The Traveling Salesman: Classical Tools and Recent Advances

Lecture 3: s-t Path TSP

Anke van Zuylen

XV Summer School in Discrete Mathematics

Valparaiso, January 6-10, 2020

WILLIAM & MARY

CHARTERED 1693

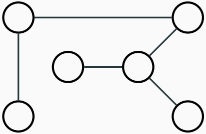
Introduction

Recall: Christofides-Serdyukov's Tree+Matching Algorithm

- Compute minimum spanning tree (MST) T on G. *"Connectivity"*

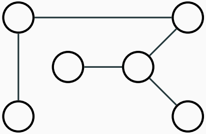
Recall: Christofides-Serdyukov's Tree+Matching Algorithm

- Compute minimum spanning tree (MST) T on G. *"Connectivity"*



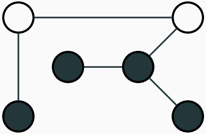
Recall: Christofides-Serdyukov's Tree+Matching Algorithm

- Compute minimum spanning tree (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Compute a minimum-cost matching M on Odd_T. *"Parity correction"*



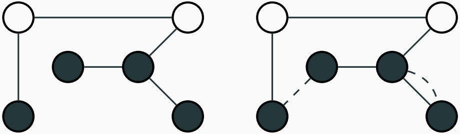
Recall: Christofides-Serdyukov's Tree+Matching Algorithm

- Compute minimum spanning tree (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Compute a minimum-cost matching M on Odd_T. *"Parity correction"*



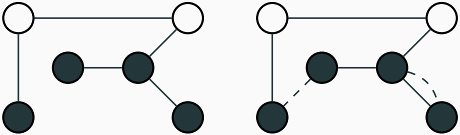
Recall: Christofides-Serdyukov's Tree+Matching Algorithm

- Compute minimum spanning tree (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Compute a minimum-cost matching M on Odd_T. *"Parity correction"*



Recall: Christofides-Serdyukov's Tree+Matching Algorithm

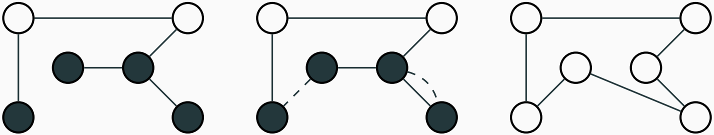
- Compute minimum spanning tree (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Compute a minimum-cost matching M on Odd_T. *"Parity correction"*
- "Shortcut" Eulerian traversal in resulting Eulerian graph $(V, T \sqcup M)$[1].



[1] $\sqcup$ is the disjoint union, so $T \sqcup M$ has two copies of e if $e \in T$ and $e \in M$.

Recall: Christofides-Serdyukov's Tree+Matching Algorithm

- Compute minimum spanning tree (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Compute a minimum-cost matching M on Odd_T. *"Parity correction"*
- "Shortcut" Eulerian traversal in resulting Eulerian graph $(V, T \sqcup M)$[1].



[1] $\sqcup$ is the disjoint union, so $T \sqcup M$ has two copies of e if $e \in T$ and $e \in M$.

Recall: Analysis

Theorem (Christofides (1976), Serdyukov (1978))

The Tree+Matching algorithm is a $\frac{3}{2}$-approximation algorithm for the TSP.

Let $c(T)$ be the cost of the edges in the MST, $c(M)$ the cost of the edges in the matching of Odd_T, and let OPT be the cost of the optimal tour.

Lemma

$$c(T) \leq OPT.$$

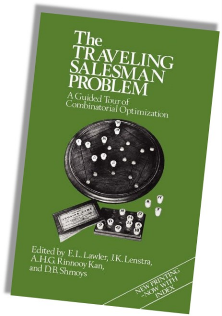
Lemma

$$c(M) \leq \frac{1}{2} OPT.$$

s-t Path TSP

p.145-180: *Performance guarantees for heuristics.*
Johnson and Papadimitriou (1985)

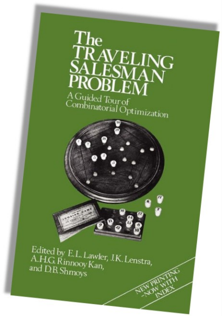
Exercises:
13. Let s, t be fixed vertices. Show that a traveling salesman path (a path that visits each city exactly once) from s to t, whose length is at most (3/2) times the optimal path length can be found using a Christofides-like algorithm.



s-t Path TSP

p.145-180: *Performance guarantees for heuristics.*
Johnson and Papadimitriou (1985)

Exercises:
13. Let s, t be fixed vertices. Show that a traveling salesman path (a path that visits each city exactly once) from s to t, whose length is at most (3/2) times the optimal path length can be found using a Christofides-like algorithm.



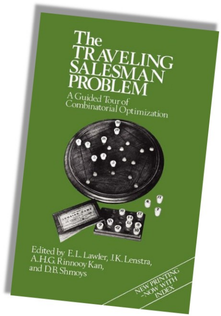
???

s-t Path TSP

p.145-180: *Performance guarantees for heuristics.*
Johnson and Papadimitriou (1985)

Exercises:

~~13. Let s, t be fixed vertices. Show that a traveling salesman path (a path that visits each city exactly once) from s to t, whose length is at most (3/2) times the optimal path length can be found using a Christofides-like algorithm.~~

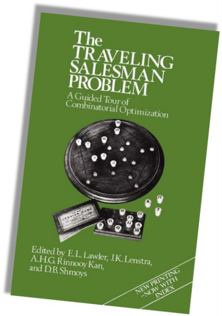


???

s-t Path TSP

p.145-180: *Performance guarantees for heuristics.*
Johnson and Papadimitriou (1985)

Exercises:
13. (a) Show that a traveling salesman path (a path that visits each city exactly once), whose length is at most (3/2) times the optimal path length can be found using a Christofides-like algorithm. (*Hint:* It may be necessary to add points to the instance so the desired matching exists.) (b) Show that the same bound can be obtained when one of the endpoints of the path is specified in advance. (c) What if *both* endpoints are fixed in advance?!



s-t Path TSP

Input:

- A complete undirected graph $G = (V, E)$;
- Start vertex $s \in V$, end vertex $t \in V$;
- Edge costs $c(e) \equiv c(i, j) \geq 0$ for all $e = (i, j) \in E$;
- Edge costs satisfy the **triangle inequality**: $c(i, j) \leq c(i, k) + c(k, j)$ for all i, j, k.

Goal: Find a min-cost path from s to t that visits all other vertices in between.

A First Result

Eulerian Path

There is an Eulerian path that starts at s, ends at t, and visits every edge exactly once iff s and t have odd-degree and all other vertices have even degree.

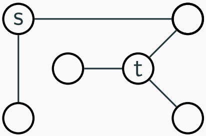
Eulerian Path

There is an Eulerian path that starts at s, ends at t, and visits every edge exactly once iff s and t have odd-degree and all other vertices have even degree.

Because edge cost satisfy triangle inequality, an Eulerian path from s to t can be shortcut to an s-t traveling salesman path without increasing the cost.

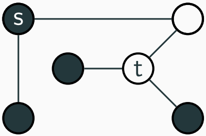
Tree+Matching Algorithm

- Compute *minimum spanning tree* (MST) T on G. *"Connectivity"*



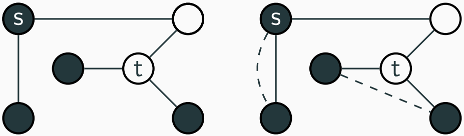
Tree+Matching Algorithm

- Compute *minimum spanning tree* (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Let $\text{W}_T = \text{Odd}_T \triangle \{s, t\}$.



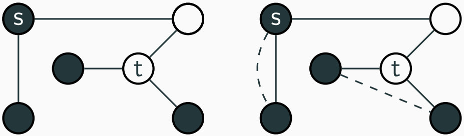
Tree+Matching Algorithm

- Compute *minimum spanning tree* (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Let $\mathrm{W}_T = \mathrm{Odd}_T \triangle \{s, t\}$. Compute a *minimum-cost matching* M of W_T. *"Parity correction"*



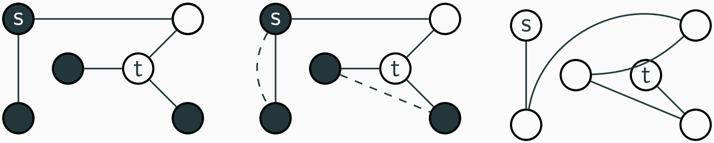
Tree+Matching Algorithm

- Compute *minimum spanning tree* (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Let $\mathrm{W}_T = \mathrm{Odd}_T \triangle \{s, t\}$. Compute a *minimum-cost matching* M of W_T. *"Parity correction"*
- "Shortcut" Eulerian path in $(V, T \sqcup M)$.

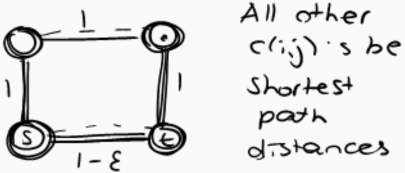


Tree+Matching Algorithm

- Compute *minimum spanning tree* (MST) T on G. *"Connectivity"*
- Let Odd_T be the odd-degree vertices of T. Let $\mathrm{W}_T = \mathrm{Odd}_T \triangle \{s, t\}$. Compute a *minimum-cost matching* M of W_T. *"Parity correction"*
- "Shortcut" Eulerian path in $(V, T \sqcup M)$.



Analysis of Tree+Matching Algorithm



Let $c(T)$ and $c(M)$ be the cost of the MST and min-cost matching of W_T, and OPT the cost of the min-cost s-t traveling salesman path.

Are the following true or false:

(a) $c(T) \leq OPT$? ✓

(b) $c(M) \leq \frac{1}{2}OPT$? ✗

Analysis of Tree+Matching Algorithm

Theorem (Hoogeveen (1990))

The Tree+Matching algorithm is a $\frac{5}{3}$-approximation algorithm for the s-t path TSP.

The theorem follows from the following two lemmas.

Lemma

$$c(T) \leq OPT \quad \checkmark$$

.

Lemma

$$c(M) \leq \frac{1}{3}(c(T) + OPT) \leq \frac{2}{3} OPT \quad \checkmark$$

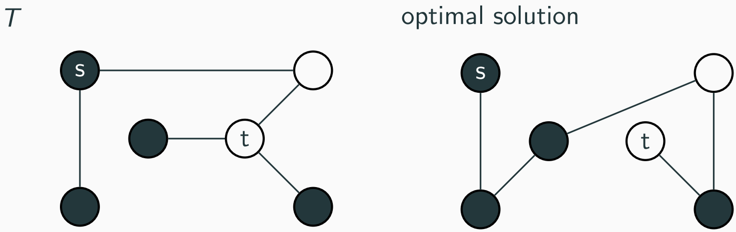
.

Proof of Second Lemma

Rather than a W_T-matching, we consider a W_T-join: A set of edges J is a W_T-join, if every vertex in W_T has odd degree in J, and every vertex in $V \setminus W_T$ has even degree in J.

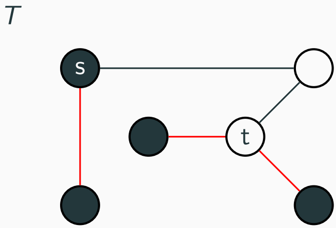
Alternative definition: J is a W_T-join, if every vertex has the "correct" degree parity in $T \sqcup J$ (i.e., s, t have odd degree, every other vertex has even degree).

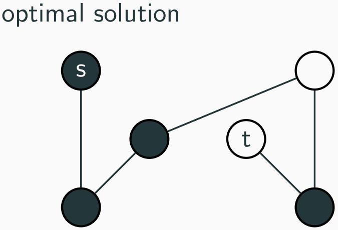
Proof of Second Lemma



Let P be the path between s and t in T (the edges in T that are not red).

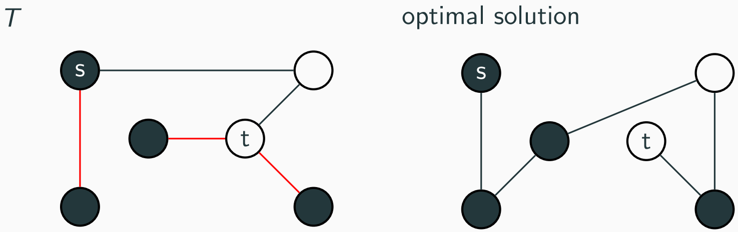
Proof of Second Lemma





Let P be the path between s and t in T (the edges in T that are not red). Observe that red edges in T form a W_T-join.

Proof of Second Lemma

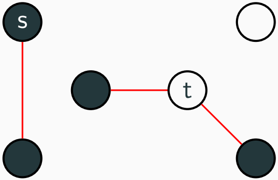


Let P be the path between s and t in T (the edges in T that are not red). Observe that red edges in T form a W_T-join.

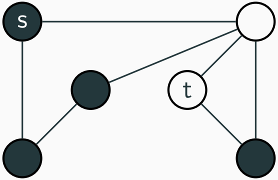
Move the edges from P to the optimal solution, to get an Eulerian graph instead of a path.

Proof of Second Lemma

$T \setminus P$



optimal solution plus P

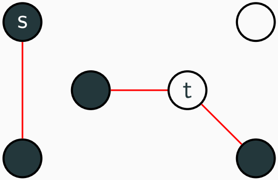


Let P be the path between s and t in T (the edges that are not red). Observe that red edges in T form a W_T-join.

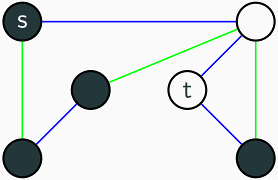
Move the edges from P to the optimal solution, to get an Eulerian graph instead of a path. A traversal of this graph can be used to get two W_T-joins (as we saw in the Christofides-Serdyukov analysis for the TSP).

Proof of Second Lemma

$T \setminus P$



optimal solution plus P

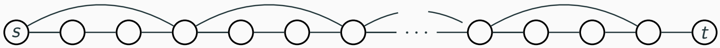


Let P be the path between s and t in T (the edges that are not red). Observe that red edges in T form a W_T-join.

Move the edges from P to the optimal solution, to get an Eulerian graph instead of a path. A traversal of this graph can be used to get two W_T-joins (as we saw in the Christofides-Serdyukov analysis for the TSP).

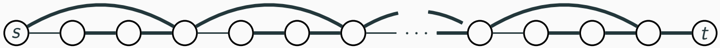
Tight Example

The analysis is tight, even for graph-TSP instances. Consider the graph-TSP instance below: cost $c(i,j)$ is number of edges in shortest i-j path in graph.



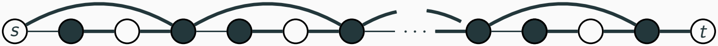
Tight Example

The analysis is tight, even for graph-TSP instances. Consider the graph-TSP instance below: cost $c(i, j)$ is number of edges in shortest i-j path in graph.



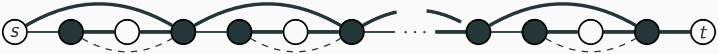
Tight Example

The analysis is tight, even for graph-TSP instances. Consider the graph-TSP instance below: cost $c(i,j)$ is number of edges in shortest i-j path in graph.



Tight Example

The analysis is tight, even for graph-TSP instances. Consider the graph-TSP instance below: cost $c(i, j)$ is number of edges in shortest i-j path in graph.



Improving These Results

The tight example shows that a minimum spanning tree might not be the best tree to start with.

Improving These Results

The tight example shows that a minimum spanning tree might not be the best tree to start with.

We would like a tree of cost at most (approximately) OPT, for which parity correction is cheaper than $\frac{2}{3}OPT$.

Improving These Results

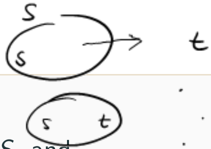
The tight example shows that a minimum spanning tree might not be the best tree to start with.

We would like a tree of cost at most (approximately) OPT, for which parity correction is cheaper than $\frac{2}{3}OPT$.

It turns out linear programming can help!

Linear Programs for the s-t Path TSP

LP Relaxation for s-t Path TSP



Let $\delta(S)$ be the set of edges with exactly one endpoint in S, and $x(E') \equiv \sum_{e \in E'} x(e)$.

Call $\delta(S)$ an <u>s-t cut</u> if $s \in S, t \notin S$ (or if $s \notin S, t \in S$). Call $\delta(S)$ a <u>non s-t cut</u> if $s \notin S, t \notin S$ (or if $s, t \in S$).

$$\text{Min} \quad \sum_{e \in E} c(e)x(e)$$

subject to:

$$x(\delta(i)) = \begin{cases} 1 & i = s,t \\ 2 & i \neq s,t \end{cases}$$

$$x(\delta(S)) \geq \begin{cases} 1 & s\text{-}t \text{ cut } \delta(S) \\ 2 & \text{non } s\text{-}t \text{ cut } \delta(S) \end{cases}$$

$$0 \leq x(e) \leq 1, \qquad \forall e \in E.$$

LP Relaxation for s-t Path TSP

Let $\delta(S)$ be the set of edges with exactly one endpoint in S, and $x(E') \equiv \sum_{e \in E'} x(e)$.

Call $\delta(S)$ an <u>s-t cut</u> if $s \in S, t \notin S$ (or if $s \notin S, t \in S$). Call $\delta(S)$ a <u>non s-t cut</u> if $s \notin S, t \notin S$ (or if $s, t \in S$).

$$\text{Min} \quad \sum_{e \in E} c(e)x(e)$$

subject to:

$$x(\delta(i)) = \begin{cases} 1, & \forall i = s, t, \\ 2, & \forall i \neq s, t, \end{cases}$$

$$x(\delta(S)) \geq \begin{cases} 1, & \forall s\text{-}t \text{ cuts } \delta(S), \\ 2, & \forall \text{non } s\text{-}t \text{ cuts } \delta(S), \end{cases}$$

$$0 \leq x(e) \leq 1, \qquad \forall e \in E.$$

Back to Analysis of Tree+Matching Algorithm

Theorem (An, Kleinberg, Shmoys (2012))

The Tree+Matching algorithm returns an s-t traveling salesman path solution of cost at most $\frac{5}{3} OPT_{LP}$.

The theorem follows from the following two lemmas, where T is an MST and M is a min-cost matching of W_T.

Lemma

$$c(T) \leq OPT_{LP}$$

Lemma

$$c(M) \leq \frac{2}{3} OPT_{LP}$$

Recall: the Spanning Tree LP

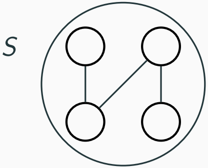
The minimum spanning tree can be found as the solution to the following LP (Edmonds 1971):

$$\text{Minimize} \quad \sum_{e \in E} c(e) z(e)$$

subject to:

$$z(E) = |V| - 1,$$
$$z(E(S)) \leq |S| - 1, \quad \forall |S| \subseteq V, |S| \geq 2,$$
$$z(e) \geq 0, \quad \forall e \in E,$$

where $E(S)$ is the set of all edges with both endpoints in S.



Proof of First Lemma: $c(T) \leq OPT_{LP}$

s-t path TSP LP

$$\text{Min} \sum_{e \in E} c(e)x(e)$$

$$x(\delta(i)) = \begin{cases} 1, & \forall i = s,t, \\ 2, & \forall i \neq s,t, \end{cases}$$

$$x(\delta(S)) \geq \begin{cases} 1, & \forall s\text{-}t \text{ cuts } \delta(S), \\ 2, & \forall \text{non } s\text{-}t \text{ cuts } \delta(S), \end{cases}$$

$$0 \leq x(e) \leq 1, \quad \forall e \in E.$$

Spanning tree LP

$$\text{Min} \sum_{e \in E} c(e)z(e)$$

$z(E) = |V| - 1$ ✓ (i)

$z(E(S)) \leq |S| - 1 \quad \forall S \subset V, |S| \geq 2$ ✓ (ii)

$z(e) \geq 0 \quad \forall e \in E.$

Let x^* be opt. solution to s-t path TSP LP.

Want to show: x^* feasible for spanning tree LP.

(i) $x^*(E) = \frac{1}{2} \sum_{i \in V} x^*(\delta(i)) = \frac{1}{2}(2|V| - 2) = |V| - 1$

(ii) $\sum_{i \in S} x^*(\delta(i)) = 2x^*(E(S)) + x^*(\delta(S))$

$\sum_{i \in S} x^*(\delta(i))$: $\leq 2|S|$ if non s-t cut; $\leq 2|S| - 1$ if s-t cut

$x^*(\delta(S))$: ≥ 1 if s-t cut; ≥ 2 if non s-t cut

Recall: the Matching LP

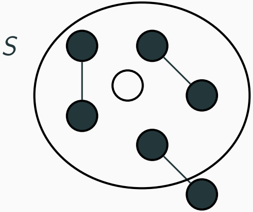
If edge costs are non-negative and satisfy the triangle inequality, a minimum-cost matching of W can be found as the solution to the following LP (Edmonds 1965):

$$\text{Minimize} \quad \sum_{e \in E} c(e) z(e)$$

subject to:

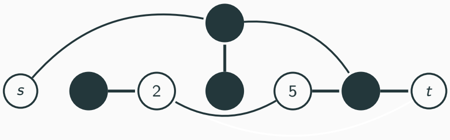
$$z(\delta(S)) \geq 1 \qquad \forall S \subset V, |S \cap W| \text{ odd.}$$

$$z(e) \geq 0 \qquad \forall e \in E.$$



Matching of W_T

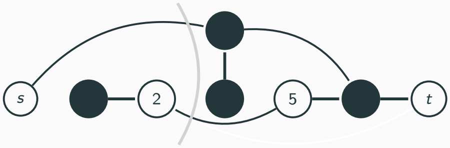
Recall that $\mathsf{W}_T = \mathsf{Odd}_T \triangle \{s, t\}$ is the set of vertices whose parity needs to be fixed.



We have a constraint $z(\delta(S)) \geq 1$ for $S \subset V$ when $|S \cap \mathsf{W}_T|$ is odd,

Matching of W_T

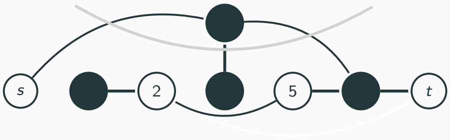
Recall that $\mathrm{W}_T = \mathrm{Odd}_T \triangle \{s, t\}$ is the set of vertices whose parity needs to be fixed.



We have a constraint $z(\delta(S)) \geq 1$ for $S \subset V$ when $|S \cap \mathrm{W}_T|$ is odd,

Matching of W_T

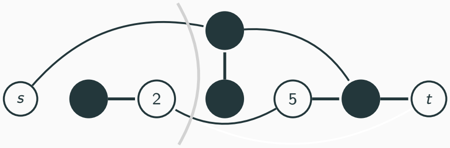
Recall that $W_T = \text{Odd}_T \triangle \{s, t\}$ is the set of vertices whose parity needs to be fixed.



We have a constraint $z(\delta(S)) \geq 1$ for $S \subset V$ when $|S \cap W_T|$ is odd,

Matching of W_T

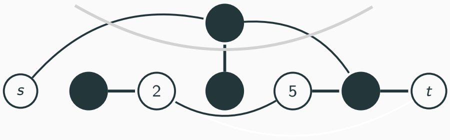
Recall that $W_T = \mathrm{Odd}_T \triangle \{s, t\}$ is the set of vertices whose parity needs to be fixed.



We have a constraint $z(\delta(S)) \geq 1$ for $S \subset V$ when $|S \cap W_T|$ is odd, i.e., if T does not have the parity in $\delta(S)$ that an s-t traveling salesman path should have.

Matching of W_T

Recall that $W_T = \mathrm{Odd}_T \triangle \{s, t\}$ is the set of vertices whose parity needs to be fixed.



We have a constraint $z(\delta(S)) \geq 1$ for $S \subset V$ when $|S \cap W_T|$ is odd, i.e., if T does not have the parity in $\delta(S)$ that an s-t traveling salesman path should have.

W_T- Matching LP

If edge costs are non-negative and satisfy the triangle inequality, a minimum-cost matching of W_T can be found as a solution to the following LP:

$$\text{Minimize} \qquad \sum_{e \in E} c(e) z(e)$$

subject to:

$$\begin{aligned} z(\delta(S)) &\geq 1 \quad \forall s\text{-}t \text{ cuts } \delta(S) : |\delta(S) \cap T| \text{ even}, \\ z(\delta(S)) &\geq 1 \quad \forall \text{non } s\text{-}t \text{ cuts } \delta(S) : |\delta(S) \cap T| \text{ odd}, \\ z(e) &\geq 0 \quad \forall e \in E. \end{aligned}$$

Proof of Second Lemma: $c(M) \leq \frac{2}{3} OPT_{LP}$

$$x(\delta(i)) = \begin{cases} 1, & \forall i = s, t, \\ 2, & \forall i \neq s, t, \end{cases}$$

$$x(\delta(S)) \geq \begin{cases} 1, & \forall s\text{-}t \text{ cuts } \delta(S), \\ 2, & \forall \text{non } s\text{-}t \text{ cuts } \delta(S), \end{cases}$$

$$0 \leq x(e) \leq 1, \quad \forall e \in E.$$

$$z(\delta(S)) \geq 1, \quad \forall s\text{-}t \text{ cuts } \delta(S)\text{: } |\delta(S) \cap T| \text{ even},$$

$$z(\delta(S)) \geq 1, \quad \forall \text{non } s\text{-}t \text{ cuts } \delta(S)\text{: } |\delta(S) \cap T| \text{ odd},$$

$$z(e) \geq 0, \quad \forall e \in E.$$

Proof of second lemma is left as an exercise (you will show that $c(M) \leq \frac{1}{3}(c(T) + OPT_{LP})$ which, together with the first lemma, implies the second lemma).

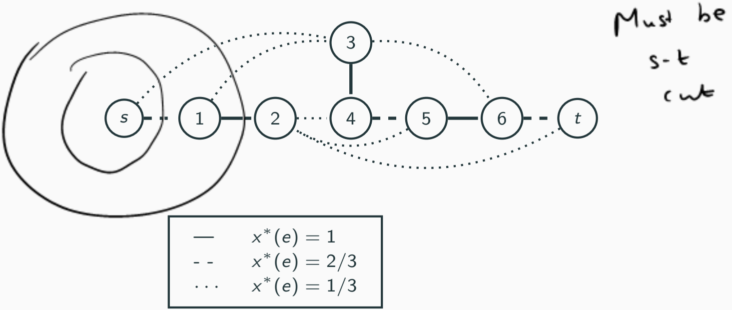
Obstacle to Improving the Bound: Even Narrow Cuts

No Even Narrow Cuts – No Problem

Let x^* be an optimal solution for the s-t path TSP LP.

Definition

A cut $\delta(S)$ is narrow if $x^*(\delta(S)) < 2$.



No Even Narrow Cuts – No Problem

Observation

Let T be a spanning tree, and $W_T = Odd_T \triangle \{s, t\}$.

If $c(T) \leq OPT_{LP}$, and there is no narrow cut $\delta(S)$ for which $|\delta(S) \cap T|$ is even, then adding a minimum-cost W_T-matching to T gives an s-t traveling salesman path of cost at most $\frac{3}{2} OPT_{LP}$.

No Even Narrow Cuts – No Problem

Observation

Let T be a spanning tree, and $W_T = Odd_T \triangle \{s, t\}$.

If $c(T) \leq OPT_{LP}$, and there is no narrow cut $\delta(S)$ for which $|\delta(S) \cap T|$ is even, then adding a minimum-cost W_T-matching to T gives an s-t traveling salesman path of cost at most $\frac{3}{2}OPT_{LP}$.

Reason: Narrow cuts are s-t cuts. The W_T-matching LP has no constraint for such a cut if $|\delta(S) \cap T|$ is odd.

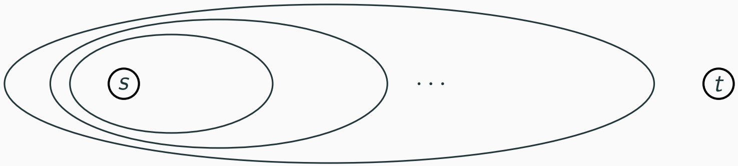
Since $\frac{1}{2}x^*(\delta(S)) \geq 1$ for all other cuts, $\frac{1}{2}x^*$ is feasible for the W_T-matching LP if there are no narrow cuts in which the tree is even.

Narrow Cuts Are Nested

Theorem (An, Kleinberg, Shmoys (2012))

If $\delta(S_1), \delta(S_2)$ are narrow cuts, $S_1 \neq S_2$, then either $S_1 \subset S_2$ or $S_2 \subset S_1$.

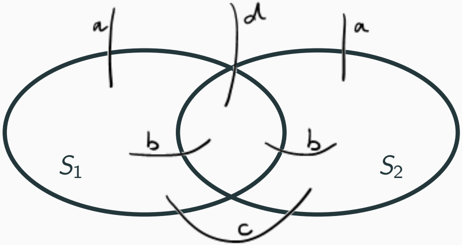
So the narrow cuts look like $s \in S_1 \subset S_2 \subset \cdots \subset S_k \subset V$.



Proof

First need to show that

$$x^*(\delta(S_1)) + x^*(\delta(S_2)) \geq x^*(\delta(S_1 \setminus S_2)) + x^*(\delta(S_2 \setminus S_1)).$$

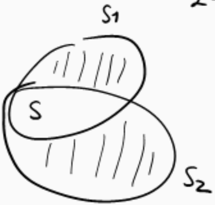


	LHS	RHS
a	1	1
b	1	1
c	2	2
d	2	0

Proof

Theorem (An, Kleinberg, Shmoys (2012))

If S_1, S_2 are narrow cuts, $S_1 \neq S_2$, then either $S_1 \subset S_2$ or $S_2 \subset S_1$.



$$2+2 > x^*(\delta(S_1)) + x^*(\delta(S_2)) \geq$$

$$x^*(\delta(S_1 \setminus S_2)) + x^*(\delta(S_2 \setminus S_1))$$

$$\geq 2+2$$

↑ not s-t cuts

Contradiction!

Improvements

An, Kleinberg and Shmoys propose the Best-of-Many algorithm, and the nested structure of narrow cuts will be important in the analysis. We discuss this in the next lecture.

In this lecture, we discuss a very nice and simple result by Gao (2013) for s-t path TSP in Graph-TSP instances that also exploits the nested structure of narrow cuts.

s-t Path TSP on Graph-TSP Instances

LP Relaxation for s-t path TSP on a Graph-TSP instance

only use edges of cost 1 but you can use them more than once $(x(e) > 1)$

$$\text{Min} \quad \sum_{e \in E} x(e)$$

subject to:

$$x(\delta(S)) \geq \begin{cases} 1, & \forall s\text{-}t \text{ cuts } \delta(S), \\ 2, & \forall \text{non } s\text{-}t \text{ cuts } \delta(S), \end{cases}$$

$$x(e) \geq 0, \qquad \forall e \in E.$$

Let x^* be an optimal LP solution, OPT_{LP} the optimal objective value.

Choosing T

Let $E(x^*) = \{e \in E : x^*(e) > 0\}$ be the support of LP solution x^*. By definition, all edges in $E(x^*)$ have cost 1.

Claim

$OPT_{LP} \geq |V| - 1$.

$$OPT_{LP} = x^*(E) = \frac{1}{2} \sum_{i \in V} x^*(\delta(i)) \geq \frac{1}{2}(2|V| - 2) = |V| - 1$$

Choosing T

Let $E(x^*) = \{e \in E : x^*(e) > 0\}$ be the support of LP solution x^*. By definition, all edges in $E(x^*)$ have cost 1.

Claim

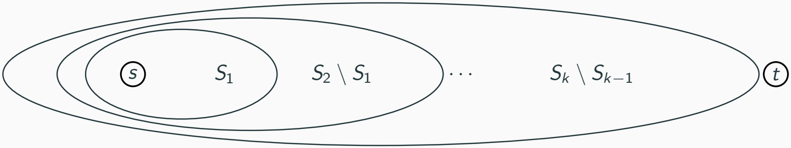
$OPT_{LP} \geq |V| - 1$.

So any spanning tree T in $(V, E(x^*))$ satisfies $c(T) \leq OPT_{LP}$.

Goal: choose a spanning tree in $(V, E(x^*))$ that is odd in each narrow cut, so that $c(T) \leq OPT_{LP}$ and $c(M) \leq \frac{1}{2} OPT_{LP}$ for a minimum-cost W_T-matching M.

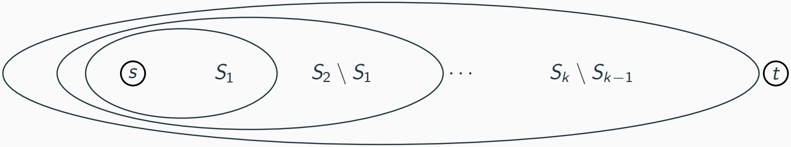
Narrow Cuts

The proof of the preceding theorem still applies, so the narrow cuts are $s \in S_1 \subset S_2 \subset \cdots \subset S_k \subset V$.



Narrow Cuts

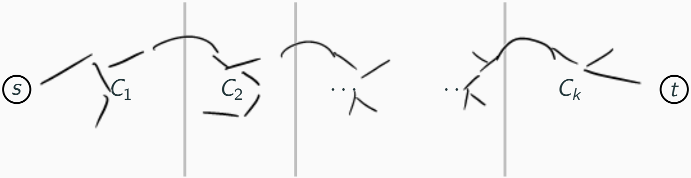
The proof of the preceding theorem still applies, so the narrow cuts are $s \in S_1 \subset S_2 \subset \cdots \subset S_k \subset V$.



Let $S_0 \equiv \emptyset$, $S_{k+1} \equiv V$, $C_i \equiv S_i \setminus S_{i-1}$.

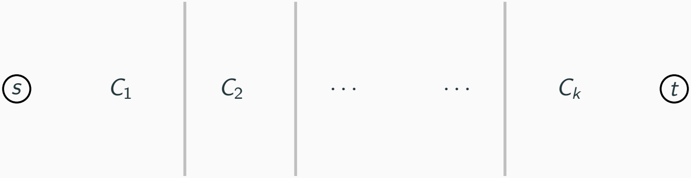
Narrow Cuts

New representation:



Each narrow cut $\delta(S_i)$ is indicated by a gray line; $S_i = \bigcup_{j=1}^{i} C_j$ is all vertices to the left of the line.

Gao Tree



A Gao tree T_{Gao} is a subset of $E(x^*)$ and consists of a spanning tree T_{C_i} on each C_i plus a single edge from C_i to C_{i+1} for every i.

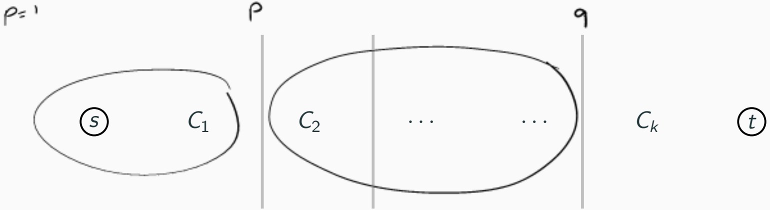
A Gao Tree Exists

Let $H = (V, E(x^*))$ the support graph of x^*, $H(S)$ the graph induced by a set S of vertices.

The fact that a Gao-tree exists follows from the following lemma.

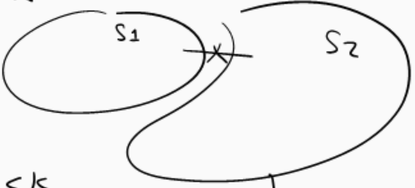
Lemma (Gao (2013))

For $1 \leq p \leq q \leq k$, $H\left(\bigcup_{p \leq i \leq q} C_i\right)$ is connected.



Proof By contradiction

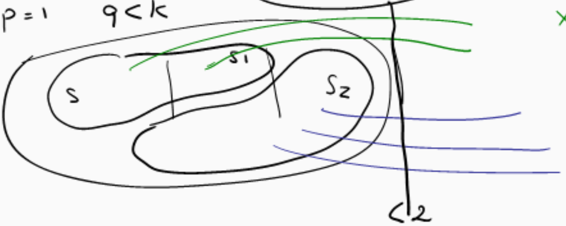
$p = 1, \; q = k$



no edge from S_1 to S_2 with $x^*(e) > 0$

contradicts $x^*(\delta(S_1)) \geq$ 1 if s-t cut; 2 non s-t cut

$p = 1 \quad q < k$

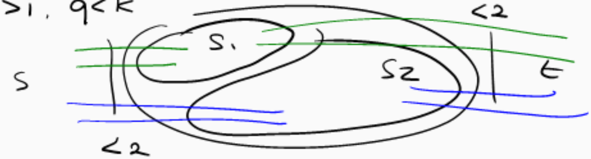


$x^*(\delta(S_1)) \geq 1$

$x^*(\delta(S_2)) \geq 2$

All cross same narrow cut $\rightarrow\leftarrow$

$p > 1, \; q < k$



$x^*(\delta(S_1)) \geq 2$

$x^*(\delta(S_2)) \geq 2$

All cross one of the two narrow cuts indicated $\rightarrow\leftarrow$

Theorem

We just proved the following theorem.

Theorem (Sebő-Vygen (2014), Gao (2013))

There exists a polynomial-time algorithm that finds a solution to any s-t path graph-TSP instance of cost at most $\frac{3}{2} OPT_{LP}$.

- First proved by Sebő and Vygen (2014) (using a completely different technique).

Theorem

Journal version (conference version 2012)

We just proved the following theorem.

Theorem (Sebő-Vygen (2014), Gao (2013))

There exists a polynomial-time algorithm that finds a solution to any s-t path graph-TSP instance of cost at most $\frac{3}{2} OPT_{LP}$.

- First proved by Sebő and Vygen (2014) (using a completely different technique).
- The analysis we saw was due to Gao (2013).

Theorem

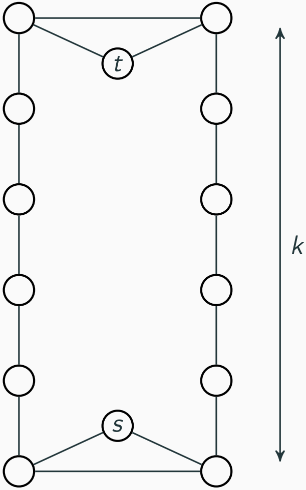
We just proved the following theorem.

Theorem (Sebő-Vygen (2014), Gao (2013))

There exists a polynomial-time algorithm that finds a solution to any s-t path graph-TSP instance of cost at most $\frac{3}{2} OPT_{LP}$.

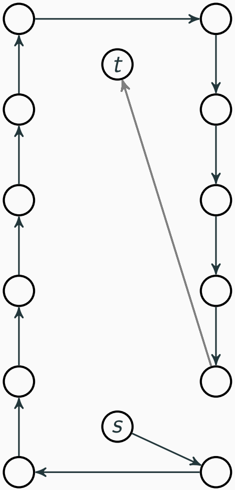
- First proved by Sebő and Vygen (2014) (using a completely different technique).
- The analysis we saw was due to Gao (2013).
- We will finish today by showing the integrality gap $\sup \frac{OPT}{OPT_{LP}}$ is at least $\frac{3}{2}$ for graph-TSP instances of s-t path TSP, which, combined with the result above, shows that the integrality gap is exactly $\frac{3}{2}$ for these instances.

Integrality Gap is At Least $3/2$ for s-t Path TSP



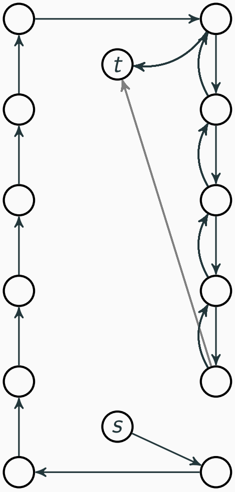
Edges indicated have $c(e) = 1$.
For any i, j not connected by an edge, $c(i, j)$ is the shortest path distance between i and j.

Integrality Gap is At Least $3/2$ for s-t Path TSP



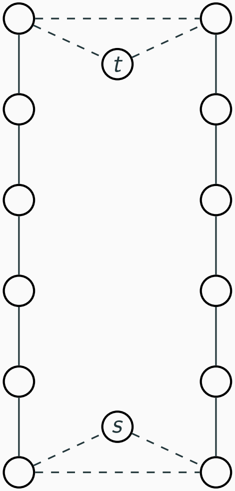
Optimal s-t traveling salesman path has cost $3k - 1$.

Integrality Gap is At Least $3/2$ for s-t Path TSP



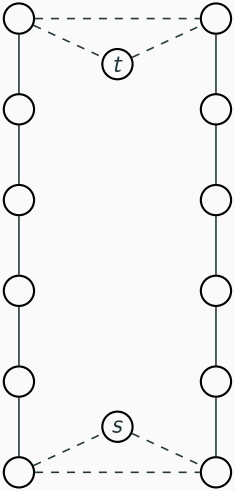
Optimal s-t traveling salesman path has cost $3k - 1$.

Integrality Gap is At Least $3/2$ for s-t Path TSP



Let $x(e) = 1$ for the non-dashed edges, and $x(e) = \frac{1}{2}$ for the dashed edges. This is a feasible (and optimal) solution to the s-t path TSP LP!

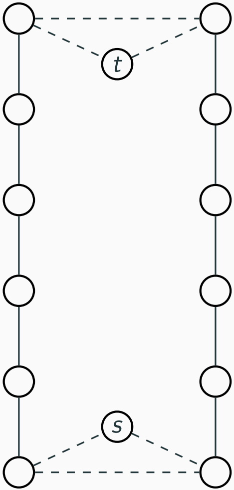
Integrality Gap is At Least $3/2$ for s-t Path TSP



Let $x(e) = 1$ for the non-dashed edges, and $x(e) = \frac{1}{2}$ for the dashed edges. This is a feasible (and optimal) solution to the s-t path TSP LP!

So $OPT_{LP} = 2(k-1) + 6(\frac{1}{2}) = 2k - 1$, whereas $OPT = 3k - 1$.

Integrality Gap is At Least $3/2$ for s-t Path TSP



Let $x(e) = 1$ for the non-dashed edges, and $x(e) = \frac{1}{2}$ for the dashed edges. This is a feasible (and optimal) solution to the s-t path TSP LP!

So $OPT_{LP} = 2(k-1) + 6(\frac{1}{2}) = 2k - 1$, whereas $OPT = 3k - 1$.

$$\sup \frac{OPT}{OPT_{LP}} \geq \lim_{k \to \infty} \frac{3k-1}{2k-1} = \frac{3}{2}.$$