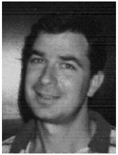
The Stackelberg Minimum Spanning Tree Game on Planar and Bounded-Treewidth Graphs


Jean Cardinal


Erik Demaine


Samuel Fiorini

Gwenaël Joret


Ilan Newman


Oren Weimann

Problem Statement

Stackelberg game, one leader and one follower

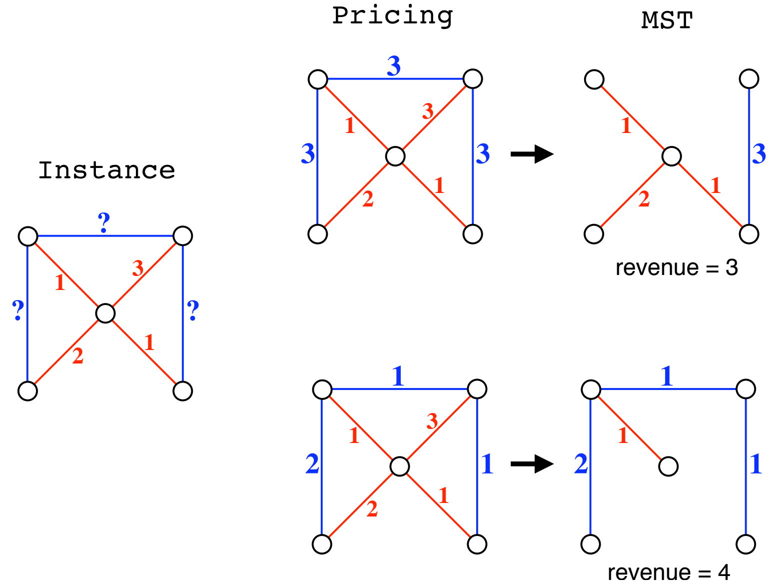
Given a graph G with red and blue edges

- Each red edge has a fixed cost $c(e)$
- The leader has to set a price $p(e)$ for each blue edge
- The follower then computes a MST with the resulting weights

Goal: maximize total weight of the blue edges in a MST

(= profit of leader)

Example



Assumption: blues have priority over reds of same weight

Stackelberg Games & Combinatorial Optimization

Stackelberg Shortest Path Game

- follower computes a shortest s–t path in a *directed* graph
- $O(\log |E(G)|)$-approx. algorithm — Roch, Savard, Marcotte 2005
- NP-hard — Roch, Savard, Marcotte 2005
- APX-hard — J. 2008
- NP-hard to approximate within $2 - \varepsilon$
  Briest and Khanna 2009 & Chalermsook, Lekhanukit, Nanongkai 2009
- Polyhedral studies — (numerous papers)
- Variants and special cases
  - River tarification problem
    Bouhtou, Grigoriev, van Hoesel, van der Kraaij, Spieksma, Uetz 2007
  - Highway problem — Heilporn, Labbé, Marcotte, Savard 2007

Stackelberg Games & Combinatorial Optimization

Stackelberg Shortest Paths Tree Game (symmetric & assymetric)

Bilò, Gualà, Proietti, Widmayer 2008 & Bilò, Gualà, Proietti 2009

Stackelberg Bipartite Vertex Cover Game — Briest, Hoefer, Krysta 2008

Stackelberg Minimum Spanning Tree Game — CDFJLNW 2007

$b := \#$blue edges

$c_1 \leq c_2 \leq \cdots \leq c_k$ red costs

- $O(\log n)$-approx. algorithm (*single price* algorithm)
  More precisely: $\min\{k, 1 + \ln b, 1 + \ln \frac{c_k}{c_1}\}$-approx.
- NP- and APX-hard, even when $k = 2$
- Integrality gap of natural LP-relaxation matches guarantee of *single price*

This Talk

Special cases of Stackelberg Minimum Spanning Tree Game:
G is planar / G has bounded treewidth

- NP-hard on planar graphs
- can be solved in poly-time on graphs of bounded treewidth
  (NB: not a FPT algorithm)

This Talk

Special cases of Stackelberg Minimum Spanning Tree Game:
G is planar / G has bounded treewidth

- NP-hard on planar graphs
- can be solved in poly-time on graphs of bounded treewidth
  (NB: not a FPT algorithm)

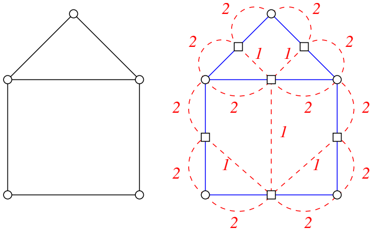
Motivation:

- Baker's decomposition of planar graphs in onion layers
- Stackelberg games and dynamic programs do not mix well
- Even for series-parallel graphs the problem is not trivial

NP-Hardness on Planar Graphs

Reduction from Minimum Connected Vertex Cover

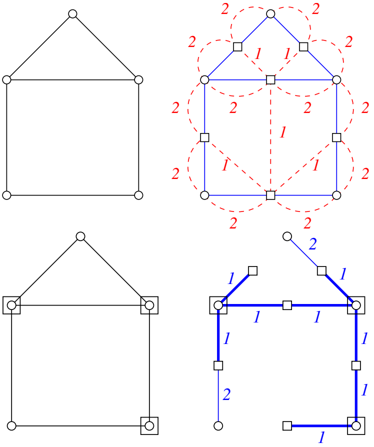
- NP-hard even if G planar with maximum degree 4   Garey and Johnson 1979



NP-Hardness on Planar Graphs

Reduction from Minimum Connected Vertex Cover

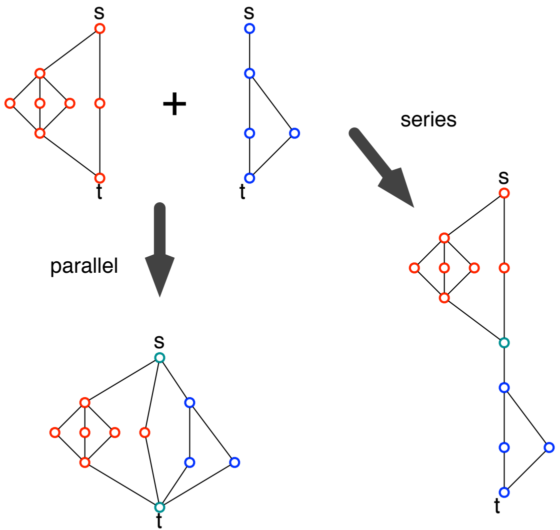
- NP-hard even if G planar with maximum degree 4 Garey and Johnson 1979



$\exists$ cover of size $\leq t \Leftrightarrow \exists$ pricing giving $\geq |E(G)| + 2|V(G)| - t - 1$

Series-Parallel Graphs

(G, s, t) series-parallel if $G \cong K_2$, or (G, s, t) results from a *series* or *parallel* composition:



A Lemma

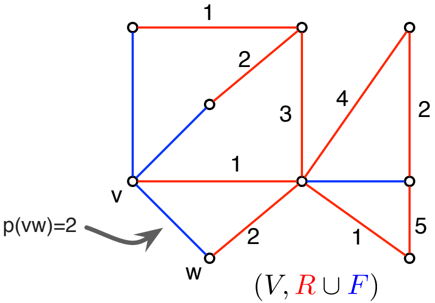
Assume:

- F *acyclic* subset of blue edges
- $\exists$ red spanning tree

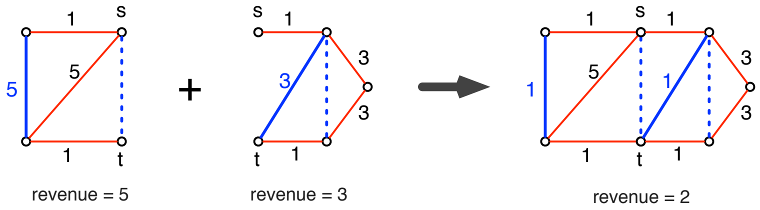
$\Rightarrow \exists$ unique pricing maximizing revenue over all solutions where follower buys F:

$$p(vw) = \min \left\{ \max_{e \in P \cap R} c(e) \mid P \in \widetilde{\mathcal{P}}(G, F, v, w) \right\} \qquad \forall vw \in F$$

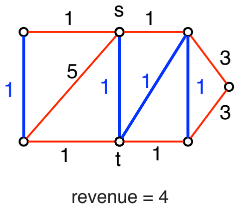
where $\widetilde{\mathcal{P}}(G, F, v, w) = \{v\text{–}w \text{ paths in } (V, R \cup F - \{vw\}) \text{ with} \geq 1 \text{ red edge}\}$



Parallel Compositions

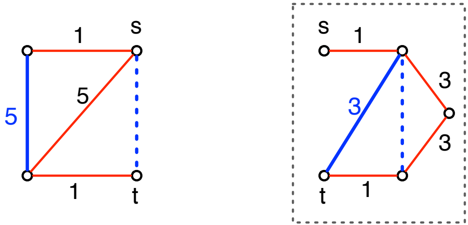


optimal solution:

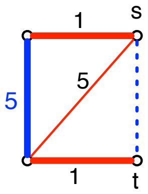


Cannot simply combine optimal solutions!

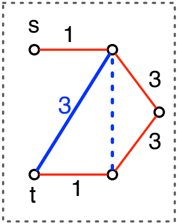
Parallel Compositions



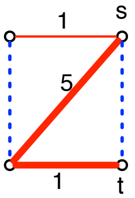
Parallel Compositions



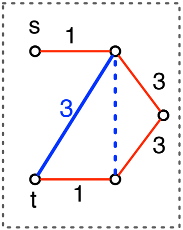
bottleneck = 1



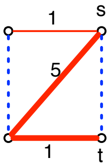
Parallel Compositions



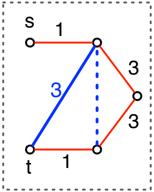
bottleneck = 5



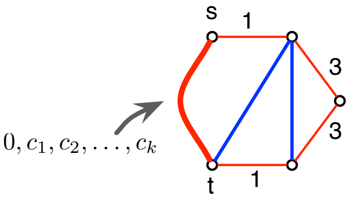
Parallel Compositions



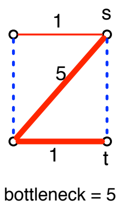
bottleneck = 5

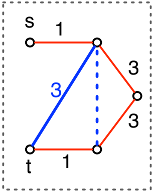


→ can prepare optimal solution for every possible bottleneck:

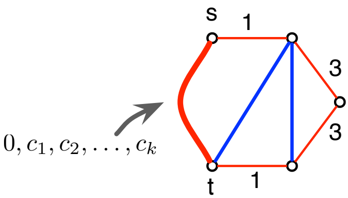


Parallel Compositions





→ can prepare optimal solution for every possible bottleneck:



Still not enough: what about bottlenecks on our side?

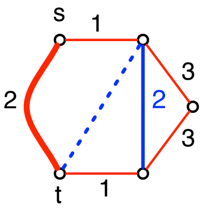
Parallel Compositions

Solution: prepare optimal solution for every

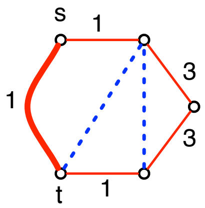
$$(i, j) \in \{0, c_1, \ldots, c_k\} \times \{0, c_1, \ldots, c_k\},$$

where

- i: internal bottleneck
- j: external bottleneck



(1,2)



(3,1)

NB: some pairs (i, j) are not feasible

Bounded Treewidth Graphs

G has treewidth $\leq \omega \iff G$ is an ω-boundaried graph

Abrahamson and Fellows 1993

ω-boundaried graph:

- ω *boundary vertices*, labeled with $1, 2, \ldots, \omega$
- operator $\emptyset$
- operator $\oplus$
- operator η
- operator ϵ
- operators that permute labels

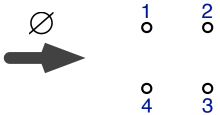
Bounded Treewidth Graphs

G has treewidth $\leq \omega \iff G$ is an ω-boundaried graph

Abrahamson and Fellows 1993

ω-boundaried graph:

- ω *boundary vertices*, labeled with $1, 2, \ldots, \omega$
- operator $\varnothing$



- operator $\oplus$
- operator η
- operator ϵ
- operators that permute labels

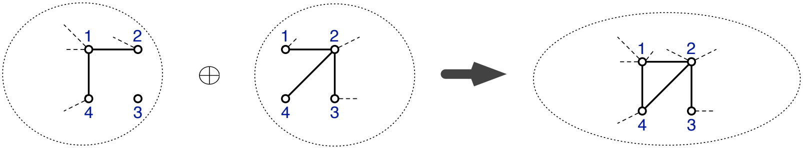
Bounded Treewidth Graphs

G has treewidth $\leq \omega \iff G$ is an ω-boundaried graph

Abrahamson and Fellows 1993

ω-boundaried graph:

- ω *boundary vertices*, labeled with $1, 2, \ldots, \omega$
- operator $\varnothing$
- operator $\oplus$



- operator η
- operator ϵ
- operators that permute labels

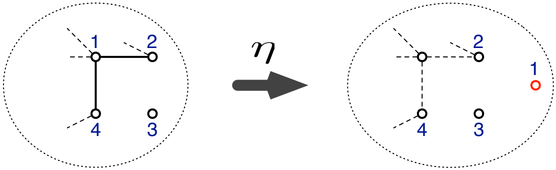
Bounded Treewidth Graphs

G has treewidth $\leq \omega \iff G$ is an ω-boundaried graph

Abrahamson and Fellows 1993

ω-boundaried graph:

- ω *boundary vertices*, labeled with $1, 2, \ldots, \omega$
- operator $\emptyset$
- operator $\oplus$
- operator η



- operator ϵ
- operators that permute labels

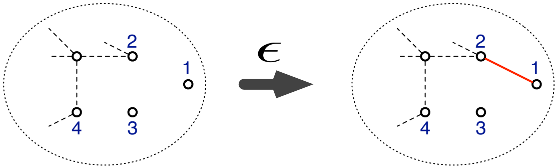
Bounded Treewidth Graphs

G has treewidth $\leq \omega \iff G$ is an ω-boundaried graph

Abrahamson and Fellows 1993

ω-boundaried graph:

- ω *boundary vertices*, labeled with $1, 2, \ldots, \omega$
- operator $\varnothing$
- operator $\oplus$
- operator η
- operator ϵ



- operators that permute labels

Bounded Treewidth Graphs

G has treewidth $\leq \omega \iff G$ is an ω-boundaried graph

Abrahamson and Fellows 1993

ω-boundaried graph:

- ω *boundary vertices*, labeled with $1, 2, \ldots, \omega$
- operator $\varnothing$
- operator $\oplus$
- operator η
- operator ϵ
- operators that permute labels

Decomposition can be found in linear time (for fixed ω)

Bodlaender 1996

Bounded Treewidth Graphs

General approach: handle each operator as in series-parallel case

Here, we compute k^{ω^2} solutions for each piece in the decomposition

Total complexity of the algorithm is $m^{O(\omega^2)}$

Conclusion

$\exists$ FPT algorithm parameterized by treewidth?

i.e. algorithm with complexity $O(f(\omega) \cdot n^c)$ for some absolute constant $c > 0$

Conclusion

$\exists$ FPT algorithm parameterized by treewidth?

i.e. algorithm with complexity $O(f(\omega) \cdot n^c)$ for some absolute constant $c > 0$

Conjecture: NO (under some reasonable complexity-theoretic assumption)

Conclusion

$\exists$ FPT algorithm parameterized by treewidth?

i.e. algorithm with complexity $O(f(\omega) \cdot n^c)$ for some absolute constant $c > 0$

Conjecture: NO (under some reasonable complexity-theoretic assumption)

For general graphs:

- APX-hard
- can be approximated within $\min\{k, 1 + \ln b, 1 + \ln \frac{c_k}{c_1}\}$

$\exists$ constant-factor approx. algorithm?

Thank You!