VI. Approximation Algorithms: Travelling Salesman Problem

Thomas Sauerwald
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.

Formal Definition
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 \( (u, v) \in E \).
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.

$3 + 2 + 1 + 3 = 9$
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.

\[
2 + 4 + 1 + 1 = 8
\]
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!
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
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 \( \frac{(n - 1)!}{2} \)

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.

- **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 \( (n - 1)!/2 \)

**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:
  \[ \forall u, v, w \in V : \quad c(u, w) \leq c(u, v) + c(v, w). \]
  
  Even this version is NP hard (Ex. 35.2-2)

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

Additional constraint to cut the solution space of the LP
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)

\[
\begin{align*}
\max & \quad \frac{1}{3} x + y \\
\text{s.t.} & \quad 2x_1 - 9x_2 \leq -27 \\
& \quad x_2 \leq 3 \\
& \quad 4x_1 + 9x_2 \leq 36 \\
& \quad x \geq 0
\end{align*}
\]

Additional constraint to cut the solution space of the LP
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)

### Graphical Representation

- Plot of linear inequalities:
  - $2x_1 - 9x_2 \leq -27$
  - $\max \frac{1}{3}x + y$
  - $x_2 \leq 3$
  - $4x_1 + 9x_2 \leq 36$

- Key points:
  - $(2.25, 3)$

- Additional constraint to cut the solution space of the LP
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**)

### Graphical Representation

- Graph showing the linear program constraints:
  - \(2x_1 - 9x_2 \leq -27\)
  - \(4x_1 + 9x_2 \leq 36\)
  - \(x_2 \leq 3\)
  - \(\max \frac{1}{3} x + y\)

- Points: (2, 3)

Additional constraint to cut the solution space of the LP

More cuts are needed to find integral solution
Outline

Introduction

General TSP

Metric TSP
Hardness of Approximation

Theorem 35.3

If \( P \neq \text{NP} \), then for any constant \( \rho \geq 1 \), there is no polynomial-time approximation algorithm with approximation ratio \( \rho \) for the 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:
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.
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.
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.
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 $(u, v) \in E'$:

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

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

$$c(T) \geq (\rho \cdot |V| + 1) + (|V| - 1) = (\rho + 1) |V|.$$
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 \( \not\in E \), \( \Rightarrow c(T) \geq (\rho |V| + 1) + (|V| - 1) = (\rho + 1) |V| \).

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).

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 ≠ 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':

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

\[G = (V, E)\]
\[G' = (V, E')\]
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 \((u, v) \in E'\):

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

\( G = (V, E) \)

\( \rho \cdot 4 + 1 \)

\( G' = (V, E') \)
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!

Let $G = (V, E)$ and $G' = (V, E')$.
**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 \(|E|\)!

\[G = (V, E)\]

\[\rho \cdot 4 + 1\]

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

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

![Graph reduction example](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 $|V|$.
**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 \(|V|\).

![Diagram of reduction from hamiltonian cycle to TSP](image-url)
Hardness of Approximation

**Theorem 35.3**

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:**

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 \( \notin E \),

\[
\begin{array}{c}
G = (V, E) \\
\xrightarrow{\text{Reduction}} \\
\rho \cdot 4 + 1 \\
G' = (V, E')
\end{array}
\]
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 \( \not\in E \).
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 \( \notin E \),

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

\[ G = (V, E) \quad \rightarrow \quad \text{Reduction} \quad \rho \cdot 4 + 1 \quad 1 \quad 1 \quad \rho \cdot 4 + 1 \quad 1 \quad G' = (V, E') \]
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 $\notin E$.

\[ G = (V, E) \quad \text{Reduction} \quad G' = (V, E') \]
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 \( \notin E \),
  \[
  \Rightarrow c(T) \geq (\rho |V| + 1) + (|V| - 1)
  \]

\[
\begin{align*}
G = (V, E) & \quad \text{Reduction} \\
\rho \cdot 4 + 1 & \\
1 & \\
1 & \\
G' = (V, E') &
\end{align*}
\]
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 $\notin E$,

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

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

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

- Gap of $\rho + 1$ between tours which are using only edges in $G$ and those which don’t.
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 \( \notin E \),

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

- **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)
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 $\notin E$, 
  \[
  \Rightarrow c(T) \geq (\rho |V| + 1) + (|V| - 1) = (\rho + 1)|V|.
  \]

- 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)
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!

VI. Travelling Salesman Problem

instances of Hamilton

instances of TSP
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

instances of Hamilton  
iinstances 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

VI. 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

VI. 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

VI. Travelling Salesman Problem
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

General Method to prove inapproximability results!

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
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.

\[ \text{APPX-TSP-TOUR}(G, c) \]
1. select a vertex \( r \in G.V \) to be a “root” vertex
2. compute a minimum spanning tree \( T \) for \( G \) from root \( r \)
   using \( \text{MST-PRIM}(G, c, r) \)
3. let \( H \) be a list of vertices, ordered according to when they are first visited
   in a preorder tree walk of \( T \)
4. 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 \( T \) for \( G \) from root \( r \)
   using \( \text{MST-PRIM}(G, c, r) \)
3. let \( H \) be a list of vertices, ordered according to when they are first visited
   in a preorder tree walk of \( T \)
4. 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 \( r \in G.V \) to be a “root” vertex
2. compute a minimum spanning tree \( T \) for \( G \) from root \( r \)
   using \textsc{MST-Prim}(G, c, r)
3. let \( H \) be a list of vertices, ordered according to when they are first visited in a preorder tree walk of \( T \)
4. \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

Solution has cost ≈ 19.704 - not optimal!
Better solution, yet still not optimal!
This is the optimal solution (cost ≈ 14.715).
1. Compute MST
Run of **APPROX-TSP-TOUR**

1. Compute MST ✓

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

1. Compute MST ✓
2. Perform preorder walk on MST

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 ✓
2. Perform preorder walk on MST ✓
Run of **APPROX-Tsp-TOUR**

![Diagram of a graph with vertices and edges labeled as a, b, c, d, e, f, g, h.]

1. Compute MST ✓
2. Perform preorder walk on MST ✓
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 ✓
2. Perform preorder walk on MST ✓
3. Return list of vertices according to the preorder tree walk

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

1. Compute MST ✓
2. Perform preorder walk on MST ✓
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 \(\approx 14.715\)).
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**
Run of **APPROX-TSP-TOUR**

1. Compute MST ✓
2. Perform preorder walk on MST ✓
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 ✓
2. Perform preorder walk on MST ✓
3. Return list of vertices according to the preorder tree walk

Solution has cost ≈ 19.704 - not optimal!

Better solution, yet still not optimal!

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

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

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

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 ✓
2. Perform preorder walk on MST ✓
3. Return list of vertices according to the preorder tree walk ✓
Run of \textbf{APPROX-TSP-TOUR}

Better solution, yet still not optimal!

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

1. Compute MST ✓
2. Perform preorder walk on MST ✓
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

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 ✓
VI. Travelling Salesman Problem

Metric TSP
VI. Travelling Salesman Problem

Optimal Solution: Objective 699

Metric TSP
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 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:
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:

solution $H$ of **APPROX-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:

Proof:

![Solution H of APPROX-TSP](image1)

![Optimal solution H*](image2)

solution $H$ of `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

![Diagram of solution $H$ and optimal solution $H^*$](image-url)
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

---

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

**Spanning Tree $T$ as a subset of $H^*$**
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

![Diagram of solution $H$ and spanning tree $T$ as subsets of $H^*$]
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 $c(T) \leq c(H^*)$
**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 $c(T) \leq c(H^*)$

![Diagram of a graph with vertices a, b, c, d, e, f, g, h, showing a solution $H$ of APPROX-TSP and a spanning tree $T$ as a subset of $H^*$, with edge costs being non-negative.]

**Note:** Exploiting that all edge costs are non-negative!
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 \( c(T) \leq c(H^*) \)
- Let \( W \) be the full walk of the minimum spanning tree \( T_{\text{min}} \) (including repeated visits)

![Diagram of graph with nodes labeled a, b, c, d, e, f, g, h.](image)

- Solution \( H \) of \texttt{APPROX-TSP}
- Optimal solution \( H^* \)
**Proof of the Approximation Ratio**

**Theorem 35.2**

\textsf{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)

\[ W = (a, b, c, b, h, d, e, f, e, g, e, d, a) \]

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)

\[
\begin{align*}
\text{Walk } W &= (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a) \\
\text{optimal solution } H^* &= (a, d, e, f, e, g, e, d, a)
\end{align*}
\]
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)
  \[ \Rightarrow \text{ Full walk traverses every edge exactly twice, so} \]

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
  $$c(W) = 2c(T_{\text{min}})$$

![Diagram of a graph with vertices and edges labeled](image)

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

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
  - 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 $T_{\text{min}}$ (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 $H^*$**
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.
  - This 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).
  - The 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 $T_{\text{min}}$ (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 $H$

![Diagram showing a graph with vertices and edges labeled with letters a, b, c, d, e, f, g, h, and the proof steps illustrated](image)

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

**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 $c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)$ exploiting triangle inequality!
- Deleting duplicate vertices from $W$ yields a tour $H$ with smaller cost:

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

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 \( H \) with smaller cost:

  \[ c(H) \leq c(W) \]

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

optimal solution \( H^* \)

exploiting triangle inequality!
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 \( H \) with smaller cost:
  \[ c(H) \leq c(W) \leq 2c(H^*) \]

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

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

![Diagram of tour and spanning tree](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:
  \[
  c(H) \leq c(W) \leq 2c(H^*)
  \]

![Diagram](image-url)

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

optimal solution \( H^* \)  

exploiting 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.
Theorem 35.2

\textsc{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 $T$ for $G$ from root $r$
3: using MST-PRIM($G, c, r$)
4: compute a perfect matching $M$ with minimum weight in the complete graph
5: over the odd-degree vertices in $T$
6: let $H$ be a list of vertices, ordered according to when they are first visited
7: in a Eulearian circuit of $T \cup M$
8: return $H$
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 $T$ for $G$ from root $r$
3: using MST-PRIM($G, c, r$)
4: compute a perfect matching $M$ with minimum weight in the complete graph over the odd-degree vertices in $T$
5: let $H$ be a list of vertices, ordered according to when they are first visited
6: in a Eulearian circuit of $T \cup M$
7: **return** $H$

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

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

1. Compute MST
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ 
3. Find an Eulerian Circuit
4. Transform the Circuit into a Hamiltonian Cycle

All vertices in $T \cup M$ have even degree!
Run of Christofides

1. Compute MST

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

1. Compute MST

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

1. Compute MST ✓

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

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$
Run of Christofides

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$

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

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

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$
1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$
Run of CHRISTOFIDES

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
Run of \textbf{CHRISTOFIDES}

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit

\textbf{All vertices in $T \cup M$ have even degree!}
Run of CHRISTOFIDES

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T ✓$
3. Find an Eulerian Circuit ✓

All vertices in $T \cup M$ have even degree!
Run of CHRISTOFIDES

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
4. Transform the Circuit into a Hamiltonian Cycle
Run of CHRISTOFIDES

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
4. Transform the Circuit into a Hamiltonian Cycle
Run of Christofides

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
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 ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
4. Transform the Circuit into a Hamiltonian Cycle
Run of **CHRISTOFIDES**

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
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 ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
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 ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
4. Transform the Circuit into a Hamiltonian Cycle

Solution has cost $\approx 15.54$ within 10% of the optimum!
1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
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 ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
4. Transform the Circuit into a Hamiltonian Cycle
Run of Christofides

1. Compute MST ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
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 ✓
2. Add a minimum-weight perfect matching $M$ of the odd vertices in $T$ ✓
3. Find an Eulerian Circuit ✓
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 ✓
2. Add a minimum-weight perfect matching \( M \) of the odd vertices in \( T \) ✓
3. Find an Eulerian Circuit ✓
4. Transform the Circuit into a Hamiltonian Cycle ✓
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.

“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
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