6.889	Lecture 16	November 7, 2011

Review of PTASes seen so far:

- Baker's approach for "local" problems
  ↳ deletion decomposition view
- Bidimensionality
- Klein's framework for connectivity problems
  ↳ contraction decomposition

Review of Klein's framework:

① Spanner step: find a subgraph G_S of G s.t.

i) $\ell(G_S) \leq c \cdot OPT$

ii) G_S contains a $(1+\varepsilon)$-approximate solution

② Contraction Decomposition: partition the edges of G_S into k parts, s.t. contracting any part gives a graph of treewidth $O(k)$.

Q_1 Q_2 . . . Q_i . . . Q_k

contract $\longrightarrow$ tw $O(k)$

Choose a part Q^* of minimum weight.

Then $\ell(Q^*) \leq \frac{\ell(G_S)}{k} \leq \frac{c \cdot OPT}{k} \leq \varepsilon \cdot OPT$

by choosing $k \geq \frac{c}{\varepsilon}$.

Recall that the algorithm for contraction decomp. in planar graphs is "Baker in the dual" but the analysis is different $\longrightarrow$ see Lecture 15

③ Dynamic Programming: Solve the problem optimally on G_S / Q^* by DP on a graph of tw $O(k)$.

④ Lifting: Convert the solution found in Step ③ to a solution in G_S by incorporating Q^*.

Notes: – the problems considered in this framework should be contraction-monotone, so that

$$OPT(G_S/Q^*) \leq OPT(G_S)$$

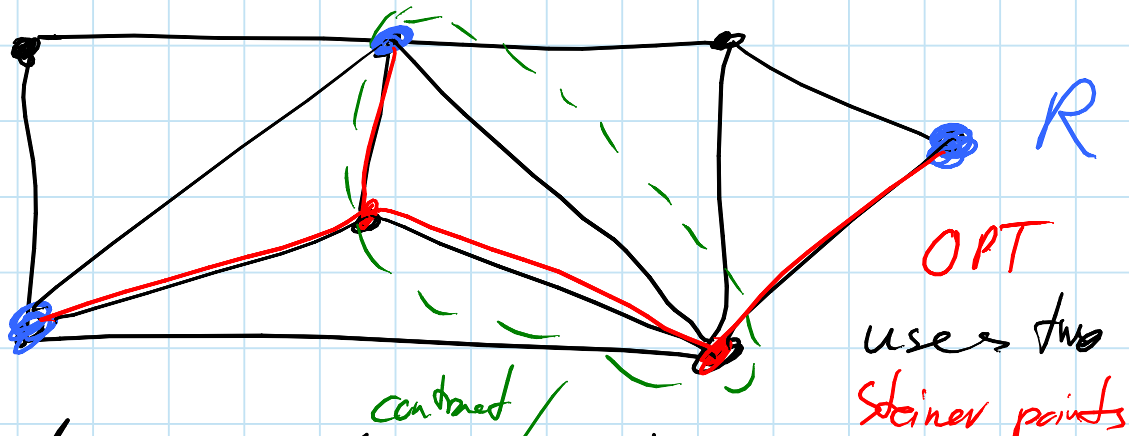
– lifting for TSP: upon uncontracting Q^*, the solution might not be a tour but it can be turned into a tour by using each edge of Q^* at most twice.

Steiner tree

Given: graph G, set of terminals $R \subseteq V(G)$
nonnegative edge weights

Goal: find a subgraph of G of minimum weight that spans R.

→ solution is always a tree
→ if $R = V(G)$ then this is MST
→ if $|R| = 2$ then this is shortest path



- NP-hard even in planar graphs
- 1.37-approx. in general graphs
- no PTAS in general graphs unless P=NP.

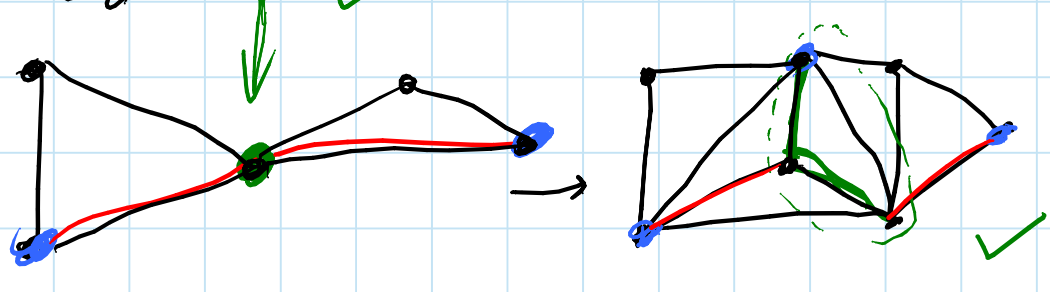
PTAS:

① Spanner ?

② contraction-decomp. ✓
(is contraction monotone)

③ DP on bounded tw ✓

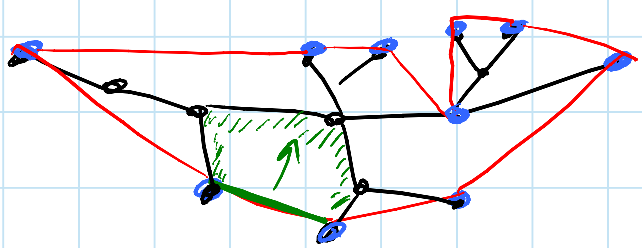
④ Lifting:
(uses Q^K at most once)



Spanner for Steiner tree [Borradaile, Klein, Mathieu 2007 / 2009]

① 2-approximation:

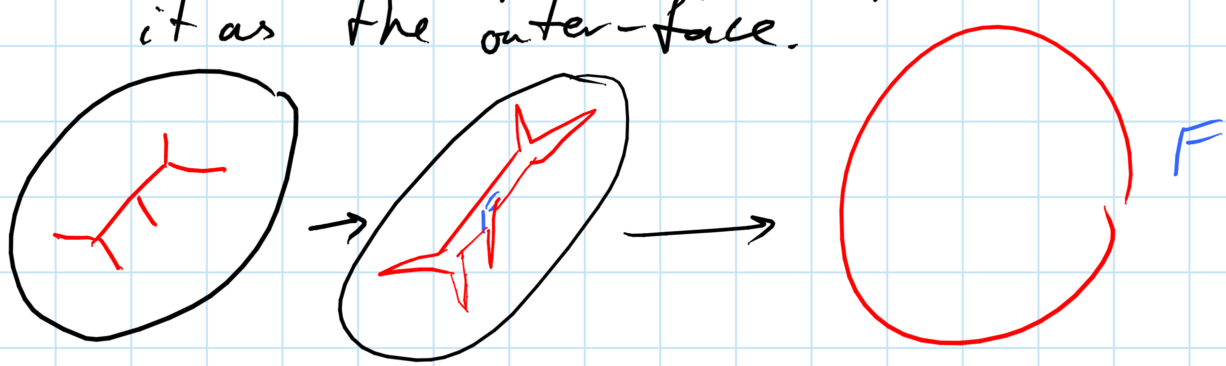
- construct distance network of terminals, that is a complete graph on R where the weight of each edge uv is $\mathrm{dist}_G(u,v)$.
- find a MST of the distance network
- take union of shortest paths corresponding to MST as 2-approximation



each edge of tree charged to at most twice

- can be implemented in $O(n)$ using reduced distance network [Mehlhorn '88, Tazari-Müller-Hannemann '09]

(1') split the graph along 2-approx. and think of it as the outer-face.

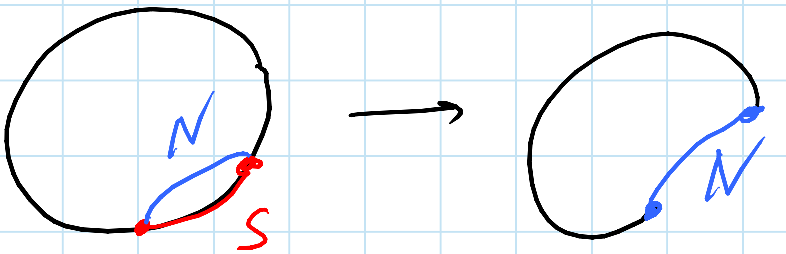


(2) Strip Decomposition

A path P between u and v is called $(1+\varepsilon)$-short iff

$$\ell(P) \leq (1+\varepsilon)\, dist_G(u,v)$$

- consider a minimal subpath S of current boundary which is not $(1+\varepsilon)$-short. Find a shortest path N between the endpoints of S.
Replace current boundary by removing S and adding N:



The part of G enclosed between S and N is called a strip. S is called the south boundary of the strip and N the north boundary.

N is a shortest path and every proper subpath of S is $(1+\varepsilon)$-short.

Lemma: The total length of all north boundaries is at most $\varepsilon^{-1} \cdot \ell(F)$ where F denotes the outer-face.

→ proof as in TSP-spanner! see prev. lecture.

⇒ Corollary: The total length of all strip boundaries is at most $8\varepsilon^{-1} \cdot OPT$.

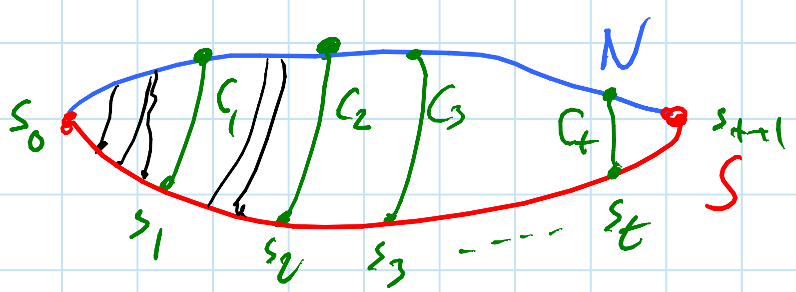
$\boxed{\text{→ add all strip boundaries to the spanner.}}$

↳ strips can be found in $O(n \log n)$ time using a variant of MSSP.

---

③ Finding Columns in a Strip:

- call the leftmost vertex on the south boundary s_0
- traverse the south boundary S from left to right.
- let s_1 be the first vertex such that $dist_S(s_0, s_1) > \varepsilon \cdot dist_{strip}(s_1, N)$

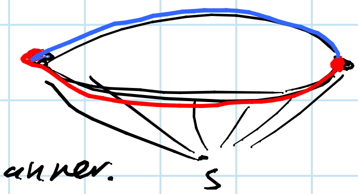


- continue to find such paths $P_1, \ldots, P_t$

→ $\ell(C_1) + \ell(C_2) + \ldots + \ell(C_t) < \frac{\ell(S)}{\varepsilon}$

- let C_0 be the trivial path $\{s_0\}$ and C_{t+1} the trivial path $\{s_{t+1}\}$. The paths $C_0, \dots, C_{t+1}$ are called the columns of the strip.

- total weight of all columns is at most $8\varepsilon^{-2} \cdot OPT$.
- for each strip, we can find all columns at once via a single-source shortest-paths computation from a super source.



- do ~~not~~ add columns to spanner.

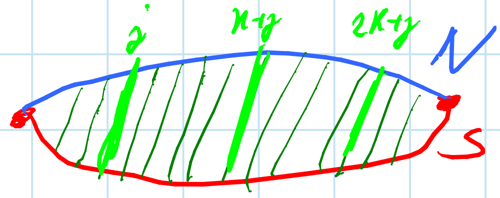
---

④ Adding Super-columns.

- define $\kappa = 8\varepsilon^{-3}$.
- let $C_0, \dots, C_{t+1}$ be the columns in one strip.
- label the columns periodically by $0, \dots, \kappa-1$, that is $\text{label}(C_i) = i \bmod \kappa$.
- choose a label j from $\{0, \dots, \kappa-1\}$ such that the total weight of all columns with label j is minimal among all choices of j.
- call the columns with label j super-columns

$\rightarrow$ total weight of all super-columns
$$\leq \frac{1}{\kappa} \cdot (\text{total weight of all columns}) \leq \varepsilon \cdot OPT$$

$\boxed{\rightarrow \text{add all super-columns to the spanner.}}$

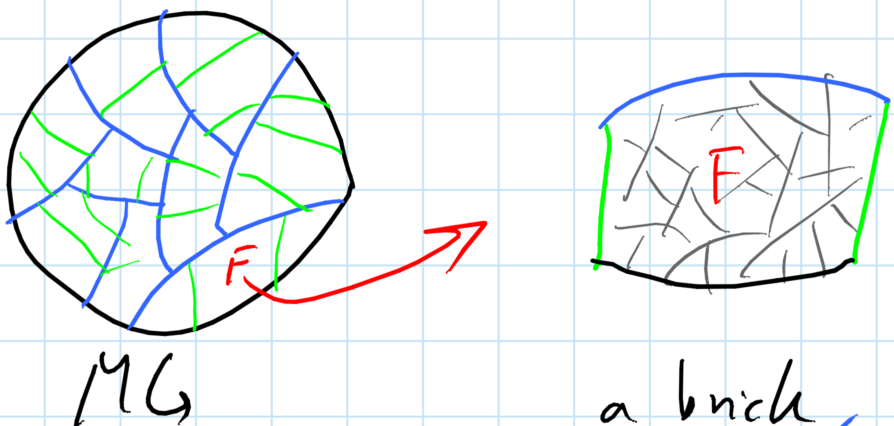


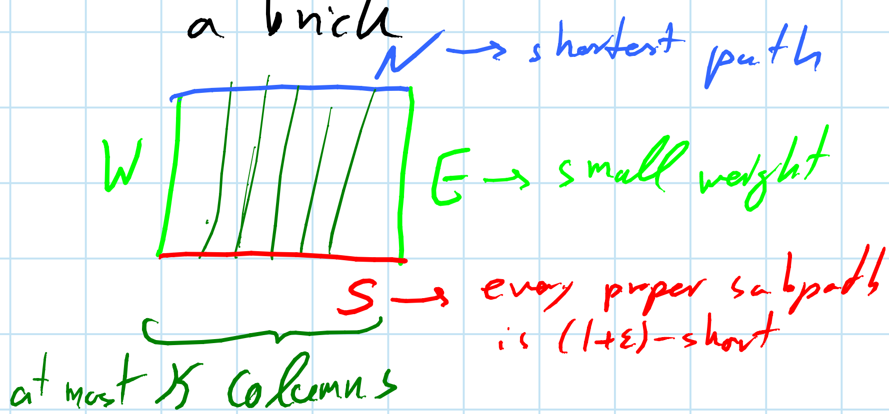
Super-columns

Intermission: The Mortar Graph and Bricks

We define the mortar graph MG to be the union of the strip boundaries and super-columns.

Let F be a face of the mortar graph. The part of G that is embedded inside F including F is called a brick. F is called the mortar boundary of the brick.





Lemma: The mortar graph contains all the terminals and its weight is at most $9\varepsilon^{-1}\cdot OPT$. It can be computed in $O(n \log n)$ time. ✓

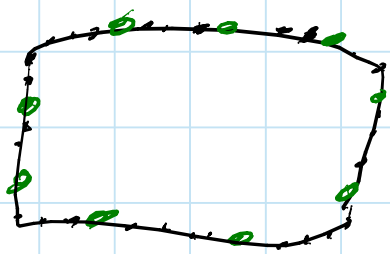
(5) Designating Portals

Let $\theta = 2\alpha(\varepsilon) \cdot 9\varepsilon^{-2}$ where $\alpha(\varepsilon) = O(\varepsilon^{-5.5})$ to be determined later.

For each brick B greedily select a set of portals on the mortar boundary ∂B such that for every vertex $x \in \partial B$ there is a portal $v \in \partial B$ with

$$\mathrm{dist}_{\partial B}(x, v) \leq \ell(\partial B) / \theta .$$

Lemma: There are at most θ portals on ∂B.

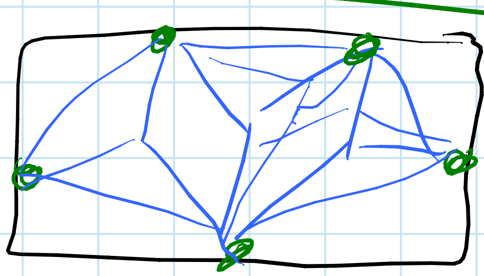


brick B with θ portals

---

(6) Adding Steiner Trees

For each brick B and each subset X of portals of B add the optimal Steiner tree that connects X inside B to the spanner.



- Each such Steiner tree can be computed in poly time via DP since terminals are on one face [Bern '90]

This completes the construction of the spanner.

Lemma: The total length of the spanner is at most

$$(1+2^{\theta+1})\, 9\, \varepsilon^{-1}\, OPT.$$

Proof: The length of any Steiner tree inside a brick B is bounded by $\ell(\partial B)$.
The number of Steiner trees in B is at most 2^{θ} and so their total length in all bricks is bounded by

$$\sum_{\text{bricks } B} 2^{\theta} \cdot \ell(\partial B) = 2^{\theta} \sum_{\text{bricks } B} \ell(\partial B) = 2^{\theta} \cdot 2\,\ell(MG)$$

The total weight of the spanner is thus

$$\ell(MG) + 2^{\theta+1} \cdot \ell(MG) \leq (1+2^{\theta+1})\, 9\, \varepsilon^{-1}$$

☺

Still need to show that the spanner contains a $(1+c\varepsilon)$-approximate solution to the original problem. This is done via the Structure Theorem which is presented in the next lecture. It essentially states that choosing θ portals on each sufficient to achieve a $(1+\varepsilon)$-approximate solution, that is, the optimal Steiner tree can be transformed in such a way that it has at most $\alpha(\varepsilon)$ joining vertices with each brick boundary.

Other Problems for Mortar Graph – Brick Decomp.

- Subset TSP: Given a set of terminals R in G, find a shortest tour that visits R.
  - same framework applies
  - sufficient to add shortest paths between portals instead of Steiner trees
  - needs a separate structure theorem [BDT '09]
  - a subset TSP-spanner can be constructed directly in $O(n \log n)$ time [Klein '06]
- 2-edge-connected sub-multigraph:
  Given a set of terminals R in G, find a shortest 2-edge-connected subgraph of G that includes R. May use edges twice to achieve connectivity but have to pay twice.
  - same framework applies but needs separate structure theorem [BK '08]
- multi-terminal cut: dual framework applies
- Steiner Forest: framework applies but requires additional techniques, e.g. Prize-Collecting Clustering [BHM '10, EKM '12]
- Euclidean Steiner tree with obstacles in the plane: can be reduced to planar instance [MT '10]

Note: There is a more efficient version of the PTAS that does not compute the spanner but works directly with the mortar graph-brick decomposition using dynamic programming on the bricks. Its running time is $2^{O(\epsilon^{-9})} \cdot n \log n$.

This algorithm has been implemented and extensive experiments have been conducted on its various parameters. The main idea that made the implementation possible was to choose $\theta = 5$ instead of $O(\epsilon^{-7.5})$. Surprisingly we obtain excellent results with this choice even though we sacrifice the theoretical approximation guarantee.

[TM '09]

References

[BDT09] Glencora Borradaile, Erik D. Demaine, and Siamak Tazari. Polynomial-time approximation schemes for subset-connectivity problems in bounded-genus graphs. In *STACS '09: Proceedings of the 26th Symposium on Theoretical Aspects of Computer Science*, pages 171–182, 2009.

[Ber90] Marshall Bern. Faster exact algorithms for Steiner trees in planar networks. *Networks*, 20:109–120, 1990.

[BHKM12] MohammadHossein Bateni, MohammadTaghi Hajiaghayi, Philip Klein, and Claire Mathieu. A polynomial-time approximation scheme for planar multiway cut. In *SODA '12: Proceedings of the 23rd annual ACM-SIAM Symposium on Discrete Algorithms*, page to appear. ACM, 2012.

[BHM10] MohammadHossein Bateni, MohammadTaghi Hajiaghayi, and Dániel Marx. Approximation schemes for Steiner forest on planar graphs and graphs of bounded treewidth. In *STOC '10: Proceedings of the 42nd annual ACM Symposium on Theory of Computing*, pages 211–220. ACM, 2010.

[BK08] G. Borradaile and P. Klein. The two-edge connectivity survivable network problem in planar graphs. In *ICALP '08: Proceedings of the 35th International Colloquium on Automata, Languages and Programming*, volume 5125 of *Lecture Notes in Computer Science*, pages 485–501. Springer, 2008.

[BKM09] Glencora Borradaile, Philip N. Klein, and Claire Mathieu. An $O(n \log n)$ approximation scheme for Steiner tree in planar graphs. *ACM Transactions on Algorithms*, 5(3), 2009.

[EKM12] David Eisenstat, Philip Klein, and Claire Mathieu. An efficient polynomial-time approximation scheme for Steiner forest in planar graphs. In *SODA '12: Proceedings of the 23rd annual ACM-SIAM Symposium on Discrete Algorithms*, page to appear. ACM, 2012.

[Kle05] Philip N. Klein. A linear-time approximation scheme for TSP for planar weighted graphs. In *FOCS '05: Proceedings of the 46th annual IEEE Symposium on Foundations of Computer Science*, pages 146–155, 2005.

[Kle06] Philip N. Klein. A subset spanner for planar graphs, with application to subset TSP. In *STOC '06: Proceedings of the 38th annual ACM Symposium on Theory of Computing*, pages 749–756, 2006.

[Meh88] K. Mehlhorn. A faster approximation algorithm for the Steiner problem in graphs. *Information Processing Letters*, 27:125–128, 1988.

[MT10] Matthias Müller-Hannemann and Siamak Tazari. A near linear time approximation scheme for Steiner tree among obstacles in the plane. *Computational Geoemtry: Theory and Applications*, 43:395–409, May 2010. Special Issue on the 10th Workshop on Algorithms and Data Structures (WADS 2007).

[Taz10] Siamak Tazari. *Algorithmic Graph Minor Theory: Approximation, Parameterized Complexity, and Practical Aspects.* PhD thesis, Humboldt-Univeristät zu Berlin, Berlin, Germany, 2010.

[TM09a] Siamak Tazari and Matthias Müller-Hannemann. Dealing with large hidden constants: Engineering a planar steiner tree PTAS. In *ALENEX '09: Proceedings of the 10th International Workshop on Algorithm Engineering and Experiments*, pages 120–131. SIAM, 2009.

[TM09b] Siamak Tazari and Matthias Müller-Hannemann. Shortest paths in linear time on minor-closed graph classes, with an application to Steiner tree approximation. *Discrete Applied Mathematics*, 157:673–684, 2009.