An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem

Gerold Jäger
University Halle, Germany

April 5, 2006

joint work with:
Boris Goldengorin, Paul Molitor, Dirk Richter
Overview

1 Introduction
   • Motivation
   • STSP and ATSP
   • Tolerances

2 Helsgaun’s Heuristic
   • $k$-Optimality
   • Improvements by Lin and Kernighan
   • Improvements by Helsgaun
   • Held-Karp Approximation
   • $\alpha$-Values
3 Modified Helsgaun’s Heuristic

- Costs
- SMTE Tolerance
- SSP3AX Tolerance
- D2OPT Tolerance
- LAP Tolerance
- Double Bridge Technique
- Backbones and Pseudo-Backbones
Motivation

Shortest tour through 15112 cities in Germany

Board with 2392 bores
TSP

Input: \( C \in \mathbb{R}^{n \times n} \) cost matrix, edge \((i, j)\) with costs \(c_{ij}\)

Output: Tour \( T = (i_1, \ldots, i_n) \) with \(i_j \neq i_k\) and \(i_j \in \{1, 2, \ldots n\}\), so that

\[
c(T) = \sum c_{i_k i_{k+1}} + c_{i_n i_1} \text{ minimum}
\]

- Symmetric (STSP) : \(\Leftrightarrow \forall i, j : c_{ij} = c_{ji}\)
- Otherwise ATSP
- Metric case: \(\Leftrightarrow \forall i, j, k : c_{ik} \leq c_{ij} + c_{jk} \) (triangle inequality)
Transformation from ATSP to STSP

Transformation by Jonker & Volgenant

Let $D = (d_{ij}) \in R^{2n,2n}$, $M = \max(c_{ij}) + 1$

$$\forall i,j \in \{1, \ldots, n\} : d_{n+i,j} := d_{j,n+i} := \begin{cases} c_{ij} & i \neq j \\ -M & i = j \end{cases}$$

All others $d_{ij} := M$

$\Rightarrow$ Efficient algorithm for STSP also efficient for ATSP.

- Disadvantage:
  - Problem size doubled.
  - Number of matrix elements quadrupled.
Example: \(\text{ATSP}(P)\) vs \(\text{STSP}(P)\)
Tolerances for TSP

Set of all optimum tours with costs $c$

$$\mathcal{I}_c := \{ T_{opt} \mid T_{opt} \text{ tour, } c(T_{opt}) = \min_{T \text{ ist Tour}} c(T) \}$$

Observations for $x \in E$:

- $c(x)$ sufficiently small $\Rightarrow \exists T_{opt} \in \mathcal{I}_c : x \in T_{opt}$
- $c(x)$ smaller $\Rightarrow x \in \bigcap \mathcal{I}_c$
- $c(x)$ sufficiently large $\Rightarrow \exists T_{opt} \in \mathcal{I}_c : x \notin T_{opt}$
- $c(x)$ large $\Rightarrow x \notin \bigcup \mathcal{I}_c$
Tolerances for TSP

Manipulation of cost function $c$
Let $x, y \in E$.

$$c_{\alpha,x}(y) := \begin{cases} c(x) + \alpha & \text{if } x = y \\ c(y) & \text{otherwise} \end{cases}$$

Upper tolerance $o_T$ and lower tolerance $u_T$
Let $T \in \mathcal{T}_c$ optimum tour, $x \in T$ and $y \notin T$.

$$o_T(x) := \sup \{ \alpha \in \mathbb{R} \mid T \in \mathcal{T}_{c+\alpha,x} \}$$
$$u_T(y) := \sup \{ \alpha \in \mathbb{R} \mid T \in \mathcal{T}_{c-\alpha,y} \}$$
Characteristics of Tolerances

Tolerances are independent from a special solution

Let \( x, y \in E \) and \( \exists T_1, T_2 \in \mathcal{T}_c \) with \( x \in T_1 \) and \( y \notin T_2 \)...

\[
\begin{align*}
\sigma_{T_1}(x) &= \sigma(x) = \sup\{ \alpha \in \mathbb{R} \mid x \in \bigcup \mathcal{T}_{c+\alpha, x} \} \\
\mu_{T_2}(y) &= \mu(y) = \sup\{ \alpha \in \mathbb{R} \mid y \in \bigcup \mathcal{T}_{c-\alpha, y} \}
\end{align*}
\]

Conclusions:

- Tolerances of an optimum tour characterize whole problem.
  
  → Only one solution from \( \mathcal{T}_c \) is necessary for computation.
Forbidding and forcing edges

Let $in, out \subseteq E$ with $in \cap out = \emptyset$.

$$D|_{in}^{out} := \{ T \mid T \text{ is tour, } in \subseteq T, out \cap T = \emptyset \}$$

$$Tc|_{in}^{out} := \{ T_{opt} \mid T_{opt} \in D|_{in}^{out}, c(T_{opt}) = \min_{T \in D|_{in}^{out}} c(T) \}$$

Computation of tolerances for TSP

Let $T \in Tc, x \in T, y \notin T, T_1 \in Tc|_x^\emptyset$ and $T_2 \in Tc|_y^\emptyset$.

$$o(x) = c(T_1) - c(T)$$

$$u(y) = c(T_2) - c(T)$$
Example

\[ T_{opt} = (a, b, c, d) \]
\[ o(a, b) = o(d, c) = c(a, c, b, d) - c(T_{opt}) = 17 - 10 = 7 \]
\[ o(b, c) = o(a, d) = c(a, c, d, b) - c(T_{opt}) = 15 - 10 = 5 \]
\[ u(a, c) = u(b, d) = c(a, c, d, b) - c(T_{opt}) = 5 \]
**k-Optimality**

**k-Opt step**

Exchange of $k$ edges from a tour leading to a new tour.

**k-optimum tour**

A tour is called $k$-optimum, if there is no $r$-OPT with $2 \leq r \leq k$ improving the tour.
Each $k$-optimum tour is also $(k - 1)$-optimum

The larger $k$, the "better" is the tour

Implementation: $O(n^k)$

In practice: $k \leq 5$
Restriction of search

- Do not add already omitted edges.
- Do not omit already added edges.
- Stop, if a tour has been found earlier. → Saves time for a tour, which could also not be improved earlier.
- Consider only edges as 5 nearest neighbors of vertices.
- Our aim: replace edges by different (tolerant) edges.
Improvements by Helsgaun

- Apply at once improving $k$-OPTs (without "collecting").
- Restriction to the 5 nearest neighbors not reasonable.
  → Instead of 5 nearest edges use edges with largest priority.
- Use $n$ trials of the following steps:
  - **Step 1:** Start from a random vertex.
  - Find the edge with the largest priority starting from this vertex not creating a cycle.
  - Repeat this process until we have a starting tour.
  - **Step 2:** Apply $k$-OPTs ($k \leq 5$) with 5 edges of the largest priority to each vertex.
  - Do these OPTs until you get no improvement any more (local minimum).
  - **Step 3:** Increase the priority of the edges in this local minimum tour.
  - **Step 4:** Go to Step 1.
Lower Bounds

MST
Minimum Spanning Tree.

Minimum 1-tree
Let $G = (V, E)$ be a graph with cost matrix $C$, $v_1 \in V$ and $M$ a MST for a graph $G \setminus \{v_1\}$, then $M$ with two shortest edges of $v_1$ is a minimum 1-tree.

→ Minimum 1-tree gives a lower bound for STSP (in polynomial time).
Observation: 1-tree and optimum tour have many edges in common.
Held-Karp Approximation

- Improving 1-tree via transformation of cost matrix.
- Addition of $x$ does not change optimum tour (but minimum 1-tree).

![Diagram](image-url)
Held-Karp Approximation

- In general: $\phi : C \rightarrow D$ with $d_{ij} = c_{ij} + \pi_i + \pi_j$.
- Set $\pi := (\pi_1, ..., \pi_n)$ and $\omega(\pi) := c(T_{\pi}) - 2 \sum \pi_i$ (re-transformation).
- $T_{\pi}$ minimum 1-tree for $D$, $\omega(\pi)$ lower bound for STSP.
- $\omega(\pi) \rightarrow \max$ via subgradient optimization.
  $\rightarrow$ Held-Karp bound.
$\alpha$-Values

Let $T$ be minimum 1-tree and $T_{ij}$ minimum 1-tree with $(i,j) \in T_{ij}$.

**$\alpha$-Values**

Set $\alpha(i,j) := c(T_{ij}) - c(T)$.

**Lemma**

- $\alpha(i,j) \geq 0$.
- $(i,j) \in T \Rightarrow T_{ij} = T \Rightarrow \alpha(i,j) = 0$.

- $\alpha$-values can be computed in $O(n^2)$.
- Observation: $\alpha$-values are lower tolerances to the 1-tree!
Tests

- Replace tolerance to the 1-tree by costs or different tolerances.

- Tests with:
  - About 70% of symmetric TSPLIB examples.
  - Unsolved VLSI examples.
An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem

Gerold Jäger  University Halle, Germany
SMTE Tolerance

Let $a, b \in V$. Generalization of relaxed tolerance.

SMTE Tolerance (Jop Sibeyn)

$$Sort_c(E(a)) = [k_1, k_2, \ldots, k_n] \text{ with } c(k_i) \leq c(k_{i+1}),$$

$$Sort_c(E(b)) = [l_1, l_2, \ldots, l_n] \text{ with } c(l_i) \leq c(l_{i+1}).$$

$$tol_{SMTE}(a, b) = \begin{cases}  
  c(k_3) - c(k_1) & \text{if } c(a, b) = c(k_1) \\
  c(k_3) - c(k_2) & \text{if } c(a, b) = c(k_2) \\
  c(k_2) - c(a, b) & \text{otherwise} 
\end{cases}$$

$$+ \begin{cases}  
  c(l_3) - c(l_1) & \text{if } c(b, a) = c(l_1) \\
  c(l_3) - c(l_2) & \text{if } c(b, a) = c(l_2) \\
  c(l_2) - c(b, a) & \text{otherwise} 
\end{cases}$$
Modified Helsgaun’s Heuristic

SMTE Tolerance in Comparison

average over all problems

Gerold Jäger University Halle, Germany
An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem
SSP3AX Tolerance, \( X \in \{1, \ldots, n - 2\} \)

- Tolerance to \( X \) cheapest alternative paths.
- Metric: edge + cheapest path form triangle.

**SSP3AX Tolerance**

Let \( a, b \in V \), \( \text{Sort}_{SSP3}(V \setminus \{a, b\}) = [v_1, v_2, \ldots, v_{n-2}] \) with \( c(a, v_i) + c(v_i, b) \leq c(a, v_{i+1}) + c(v_{i+1}, b) \).

\[
\text{tol}_{SSP3AX}(a, b) = \sum_{i=1}^{X} [c(a, v_i) + c(v_i, b)] - X \cdot c(a, b)
\]
SSP3AX Tolerance: Determining Optimum X

average over all problems

Gerold Jäger  University Halle, Germany  An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem
An Improved Helsgaun’s Heuristic for the Symmetric Travelling Salesman Problem

SSP3A5 Tolerance in Comparison

average over all problems

<table>
<thead>
<tr>
<th>trials in percentage</th>
<th>avg. excess over optimum in percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SLKH</td>
</tr>
</tbody>
</table>

Gerold Jäger  University Halle, Germany
D2OPT Tolerance: Idea

**Restricted 2-OPT neighborhood**

$2\text{-OPT}(T, k) =$ Tours from applying 2-OPT to $T$ ($k \notin T$) by edge $k$

**D2OPT Tolerance**

Given a tour $T$ and edge $k \in E \setminus T$.

$tol_{D2OPT}(T, k) := \sup\{\alpha \mid \exists G \in 2\text{-OPT}(T, k), c_{\alpha, k}(G) \leq c_{\alpha, k}(T)\}$

- Tolerance depends on tour $T$. 

An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem
D2OPT Tolerance: Computation

Computation of D2OPT Tolerance

\[
\text{tol}_{D2OPT}(T, k) = \min\{c(alt11) + c(alt22) - c(1, 6), c(alt12) + c(alt21) - c(3, 4)\} - c(k)
\]
D2OPT Tolerance: Improvements

- Problem: Updates expensive, $O(rn)$ per r-OPT

→ Update nodes later (D2OPT*XPD, dirty nodes).
  Update, if more than X% nodes "dirty".

- Further Idea: Prefer cheap edges
  i.e. merge with costs of edge (additional multiplications!).
An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem

Effect of Dirty Nodes

**average over all problems**

![Graph showing the effect of dirty nodes on the average over all problems.](image-url)

- **D2OPT**
- **D2OPT1N5PD**
- **D2OPTNN10PD**
- **D2OPTNN20PD**
- **D2OPTNN50PD**

Costs
SMTE Tolerance
SSP3AX Tolerance
D2OPT Tolerance
LAP Tolerance
Double Bridge Technique
Backbones and Pseudo-Backbones
LAP Tolerance

- Tolerance to linear assignment problem.
- Problem 1: $O(n^2)$ memory.
  - Only medium problems (up to 5000 nodes).
- Problem 2: Tolerance for asymmetric problems.
  - Bad for symmetric problems.
Modified Helsgaun’s Heuristic

LAP Tolerance in Comparison

average over all problems

<table>
<thead>
<tr>
<th>trials in percentage</th>
<th>SLKH</th>
<th>SMTE</th>
<th>D2OPTNN</th>
<th>SSP3A5</th>
<th>SNN</th>
<th>SLAP</th>
<th>D2OPTMTENN</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
<td>0.25</td>
</tr>
<tr>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
<td>0.75</td>
</tr>
<tr>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
<td>1.25</td>
</tr>
<tr>
<td>1.50</td>
<td>1.50</td>
<td>1.50</td>
<td>1.50</td>
<td>1.50</td>
<td>1.50</td>
<td>1.50</td>
<td>1.50</td>
</tr>
<tr>
<td>1.75</td>
<td>1.75</td>
<td>1.75</td>
<td>1.75</td>
<td>1.75</td>
<td>1.75</td>
<td>1.75</td>
<td>1.75</td>
</tr>
<tr>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
<td>2.00</td>
</tr>
<tr>
<td>2.25</td>
<td>2.25</td>
<td>2.25</td>
<td>2.25</td>
<td>2.25</td>
<td>2.25</td>
<td>2.25</td>
<td>2.25</td>
</tr>
<tr>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
<td>2.50</td>
</tr>
<tr>
<td>2.75</td>
<td>2.75</td>
<td>2.75</td>
<td>2.75</td>
<td>2.75</td>
<td>2.75</td>
<td>2.75</td>
<td>2.75</td>
</tr>
<tr>
<td>3.00</td>
<td>3.00</td>
<td>3.00</td>
<td>3.00</td>
<td>3.00</td>
<td>3.00</td>
<td>3.00</td>
<td>3.00</td>
</tr>
<tr>
<td>3.50</td>
<td>3.50</td>
<td>3.50</td>
<td>3.50</td>
<td>3.50</td>
<td>3.50</td>
<td>3.50</td>
<td>3.50</td>
</tr>
<tr>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
<td>3.75</td>
</tr>
<tr>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
<td>4.00</td>
</tr>
<tr>
<td>4.50</td>
<td>4.50</td>
<td>4.50</td>
<td>4.50</td>
<td>4.50</td>
<td>4.50</td>
<td>4.50</td>
<td>4.50</td>
</tr>
<tr>
<td>4.75</td>
<td>4.75</td>
<td>4.75</td>
<td>4.75</td>
<td>4.75</td>
<td>4.75</td>
<td>4.75</td>
<td>4.75</td>
</tr>
<tr>
<td>5.00</td>
<td>5.00</td>
<td>5.00</td>
<td>5.00</td>
<td>5.00</td>
<td>5.00</td>
<td>5.00</td>
<td>5.00</td>
</tr>
<tr>
<td>5.25</td>
<td>5.25</td>
<td>5.25</td>
<td>5.25</td>
<td>5.25</td>
<td>5.25</td>
<td>5.25</td>
<td>5.25</td>
</tr>
<tr>
<td>5.50</td>
<td>5.50</td>
<td>5.50</td>
<td>5.50</td>
<td>5.50</td>
<td>5.50</td>
<td>5.50</td>
<td>5.50</td>
</tr>
<tr>
<td>5.75</td>
<td>5.75</td>
<td>5.75</td>
<td>5.75</td>
<td>5.75</td>
<td>5.75</td>
<td>5.75</td>
<td>5.75</td>
</tr>
<tr>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
<td>6.00</td>
</tr>
<tr>
<td>6.50</td>
<td>6.50</td>
<td>6.50</td>
<td>6.50</td>
<td>6.50</td>
<td>6.50</td>
<td>6.50</td>
<td>6.50</td>
</tr>
<tr>
<td>6.75</td>
<td>6.75</td>
<td>6.75</td>
<td>6.75</td>
<td>6.75</td>
<td>6.75</td>
<td>6.75</td>
<td>6.75</td>
</tr>
<tr>
<td>7.00</td>
<td>7.00</td>
<td>7.00</td>
<td>7.00</td>
<td>7.00</td>
<td>7.00</td>
<td>7.00</td>
<td>7.00</td>
</tr>
<tr>
<td>7.25</td>
<td>7.25</td>
<td>7.25</td>
<td>7.25</td>
<td>7.25</td>
<td>7.25</td>
<td>7.25</td>
<td>7.25</td>
</tr>
<tr>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
<td>7.50</td>
</tr>
<tr>
<td>7.75</td>
<td>7.75</td>
<td>7.75</td>
<td>7.75</td>
<td>7.75</td>
<td>7.75</td>
<td>7.75</td>
<td>7.75</td>
</tr>
<tr>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
<td>8.00</td>
</tr>
<tr>
<td>8.25</td>
<td>8.25</td>
<td>8.25</td>
<td>8.25</td>
<td>8.25</td>
<td>8.25</td>
<td>8.25</td>
<td>8.25</td>
</tr>
<tr>
<td>8.50</td>
<td>8.50</td>
<td>8.50</td>
<td>8.50</td>
<td>8.50</td>
<td>8.50</td>
<td>8.50</td>
<td>8.50</td>
</tr>
<tr>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
<td>8.75</td>
</tr>
<tr>
<td>9.00</td>
<td>9.00</td>
<td>9.00</td>
<td>9.00</td>
<td>9.00</td>
<td>9.00</td>
<td>9.00</td>
<td>9.00</td>
</tr>
</tbody>
</table>

Gerold Jäger  University Halle, Germany
An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem
Helsgaun creates in each trial a new starting tour for leaving the last local minimum tour.

This costs much time for creating a (bad) starting tour and applying many OPT steps to reach again small tours.

Another possibility is to apply (if we have applied \( k \)-OPT steps) one (or more) \((k + 1)\)-OPT step making the tour worse.

For \( k = 3 \) the following double bridge step (4-OPT step) shows a good quality (see Johnson, McGeoch):

For Helsgaun we generalize this step to a 6-OPT step.
Backbones and Pseudo-Backbones

- Edges are called *backbones*, if they appear in all optimal solutions of a TSP.
- Edges are called *pseudo-backbones*, if they appear in all pseudo-optimal solutions of a TSP. (see Zhang and Looks)
- A local minimum after each trial in Helsgaun’s heuristic can be viewed as a pseudo-optimal tour.
- This leads to the following modification of Helsgaun:
  - Start some pre-trials ($X\%$ percent of all trials).
  - The pre-trials determine pseudo-backbones or almost pseudo-backbones, i.e. edges that appear in many of the pseudo-optimal tours.
  - The new priority function is computed by the percentage of an edge in appearing in pseudo-optimal tours.
Pseudo-Backbone Double-Bridge Version in Comparison

Modified Helsgaun’s Heuristic

Pseudo-Backbone Double-Bridge Version in Comparison

average over all problems

Gerold Jäger University Halle, Germany
An Improved Helsgaun’s Heuristic for the Symmetric Traveling Salesman Problem
This version has improved the upper bound (21537 to 21535) of one of the famous VLSI instances.

Helsgaun found only a tour of length 21542 in three times of our time.