V. Approx. Algorithms: Travelling Salesman Problem

Thomas Sauerwald

Easter 2019
Outline

Introduction

General TSP

Metric TSP
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

Given: A complete undirected graph \( G = (V, E) \) with nonnegative integer cost \( c(u, v) \) for each edge \((u, v) \in E\).

Goal: Find a hamiltonian cycle of \( G \) with minimum cost.

Formal Definition

Solution space consists of at most \( n! \) possible tours. Actually the right number is \((n-1)!)/2\.

Metric TSP: costs satisfy triangle inequality:
\[
\forall u, v, w \in V: c(u, w) \leq c(u, v) + c(v, w).
\]

Euclidean TSP: cities are points in the Euclidean space, costs are equal to their (rounded) Euclidean distance.
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

Formal Definition

- **Given**: A complete undirected graph \( G = (V, E) \) with nonnegative integer cost \( c(u, v) \) for each edge \( (u, v) \in E \)

---

Special Instances

Even this version is NP hard (Ex. 35.2-2)
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

Formal Definition

- **Given**: A complete undirected graph $G = (V, E)$ with nonnegative integer cost $c(u, v)$ for each edge $(u, v) \in E$
- **Goal**: Find a hamiltonian cycle of $G$ with minimum cost.
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

**Formal Definition**
- **Given**: A complete undirected graph $G = (V, E)$ with nonnegative integer cost $c(u, v)$ for each edge $(u, v) \in E$
- **Goal**: Find a hamiltonian cycle of $G$ with minimum cost.
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

Formal Definition

- **Given**: A complete undirected graph $G = (V, E)$ with nonnegative integer cost $c(u, v)$ for each edge $(u, v) \in E$
- **Goal**: Find a hamiltonian cycle of $G$ with minimum cost.

![Graph Example]

$3 + 2 + 1 + 3 = 9$
The Traveling Salesman Problem (TSP)

*Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.*

Formal Definition

- **Given:** A complete undirected graph \( G = (V, E) \) with nonnegative integer cost \( c(u, v) \) for each edge \( (u, v) \in E \)
- **Goal:** Find a hamiltonian cycle of \( G \) with minimum cost.
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

**Formal Definition**

- **Given**: A complete undirected graph $G = (V, E)$ with nonnegative integer cost $c(u, v)$ for each edge $(u, v) \in E$
- **Goal**: Find a hamiltonian cycle of $G$ with minimum cost.

Solution space consists of at most $n!$ possible tours!

Actual number is $\frac{(n-1)!}{2}$.
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

Formal Definition

- **Given:** A complete undirected graph \( G = (V, E) \) with nonnegative integer cost \( c(u, v) \) for each edge \((u, v) \in E\)
- **Goal:** Find a hamiltonian cycle of \( G \) with minimum cost.

Solution space consists of at most \( n! \) possible tours!

Actually the right number is \((n - 1)!/2\)
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

Formal Definition

- **Given:** A complete undirected graph \( G = (V, E) \) with nonnegative integer cost \( c(u, v) \) for each edge \( (u, v) \in E \)
- **Goal:** Find a hamiltonian cycle of \( G \) with minimum cost.

Solution space consists of at most \( n! \) possible tours!

Actually the right number is \( (n - 1)! / 2 \)

Special Instances

Even this version is NP hard (Ex. 35.2-2)
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

**Formal Definition**
- **Given:** A complete undirected graph $G = (V, E)$ with nonnegative integer cost $c(u, v)$ for each edge $(u, v) \in E$
- **Goal:** Find a hamiltonian cycle of $G$ with minimum cost.

Solution space consists of at most $n!$ possible tours!

Actually the right number is $(n - 1)!/2$

**Special Instances**
- **Metric TSP:** costs satisfy triangle inequality:
  \[
  \forall u, v, w \in V : \quad c(u, w) \leq c(u, v) + c(v, w).
  \]
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

Formal Definition

- **Given**: A complete undirected graph $G = (V, E)$ with nonnegative integer cost $c(u, v)$ for each edge $(u, v) \in E$
- **Goal**: Find a hamiltonian cycle of $G$ with minimum cost.

Solution space consists of at most $n!$ possible tours!

Actually the right number is $(n - 1)!/2$

Special Instances

- **Metric TSP**: costs satisfy triangle inequality:
  \[ \forall u, v, w \in V :\quad c(u, w) \leq c(u, v) + c(v, w). \]
- **Euclidean TSP**: cities are points in the Euclidean space, costs are equal to their (rounded) Euclidean distance
The Traveling Salesman Problem (TSP)

Given a set of cities along with the cost of travel between them, find the cheapest route visiting all cities and returning to your starting point.

**Formal Definition**

- **Given:** A complete undirected graph \( G = (V, E) \) with nonnegative integer cost \( c(u, v) \) for each edge \( (u, v) \in E \)
- **Goal:** Find a hamiltonian cycle of \( G \) with minimum cost.

Solution space consists of at most \( n! \) possible tours!

Actually the right number is \((n - 1)!/2\)

**Special Instances**

- **Metric TSP:** costs satisfy triangle inequality:

\[
\forall u, v, w \in V: \quad c(u, w) \leq c(u, v) + c(v, w).
\]

- **Euclidean TSP:** cities are points in the Euclidean space, costs are equal to their (rounded) Euclidean distance

Even this version is NP hard (Ex. 35.2-2)
Dantzig, Fulkerson and Johnson found an optimal tour through 42 cities.

http://www.math.uwaterloo.ca/tsp/history/img/dantzig_big.html
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable \( x(u, v) = 1 \) iff tour goes between \( u \) and \( v \))
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable $x(u, v) = 1$ iff tour goes between $u$ and $v$)
2. Solve the linear program. If the solution is integral and forms a tour, stop. Otherwise find a new constraint to add (cutting plane)
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable $x(u, v) = 1$ iff tour goes between $u$ and $v$)
2. Solve the linear program. If the solution is integral and forms a tour, stop.
   Otherwise find a new constraint to add (cutting plane)
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable $x(u, v) = 1$ iff tour goes between $u$ and $v$)
2. Solve the linear program. If the solution is integral and forms a tour, stop. Otherwise find a new constraint to add (cutting plane)
The Dantzig-Fulkerson-Johnson Method

1. Create a **linear program** (variable $x(u, v) = 1$ iff tour goes between $u$ and $v$)
2. Solve the linear program. If the solution is integral and forms a tour, stop. Otherwise find a new constraint to add (**cutting plane**)
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable $x(u, v) = 1$ iff tour goes between $u$ and $v$)
2. Solve the linear program. If the solution is integral and forms a tour, stop. Otherwise find a new constraint to add (cutting plane)

![Graph showing linear program constraints and a solution point](image)
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable \( x(u, v) = 1 \) iff tour goes between \( u \) and \( v \))
2. Solve the linear program. If the solution is integral and forms a tour, stop. Otherwise find a new constraint to add (cutting plane)

\[
\begin{align*}
\max \frac{1}{3} x + y \\
2x_1 - 9x_2 &\leq -27 \\
x_2 &\leq 3 \\
4x_1 + 9x_2 &\leq 36
\end{align*}
\]
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable $x(u, v) = 1$ iff tour goes between $u$ and $v$)
2. Solve the linear program. If the solution is integral and forms a tour, stop. Otherwise find a new constraint to add (cutting plane)

$$\begin{align*}
\max & \quad \frac{1}{3}x + y \\
\text{s.t.} & \quad 2x_1 - 9x_2 \leq -27 \\
& \quad x_2 \leq 3 \\
& \quad 4x_1 + 9x_2 \leq 36
\end{align*}$$

Additional constraint to cut the solution space of the LP

V. Travelling Salesman Problem

Introduction
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable \( x(u, v) = 1 \) iff tour goes between \( u \) and \( v \))
2. Solve the linear program. If the solution is integral and forms a tour, stop.
   Otherwise find a new constraint to add (cutting plane)

\[
\begin{align*}
2x_1 - 9x_2 &\leq -27 \\
\max \frac{1}{3}x + y \\
x_2 &\leq 3 \\
4x_1 + 9x_2 &\leq 36
\end{align*}
\]
The Dantzig-Fulkerson-Johnson Method

1. Create a linear program (variable $x(u, v) = 1$ iff tour goes between $u$ and $v$)
2. Solve the linear program. If the solution is integral and forms a tour, stop. Otherwise find a new constraint to add (cutting plane)

V. Travelling Salesman Problem

Introduction
Outline

Introduction

General TSP

Metric TSP
Hardness of Approximation

**Theorem 35.3**

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

---

**Proof:**

Idea: Reduction from the hamiltonian-cycle problem.

Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem. Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:

$$c(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E, \\ \rho |V| + 1 & \text{otherwise.} \end{cases}$$

If $G$ has a hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.

If $G$ does not have a hamiltonian cycle, then any tour $T$ must use some edge $\not\in E$, so

$$c(T) \geq (\rho |V| + 1) + (|V| - 1) = (\rho + 1) |V|.$$ 

Gap of $\rho + 1$ between tours which are using only edges in $G$ and those which don't.

$\rho$-Approximation of TSP in $G'$ computes hamiltonian cycle in $G$ (if one exists).

Large weight will render this edge useless!

Can create representations of $G'$ and $c$ in time polynomial in $|V|$ and $|E|$!
If P ≠ NP, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:
Hardness of Approximation

Theorem 35.3

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Theorem 35.3

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

**Idea:** Reduction from the hamiltonian-cycle problem.

**Proof:**
- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
Hardness of Approximation

Theorem 35.3
If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

**Proof:**
- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem

\[ G = (V, E) \]
Hardness of Approximation

Theorem 35.3

If P ≠ NP, then for any constant ρ ≥ 1, there is no polynomial-time approximation algorithm with approximation ratio ρ for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let G = (V, E) be an instance of the hamiltonian-cycle problem.
- Let G' = (V, E') be a complete graph with costs for each (u, v) ∈ E':

\[ c(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E, \\ ρ |V| + 1 & \text{otherwise}. \end{cases} \]

If G has a hamiltonian cycle H, then (G', c) contains a tour of cost |V|.

If G does not have a hamiltonian cycle, then any tour T must use some edge \( \notin E \), \( \Rightarrow c(T) \geq (ρ |V| + 1) + (|V| - 1) = (ρ + 1)|V| \).

Gap of $ρ + 1$ between tours which are using only edges in G and those which don't.

$ρ$-Approximation of TSP in G' computes hamiltonian cycle in G (if one exists).

Large weight will render this edge useless!

Can create representations of G' and c in time polynomial in |V| and |E|!
Hardness of Approximation

If \( P \neq NP \), then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

**Theorem 35.3**

**Proof:**

**Idea:** Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem.
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):

\[
c(u, v) = \begin{cases} 
1 & \text{if } (u, v) \in E, \\
\rho |V| + 1 & \text{otherwise.}
\end{cases}
\]

If \( G \) has a hamiltonian cycle \( H \), then \((G', c)\) contains a tour of cost \(|V|\).

If \( G \) does not have a hamiltonian cycle, then any tour \( T \) must use some edge \( \notin E \), \( \Rightarrow c(T) \geq (\rho |V| + 1) + (|V| - 1) = (\rho + 1)|V| \).

Gap of \( \rho + 1 \) between tours which are using only edges in \( G \) and those which don't:

- \( \rho \)-Approximation of TSP in \( G' \) computes hamiltonian cycle in \( G \) (if one exists).

Large weight will render this edge useless!

Can create representations of \( G' \) and \( c \) in time polynomial in \(|V| \) and \(|E| \)!
Hardness of Approximation

**Theorem 35.3**

If P \( \neq \) NP, then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

**Proof:**

**Idea:** Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):
  
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]

\[G = (V, E)\]

\[G' = (V, E')\]
Hardness of Approximation

Theorem 35.3

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:

$$c(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E, \\ \rho |V| + 1 & \text{otherwise}. \end{cases}$$

$G = (V, E)$

$G' = (V, E')$
Hardness of Approximation

Theorem 35.3
If P \neq NP, then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):
  \[
c(u, v) = \begin{cases} 
1 & \text{if } (u, v) \in E, \\
\rho |V| + 1 & \text{otherwise.}
\end{cases}
\]

Large weight will render this edge useless!

Reduction

\[ G = (V, E) \]

\[ G' = (V, E') \]
Hardness of Approximation

Theorem 35.3

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:

$$c(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E, \\ \rho |V| + 1 & \text{otherwise.} \end{cases}$$

Can create representations of $G'$ and $c$ in time polynomial in $|V|$ and $|E|$!

V. Travelling Salesman Problem General TSP
Hardness of Approximation

Theorem 35.3

If P \neq NP, then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem.
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):

\[
c(u, v) = \begin{cases} 
1 & \text{if } (u, v) \in E, \\
\rho |V| + 1 & \text{otherwise.}
\end{cases}
\]

\( G = (V, E) \) \hspace{1cm} \text{Reduction} \hspace{1cm} \rho \cdot 4 + 1 \hspace{1cm} G' = (V, E') \)
Hardness of Approximation

**Theorem 35.3**

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

**Proof:**

**Idea:** Reduction from the hamiltonian-cycle problem.

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:
  \[ c(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E, \\ \rho |V| + 1 & \text{otherwise.} \end{cases} \]

- If $G$ has a hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.

\[ G = (V, E) \quad \text{Reduction} \quad G' = (V, E') \]

\[ \rho \cdot 4 + 1 \]
Hardness of Approximation

**Theorem 35.3**

If \( P \neq NP \), then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

**Proof:**


- Let \( G = (V, E) \) be an instance of the Hamiltonian-cycle problem.
- Let \( G' = (V, E') \) be a complete graph with costs for each \( (u, v) \in E' \):
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]

- If \( G \) has a Hamiltonian cycle \( H \), then \( (G', c) \) contains a tour of cost \( |V| \).

\[G = (V, E)\]  
\[\xrightarrow{\text{Reduction}}\]  
\[G' = (V, E')\]  

\[\rho \cdot 4 + 1\]
Hardness of Approximation

If \( P \neq NP \), then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

**Theorem 35.3**

**Proof:**

- **Idea:** Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem.
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]
- If \( G \) has a hamiltonian cycle \( H \), then \((G', c)\) contains a tour of cost \(|V|\).

**Diagram:**

- **Original Graph:** \( G = (V, E) \)
- **Reduction:** \( \rho \cdot 4 + 1 \)
- **New Graph:** \( G' = (V, E') \)
Hardness of Approximation

Theorem 35.3

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:

$$c(u, v) = \begin{cases} 1 & \text{if } (u, v) \in E, \\ \rho |V| + 1 & \text{otherwise.} \end{cases}$$

- If $G$ has a hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$. 

\[ G = (V, E) \quad \xrightarrow{\text{Reduction}} \quad G' = (V, E') \]
Hardness of Approximation

Theorem 35.3

If P \( \neq \) NP, then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

Proof:

**Idea:** Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem.
- Let \( G' = (V, E') \) be a complete graph with costs for each \( (u, v) \in E' \):
  
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise.} 
  \end{cases}
  \]

- If \( G \) has a hamiltonian cycle \( H \), then \( (G', c) \) contains a tour of cost \( |V| \).
- If \( G \) does not have a hamiltonian cycle, then any tour \( T \) must use some edge \( \notin E \).

![Graph Reduction](image)
Hardness of Approximation

Theorem 35.3
If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:


- Let $G = (V, E)$ be an instance of the Hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise.}
  \end{cases}
  \]

- If $G$ has a Hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$
- If $G$ does not have a Hamiltonian cycle, then any tour $T$ must use some edge $\not\in E$,
Theorem 35.3

If P $\neq$ NP, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:
  
  $$c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise.}
  \end{cases}$$

- If $G$ has a hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.
- If $G$ does not have a hamiltonian cycle, then any tour $T$ must use some edge $\not\in E$.

\[\begin{array}{c}
G = (V, E) \\
\begin{array}{c}
\text{Reduction} \\
\rho \cdot 4 + 1
\end{array} \\
G' = (V, E')
\end{array}\]
Theorem 35.3

If \( P \neq NP \), then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem.
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):
  \[
c(u, v) = \begin{cases} 
1 & \text{if } (u, v) \in E, \\
\rho |V| + 1 & \text{otherwise}.
\end{cases}
\]

- If \( G \) has a hamiltonian cycle \( H \), then \((G', c)\) contains a tour of cost \(|V|\).
- If \( G \) does not have a hamiltonian cycle, then any tour \( T \) must use some edge \( \not\in E \).

![Diagram showing reduction from Hamiltonian cycle to TSP]

\( G = (V, E) \)  \( G' = (V, E') \)
Hardness of Approximation

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

**Theorem 35.3**

**Proof:**

**Idea:** Reduction from the hamiltonian-cycle problem.

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]

- If $G$ has a hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.
- If $G$ does not have a hamiltonian cycle, then any tour $T$ must use some edge $\notin E$.

![Graph Reduction](image)
Hardness of Approximation

**Theorem 35.3**

If \( P \neq NP \), then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

**Proof:**

**Idea:** Reduction from the Hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the Hamiltonian-cycle problem.
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):
  \[
  c(u, v) = \begin{cases} 
    1 & \text{if } (u, v) \in E, \\
    \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]

- If \( G \) has a Hamiltonian cycle \( H \), then \((G', c)\) contains a tour of cost \(|V|\).
- If \( G \) does not have a Hamiltonian cycle, then any tour \( T \) must use some edge \( e \notin E \).

![Diagram](image.png)
Hardness of Approximation

Theorem 35.3

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]

- If $G$ has a hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.
- If $G$ does not have a hamiltonian cycle, then any tour $T$ must use some edge $\not\in E$,
  \[
  \Rightarrow c(T) \geq \left(\rho |V| + 1\right) + (|V| - 1)
  \]

\[
G = (V, E) \quad \xrightarrow{\text{Reduction}} \quad G' = (V, E')
\]

- $\rho \cdot 4 + 1$
Hardness of Approximation

Theorem 35.3

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

Proof:

- Let $G = (V, E)$ be an instance of the Hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:
  
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]

- If $G$ has a Hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.
- If $G$ does not have a Hamiltonian cycle, then any tour $T$ must use some edge $\notin E$,

  \[
  \Rightarrow \quad c(T) \geq (\rho |V| + 1) + (|V| - 1) = (\rho + 1)|V|.
  \]
Hardness of Approximation

**Theorem 35.3**

If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

**Proof:**

- Idea: Reduction from the hamiltonian-cycle problem.

- Let $G = (V, E)$ be an instance of the hamiltonian-cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$: 
  
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}.
  \end{cases}
  \]

- If $G$ has a hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.
- If $G$ does not have a hamiltonian cycle, then any tour $T$ must use some edge $\notin E$, 
  
  \[
  \Rightarrow \quad c(T) \geq (\rho |V| + 1) + (|V| - 1) = (\rho + 1)|V|.
  \]

- Gap of $\rho + 1$ between tours which are using only edges in $G$ and those which don’t.
Hardness of Approximation

Theorem 35.3

If \( P \neq NP \), then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the general TSP.

Proof:

Idea: Reduction from the hamiltonian-cycle problem.

- Let \( G = (V, E) \) be an instance of the hamiltonian-cycle problem
- Let \( G' = (V, E') \) be a complete graph with costs for each \((u, v) \in E'\):
  \[
  c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho|V| + 1 & \text{otherwise.}
  \end{cases}
  \]

- If \( G \) has a hamiltonian cycle \( H \), then \((G', c)\) contains a tour of cost \(|V|\)
- If \( G \) does not have a hamiltonian cycle, then any tour \( T \) must use some edge \( \not\in E \),
  \[
  \Rightarrow c(T) \geq (\rho|V| + 1) + (|V| - 1) = (\rho + 1)|V|.
  \]

- Gap of \( \rho + 1 \) between tours which are using only edges in \( G \) and those which don’t
- \( \rho \)-Approximation of TSP in \( G' \) computes hamiltonian cycle in \( G \) (if one exists)

\[
G = (V, E) \quad \text{Reduction} \quad \rho \cdot 4 + 1 \quad \text{G'} = (V, E')
\]
Hardness of Approximation

**Theorem 35.3**
If $P \neq NP$, then for any constant $\rho \geq 1$, there is no polynomial-time approximation algorithm with approximation ratio $\rho$ for the general TSP.

**Proof:**

- Let $G = (V, E)$ be an instance of the Hamiltonian cycle problem.
- Let $G' = (V, E')$ be a complete graph with costs for each $(u, v) \in E'$:
  
  $$c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho|V| + 1 & \text{otherwise}.
  \end{cases}$$

- If $G$ has a Hamiltonian cycle $H$, then $(G', c)$ contains a tour of cost $|V|$.
- If $G$ does not have a Hamiltonian cycle, then any tour $T$ must use some edge $\notin E$,
  
  $$c(T) \geq (\rho|V| + 1) + (|V| - 1) = (\rho + 1)|V|.$$

- Gap of $\rho + 1$ between tours which are using only edges in $G$ and those which don’t.
- $\rho$-Approximation of TSP in $G'$ computes Hamiltonian cycle in $G$ (if one exists) \(\square\)

![Diagram of graph reduction](image)
Proof of Theorem 35.3 from a higher perspective

instances of Hamilton  
instances of TSP
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

instances of Hamilton

instances of TSP
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

All instances with cost $\leq k$

instances of Hamilton

instances of TSP
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

All instances with cost $\leq k$

All instances with cost $> \rho \cdot k$

instances of Hamilton

instances of TSP

V. Travelling Salesman Problem General TSP
Proof of Theorem 35.3 from a higher perspective

All instances with a Hamiltonian cycle

All instances with cost $\leq k$

All instances with cost $> \rho \cdot k$

instances of Hamilton

instances of TSP

V. Travelling Salesman Problem General TSP
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

All instances with cost $\leq k$

All instances with cost $> \rho \cdot k$

instances of Hamilton

instances of TSP
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

All instances with cost $\leq k$

All instances with cost $> \rho \cdot k$

General Method to prove inapproximability results!

instances of Hamilton  \hspace{1cm}  instances of TSP
Outline

Introduction

General TSP

Metric TSP
Metric TSP (TSP Problem with the Triangle Inequality)

Idea: First compute an MST, and then create a tour based on the tree.
**Metric TSP (TSP Problem with the Triangle Inequality)**

Idea: First compute an MST, and then create a tour based on the tree.

\[
\text{APPROX-TSP-TOUR}(G, c)
\]

1: select a vertex \( r \in G. V \) to be a “root” vertex
2: compute a minimum spanning tree \( T_{\text{min}} \) for \( G \) from root \( r \)
3: using \( \text{MST-PRIM}(G, c, r) \)
4: let \( H \) be a list of vertices, ordered according to when they are first visited
5: in a preorder walk of \( T_{\text{min}} \)
6: return the hamiltonian cycle \( H \)
Metric TSP (TSP Problem with the Triangle Inequality)

**Idea:** First compute an MST, and then create a tour based on the tree.

\[
\text{APPROX-TSP-TOUR}(G, c)
\]

1. select a vertex \( r \in G.V \) to be a “root” vertex
2. compute a minimum spanning tree \( T_{\text{min}} \) for \( G \) from root \( r \)
3. using \( \text{MST-PRIM}(G, c, r) \)
4. let \( H \) be a list of vertices, ordered according to when they are first visited
5. in a preorder walk of \( T_{\text{min}} \)
6. **return** the hamiltonian cycle \( H \)

Runtime is dominated by \( \text{MST-PRIM} \), which is \( \Theta(V^2) \).
Metric TSP (TSP Problem with the Triangle Inequality)

Idea: First compute an MST, and then create a tour based on the tree.

\textsc{Approx-Tsp-Tour}(G, c)
1: select a vertex \( r \in G \cdot V \) to be a “root” vertex
2: compute a minimum spanning tree \( T_{\text{min}} \) for \( G \) from root \( r \)
3: using \textsc{Mst-Prim}(G, c, r)
4: let \( H \) be a list of vertices, ordered according to when they are first visited
5: in a preorder walk of \( T_{\text{min}} \)
6: \textbf{return} the hamiltonian cycle \( H \)

Runtime is dominated by \textsc{Mst-Prim}, which is \( \Theta(V^2) \).

Remember: In the Metric-TSP problem, \( G \) is a complete graph.
Run of APPROX-TSP-TOUR

Solution has cost \( \approx 19.704 \) - not optimal!
Better solution, yet still not optimal!
This is the optimal solution (cost \( \approx 14.715 \)).

1. Compute MST 
2. Perform preorder walk on MST 
3. Return list of vertices according to the preorder tree walk
Run of APPROX-TSP-TOUR

1. Compute MST $T_{\text{min}}$

Solution has cost $\approx 19.704$ - not optimal!

Better solution, yet still not optimal!

This is the optimal solution (cost $\approx 14.715$).
Run of APPROX-TSP-TOUR

1. Compute MST $T_{\text{min}}$
Run of APPROX-TSP-TOUR

1. Compute MST $T_{min}$ ✓

Solution has cost $\approx 19.704$ - not optimal!
Better solution, yet still not optimal!
This is the optimal solution (cost $\approx 14.715$).

V. Travelling Salesman Problem

Metric TSP
Run of **APPROX-TSP-TOUR**

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$

Solution has cost $\approx 19.704$ - not optimal!

Better solution, yet still not optimal!

This is the optimal solution (cost $\approx 14.715$).
Run of APPROX-TSP-TOUR

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
Run of APPROX-TSP-TOUR

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk

Solution has cost $\approx 19.704$ - not optimal!
Better solution, yet still not optimal!
This is the optimal solution (cost $\approx 14.715$).
Run of **APPROX-TSP-TOUR**

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk

Solution has cost $\approx 19.704$ - not optimal!

Better solution, yet still not optimal!

This is the optimal solution (cost $\approx 14.715$).
Run of **APPROX-TSP-TOUR**

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk

| V. Travelling Salesman Problem | Metric TSP | 11 |
Run of **APPROX-Tsp-TOUR**

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk

Solution has cost $\approx 19.704$ - not optimal!
Better solution, yet still not optimal!
This is the optimal solution (cost $\approx 14.715$).
Run of \textbf{APPROX-TSP-TOUR}

1. Compute MST $T_{\text{min}} \checkmark$
2. Perform preorder walk on MST $T_{\text{min}} \checkmark$
3. Return list of vertices according to the preorder tree walk

\begin{itemize}
  \item Solution has cost $\approx 19.704$ - not optimal!
  \item Better solution, yet still not optimal!
  \item This is the optimal solution (cost $\approx 14.715$).
\end{itemize}
Run of **APPROX-TSP-TOUR**

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk

Solution has cost $\approx 19.704$ - not optimal!
Better solution, yet still not optimal!
This is the optimal solution (cost $\approx 14.715$).
Run of APPROX-TSP-TOUR

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk

Solution has cost $\approx 19.704$ - not optimal!

Better solution, yet still not optimal!

This is the optimal solution (cost $\approx 14.715$).
Run of APPROX-TSP-TOUR

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk

Solution has cost $\approx 19.704$ - not optimal!
Better solution, yet still not optimal!
This is the optimal solution (cost $\approx 14.715$).
Run of \textbf{APPROX-TSP-TOUR}

1. Compute \text{MST} \ T_{\text{min}}  
2. Perform preorder walk on \text{MST} \ T_{\text{min}}  
3. Return list of vertices according to the preorder tree walk  

\begin{itemize}
  \item Solution has cost $\approx 19.704$ - not optimal!
  \item Better solution, yet still not optimal!
  \item This is the optimal solution (cost $\approx 14.715$).
\end{itemize}
Run of APPROX-TSP-TOUR

Solution has cost $\approx 19.704$ - not optimal!

1. Compute MST $T_{\text{min}} \checkmark$
2. Perform preorder walk on MST $T_{\text{min}} \checkmark$
3. Return list of vertices according to the preorder tree walk $\checkmark$
Run of APPROX-TSP-TOUR

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk ✓
Run of APPROX-TSP-TOUR

Better solution, yet still not optimal!

1. Compute MST $T_{\text{min}}$ ✓
2. Perform preorder walk on MST $T_{\text{min}}$ ✓
3. Return list of vertices according to the preorder tree walk ✓
Run of **APPROX-TSP-TOUR**

1. Compute MST $T_{\text{min}} \checkmark$
2. Perform preorder walk on MST $T_{\text{min}} \checkmark$
3. Return list of vertices according to the preorder tree walk \checkmark
Run of **APPROX-TSP-TOUR**

This is the optimal solution (cost \(\approx 14.715\)).

1. Compute MST \(T_{\text{min}}\) ✓
2. Perform preorder walk on MST \(T_{\text{min}}\) ✓
3. Return list of vertices according to the preorder tree walk ✓
Approximate Solution: Objective 921
Optimal Solution: Objective 699

V. Travelling Salesman Problem

Metric TSP
Proof of the Approximation Ratio

**Theorem 35.2**

`APPROX-TSP-TOUR` is a polynomial-time \( 2 \)-approximation for the traveling-salesman problem with the triangle inequality.
Proof of the Approximation Ratio

Theorem 35.2

\textsc{Approx-TSP-Tour} is a polynomial-time \textit{2-approximation} for the traveling-salesman problem with the triangle inequality.

Proof:
Proof of the Approximation Ratio

Theorem 35.2

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:

- Consider the optimal tour $H^*$ and remove an arbitrary edge $\Rightarrow$ yields a spanning tree $T$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
- Full walk traverses every edge exactly twice, so $c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)$
- Deleting duplicate vertices from $W$ yields a tour $H$
- $c(H) \leq c(W) \leq 2c(H^*)$

Exploiting that all edge costs are non-negative!

Exploiting triangle inequality!
Proof of the Approximation Ratio

Theorem 35.2

\texttt{APPROX-TSP-TOUR} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:

Consider the optimal tour $H^*$ and remove an arbitrary edge $\Rightarrow$ yields a spanning tree $T$ and let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits) $\Rightarrow$ full walk traverses every edge exactly twice, so $c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)$.

Deleting duplicate vertices from $W$ yields a tour $H$ of $\text{APPROX-TSP}$ $c(H) \leq c(W) \leq 2c(H^*)$ exploiting that all edge costs are non-negative! exploiting triangle inequality!

solution $H$ of $\text{APPROX-TSP}$

optimal solution $H^*$
Proof of the Approximation Ratio

**Theorem 35.2**

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:
- Consider the optimal tour $H^*$ and remove an arbitrary edge.
**Proof of the Approximation Ratio**

**Theorem 35.2**

\textsc{Approx-Tsp-Tour} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge.

![Diagram](image)

- Solution $H$ of \textsc{Approx-Tsp}.
- Spanning tree $T$ as a subset of $H^*$.
Proof of the Approximation Ratio

**Theorem 35.2**

\textsc{Approx-Tsp-Tour} is a polynomial-time \textit{2-approximation} for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge
- \Rightarrow yields a \textit{spanning tree} $T$ and

\[ c(W) = 2c(T_{\min}) \leq 2c(H^*) \]

Deleting duplicate vertices from $W$ yields a tour $H$

\[ c(H) \leq c(W) \leq 2c(H^*) \]

exploiting that all edge costs are non-negative!
Proof of the Approximation Ratio

Theorem 35.2

\texttt{APPROX-TSP-TOUR} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:

- Consider the optimal tour $H^*$ and remove an arbitrary edge

\[ \Rightarrow \text{ yields a spanning tree } T \text{ and } c(T) \leq c(H^*) \]
**Proof of the Approximation Ratio**

**Theorem 35.2**

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge
  \[ \Rightarrow \] yields a spanning tree $T$ and $c(T) \leq c(H^*)$

- Exploiting that all edge costs are non-negative!

---

**Solution $H$ of APPROX-TSP**

**Spanning tree $T$ as a subset of $H^*$**
Proof of the Approximation Ratio

**Theorem 35.2**

\textsc{APPROX-TSP-TOUR} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge
- \( \Rightarrow \) yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
Proof of the Approximation Ratio

**Theorem 35.2**

\textsc{Approx-TSP-Tour} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge
  \[ \Rightarrow \text{yields a spanning tree } T \text{ and } c(T) \leq c(H^*) \]
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)

![Diagram](image.png)

- minimum spanning tree $T_{\text{min}}$
- optimal solution $H^*$
Proof of the Approximation Ratio

Theorem 35.2

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:
- Consider the optimal tour $H^*$ and remove an arbitrary edge
  $\Rightarrow$ yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)

Walk $W = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a)$

optimal solution $H^*$
Proof of the Approximation Ratio

**Theorem 35.2**

\textsc{Approx-Tsp-Tour} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:

- Consider the optimal tour $H^*$ and remove an arbitrary edge
  \Rightarrow yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
  \Rightarrow Full walk traverses every edge exactly twice, so

Walk $W = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a)$

optimal solution $H^*$
Proof of the Approximation Ratio

**Theorem 35.2**

**APP-SP** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:
- Consider the optimal tour $H^*$ and remove an arbitrary edge
  $\Rightarrow$ yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{min}$ (including repeated visits)
  $\Rightarrow$ Full walk traverses every edge exactly twice, so
  \[ c(W) = 2c(T_{min}) \]

Walk $W = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a)$

optimal solution $H^*$
**Theorem 35.2**

\textsc{APPROX-TSP-TOUR} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**
- Consider the optimal tour \( H^* \) and remove an arbitrary edge
  \( \Rightarrow \) yields a spanning tree \( T \) and \( c(T) \leq c(H^*) \)
- Let \( W \) be the full walk of the minimum spanning tree \( T_{\text{min}} \) (including repeated visits)
  \( \Rightarrow \) Full walk traverses every edge exactly twice, so
  \[ c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*) \]

Walk \( W = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a) \)  
Optimal solution \( H^* \)
**Proof of the Approximation Ratio**

**Theorem 35.2**

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge.

  $\Rightarrow$ yields a spanning tree $T$ and $c(T) \leq c(H^*)$

- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits).

  $\Rightarrow$ Full walk traverses every edge exactly twice, so

  $c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)$

- Deleting duplicate vertices from $W$ yields a tour $H$

---

**Walk** $W = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a)$

**Optimal solution** $H^* = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a)$
Proof of the Approximation Ratio

**Theorem 35.2**

APPROX-TSP-TOUR is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge
  - yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
  - Full walk traverses every edge exactly twice, so
    \[
    c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)
    \]
- Deleting duplicate vertices from $W$ yields a tour $H$

Walk $W = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a)$

Optimal solution $H^*$
Proof of the Approximation Ratio

**Theorem 35.2**

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**
- Consider the optimal tour $H^*$ and remove an arbitrary edge
  $\Rightarrow$ yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{min}$ (including repeated visits)
  $\Rightarrow$ Full walk traverses every edge exactly twice, so
  $c(W) = 2c(T_{min}) \leq 2c(T) \leq 2c(H^*)$
- Deleting duplicate vertices from $W$ yields a tour $H$

Walk $W = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a)$

optimal solution $H^*$
**Theorem 35.2**

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:
- Consider the optimal tour $H^*$ and remove an arbitrary edge
  - yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
  - Full walk traverses every edge exactly twice, so
    
    $$c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)$$
- Deleting duplicate vertices from $W$ yields a tour $H$

![Diagram of a graph showing nodes and edges, with a tour and optimal solution highlighted.](image)
Proof of the Approximation Ratio

**Theorem 35.2**

\textsc{APPROX-TSP-TOUR} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour \( H^* \) and remove an arbitrary edge
  \( \Rightarrow \) yields a spanning tree \( T \) and \( c(T) \leq c(H^*) \)
- Let \( W \) be the full walk of the minimum spanning tree \( T_{\text{min}} \) (including repeated visits)
  \( \Rightarrow \) Full walk traverses every edge exactly twice, so
  \[ c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*) \]
- Deleting duplicate vertices from \( W \) yields a tour \( H \) with smaller cost:

![Diagram](image)

Tour \( H = (a, b, c, h, d, e, f, g, a) \)

optimal solution \( H^* \)
**Theorem 35.2**

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**
- Consider the optimal tour $H^*$ and remove an arbitrary edge
  - yields a spanning tree $T$ and $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
  - Full walk traverses every edge exactly twice, so
    \[ c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*) \]
- Deleting duplicate vertices from $W$ yields a tour $H$ with smaller cost:
  \[ c(H) \leq c(W) \]

Tour $H = (a, b, c, h, d, e, f, g, a)$

optimal solution $H^*$
Proof of the Approximation Ratio

**Theorem 35.2**

\textsc{APPROX-TSP-TOUR} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

\textbf{Proof:}

- Consider the optimal tour \(H^*\) and remove an arbitrary edge
  \[\Rightarrow\] yields a spanning tree \(T\) and \(c(T) \leq c(H^*)\)
- Let \(W\) be the full walk of the minimum spanning tree \(T_{\text{min}}\) (including repeated visits)
  \[\Rightarrow\] Full walk traverses every edge exactly twice, so
  \[c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)\]
- Deleting duplicate vertices from \(W\) yields a tour \(H\) with smaller cost:
  \[c(H) \leq c(W) \leq 2c(H^*)\]

```
V. Travelling Salesman Problem
Metric TSP
14
```
**Proof of the Approximation Ratio**

**Theorem 35.2**

\( \text{APPROX-TSP-TOUR} \) is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Proof:
- Consider the optimal tour \( H^* \) and remove an arbitrary edge

\[ \Rightarrow \text{yields a spanning tree } T \text{ and } c(T) \leq c(H^*) \]
- Let \( W \) be the full walk of the minimum spanning tree \( T_{\text{min}} \) (including repeated visits)

\[ \Rightarrow \text{Full walk traverses every edge exactly twice, so} \]

\[ c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*) \]

- Deleting duplicate vertices from \( W \) yields a tour \( H \) with smaller cost:

\[ c(H) \leq c(W) \leq 2c(H^*) \]
Proof of the Approximation Ratio

Theorem 35.2

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

**Proof:**

- Consider the optimal tour $H^*$ and remove an arbitrary edge.
  \[ \Rightarrow \text{ yields a spanning tree } T \text{ and } c(T) \leq c(H^*) \]
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits).
  \[ \Rightarrow \text{ Full walk traverses every edge exactly twice, so } c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*) \]

- Deleting duplicate vertices from $W$ yields a tour $H$ with smaller cost:
  \[ c(H) \leq c(W) \leq 2c(H^*) \]

![Diagram](image)

Tour $H = (a, b, c, h, d, e, f, g, a)$

optimal solution $H^*$
Christofides Algorithm

Theorem 35.2

\textsc{APPROX-TSP-TOUR} is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.
Christofides Algorithm

**Theorem 35.2**

APPROX-TSP-TOUR is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Can we get a better approximation ratio?
Christofides Algorithm

**Theorem 35.2**

**APPROX-TSP-TOUR** is a polynomial-time \( 2 \)-approximation for the traveling-salesman problem with the triangle inequality.

Can we get a better approximation ratio?

**CHRISTOFIDES** \((G, c)\)

1: select a vertex \( r \in G. V \) to be a “root” vertex
2: compute a minimum spanning tree \( T_{\text{min}} \) for \( G \) from root \( r \)
3: using MST-PRIM\((G, c, r)\)
4: compute a perfect matching \( M_{\text{min}} \) with minimum weight in the complete graph
5: over the odd-degree vertices in \( T_{\text{min}} \)
6: let \( H \) be a list of vertices, ordered according to when they are first visited
7: in a Eulearian circuit of \( T_{\text{min}} \cup M_{\text{min}} \)
8: return the hamiltonian cycle \( H \)
Christofides Algorithm

Theorem 35.2

**APPROX-TSP-TOUR** is a polynomial-time 2-approximation for the traveling-salesman problem with the triangle inequality.

Can we get a better approximation ratio?

**CHRISTOFIDES**$(G, c)$
1: select a vertex $r \in G.V$ to be a “root” vertex
2: compute a minimum spanning tree $T_{\text{min}}$ for $G$ from root $r$
3: using $\text{MST-PRIM}(G, c, r)$
4: compute a perfect matching $M_{\text{min}}$ with minimum weight in the complete graph
5: over the odd-degree vertices in $T_{\text{min}}$
6: let $H$ be a list of vertices, ordered according to when they are first visited
7: in a Eulearian circuit of $T_{\text{min}} \cup M_{\text{min}}$
8: return the hamiltonian cycle $H$

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.
Run of Christofides

V. Travelling Salesman Problem

Solution has cost $\approx 15.54$ within $10\%$ of the optimum!

1. Compute MST $T_{\min}$

2. Add a minimum-weight perfect matching $M_{\min}$ of the odd vertices in $T_{\min}$

3. Find an Eulerian Circuit in $T_{\min} \cup M_{\min}$

4. Transform the Circuit into a Hamiltonian Cycle

All vertices in $T_{\min} \cup M_{\min}$ have even degree!
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$

Solution has cost $\approx 15.54$ - within $10\%$ of the optimum!
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$

Solution has cost $\approx 15.54$ within $10\%$ of the optimum!
Run of **CHRISTOFIDES**

1. Compute MST $T_{\text{min}}$ ✓

Solution has cost $\approx 15.54$ within 10% of the optimum!
1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$
1. Compute MST $T_{\text{min}}$
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$
1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$
1. Compute MST $T_{\text{min}} \checkmark$
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$
Run of CHRISTOFIDES

1. Compute MST $T_{min}$
2. Add a minimum-weight perfect matching $M_{min}$ of the odd vertices in $T_{min}$
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$

All vertices in $T_{\text{min}} \cup M_{\text{min}}$ have even degree!
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$

All vertices in $T_{\text{min}} \cup M_{\text{min}}$ have even degree!
Run of **CHRISTOFIDES**

1. Compute MST $T_{\text{min}} \checkmark$
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}} \checkmark$
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}} \checkmark$
4. Transform the Circuit into a Hamiltonian Cycle
Run of **CHRISTOFIDES**

1. Compute MST $T_{\text{min}}$
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ within 10% of the optimum!
Run of Christofides

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ within $10\%$ of the optimum!
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ within 10% of the optimum!
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ - within 10% of the optimum!
Run of *CHRISTOFIDES*

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ - within 10% of the optimum!
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ within 10\% of the optimum!
Run of Christofides

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ - within 10% of the optimum!
Run of CHRISTOFIDES

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle
Run of Christofides

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle ✓
Run of CHRISTOFIDES

Solution has cost $\approx 15.54$ - within 10% of the optimum!

1. Compute MST $T_{\text{min}}$ ✓
2. Add a minimum-weight perfect matching $M_{\text{min}}$ of the odd vertices in $T_{\text{min}}$ ✓
3. Find an Eulerian Circuit in $T_{\text{min}} \cup M_{\text{min}}$ ✓
4. Transform the Circuit into a Hamiltonian Cycle ✓
Proof of the Approximation Ratio

Theorem (Christofides’76)
There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.
Proof of the Approximation Ratio

Theorem (Christofides’76)

There is a polynomial-time \( \frac{3}{2} \)-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):
Proof of the Approximation Ratio

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):

- As before, let $H^*$ denote the optimal tour

Proof is quite similar to the previous analysis
Proof of the Approximation Ratio

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):

- As before, let $H^*$ denote the optimal tour.
- The Eulerian Circuit $W$ uses each edge of the minimum spanning tree $T_{\text{min}}$ and the minimum-weight matching $M_{\text{min}}$ exactly once:

$$c(W) \leq c(H^*) + c(M_{\text{min}}) \leq c(H^*) + \frac{1}{2} c(H^*) = \frac{3}{2} c(H^*).$$

(1)
Proof of the Approximation Ratio

There is a polynomial-time \( \frac{3}{2} \)-approximation algorithm for the travelling salesman problem with the triangle inequality.

**Theorem (Christofides’76)**

Proof is quite similar to the previous analysis

**Proof (Approximation Ratio):**

- As before, let \( H^* \) denote the optimal tour
- The Eulerian Circuit \( W \) uses each edge of the minimum spanning tree \( T_{\text{min}} \) and the minimum-weight matching \( M_{\text{min}} \) exactly once:

\[
c(W) = c(T_{\text{min}}) + c(M_{\text{min}}) \leq c(H^*) + c(M_{\text{min}})
\]  

(1)
Proof of the Approximation Ratio

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):

Proof is quite similar to the previous analysis

- As before, let $H^*$ denote the optimal tour
- The Eulerian Circuit $W$ uses each edge of the minimum spanning tree $T_{\text{min}}$ and the minimum-weight matching $M_{\text{min}}$ exactly once:
  \[
  c(W) = c(T_{\text{min}}) + c(M_{\text{min}}) \leq c(H^*) + c(M_{\text{min}})
  \] (1)
- Let $H_{\text{odd}}^*$ be an optimal tour on the odd-degree vertices in $T_{\text{min}}$
Proof of the Approximation Ratio

Theorem (Christofides’76)
There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):
Proof is quite similar to the previous analysis

- As before, let $H^*$ denote the optimal tour
- The Eulerian Circuit $W$ uses each edge of the minimum spanning tree $T_{\text{min}}$ and the minimum-weight matching $M_{\text{min}}$ exactly once:
  \[ c(W) = c(T_{\text{min}}) + c(M_{\text{min}}) \leq c(H^*) + c(M_{\text{min}}) \]  
  \[ (1) \]
- Let $H_{\text{odd}}^*$ be an optimal tour on the odd-degree vertices in $T_{\text{min}}$
- Taking edges alternately, we obtain two matchings $M_1$ and $M_2$ such that $c(M_1) + c(M_2) = c(H_{\text{odd}}^*)$
Proof of the Approximation Ratio

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):

Proof is quite similar to the previous analysis

- As before, let $H^*$ denote the optimal tour
- The Eulerian Circuit $W$ uses each edge of the minimum spanning tree $T_{\text{min}}$ and the minimum-weight matching $M_{\text{min}}$ exactly once:

$$c(W) = c(T_{\text{min}}) + c(M_{\text{min}}) \leq c(H^*) + c(M_{\text{min}})$$  \hspace{1cm} (1)

- Let $H^*_{\text{odd}}$ be an optimal tour on the odd-degree vertices in $T_{\text{min}}$
- Taking edges alternately, we obtain two matchings $M_1$ and $M_2$ such that $c(M_1) + c(M_2) = c(H^*_{\text{odd}})$
- By shortcutting and the triangle inequality,
Proof of the Approximation Ratio

Theorem (Christofides’76)
There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):

- As before, let $H^*$ denote the optimal tour.
- The Eulerian Circuit $W$ uses each edge of the minimum spanning tree $T_{\text{min}}$ and the minimum-weight matching $M_{\text{min}}$ exactly once:

$$c(W) = c(T_{\text{min}}) + c(M_{\text{min}}) \leq c(H^*) + c(M_{\text{min}}) \quad (1)$$

- Let $H^*_\text{odd}$ be an optimal tour on the odd-degree vertices in $T_{\text{min}}$.
- Taking edges alternately, we obtain two matchings $M_1$ and $M_2$ such that

$$c(M_1) + c(M_2) = c(H^*_\text{odd})$$

- By shortcutting and the triangle inequality,

$$c(M_{\text{min}}) \leq \frac{1}{2} c(H^*_\text{odd}) \leq \frac{1}{2} c(H^*). \quad (2)$$
Proof of the Approximation Ratio

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):

- As before, let $H^*$ denote the optimal tour.
- The Eulerian Circuit $W$ uses each edge of the minimum spanning tree $T_{\text{min}}$ and the minimum-weight matching $M_{\text{min}}$ exactly once:
  \[
  c(W) = c(T_{\text{min}}) + c(M_{\text{min}}) \leq c(H^*) + c(M_{\text{min}})
  \]  
  \[\text{(1)}\]

- Let $H^*_{\text{odd}}$ be an optimal tour on the odd-degree vertices in $T_{\text{min}}$.
- Taking edges alternately, we obtain two matchings $M_1$ and $M_2$ such that
  \[
  c(M_1) + c(M_2) = c(H^*_{\text{odd}})
  \]
  \[
  \text{By shortcutting and the triangle inequality,}
  \]
  \[
  c(M_{\text{min}}) \leq \frac{1}{2} c(H^*_{\text{odd}}) \leq \frac{1}{2} c(H^*)
  \]
  \[\text{(2)}\]

- Combining 1 with 2 yields

Proof is quite similar to the previous analysis.
Proof of the Approximation Ratio

Theorem (Christofides’76)
There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Proof (Approximation Ratio):

- As before, let $H^*$ denote the optimal tour.
- The Eulerian Circuit $W$ uses each edge of the minimum spanning tree $T_{\text{min}}$ and the minimum-weight matching $M_{\text{min}}$ exactly once:
  \[ c(W) = c(T_{\text{min}}) + c(M_{\text{min}}) \leq c(H^*) + c(M_{\text{min}}) \]  
  \[ (1) \]

- Let $H^*_{\text{odd}}$ be an optimal tour on the odd-degree vertices in $T_{\text{min}}$.
- Taking edges alternately, we obtain two matchings $M_1$ and $M_2$ such that
  \[ c(M_1) + c(M_2) = c(H^*_{\text{odd}}) \]
- By shortcutting and the triangle inequality,
  \[ c(M_{\text{min}}) \leq \frac{1}{2} c(H^*_{\text{odd}}) \leq \frac{1}{2} c(H^*) \]
  \[ (2) \]
- Combining 1 with 2 yields
  \[ c(W) \leq c(H^*) + c(M_{\text{min}}) \leq c(H^*) + \frac{1}{2} c(H^*) = \frac{3}{2} c(H^*) \]
Concluding Remarks

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.
Concluding Remarks

**Theorem (Christofides’76)**

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

**Theorem (Arora’96, Mitchell’96)**

There is a PTAS for the Euclidean TSP Problem.
Concluding Remarks

Theorem (Christofides’76)
There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Both received the Gödel Award 2010

Theorem (Arora’96, Mitchell’96)
There is a PTAS for the Euclidean TSP Problem.
Concluding Remarks

Theorem (Christofides’76)

There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

Both received the Gödel Award 2010

Theorem (Arora’96, Mitchell’96)

There is a PTAS for the Euclidean TSP Problem.

“Christos Papadimitriou told me that the traveling salesman problem is not a problem. It’s an addiction.”

Jon Bentley 1991
Concluding Remarks

**Theorem (Christofides’76)**
There is a polynomial-time $\frac{3}{2}$-approximation algorithm for the travelling salesman problem with the triangle inequality.

*Both received the Gödel Award 2010*

**Theorem (Arora’96, Mitchell’96)**
There is a PTAS for the Euclidean TSP Problem.

“Christos Papadimitriou told me that the traveling salesman problem is not a problem. It’s an addiction.”

Jon Bentley 1991