Improving Christofides' Algorithm for the s-t Path TSP

Hyung-Chan An

Joint work with Bobby Kleinberg and David Shmoys

Metric TSP

- Metric (circuit) TSP
  - Given a weighted graph $G = (V, E)$ ($c : E \to \mathbb{R}_+$), find a minimum Hamiltonian circuit
  - Triangle inequality holds
  - Christofides (1976) gave a $3/2$-approximation algorithm

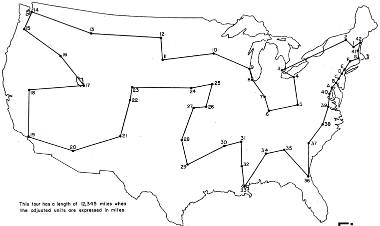


FIG. 16. The optimal tour of 49 cities.

Figure from [Dantzig, Fulkerson, Johnson 1954]

Metric TSP

- Metric (circuit) TSP
  - Given a weighted graph $G = (V, E)$ $(c : E \to \mathbb{R}_+)$, find a minimum Hamiltonian circuit
  - Triangle inequality holds
  - Christofides (1976) gave a $3/2$-approximation algorithm
    - No better performance guarantee known

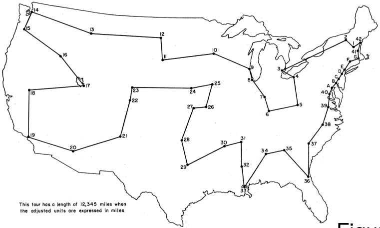


FIG. 16. The optimal tour of 49 cities.

Figure from [Dantzig, Fulkerson, Johnson 1954]

Metric *s*-*t* Path TSP

- Metric *s-t path* TSP
  - Given a weighted graph $G = (V, E)$ $(c : E \to \mathbb{R}_+)$ with *endpoints* $s, t \in V$, find a minimum s-t Hamiltonian *path*
  - Triangle inequality holds
  - Hoogeveen (1991) showed that Christofides' algorithm is a $5/3$-approximation algorithm and this bound is tight

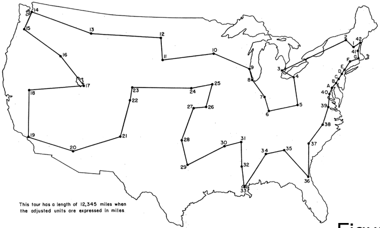


FIG. 16. The optimal tour of 49 cities.

Figure from [Dantzig, Fulkerson, Johnson 1954]

Our Main Result

Theorem

Christofides' algorithm can be improved to yield a deterministic ϕ-approximation algorithm for the s-t path TSP for an arbitrary metric, where $\phi = \frac{1+\sqrt{5}}{2}$ is the golden ratio ($\phi < 1.6181$)

Recent Exciting Improvements

- Recent improvements for unit-weight graphical metric TSP
  - Shortest path metric in an underlying unweighted graph
  - Better approximation than Christofides' ([Oveis Gharan, Saberi, Singh 2011], [Mömke, Svensson 2011], [Mucha 2011], [Sebő, Vygen 2012])

Recent Exciting Improvements

- Recent improvements for unit-weight graphical metric TSP
  - Shortest path metric in an underlying unweighted graph
  - Better approximation than Christofides' ([Oveis Gharan, Saberi, Singh 2011], [Mömke, Svensson 2011], [Mucha 2011], [Sebő, Vygen 2012])
- Our algorithm for the s-t path TSP improves Christofides' for an *arbitrary* metric

Recent Exciting Improvements

- Recent improvements for unit-weight graphical metric TSP
  - Shortest path metric in an underlying unweighted graph
  - Better approximation than Christofides'
    ([Oveis Gharan, Saberi, Singh 2011],
    [Mömke, Svensson 2011], [Mucha 2011],
    [Sebő, Vygen 2012])
  - Techniques can be successfully applied to both variants
- Our algorithm for the s-t path TSP improves Christofides' for an *arbitrary* metric
  - Can our techniques be extended to the circuit variant?

Can Randomization Beat Christofides?

Can Randomization Beat Christofides?

- Find minimum span. tree $\mathscr{T}_{\text{min}}$
- Augment $\mathscr{T}_{\text{min}}$ into a low-cost Eulerian circuit/path
- Transform it into a Hamiltonian circuit/path of no greater cost

Can Randomization Beat Christofides?

- Find minimum span. tree $\mathscr{T}_{\min}$
- Augment $\mathscr{T}_{\min}$ into a low-cost Eulerian circuit/path
- Transform it into a Hamiltonian circuit/path of no greater cost

- Choose *random* span. tree $\mathscr{T}$
- Augment $\mathscr{T}$ into a low-cost Eulerian circuit/path
- Transform it into a Hamiltonian circuit/path of no greater cost

---

- Asadpour, Goemans, Mądry, Oveis Gharan, Saberi 2010:
  - $O(\log n / \log\log n)$-approx for ATSP
- Oveis Gharan, Saberi, Singh 2011
  - Conjectured $(3/2 - \epsilon)$-approx

Can Randomization Beat Christofides?

- Find minimum span. tree $\mathscr{T}_{\min}$
- Augment $\mathscr{T}_{\min}$ into a low-cost Eulerian circuit/path
- Transform it into a Hamiltonian circuit/path of no greater cost

- Choose *random* span. tree $\mathscr{T}$
- Augment $\mathscr{T}$ into a low-cost Eulerian circuit/path
- Transform it into a Hamiltonian circuit/path of no greater cost

---

- Asadpour, Goemans, Mądry, Oveis Gharan, Saberi 2010:
  - $O(\log n / \log\log n)$-approx for ATSP
- Oveis Gharan, Saberi, Singh 2011
  - Conjectured $(3/2 - \epsilon)$-approx
  - Proved for unit-weight graphical metric

Can Randomization Beat Christofides?

- Find minimum span. tree $\mathscr{T}_{\text{min}}$
- Augment $\mathscr{T}_{\text{min}}$ into a low-cost Eulerian circuit/path
- Transform it into a Hamiltonian circuit/path of no greater cost

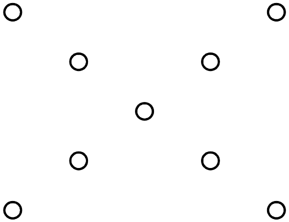
- Choose *random* span. tree $\mathscr{T}$
- Augment $\mathscr{T}$ into a low-cost Eulerian circuit/path
- Transform it into a Hamiltonian circuit/path of no greater cost

---

- Asadpour, Goemans, Mądry, Oveis Gharan, Saberi 2010:
  - $O(\log n / \log\log n)$-approx for ATSP
- Oveis Gharan, Saberi, Singh 2011
  - Conjectured $(3/2 - \epsilon)$-approx
  - Proved for unit-weight graphical metric
- ϕ-approx for *s*-*t* path TSP
  - Arbitrary metric
  - Simpler random choice

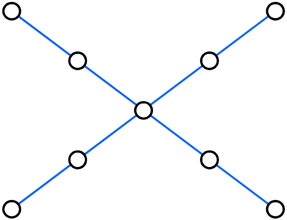
Christofides' Algorithm

- Christofides' algorithm



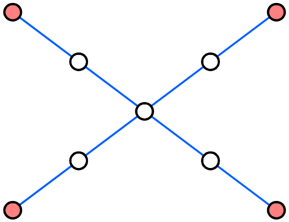
Christofides' Algorithm

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\min}$



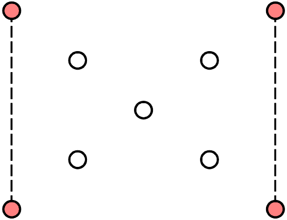
Christofides' Algorithm

- Christofides' algorithm
  - Find a minimum spanning tree $\mathcal{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T is the set of odd-degree vertices in $\mathcal{T}_{\text{min}}$



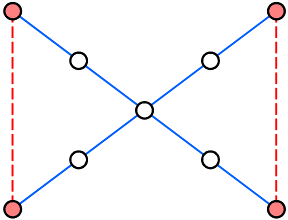
Christofides' Algorithm

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\min}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T is the set of odd-degree vertices in $\mathscr{T}_{\min}$
  - Find a minimum perfect matching M on T



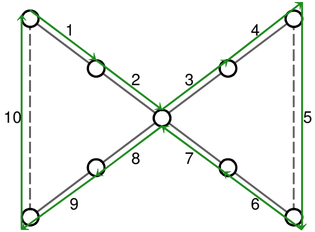
Christofides' Algorithm

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T is the set of odd-degree vertices in $\mathscr{T}_{\text{min}}$
  - Find a minimum perfect matching M on T



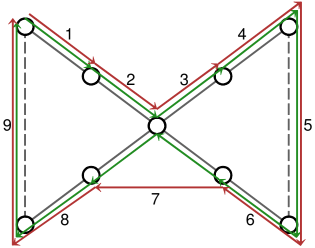
Christofides' Algorithm

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T is the set of odd-degree vertices in $\mathscr{T}_{\text{min}}$
  - Find a minimum perfect matching M on T
  - Find an Eulerian circuit of $\mathscr{T}_{\text{min}} \cup M$



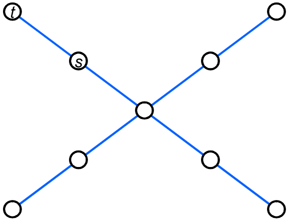
Christofides' Algorithm

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T is the set of odd-degree vertices in $\mathscr{T}_{\text{min}}$
  - Find a minimum perfect matching M on T
  - Find an Eulerian circuit of $\mathscr{T}_{\text{min}} \cup M$
  - Shortcut it into a Hamiltonian circuit H



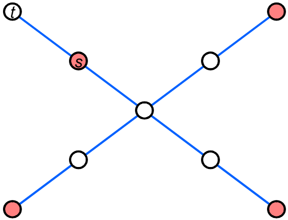
Christofides' Algorithm, for *s*-*t* path TSP

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., *T is the set of even-degree endpoints and other odd-degree vertices in $\mathscr{T}_{\text{min}}$*
  - Find a minimum perfect matching M on T
  - Find an s-t Eulerian *path* of $\mathscr{T}_{\text{min}} \cup M$
  - Shortcut it into an s-t Hamiltonian *path*



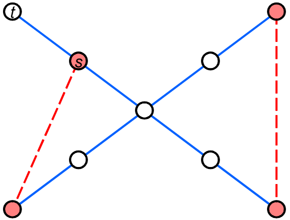
Christofides' Algorithm, for *s*-*t* path TSP

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T *is the set of even-degree endpoints and other odd-degree vertices in* $\mathscr{T}_{\text{min}}$
  - Find a minimum perfect matching M on T
  - Find an *s*-*t* Eulerian *path* of $\mathscr{T}_{\text{min}} \cup M$
  - Shortcut it into an *s*-*t* Hamiltonian *path*



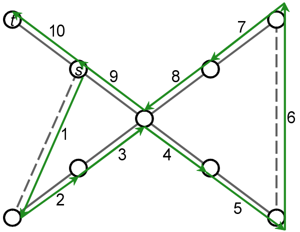
Christofides' Algorithm, for *s*-*t* path TSP

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T *is the set of even-degree endpoints and other odd-degree vertices in* $\mathscr{T}_{\text{min}}$
  - Find a minimum perfect matching M on T
  - Find an *s*-*t* Eulerian *path* of $\mathscr{T}_{\text{min}} \cup M$
  - Shortcut it into an *s*-*t* Hamiltonian *path*



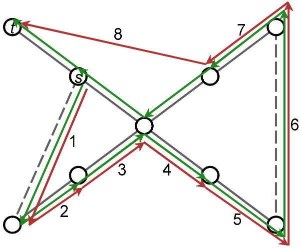
Christofides' Algorithm, for *s*-*t* path TSP

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\text{min}}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T *is the set of even-degree endpoints and other odd-degree vertices in* $\mathscr{T}_{\text{min}}$
  - Find a minimum perfect matching M on T
  - Find an *s*-*t* Eulerian *path* of $\mathscr{T}_{\text{min}} \cup M$
  - Shortcut it into an *s*-*t* Hamiltonian *path*



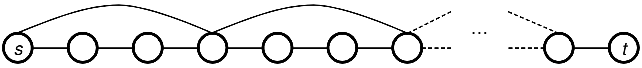
Christofides' Algorithm, for *s*-*t* path TSP

- Christofides' algorithm
  - Find a minimum spanning tree $\mathscr{T}_{\min}$
  - Let T be the set of vertices with "wrong" parity of degree: i.e., T *is the set of even-degree endpoints and other odd-degree vertices in* $\mathscr{T}_{\min}$
  - Find a minimum perfect matching M on T
  - Find an *s*-*t* Eulerian *path* of $\mathscr{T}_{\min} \cup M$
  - Shortcut it into an *s*-*t* Hamiltonian *path*



Path-variant Christofides' algorithm

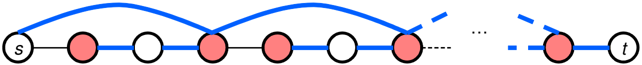
- Path-variant Christofides' algorithm
  - $5/3$-approximation algorithm [Hoogeveen 1991]
  - This bound is tight



  - Unit-weight graphical metric:
    distance between two vertices defined as shortest distance on this underlying unit-weight graph

Path-variant Christofides' algorithm

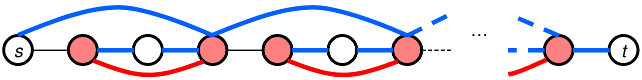
- Path-variant Christofides' algorithm
  - $5/3$-approximation algorithm [Hoogeveen 1991]
  - This bound is tight



  - Unit-weight graphical metric:
    distance between two vertices defined as shortest distance on this underlying unit-weight graph

Path-variant Christofides' algorithm

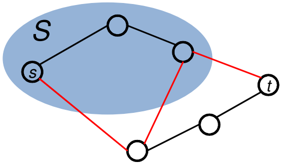
- Path-variant Christofides' algorithm
  - $5/3$-approximation algorithm [Hoogeveen 1991]
  - This bound is tight



  - Unit-weight graphical metric:
    distance between two vertices defined as shortest distance on this underlying unit-weight graph

Held-Karp Relaxation

- Held-Karp relaxation
  - $\delta(S)$ for $S \subsetneq V$ denotes the set of edges in cut $(S, \bar{S})$



  - Incidence vector χ_F of $F \subset E$ is $(\chi_F)_e := \begin{cases} 1 & \text{if } e \in F \\ 0 & \text{otherwise} \end{cases}$

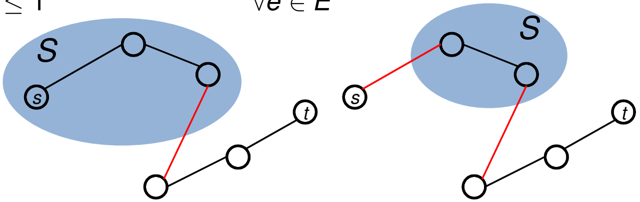
Held-Karp Relaxation

- Held-Karp relaxation

  For $G = (V, E)$ and $s, t \in V$,

$$\begin{cases} \sum_{e\in\delta(\{s\})} x_e = \sum_{e\in\delta(\{t\})} x_e = 1 & \\ \sum_{e\in\delta(\{v\})} x_e = 2, & \forall v \in V \setminus \{s,t\} \\ \sum_{e\in\delta(S)} x_e \geq 1, & \forall S \subsetneq V, |\{s,t\} \cap S| = 1 \\ \sum_{e\in\delta(S)} x_e \geq 2, & \forall S \subsetneq V, |\{s,t\} \cap S| \neq 1, S \neq \emptyset \\ 0 \leq x_e \leq 1 & \forall e \in E \end{cases}$$

$x \in \mathbb{R}^E$



Held-Karp Relaxation

- Polynomial-time solvable
- Feasible region of this LP is contained in the ST polytope

Held-Karp Relaxation

- Polynomial-time solvable
- Feasible region of this LP is contained in the ST polytope
- Held-Karp solution can be written as a convex combination of (incidence vectors of) spanning trees

Held-Karp Relaxation

- Polynomial-time solvable
- Feasible region of this LP is contained in the ST polytope
- Held-Karp solution can be written as a convex combination of (incidence vectors of) spanning trees
- Can find such a decomposition in polynomial time [Grötschel, Lovász, Schrijver 1981]

Our Algorithm

- Best-of-Many Christofides' Algorithm
  - *Compute an optimal solution x^* to the Held-Karp relaxation*
  - *Rewrite x^* as a convex comb. of spanning trees $\mathscr{T}_1, \ldots, \mathscr{T}_k$*
  - For each $\mathscr{T}_i$:
    - Let T_i be the set of vertices with "wrong" parity of degree: i.e., T_i is the set of even-degree endpoints and other odd-degree vertices in $\mathscr{T}_i$
    - Find a minimum perfect matching M_i on T_i
    - Find an s-t Eulerian path of $\mathscr{T}_i \cup M_i$
    - Shortcut it into an s-t Hamiltonian path H_i
  - Output the best Hamiltonian path

Randomized Algorithm

- Sampling Christofides' Algorithm
  - Sample $\mathscr{T}$ by choosing $\mathscr{T}_i$ with probability λ_i $(x^* = \sum_{i=1}^{k} \lambda_i \chi_{\mathscr{T}_i})$

Randomized Algorithm

- Sampling Christofides' Algorithm
  - Sample $\mathscr{T}$ by choosing $\mathscr{T}_i$ with probability λ_i ($x^* = \sum_{i=1}^{k} \lambda_i \chi_{\mathscr{T}_i}$)
- $\mathrm{E}[c(H)] \leq \rho \cdot \mathrm{OPT} \implies$
  Best-of-Many Christofides' Algorithm is ρ-approx. algorithm

Randomized Algorithm

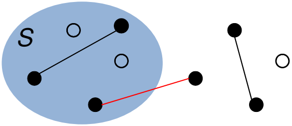
- Sampling Christofides' Algorithm
  - Sample $\mathscr{T}$ by choosing $\mathscr{T}_i$ with probability λ_i $(x^* = \sum_{i=1}^{k} \lambda_i \chi_{\mathscr{T}_i})$
- $\mathrm{E}[c(H)] \leq \rho \cdot \mathrm{OPT} \implies$ Best-of-Many Christofides' Algorithm is ρ-approx. algorithm
- $\Pr[e \in \mathscr{T}] = x_e^*$

Randomized Algorithm

- Sampling Christofides' Algorithm
  - Sample $\mathscr{T}$ by choosing $\mathscr{T}_i$ with probability λ_i $(x^* = \sum_{i=1}^{k} \lambda_i \chi_{\mathscr{T}_i})$
- $\mathrm{E}[c(H)] \leq \rho \cdot \mathrm{OPT} \implies$ Best-of-Many Christofides' Algorithm is ρ-approx. algorithm
- $\Pr[e \in \mathscr{T}] = x_e^*$
  - $\mathrm{E}[c(\mathscr{T})] = \sum_{e \in E} c_e x_e^* = c(x^*)$
  - The rest of the analysis focuses on bounding $c(M)$

Polyhedral Characterization of Matchings

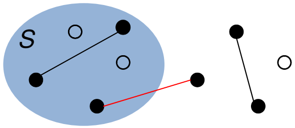
- Polyhedral characterization of matchings on T (assuming triangle inequality) [Edmonds, Johnson 1973]



$$\begin{cases} \sum_{e \in \delta(S)} y_e \geq 1, & \forall S \subset V, |S \cap T| \text{ odd} \\ y \in \mathbb{R}_+^E \end{cases}$$

Polyhedral Characterization of Matchings

- Polyhedral characterization of matchings on T (assuming triangle inequality) [Edmonds, Johnson 1973]



$$\begin{cases} \sum_{e \in \delta(S)} y_e \geq 1, & \forall S \subset V, |S \cap T| \text{ odd} \\ y \in \mathbb{R}^E_+ \end{cases}$$

- Call a feasible solution a *fractional matching*; its cost upper-bounds $c(M)$

Proof of $5/3$-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  $x^* :=$ optimal Held-Karp solution

Proof of $5/3$-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  $x^* :=$ optimal Held-Karp solution
- Take $y := \alpha\chi_{\mathscr{T}} + \beta x^*$ for $\alpha = \beta = \frac{1}{3}$

Proof of $5/3$-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  $x^* :=$ optimal Held-Karp solution
- Take $y := \alpha\chi_{\mathscr{T}} + \beta x^*$ for $\alpha = \beta = \frac{1}{3}$

$$\text{(Matching)} \quad \begin{cases} \displaystyle\sum_{e \in \delta(S)} y_e \geq 1, & \forall S \subset V, |S \cap T| \text{ odd} \\ y \in \mathbb{R}_+^E \end{cases}$$

Proof of 5/3-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  $x^* :=$ optimal Held-Karp solution
- Take $y := \alpha\chi_{\mathscr{T}} + \beta x^*$ for $\alpha = \beta = \frac{1}{3}$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities		1	
LB on nonseparating cut capacities		2	

$$
\text{(Held-Karp)} \quad \begin{cases}
\sum_{e\in\delta(\{s\})} x_e = \sum_{e\in\delta(\{t\})} x_e = 1 & \\
\sum_{e\in\delta(\{v\})} x_e = 2, & \forall v \in V \setminus \{s,t\} \\
\sum_{e\in\delta(S)} x_e \geq 1, & \forall S \subsetneq V, |\{s,t\} \cap S| = 1 \\
\sum_{e\in\delta(S)} x_e \geq 2, & \forall S \subsetneq V, |\{s,t\} \cap S| \neq 1, S \neq \emptyset \\
0 \leq x_e \leq 1 & \forall e \in E
\end{cases}
$$

Proof of $5/3$-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  $x^* :=$ optimal Held-Karp solution
- Take $y := \alpha\chi_{\mathscr{T}} + \beta x^*$ for $\alpha = \beta = \frac{1}{3}$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities	1	1	
LB on nonseparating cut capacities	1	2	

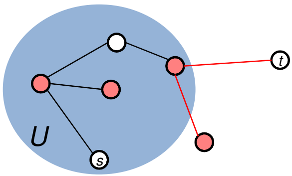
Proof of $5/3$-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  $x^* :=$ optimal Held-Karp solution
- Take $y := \alpha\chi_{\mathscr{T}} + \beta x^*$ for $\alpha = \beta = \frac{1}{3}$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities	2	1	
LB on nonseparating cut capacities	1	2	

Lemma

An s-t cut $(U, \bar{U})$ that is odd w.r.t. T (i.e., $|U \cap T|$ is odd) has at least two tree edges in it



Proof of $5/3$-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  x^* := optimal Held-Karp solution
- Take $y := \alpha\chi_{\mathscr{T}} + \beta x^*$ for $\alpha = \beta = \frac{1}{3}$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities	2	1	$2\alpha + \beta = 1$
LB on nonseparating cut capacities	1	2	$\alpha + 2\beta = 1$

Proof of $5/3$-approximation

- Want: a fractional matching y with $\mathrm{E}[c(y)] \leq \frac{2}{3}c(x^*)$
  $x^* :=$ optimal Held-Karp solution
- Take $y := \alpha\chi_{\mathscr{T}} + \beta x^*$ for $\alpha = \beta = \frac{1}{3}$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities	2	1	$2\alpha + \beta = 1$
LB on nonseparating cut capacities	1	2	$\alpha + 2\beta = 1$

- $\mathrm{E}[c(y)] = \alpha\mathrm{E}[c(\chi_{\mathscr{T}})] + \beta c(x^*) = (\alpha + \beta)c(x^*)$
- $\mathrm{E}[c(H)] \leq \mathrm{E}[c(\mathscr{T})] + \mathrm{E}[c(M)] \leq (1 + \alpha + \beta)c(x^*)$

Theorem

The given algorithm is a $(1 + \alpha + \beta)$*-approximation algorithm*

Improvement upon $5/3$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities	2	1	$2\alpha + \beta$
LB on nonseparating cut capacities	1	2	$\alpha + 2\beta$

- Perturb α and β
  - In particular, decrease α by 2ϵ and increase β by ϵ

Improvement upon $5/3$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities	2	1	$2\alpha + \beta$
LB on nonseparating cut capacities	1	2	$\alpha + 2\beta$

- Perturb α and β
  - In particular, decrease α by 2ϵ and increase β by ϵ
- $E[c(y)] = (\alpha + \beta)c(x^*)$ decreases by $\epsilon c(x^*)$
- $\alpha + 2\beta$ unchanged; nonseparating cuts remain satisfied
- T-odd s-t cuts with small capacity may become violated
  - If violated, by at most $d := O(\epsilon)$

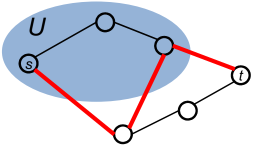
Improvement upon $5/3$

	$\chi_{\mathscr{T}}$	x^*	y
LB on T-odd s-t cut capacities	2	1	$2\alpha + \beta$
LB on nonseparating cut capacities	1	2	$\alpha + 2\beta$

- Perturb α and β
  - In particular, decrease α by 2ϵ and increase β by ϵ
- $E[c(y)] = (\alpha + \beta)c(x^*)$ decreases by $\epsilon c(x^*)$
- $\alpha + 2\beta$ unchanged; nonseparating cuts remain satisfied
- T-odd s-t cuts with small capacity may become violated
  - If violated, by at most $d := O(\epsilon)$

Definition

For $0 < \tau \leq 1$, a τ-narrow cut $(U, \bar{U})$ is an s-t cut with $\sum_{e \in \delta(U)} x^*_e < 1 + \tau$



Improvement upon $5/3$

- τ-narrow cuts may be violated when they are T-odd

Improvement upon $5/3$

- τ-narrow cuts may be violated when they are T-odd

Lemma

For any τ-narrow cut $(U, \bar{U})$, $\Pr[|U \cap T| \text{ odd}] < \tau$

Improvement upon 5/3

- τ-narrow cuts may be violated when they are T-odd

Lemma

For any τ-narrow cut $(U, \bar{U})$, $\Pr[|U \cap T| \text{ odd}] < \tau$

Proof.

- Expected number of tree edges in the cut is $< 1 + \tau$: $\sum_{e \in \delta(U)} \Pr[e \in \mathscr{T}] = \sum_{e \in \delta(U)} x_e^* < 1 + \tau$

$\square$

- $\Pr[e \in \mathscr{T}] = x_e^*$

Improvement upon $5/3$

- τ-narrow cuts may be violated when they are T-odd

Lemma

For any τ-narrow cut $(U, \bar{U})$, $\Pr[|U \cap T| \text{ odd}] < \tau$

Proof.

- Expected number of tree edges in the cut is $< 1 + \tau$: $\sum_{e \in \delta(U)} \Pr[e \in \mathscr{T}] = \sum_{e \in \delta(U)} x_e^* < 1 + \tau$
- $(U, \bar{U})$ has at least one tree edge in it
- If $(U, \bar{U})$ is odd w.r.t. T, it must have another tree edge in it

□

- $\Pr[e \in \mathscr{T}] = x_e^*$

Lemma

An s-t cut $(U, \bar{U})$ that is odd w.r.t. T (i.e., $|U \cap T|$ is odd) has at least two tree edges in it

Improvement upon $5/3$

- τ-narrow cuts may be violated when they are T-odd
- This happens with probability smaller than $\tau = O(\epsilon)$
- When this happens, the cut will have deficiency $d = O(\epsilon)$

Improvement upon $5/3$

- τ-narrow cuts may be violated when they are T-odd
- This happens with probability smaller than $\tau = O(\epsilon)$
- When this happens, the cut will have deficiency $d = O(\epsilon)$
- *Suppose* edge sets of τ-narrow cuts were disjoint

Improvement upon $5/3$

- τ-narrow cuts may be violated when they are T-odd
- This happens with probability smaller than $\tau = O(\epsilon)$
- When this happens, the cut will have deficiency $d = O(\epsilon)$

- *Suppose* edge sets of τ-narrow cuts were disjoint
- $y := \alpha\chi_{\mathscr{T}} + \beta x^* + r$

Improvement upon $5/3$

- τ-narrow cuts may be violated when they are T-odd
- This happens with probability smaller than $\tau = O(\epsilon)$
- When this happens, the cut will have deficiency $d = O(\epsilon)$

- *Suppose* edge sets of τ-narrow cuts were disjoint
- $y := \alpha\chi_{\mathscr{T}} + \beta x^* + r$
- For each e, if e is in a τ-narrow cut that is odd w.r.t. T, set $r_e := dx_e^*$

Claim y is a fractional matching

Claim $\mathrm{E}[c(r)] \leq d\tau c(x^*)$

Improvement upon $5/3$

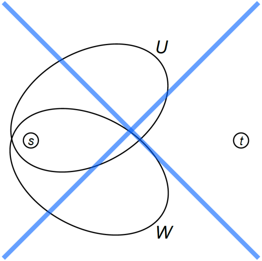
- τ-narrow cuts are not disjoint

Improvement upon $5/3$

- τ-narrow cuts are not disjoint, but "almost" disjoint

Lemma

τ-narrow cuts do not cross: i.e., for τ-narrow cuts $(U, \bar{U})$ and $(W, \bar{W})$ with $s \in U, W$, either $U \subset W$ or $W \subset U$.

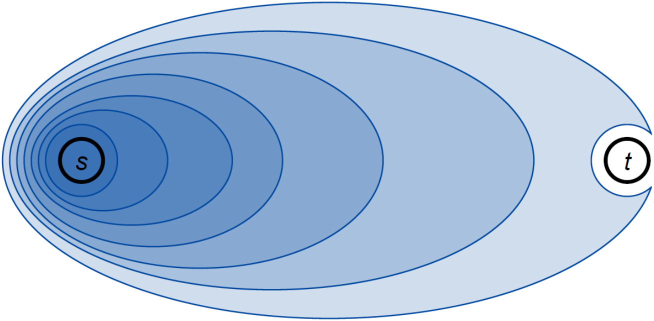


Improvement upon $5/3$

- τ-narrow cuts are not disjoint, but "almost" disjoint

Lemma

τ-narrow cuts do not cross: i.e., for τ-narrow cuts $(U, \bar{U})$ and $(W, \bar{W})$ with $s \in U, W$, either $U \subset W$ or $W \subset U$. Therefore, τ-narrow cuts constitute a layered structure.



Improvement upon $5/3$

- τ-narrow cuts are not disjoint, but "almost" disjoint

Lemma

τ-narrow cuts do not cross: i.e., for τ-narrow cuts $(U, \bar{U})$ and $(W, \bar{W})$ with $s \in U, W$, either $U \subset W$ or $W \subset U$. Therefore, τ-narrow cuts constitute a layered structure.

Lemma

Each τ-narrow cut has a "representative" edge set of capacity $\geq 1 - \frac{\tau}{2}$, and they are mutually disjoint

The Main Result

Theorem

Best-of-many Christofides' algorithm is a deterministic ϕ-approximation algorithm for the s-t path TSP for the general metric, where $\phi = \frac{1+\sqrt{5}}{2} < 1.6181$ *is the golden ratio*

Open Questions

- Circuit TSP
  - Is there a better than $3/2$-approximation algorithm?
  - Do our techniques extend to the circuit TSP?

Thank you.