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**

Given a complete undirected graph $G = (V, E)$ with nonnegative integer cost $c(u, v)$ for each edge $(u, v) \in E$, the goal is to find a Hamiltonian cycle of $G$ with minimum cost.

The solution space consists of at most $n!$ possible tours, where $n$ is the number of cities. Actually, the right number is $(n-1)!/2$.

**Metric TSP**

Costs satisfy the triangle inequality:

$$c(u, w) \leq c(u, v) + c(v, w)$$

**Euclidean TSP**

Cities are points in the Euclidean space, costs are equal to their (rounded) Euclidean distance.
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 $(u, v) \in E$
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.
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

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:

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

**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

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**
- **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](image)

- Additional constraint to cut the solution space of the LP
- \(2x_1 - 9x_2 \leq -27\)
- \(\max \frac{1}{3}x + y\)
- \(x_2 \leq 3\)
- \(4x_1 + 9x_2 \leq 36\)
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)

```
max \frac{1}{3}x + y
x_1 \leq 3

2x_1 - 9x_2 \leq -27
4x_1 + 9x_2 \leq 36
x_2 \leq 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)

Additional constraint to cut the solution space of the LP

**Linear Program:**

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

Additional constraint to cut the solution space of the LP

$2x_1 - 9x_2 \leq -27$

$\max \frac{1}{3}x + y$

$x_2 \leq 3$

$4x_1 + 9x_2 \leq 36$

$(2, 3)$
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)

\[
\begin{align*}
\text{max } & \frac{1}{3} x + y \\
\text{subject to } & 2x_1 - 9x_2 \leq -27 \\
& x_2 \leq 3 \\
& 4x_1 + 9x_2 \leq 36
\end{align*}
\]

More cuts are needed to find integral solution
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.
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:
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.

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.


- 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 $(u, v) \in E'$:

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

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

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!
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 \( \notin E \), 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 \( P \neq NP \), 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 \((u, v) \in E'\):

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

\( G = (V, E) \)

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

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

---

$G = (V, E) \quad G' = (V, E')$
**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 \((u, v) \in E'\):

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

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

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

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

---

![Reduction Diagram](image)
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 $|V|$. 

![Reduction diagram](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 \(|V|\).

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

\[
1 \quad \rho \cdot 4 + 1 \quad 1 \quad 1 \quad 1 \quad 1
\]
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 $(u, v) \in E'$:
  \[ c(u, v) = \begin{cases} 
  1 & \text{if } (u, v) \in E, \\
  \rho |V| + 1 & \text{otherwise}. 
  \end{cases} \]
- 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:**

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

**Theorem 35.3**

**Proof:** 

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

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

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

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

**Diagram:**

- $G = (V, E)$
- $G' = (V, E')$
- Reduction: $\rho \cdot 4 + 1$
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:**

**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 $\notin E$.

\[
\begin{align*}
G &= (V, E) \\
G' &= (V, E')
\end{align*}
\]

**Reduction**

$$\rho \cdot 4 + 1$$
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:** 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 \( \notin E \).

![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.


- 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 $\notin E$,

Let $G = (V, E)$

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

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

- 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 \( \notin E \),

\[\rho \cdot 4 + 1\]
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.


- 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 \( \notin E \),

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


- 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 \( \notin E \),
  \[
  \Rightarrow \quad c(T) \geq (\rho |V| + 1) + (|V| - 1)
  \]

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

VI. Travelling Salesman Problem

General 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.


- 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 \( \not\in E \),

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


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

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

- Gap of ρ + 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) \(\square\)
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!
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
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

instances of Hamilton

All instances with cost \( \leq k \)

y

instances of TSP

All instances with cost \( > \rho \cdot k \)

f(x)
f(y)

f

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

General Method to prove inapproximability results!
Proof of Theorem 35.3 from a higher perspective

All instances with a hamiltonian cycle

General Method to prove inapproximability results!

All instances with cost $\leq k$

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

instances of Hamilton

instances of TSP

VI. Travelling Salesman Problem
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.

In many practical situations, the least costly way to go from a place $u$ to a place $w$ is to go directly, with no intermediate steps. Put another way, cutting out an intermediate stop never increases the cost. We formalize this notion by saying that the cost function $c$ satisfies the triangle inequality if, for all vertices $u; v; w \in V$,

$$c(u, w) \leq c(u, v) + c(v, w).$$

The triangle inequality seems as though it should naturally hold, and it is automatically satisfied in several applications. For example, if the vertices of the graph are points in the plane and the cost of traveling between two vertices is the ordinary Euclidean distance between them, then the triangle inequality is satisfied. Furthermore, many cost functions other than Euclidean distance satisfy the triangle inequality.

As Exercise 35.2-2 shows, the traveling-salesman problem is NP-complete even if we require that the cost function satisfy the triangle inequality. Thus, we should not expect to find a polynomial-time algorithm for solving this problem exactly. Instead, we look for good approximation algorithms.

In Section 35.2.1, we examine a $2$-approximation algorithm for the traveling-salesman problem with the triangle inequality. In Section 35.2.2, we show that without the triangle inequality, a polynomial-time approximation algorithm with a constant approximation ratio does not exist unless $P = NP$.

35.2.1 The traveling-salesman problem with the triangle inequality

Applying the methodology of the previous section, we shall first compute a structure—a minimum spanning tree—whose weight gives a lower bound on the length of an optimal traveling-salesman tour. We shall then use the minimum spanning tree to create a tour whose cost is no more than twice that of the minimum spanning tree's weight, as long as the cost function satisfies the triangle inequality. The following algorithm implements this approach, calling the minimum-spanning-tree algorithm MST-PIM from Section 23.2 as a subroutine. The parameter $G$ is a complete undirected graph, and the cost function $c$ satisfies the triangle inequality.

\[ \text{A} \text{PPROX-TSP-TOUR}(G; c) \]

1. Select a vertex $r \in G$: to be a "root" vertex
2. Compute a minimum spanning tree $T$ for $G$ from root $r$ using MST-PIM $(G; c; r)$
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 \( H \)
Metric TSP (TSP Problem with the Triangle Inequality)

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

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 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 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.

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

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 \( \Theta(V^2) \).

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 ✓
1. Compute MST
Run of **APPROX-TSP-TOUR**

1. Compute 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 ✓

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

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

VI. Travelling Salesman Problem

1. Compute MST ✓
2. Perform preorder walk on MST ✓
Run of **APPROX-TSP-TOUR**

![Graph diagram]

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 \textbf{APPROX-Tsp-Tour}

1. Compute MST ✓
2. Perform preorder walk on MST ✓
3. Return list of vertices according to the preorder tree walk
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 $\approx 14.715$).
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

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

VI. Travelling Salesman Problem
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

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 **APPROX-TSP-TOUR**

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

Better solution, yet still not optimal!

Solution has cost $\approx 19.704$ - 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

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 ✓
Approximate Solution: Objective 921
Optimal Solution: Objective 699

VI. Travelling Salesman Problem

Metric TSP
Proof of the Approximation Ratio

Theorem 35.2

\textbf{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:**

Consider the optimal tour $H^*$ and remove an arbitrary edge $\Rightarrow$ yields a spanning tree $T$ and $\Rightarrow$ therefore:

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 $\text{Approx-TSP-Tour}$ minimum spanning tree $T_{\text{min}}$ (Walk $W$ = (a, b, c, b, h, b, a, d, e, f, e, g, e, d, a))

exploiting that all edge costs are non-negative! exploiting triangle inequality!
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:**

![Diagram of a solution H of APPROX-TSP](diagram.png)
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 $y$ yields a spanning tree $T$ and $\therefore$

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**

**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**

**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

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

```
              a  d  e  h
               b  f  g
              c

solution $H$ of APPROX-TSP
```

```
              a  d  e  h
               b  f  g
              c

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 to yield a spanning tree $T$ and therefore $c(T) \leq c(H^*)$.

![Graph illustrating the solution $H$ and spanning tree $T$.]

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

**Theorem 35.2**

\textbf{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 therefore $c(T) \leq c(H^*)$
- 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**

$\text{APPROX-TSP-TOUR}$ 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 $\Rightarrow$ yields a spanning tree $T$ and therefore $c(T) \leq c(H^*)$.
- 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**

**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 therefore $c(T) \leq c(H^*)$
- 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 $H^*$
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 therefore $c(T) \leq c(H^*)$
- 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 $H^*$
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 spanning tree $T$ and therefore $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
  \Rightarrow Full walk traverses every edge \textit{exactly twice}, so
  \[ c(W) = 2c(T_{\text{min}}) \]

![Diagram of the optimal tour and the spanning tree](image)
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 therefore $c(T) \leq c(H^*)$.
- 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
  \[
  c(W) = 2c(T_{\text{min}}) \leq 2c(T) \leq 2c(H^*)
  \]

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 \textit{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 therefore $c(T) \leq c(H^*)$
- Let $W$ be the full walk of the minimum spanning tree $T_{\text{min}}$ (including repeated visits)
- Full walk traverses every edge \textit{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**

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

**Proof:**

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

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 therefore $c(T) \leq c(H^*)$

- 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](#)

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 therefore \( c(T) \leq c(H^*) \)
- 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:

\[
\text{Tour } H = (a, b, c, h, d, e, f, g, a)
\]

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

**Theorem 35.2**

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

Proof:

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

\[ \text{Tour } H = (a, b, c, h, d, e, f, g, a) \]

\[ \text{optimal solution } H^* \]
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 therefore $c(T) \leq c(H^*)$
- 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 $\Rightarrow$ yields a spanning tree $T$ and therefore $c(T) \leq c(H^*)$
- 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:

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

optimal solution $H^*$
Christofides Algorithm

Theorem 35.2

\textbf{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 \(T\) 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

\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?

\begin{itemize}
  \item \textsc{Christofides}(G, c)
  \item 1: select a vertex \( r \in G \). \( V \) to be a “root” vertex
  \item 2: compute a minimum spanning tree \( T \) for \( G \) from root \( r \)
  \item 3: \hspace{1cm} using \textsc{Mst-Prim}(G, c, r)
  \item 4: compute a perfect matching \( M \) with minimum weight in the complete graph
  \item 5: \hspace{1cm} over the odd-degree vertices in \( T \)
  \item 6: let \( H \) be a list of vertices, ordered according to when they are first visited
  \item 7: \hspace{1cm} in a Eulerian circuit of \( T \cup M \)
  \item 8: \textbf{return} \( H \)
\end{itemize}

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 $T$

Solution has cost ≈ 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

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

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!
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

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

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!
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
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

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.
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