u \in U : P_u \subseteq \bigcup_{\ell \in C} P_\ell \right\}$$

Relative greedy: improving structured solutions

Invariant: $U \cup F$ is a WTAP solution.



① U $:=$ 2-approximate up-link solution s.t. the paths P_u with $u \in U$ are disjoint.
$F := \emptyset$

② As long as $w(U \cup F)$ decreases:

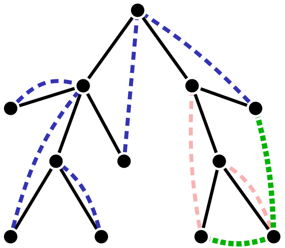
- Select a component $C \subseteq L$.
- Add C to F.
- Remove the following from U:

$$\text{Drop}_U(C) := \left\{ u \in U : P_u \subseteq \bigcup_{\ell \in C} P_\ell \right\}$$

③ Return $U \cup F$.

Relative greedy: improving structured solutions

Invariant: $U \cup F$ is a WTAP solution.



① U := 2-approximate up-link solution s.t. the paths P_u with $u \in U$ are disjoint.
$F := \emptyset$

② As long as $w(U \cup F)$ decreases:

- Select a component $C \subseteq L$.
- Add C to F.
- Remove the following from U:

$$\mathrm{Drop}_U(C) := \left\{ u \in U : P_u \subseteq \bigcup_{\ell \in C} P_\ell \right\}$$

③ Return $U \cup F$.

Choose C as a $\left\lceil \frac{1}{\varepsilon} \right\rceil$-thin link set minimizing

$$\frac{w(C)}{w(\mathrm{Drop}_U(C))}$$

The Existence of Improving Components

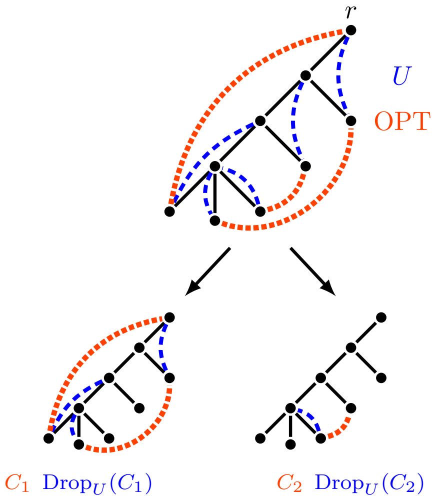
U := set of up-links s.t. the paths P_u with $u \in U$ are disjoint

Fix $\varepsilon > 0$.

Decomposition Theorem [T., Zenklusen, 2021]

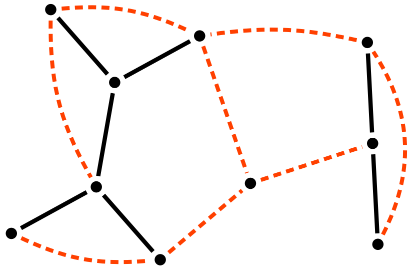
There exists a partition $\mathcal{C}$ of OPT into $\lceil 1/\varepsilon \rceil$-thin components s.t.:

$$\sum_{C \in \mathcal{C}} w(\mathrm{Drop}_U(C)) \geq (1-\varepsilon) \cdot w(U).$$



Forest Augmentation

Forest Augmentation Problem (FAP)

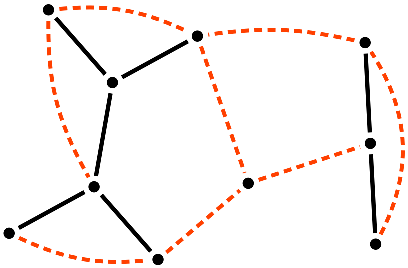


forest $G = (V, F)$

links $L \subseteq \binom{V}{2}$

Task: Find min-cardinality $S \subseteq L$ s.t. $G + S$ is 2-edge-connected.

Forest Augmentation Problem (FAP)



forest $G = (V, F)$

links $L \subseteq \binom{V}{2}$

Task: Find min-cardinality $S \subseteq L$ s.t. $G + S$ is 2-edge-connected.

Theorem [Grandoni, Jabal Ameli, T., 2022]

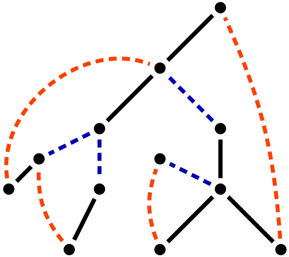
There is a 1.998-approximation algorithm for Forest Augmentation.

Prior work

- **Tree Augmentation Problem (TAP):** 1.393-approximation [Cecchetto, T., Zenklusen, 2021]
  (improving on [Nagamochi, 2003], [Even, Feldmann, Kortsarz, Nutov, 2009], [Cheriyan, Gao, 2018], [Kortaz, Nutov, 2016], [Kortaz, Nutov, 2018], [Adjashvilli, 2018], [Nutov, 2017], [Fiorini, Groß, Könemann, Sanità, 2018], [Grandoni, Kalaitzis, Zenklusen, 2018])
- **Unweighted 2-ECSS:** $\frac{4}{3}$-approximation
  [Sebő, Vygen, 2014], [Hunkenschröder, Vempala, Vetta, 2019]
  (improving on [Khuller, Vishkin, 1994], [Cheriyan, Sebő, Szigeti, 2001])
- **Matching Augmentation Problem (MAP):** $\frac{5}{3}$-approximation
  [Cheriyan, Cummings, Dippel, Zhu, 2020]
  (improving on [Cheriyan, Dippel, Grandoni, Khan, Narayan, 2020], [Bamas, Drygala, Svensson, 2022])

A naive approach

1. Complete the forest F to a spanning tree.
2. Augment the spanning tree to a 2-edge-connected graph using a ρ-approximation for TAP.

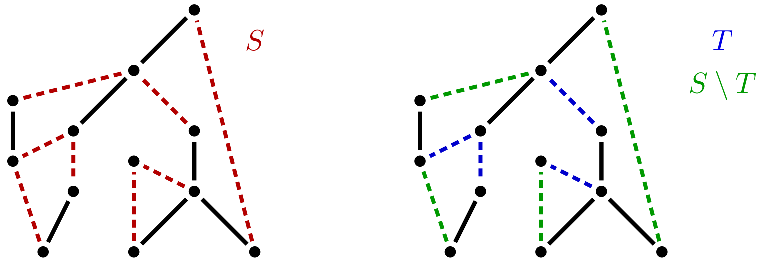


Number of edges: $|V| - |F| - 1 + \rho \cdot \mathrm{OPT}$

Issue: $|V| - |F|$ could be as large as OPT.

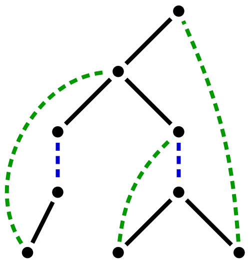
A second approach

1. Compute a 2-approximation $S \subseteq L$.
2. Choose $T \subseteq S$ such that $F \cup T$ is a spanning tree.
3. Improve the TAP solution $S \setminus T$.



Issue: We cannot always improve the TAP solution.

A better-than-2 approximation for Forest Augmentation



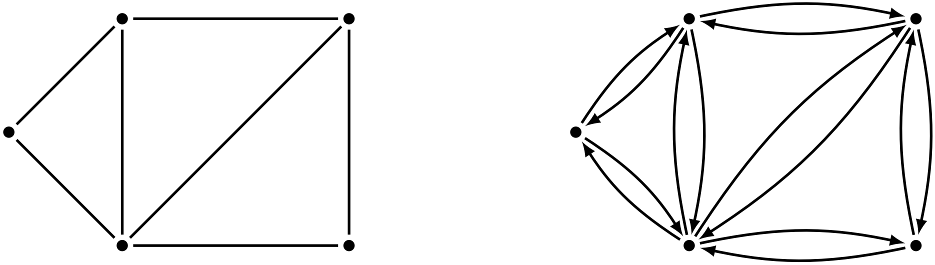
Construct TAP instance and
up-link solution U
of total cost $\leq 2 \cdot |\mathrm{OPT}|$.

U far from optimal → Use relative greedy to improve up-link solution.

U almost optimal → Then: FAP instance is “close” to MAP instance.

A 2-approximation for the Minimum Weight 2-ECSS problem

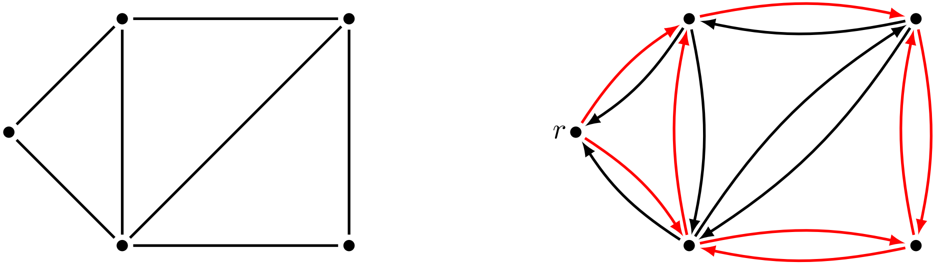
1. Replace every edge $\{a, b\}$ by directed edges (a, b) and (b, a).



A 2-approximation for the Minimum Weight 2-ECSS problem

1. Replace every edge $\{a, b\}$ by directed edges (a, b) and (b, a).
2. Fix $r \in V$ and compute cheapest directed edge set D with

$$|D \cap \delta^-(R)| \geq 2 \quad \forall \emptyset \neq R \subsetneq V \setminus \{r\}.$$

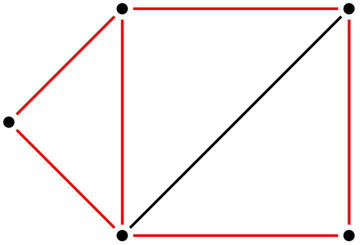


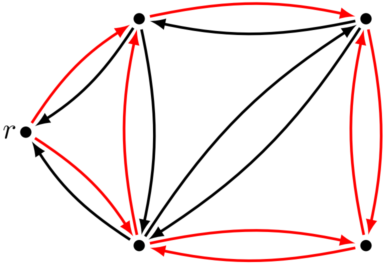
A 2-approximation for the Minimum Weight 2-ECSS problem

1. Replace every edge $\{a, b\}$ by directed edges (a, b) and (b, a).
2. Fix $r \in V$ and compute cheapest directed edge set D with

$$|D \cap \delta^-(R)| \geq 2 \quad \forall\, \emptyset \neq R \subsetneq V \setminus \{r\}.$$

3. Replace every edge $(a, b) \in D$ by $\{a, b\}$.



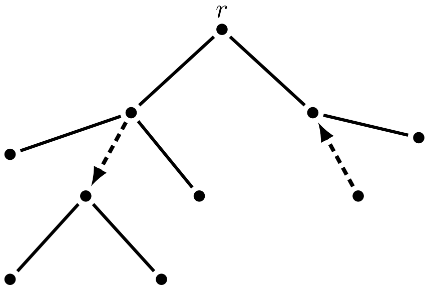


Computing a spanning tree with a cheap up-link solution

1. Fix $r \in V$ and compute cheapest directed edge set D with

$$|D \cap \delta^-(R)| \geq 2 \quad \forall \emptyset \neq R \subsetneq V \setminus \{r\}.$$

2. Choose $T \subseteq D$ such that $F \cup T$ is a spanning tree (ignoring orientations).

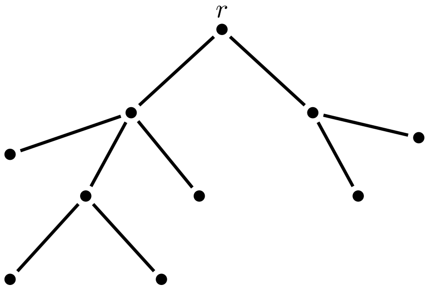


Computing a spanning tree with a cheap up-link solution

1. Fix $r \in V$ and compute cheapest directed edge set D with

$$|D \cap \delta^-(R)| \geq 2 \quad \forall\, \emptyset \neq R \subsetneq V \setminus \{r\}.$$

2. Choose $T \subseteq D$ such that $F \cup T$ is a spanning tree (ignoring orientations).



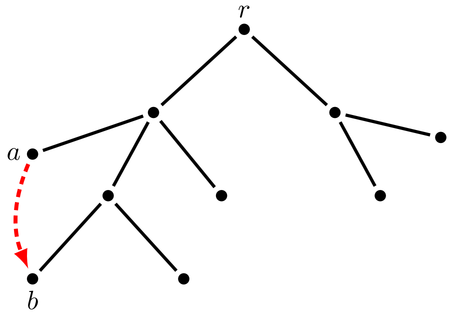
Computing a spanning tree with a cheap up-link solution

1. Fix $r \in V$ and compute cheapest directed edge set D with

$$|D \cap \delta^-(R)| \geq 2 \quad \forall\, \emptyset \neq R \subsetneq V \setminus \{r\}.$$

2. Choose $T \subseteq D$ such that $F \cup T$ is a spanning tree (ignoring orientations).
3. Up-links corresponding to links in $D \setminus T$ are a solution for the TAP instance with tree $F \cup T$.

link $(a, b) \in D \setminus T$

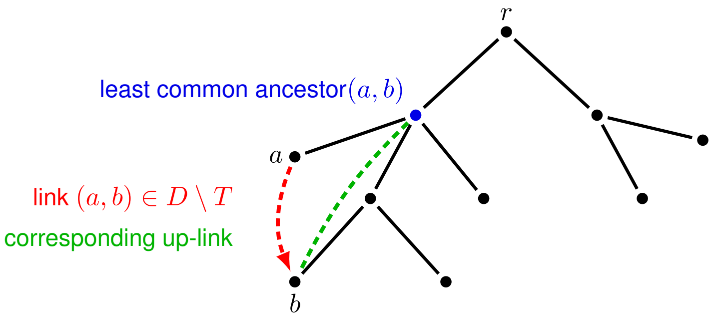


Computing a spanning tree with a cheap up-link solution

1. Fix $r \in V$ and compute cheapest directed edge set D with

$$|D \cap \delta^-(R)| \geq 2 \quad \forall\, \emptyset \neq R \subsetneq V \setminus \{r\}.$$

2. Choose $T \subseteq D$ such that $F \cup T$ is a spanning tree (ignoring orientations).
3. Up-links corresponding to links in $D \setminus T$ are a solution for the TAP instance with tree $F \cup T$.

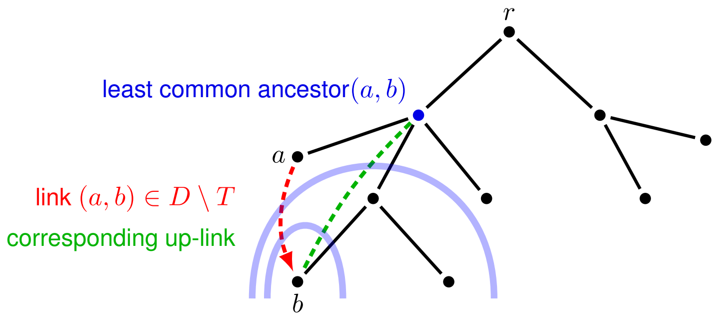


Computing a spanning tree with a cheap up-link solution

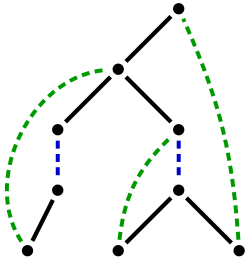
1. Fix $r \in V$ and compute cheapest directed edge set D with

$$|D \cap \delta^{-}(R)| \geq 2 \quad \forall\, \emptyset \neq R \subsetneq V \setminus \{r\}.$$

2. Choose $T \subseteq D$ such that $F \cup T$ is a spanning tree (ignoring orientations).
3. Up-links corresponding to links in $D \setminus T$ are a solution for the TAP instance with tree $F \cup T$.



A better-than-2 approximation for Forest Augmentation



Construct TAP instance and up-link solution U of total cost $\leq 2 \cdot |\mathrm{OPT}|$.

U far from optimal → Use relative greedy to improve up-link solution.

U almost optimal → Then: FAP instance is “close” to MAP instance.

Conclusions

Conclusions

Summary: best known approximation ratios

Min-weight 2-ECSS:	2-approximation
Weighted Tree Augmentation:	$(1.5 + \varepsilon)$-approximation
Forest Augmentation:	1.998-approximation
Tree Augmentation:	1.393-approximation

Open question: Is there a better-than-2 approximation for min-weight 2-ECSS?

Conclusions

Summary: best known approximation ratios

Min-weight 2-ECSS:	2-approximation
Weighted Tree Augmentation:	$(1.5 + \varepsilon)$-approximation
Forest Augmentation:	1.998-approximation
Tree Augmentation:	1.393-approximation

Open question: Is there a better-than-2 approximation for min-weight 2-ECSS?

Thank you!