THE TRAVELING SALESMAN PROBLEM: 
A STATISTICAL APPROACH

Tiago Salvador

Programa Novos Talentos em Matemática
(Fundação Calouste Gulbenkian)

Supervisor:
Prof. Manuel Cabral Morais

Lisbon, November 2009 — October 2010
Contents

1 Introduction .................................................. 1

2 History ......................................................... 3

3 Modeling ....................................................... 6

4 Computational complexity ....................................... 8

5 $\lambda$-optimal algorithm ...................................... 13

6 Statistical approach ........................................... 15
   6.1 A fundamental result in extreme-value theory .......... 15
   6.2 Description of the statistical approach ................. 17
   6.3 The three-parameter Weibull distribution ............... 18

7 Point estimation of the optimal cost ....................... 20
   7.1 Maximum Likelihood estimates ......................... 20
   7.2 Zanakis estimates ..................................... 22
   7.3 Wyckoff-Bain-Engelhardt estimates ..................... 22

8 Interval estimation for the optimal cost ................... 24
   8.1 Golden and Alt confidence interval .................... 24
   8.2 Los and Lardinois confidence interval .................. 25
   8.3 Other confidence intervals ............................ 25

9 Statistical analysis of the results of the $\lambda$-optimal and $\lambda$-optimal greedy algorithms ....... 26
   9.1 Preliminary analysis ................................... 28
9.2 Point estimates of \((a, b, c)\) .................................................. 31
9.3 Interval estimates for the optimal cost ........................................ 33
9.4 Kolmogorov-Smirnov goodness-of-fit test .................................... 35
9.5 Comparison of (average) performances of the \(\lambda\)-optimal and \(\lambda\)-optimal
greedy algorithms .......................................................... 35

10 Final remarks........................................................................... 39

Bibliography................................................................................. 41
Chapter 1

Introduction

The Traveling Salesman Problem (TSP) is simple to describe but rather difficult to solve. It can be stated as follows: consider a salesman who intends to visit once and only once each city of a given set of $N$ cities such that he ends where he started, and assume that he knows the cost of traveling between any two cities. The TSP consists in finding a sequence of cities such that the associated total traveling cost is minimal — the optimal tour — and is one of the most studied problems in computational mathematics.

Here is what lies ahead in the next nine chapters.

In Chapter 2, we see how the TSP has led to improved solution methods in many areas of mathematical optimization for over 50 years.

In Chapter 3, we focus on the rigorous modeling of the TSP, which can be done in several distinct ways. In an optimization perspective, it is usually described as an integer assignment problem with some additional constraints.

Up to now there is not any known exact polynomial-time algorithm to solve the TSP. This is somehow expected if we take into account its computational complexity, addressed in Chapter 4: the TSP is a NP-hard problem, therefore we do not expect to find such an algorithm.

A possible solution to this problem is to consider approximate algorithms with polynomial-time complexity such as the $\lambda$-optimal algorithms, described in Chapter 5.

The computational complexity of the TSP and the inability to find the optimal solution in polynomial-time justify the statistical approach we present in Chapter 6.

It is important not only to study the quality of the approximate solutions but also to compute point and interval estimates for the optimal cost. We focus our attention the
these two types of estimates in chapters 7 and 8.

In Chapter 9 we present an empirical/statistical analysis of four different instances of the TSP.

Lastly, we conclude with some final remarks in Chapter 10.
Chapter 2

History

It is hard to determine the origins of the TSP problem. According to Hoffman and Wolfe (1985, p. 5), the TSP was first mentioned in a handbook for traveling salesmen from 1832. This handbook includes examples tours through Germany and Switzerland, but no mathematical treatment had been given to the problem; curiously enough, mathematical problems related to the TSP problem were treated in the 1800’s by the Irish mathematician W. R. Hamilton and by the British mathematician Thomas Kirkman (wiki/tsp, 2009).

<table>
<thead>
<tr>
<th>Year</th>
<th>Research team</th>
<th>Size of instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1954</td>
<td>G. Dantzig, R. Fulkerson and S. Johnson</td>
<td>49 cities</td>
</tr>
<tr>
<td>1971</td>
<td>M. Held and R.M. Karp</td>
<td>64 cities</td>
</tr>
<tr>
<td>1975</td>
<td>P.M. Camerini, L. Fratta and F. Maffioli</td>
<td>67 cities</td>
</tr>
<tr>
<td>1977</td>
<td>M. Grötschel</td>
<td>120 cities</td>
</tr>
<tr>
<td>1980</td>
<td>H. Crowder and M.W. Padberg</td>
<td>318 cities</td>
</tr>
<tr>
<td>1987</td>
<td>M. Padberg and G. Rinaldi</td>
<td>532 cities</td>
</tr>
<tr>
<td>1987</td>
<td>M. Grötschel and O. Holland</td>
<td>666 cities</td>
</tr>
<tr>
<td>1987</td>
<td>M. Padberg and G. Rinaldi</td>
<td>2392 cities</td>
</tr>
<tr>
<td>1994</td>
<td>D. Applegate, R. Bixby, V. Chvátal and W. Cook</td>
<td>7397 cities</td>
</tr>
<tr>
<td>1998</td>
<td>D. Applegate, R. Bixby, V. Chvátal and W. Cook</td>
<td>13509 cities</td>
</tr>
<tr>
<td>2001</td>
<td>D. Applegate, R. Bixby, V. Chvátal and W. Cook</td>
<td>15112 cities</td>
</tr>
<tr>
<td>2004</td>
<td>D. Applegate, R. Bixby, V. Chvátal, W. Cook, and K. Helsgaun</td>
<td>24978 cities</td>
</tr>
<tr>
<td>2006</td>
<td>D. Applegate, R. Bixby, V. Chvátal and W. Cook</td>
<td>85900 cities</td>
</tr>
</tbody>
</table>

Table 2.1: Milestones in the solution of TSP instances (inspired by www.tsp.gatech.edu/history/milestone.html)
• 1930’s

It is believed that the term traveling salesman problem was introduced in mathematical circles in 1931-1932 by Hassler Whitney (Hoffman and Wolfe, 1985, p. 5). Karl Menger defined the problem, considered the obvious brute-force algorithm and observed the non-optimality of the nearest neighbor heuristic (wiki/tsp, 2009).

• 1950’s

Dantzig et al. (1954) expressed the TSP as an integer linear program and developed the cutting plane method for its solution, which was later used in the development of a general algorithm; with these new methods they solved an instance with 49 cities to optimality by constructing a tour and proving that no other tour could be shorter (wiki/tsp, 2009). This is in fact one of the problems that we will address in Chapter 9 and we shall call it Dantzig42.

• 1960’s

According to Hoffman and Wolfe (1985, pp. 9–10), Gomory proposed several algorithms, in 1958, 1960 and 1963, to solve the TSP which have continued to affect the theory on integer programming. The most popular of them is the branch and bound method.

• 1970’s and 1980’s

An important way of finding bounds for branch and bound algorithms has become known as Lagrangean relaxation. As claimed by Balas and Toth (1985, p. 373), Held and Karp were the first to use a version of this general procedure, solving an instance of the TSP with 64 cities. Karp (1972) showed that the Hamiltonian cycle problem was NP-complete, which implies the NP-hardness of TSP. This supplied a scientific explanation for the apparent computational difficulty of finding optimal tours (wiki/tsp, 2009). In the late 1970’s and 1980’s, great progress was made when Grötschel and others researchers (see Table 2.1) managed to solve instances with up to 2392 cities, using cutting planes and branch and bound.

• 1990’s and 2000’s

As reported in www.tsp.gatech.edu/concorde/index.html, in the 1990’s, Applegate, Bixby, Chvátal and Cook developed the program Concorde that has been use in
many recent record solutions. In April 2001, Applegate, Bixby, Chvátal and Cook announced the solution of a TSP through 15112 cities in Germany. The computation was carried out on a network of 110 processors and the total computer time used was 22.6 years, scaled to a Compaq EV6 Alpha processor running at 500MHz. In May 2004, the TSP problem of visiting all 24978 towns in Sweden was solved: a tour of length approximately 72500 kilometers was found and it was proven that no shorter tour exists. In April 2006 an instance with 85900 points was solved using Concorde TSP Solver, taking over 136 CPU years, scaled to a 2.4GHz AMP Opteron 250 computer node. For many other instances with millions of cities, solutions can be found that are probably within 1% of the optimal tour (wiki/tsp, 2009).
Chapter 3

Modeling

One way of modeling the TSP is as graph problem, typically an undirected graph. We can consider the cities as the graph’s vertices, the paths as the edges and, naturally, the length of a path as the edge length. Hence, our problem consists in finding the shortest Hamiltonian cycle. Usually, we model the TSP such that the graph is complete (i.e., every pair of vertices is connected by an edge). If no path exists between two cities, we add an arbitrarily long edge which will not affect the optimal tour (in this case, the shortest Hamiltonian cycle).

A more classical way to model the TSP is the following (Langevin \textit{et al.}, 1990):

\begin{equation}
\text{minimize } \sum_{i=1}^{N} \sum_{j=1}^{N} c_{ij} x_{ij} \quad (3.1)
\end{equation}

such that

\begin{equation}
\sum_{i=1}^{N} x_{ij} = 1, \quad i = 1, \ldots, N, \quad (3.2)
\end{equation}

\begin{equation}
\sum_{j=1}^{N} x_{ij} = 1, \quad j = 1, \ldots, N, \quad (3.3)
\end{equation}

\begin{equation}
0 \leq x_{ij} \leq 1, \quad i, j = 1, \ldots, N, \quad (3.4)
\end{equation}

\begin{equation}
x_{ij} \in \mathbb{Z}, \quad i, j = 1, \ldots, N, \quad (3.5)
\end{equation}

“subtour breaking constraints”. \quad (3.6)

In fact, (3.1)–(3.5) describes an assignment problem, which can be solved efficiently. Unfortunately, the TSP requires more constraints. The coefficient $c_{ij}$ represents the cost
(or the distance) to travel from city $i$ to $j$. Due to constraints (3.4) and (3.5), the variable $x_{ij}$ takes values 1 or 0. It takes value 1 if in the optimal tour the salesman must travel from city $i$ to $j$ and 0 otherwise. Constraints (3.2) and (3.3) ensure that each city is visited only once.

We now present three different ways of representing the subtour breaking constraints. As stated by Langevin et al. (1990), Dantzig et al. (1954) replaced (3.6) by

$$
\sum_{i \in Q} \sum_{j \in Q} x_{ij} \leq |Q| - 1, \quad \text{for all } Q \subseteq \{1, \ldots, N\} \text{ and } 2 \leq |Q| \leq N - 1.
$$

(3.7)

Langevin et al. (1990) also adds that Miller et al. (1960) presented a more compact variation of (3.7) by using $O(N^2)$ supplementary continuous variables:

$$
u_i - \nu_j + N x_{ij} \leq N - 1 \quad \text{for all } j \neq 1 \text{ and } i \neq j.
$$

(3.8)

Although, it is not immediately clear, the precise meaning of each variable $u_i$ is the number of visited cities between city 1 and city $i$ in the optimal tour.

Finally, Gavish and Graves (1978) described the subtour breaking constraints using a single commodity flow:

$$
N \sum_{j=1}^i y_{ij} - \sum_{j=2}^N y_{ij} = 1, \quad i = 2, \ldots, N
$$

(3.9)

$$
y_{ij} \leq (N-1)x_{ij}, \quad i = 2, \ldots, N, \quad j = 1, \ldots, N
$$

(3.10)

$$
y_{ij} \geq 0, \quad i, j = 1, \ldots, N.
$$

(3.11)

For fixed values of $x_{ij}$, the constraints given in (3.9) and (3.10) form a network flow problem and therefore the $y_{ij}$ are integers. The value of the variable $y_{ij}$ can be interpreted as indicating the number of arcs which are included in the path between city 1 and arc $(i, j)$ in the optimal tour.

Despite all of the above formulations, we will introduce one more since it will be useful in the next chapter, where we discuss the computational complexity of the TSP.

**TSP**

- **Instance:** Integer $N \geq 3$ and $N \times N$ cost/distance matrix $C = (c_{ij})$.

- **Question:** Which cyclic permutation $\pi$ of the integers from 1 to $N$ minimizes

$$
\sum_{i=1}^N c_{i\pi(i)} + c_{\pi(N)1}?
$$
Chapter 4

Computational complexity

In this chapter we shall explain how difficult it is to solve the TSP problem and follow closely Johnson and Papadimitriou (1985). The difficulty is measured in terms of the performance of the algorithm used to solve the problem. If there is an algorithm that solves the problem easily, a notion we shall define later, then the problem is an easy one.

Definition 4.1 An algorithm is a step by step procedure for solving a problem and is said to solve a problem $P$ if, given any instance $I$ of $P$ as input data, it generates the answer of $P$’s question for $I$.

Consider for instance the following algorithm:

Algorithm $A$

- **Input:** An integer $N \geq 3$ and $N \times N$ cost/distance matrix $C = (c_{ij})$.
- **Output:** The shortest tour of $N$ cities.

1: $\text{min} := \infty$
2: for all cyclic permutations $\pi$ of $\{1, ..., N\}$ do
3: $\text{cost} := \sum_{i=1}^{N} c_{\pi(i)}$
4: if $\text{cost} < \text{min}$ then
5: $\text{min} := \text{cost}$
6: $\text{besttour} := \pi$
7: end if
8: end for
9: return besttour
In order to estimate and express the running time of the algorithm just presented we shall use the $O$-notation. Let $N$ be the number of $N$ cities of the instance we are solving. Then the execution time of algorithm A is $O(N!)$. The main loop is repeated $(N - 1)!$ times, one for each cyclic permutation of $\{1, ..., N\}$. In each one of these repetitions, a tour is generated, its cost is evaluated and the $\text{min}$ and $\text{besttour}$ variables are updated if necessary. These actions take time $O(N)$, so the execution time of the algorithm is in fact $O(N!)$. If we consider that our computer can do $10^{12}$ operations per second, a 50 city TSP will take approximately \[
\frac{50!}{10^{12}} \approx 3 \times \frac{10^{64}}{10^{12}} = 3 \times 10^{52}\] seconds. As physicists estimate the time of life of the universe to be around 10 thousand million years or around $10^{17}$ seconds, solving a 50 city TSP by brute force is clearly impractical.

Relatively small values of $N$ correspond to huge values in exponential functions and reasonable values in polynomial functions, so it comes as no surprise that the classification of an algorithm as good or bad depends on whether or not it has a polynomial-time complexity. Similarly, we can classify the problems as easy or hard depending whether or not it can be solved by an algorithm with polynomial-time complexity. This is the rationale behind the development of the theory of computational complexity.

**Definition 4.2** The class $\mathbf{P}$ consists of all those decision problems for which a polynomial-time algorithm exists.

Decision problems are problems whose questions requires a “yes” or “no” answer. It is important to point out that the restriction to these type of problems in the definition of the class $\mathbf{P}$ can be done without loss of generality.

The TSP problem defined in the previous chapter is not a decision problem but can be easily reformulated as a decision problem:

**TSP decision**

- **Instance:** Integer $N \geq 3$, $N \times N$ cost/distance matrix $C = (c_{ij})$ and an integer $B \geq 0$.
- **Question:** Is there a cyclic permutation $\pi$ of the integers from 1 to $N$ such that
  \[
  \sum_{i=1}^{N} c_{i\pi(i)} \leq B?
  \]
It has been shown that there is a polynomial-time algorithm for the TSP if and only if there is a polynomial-time algorithm for TSP decision, a result that confirms that one can only consider decision problems. So the crucial question is: is TSP decision in P?

**Definition 4.3** A **nondeterministic algorithm** is like an ordinary algorithm where parallel computation is possible, creating many different computational paths.

These algorithms do not yield a solution to every possible computational path, but are guaranteed to reach a correct solution for some path (i.e., if the right choices are made on the way).

**Definition 4.4** The class **NP** consists of all those decision problems for which a polynomial-time nondeterministic algorithm exists.

It is obvious that \( P \subseteq NP \) since an “ordinary” algorithm is a nondeterministic algorithm. But what about \( NP \subseteq P \)? If so, then \( P = NP \) which means that one can simulate deterministically any nondeterministic algorithm without sacrificing more than a polynomial amount of time. Actually, proving or disproving \( P = NP \) is one of the Millennium Problems (http://www.claymath.org/millennium/), which is self-explanatory of the importance and difficulty of such a proof. One can show that \( NP \) contains the TSP decision. However, we cannot hope to show that TSP decision is not in \( P \) unless we first prove that \( P \neq NP \).

**Definition 4.5** A decision problem \( A \) is **NP-complete** if

- \( A \in NP \)
- Every problem in \( NP \) is polynomial transformable to \( A \).

The TSP decision is **NP-complete**. Furthermore, it is not difficult to prove that if \( A \) is **NP-complete**, \( A \in P \) if and only if \( P = NP \). As it is widely believed that \( P \neq NP \), then none of the **NP-complete** problems can be in \( P \), namely, the TSP decision. The TSP itself (optimization version) cannot be **NP-complete** because it is not a decision problem, even though it is equivalent in complexity to the TSP decision.

**Definition 4.6** A problem is **NP-hard** if all problems in **NP** are polynomial reducible (not necessarily transformable) to it.
It can be shown that the TSP is **NP-hard** which reinforces the idea that it is difficult to solve it in polynomial-time.

A further computational complexity analysis of the TSP requires the definition of more general complexity classes. However, if it turns out that \( P = NP \) then the discussion that follows will be meaningless.

**Definition 4.7** The complement of a decision problem is itself a decision problem which results from reversing the “yes” and “no” answers from the “original” decision problem.

For instance, the complement of TSP decision is:

**TSP complement**

- **Instance**: Integer \( N \geq 3 \), \( N \times N \) cost/distance matrix \( C = (c_{ij}) \) and an integer \( B \geq 0 \).

- **Question**: Is it true that for all cyclic permutation \( \pi \) of the integers from 1 to \( N \),

\[
\sum_{i=1}^{N} c_{i\pi(i)} > B?
\]

We are now able to define a new complexity class.

**Definition 4.8** The class **coNP** consists of all the complements of all decision problems in **NP**.

Due to the asymmetry in the definition it is expected that \( NP \neq coNP \). Unsurprisingly, this result has not been proved because it would imply that \( P \neq NP \).

Now we can use **NP** and **coNP** to define a class that appears to include even harder problems.

**Definition 4.9** The class **DP** consists of all decision problems which are the intersection of a problem in **NP** with one in **coNP**, i.e., each problem \( X \) in **DP** is defined by two problems \( X_1 \) and \( X_2 \) over the same set of instances, with \( X_1 \in NP \) and \( X_2 \in coNP \), such that the answer of \( X \) is “yes” if and only if the answer for both \( X_1 \) and \( X_2 \) is “yes”.

11
The following decision problem is in $\text{DP}$:

**Exact TSP**

- **Instance:** Integer $N \geq 3$, $N \times N$ cost/distance matrix $C = (c_{ij})$ and an integer $B \geq 0$.
- **Question:** Is the cost of an optimal tour exactly $B$?

*Exact TSP* is the intersection of the *NP* decision problem *TSP decision* and a variant of *TSP complement* which asks if for all cyclic permutation $\pi$ of the integers from 1 to $N$, $\sum_{i=1}^{N} c_{i\pi(i)} \geq B$? and is in *coNP*.

Indeed, $\text{NP} \subseteq \text{DP}$ and $\text{coNP} \subseteq \text{DP}$. The trivial problem (the problem that answers every question “yes”) is in $\text{P}$, hence is in both $\text{NP}$ and $\text{coNP}$. So given any problem in one of these classes, it can be seen as itself intersected with the trivial problem and therefore is in $\text{DP}$.

Having Definition 4.5 in mind, we define the problems $\text{DP-complete}$, which are potentially worse than *NP-complete* since $\text{NP} \subseteq \text{DP}$ and $\text{coNP} \subseteq \text{DP}$, in the natural way. If fact, in some sense they are. It can be shown that if a $\text{DP-complete}$ problem is in $\text{NP}$ or in $\text{coNP}$ then $\text{NP} = \text{coNP}$. Assuming that $\text{NP} \neq \text{coNP}$, then any $\text{DP-complete}$ problem is not in $\text{NP}$, whereas all the $\text{NP-complete}$ are.

Papadimitriou and Yannakakis (1984) have shown the following result:

**Theorem 4.10** The Exact TSP is $\text{DP-complete}$.

So the Exact TSP is even harder to solve than the *NP-complete* TSP decision.
Chapter 5

λ-optimal algorithm

As mentioned in Chapter 4, we do not expect to find a polynomial-time algorithm to solve the TSP since it is a \textbf{NP-hard} problem. For that matter, several algorithms have been proposed that do not always yield the optimal solution but instead a reasonable solution in a reasonable amount of time. Actually, there is a compromise between the time required by the algorithm and the quality of the yielded solution.

There are several approximate algorithms for the TSP, which can be divided in two groups: tour construction algorithms and iterative improvement algorithms. The Nearest Neighbor Algorithm, Savings Algorithm (Clarke and Wright, 1964) and the Convex Hull Algorithm (Golden and Stewart, 1985, p. 217) are examples in the first group. The iterative improvement algorithms usually receive as input a random tour or a tour obtained by using a tour construction algorithm. One example is the λ-optimal algorithm, whose description requires two preliminary definitions taken from Lin (1965).

\begin{definition}
A tour is said to be \( \lambda \)-optimal (or simply \( \lambda \)-opt) if it is impossible to obtain a tour with smaller cost by replacing any \( \lambda \) of its edges by any other set of \( \lambda \) edges.
\end{definition}

\begin{definition}
The \( \lambda \)-optimal neighborhood of a tour \( T \), \( N_\lambda(T) \) is the set of all tours we can obtain of \( T \) by replacing \( \lambda \) of its edges.
\end{definition}

\textbf{λ-optimal algorithm}

\begin{enumerate}
\item \textbf{Step 1} Randomly choose a tour \( T \).
\item \textbf{Step 2} If \( T \) is \( \lambda \)-optimal stop. Otherwise, go to step 3.
\end{enumerate}
Step 3 Compute $N_\lambda(T)$ and the associated costs of its tours.

Step 4 Choose $T^*$ as the tour in $N_\lambda(T)$ with smallest cost and return to step 2 with $T = T^*$.

For more details, see Lin (1965) and Morais (1994).

We shall also be interested in a variant of this algorithm. At step 4, instead of choosing the tour with smallest cost, choose the first one found that has a smaller cost then $T$. This is what we shall call the $\lambda$-optimal greedy algorithm.

In what follows, we focus on the 2-optimal, 2-optimal greedy, 3-optimal and 3-optimal greedy algorithms.
Chapter 6

Statistical approach

A statistical approach to an optimization problem requires the use of a probabilistic model to characterize the behavior of the approximate solutions.

The idea of using the Weibull distribution is not new: Golden (1977) refers that McRoberts (1966), while dealing with combinatorially explosive plant-layout problems, was the first author to do it. Since then, several authors have proceeded in a similar way: Golden (1977), Golden and Alt (1979), Los and Lardinois (1982), Kovacs and Goodin (1985) and Morais (1994).

The use of the Weibull distribution is not limited to the TSP. In fact it has been used by other authors in other combinatorial optimization problems: the covering problem (Vasko et al., 1984), the Steiner problem defined for graphs (Cerdeira, 1986) and the multicovering problem (Gonsalvez et al., 1987).

The titles of Dannenbring (1977), Ariyawansa (1980) and Derigs (1983) also suggest the use of the statistical approach.

6.1 A fundamental result in extreme-value theory

The study of the asymptotic behavior of $Y(m) = \max_{j=1,\ldots,m} Y_j$ the maximum of $Y = (Y_1, \ldots, Y_m)$, a random sample of dimension $m$ from a population $Y$ was an interesting challenge to the statisticians of the beginning of the 20th century.

In our case, we will be interested in results for the minimum which can be easily deduced since the minimum of $Y$, $Y_{(1)} = \min_{j=1,\ldots,m} Y_j$, is equal to $-\max_{j=1,\ldots,m} (-Y_j)$.

A fundamental result in the extreme-value theory is the Gnedenko theorem, which
we present here on its version for the minimum (Morais, 1994), since it will allow us to partially justify the choice of the Weibull distribution in the estimation of the optimal tour cost of the TSP.

**Theorem 6.1** Let

- \{Y_j\}_{j=1,2,...} be a sequence of independent and identically distributed random variables with common distribution function \(F_Y(y)\) and

- \(H_m(y) = 1 - [1 - F_Y(y)]^m\) the distribution function of \(Y_{(1)}\).

Then, if there is

- a non degenerate distribution function \(H(z)\) and

- real constants \(\{\lambda_m\}\) and \(\{\delta_m > 0\}\) such that

\[
\lim_{m \to \infty} H_m(\lambda_m + \delta_m z) = \lim_{m \to \infty} P \left[ \frac{Y_{(1)} - \lambda_m}{\delta_m} \leq z \right] = H(z),
\]

for every continuity point of \(H(z)\),

\(H(z)\) is one of the following types:

- **Fréchet (minimum)**
  
  \[
  \Lambda'_1(z) = \left[ 1 - e^{-(z-\alpha)} \right] \times I_{(-\infty,0)}(z) + I_{(0,+\infty)}(z) \quad (\alpha > 0);
  \]

  (6.1)

- **Weibull (minimum)**
  
  \[
  \Lambda'_2(z) = \left( 1 - e^{-z^\alpha} \right) \times I_{(0,+\infty)}(z) \quad (\alpha > 0);
  \]

  (6.2)

- **Gumbel (minimum)**
  
  \[
  \Lambda'_3(z) = \left( 1 - e^{-e^{-z}} \right) \times I_{(-\infty,+\infty)}(z) \quad (\alpha > 0).
  \]

(6.3)

This theorem does not provide an answer to the following question: how can we identify the asymptotic distribution of the minimum of the random sample \(Y\), that is, to which domain of attraction does the distribution function of \(Y\) belong?

For instance the domain of attraction of the Weibull (minimum) distribution can be characterized as follows:
Theorem 6.2 Let $\alpha(F_Y) = \inf \{y : F_Y(y) > 0\}$ and $\omega_m(F_Y) = \sup \{y : F_Y(y) \leq 1/m\}$. Then $F_Y(y)$ belongs to the domain of attraction of the Weibull (minimum) distribution if and only if:

- $\alpha(F_Y) > -\infty$ and
- $\exists \alpha > 0 : \lim_{y \to 0^+} \frac{F_Y[\alpha(F_Y) + ky]}{F_Y[\alpha(F_Y) + y]} = k^{-\alpha}, \forall k > 0$

Under these conditions,

$$\lim_{m \to \infty} P \left[ \frac{Y(1) - \lambda_m}{\delta_m} \leq z \right] = \Lambda'(z),$$

where $\lambda_m = \alpha(F_Y)$ and $\delta_m = \omega_m - \alpha(F_Y)$.

It is important to note that the fact that the range of the random variable $Y$ is limited to the left is not a necessary nor sufficient condition for $F_Y(.)$ to belong to the domain of attraction of the Weibull (minimum) distribution.

Although we do not present the domains of attraction of the remaining distributions, which can be found in Morais (1994), it is worth noting that the Gumbel (minimum) distribution is associated to random variables with range $\mathbb{R}$ and the Fréchet (minimum) distribution is related to random variables with range limited to the right.

6.2 Description of the statistical approach

Considerer an approximate algorithm $A$ for solving the TSP which will be executed $n$ times.

At iteration $i$, $A$ will yield an approximate solution to the TSP, whose cost will be represented by the random variable $X_i$. Naturally, the final solution will be the minimum among the $n$ approximate solutions, that is, $X_{(1)} = \min_{i=1,\ldots,n} X_i$.

We can justify the use of the Weibull distribution in the modeling of the approximate solutions, as McRoberts (1966) did, by treating the intermediate approximate solutions as a set say $m$ of independent random variables from a parent distribution and therefore $X_i$ will be the minimum of this set. In addition, assuming the remaining conditions of Theorem 6.1 hold, $X_i$ has a distribution function that can be approximated by one of the three limiting distribution functions defined in this theorem. Clearly, as Golden (1977) notices, there is a clearly interdependence among the intermediate solutions, hence the
independence assumption is debatable. However, Golden (1977) argues that if we assume that we are dealing with a $N$-city TSP, then there are $\frac{(N-1)!}{2}$ tours, whose costs are bounded from below by the cost of the optimal tour, thus each solution is a local minimum from a large number $m$ of possible tours. Furthermore, the solutions are independent since the initial tours are randomly chosen.

Finally, note that the Weibull distribution is the only limiting distribution from Theorem 6.1 with a range limited from below. Therefore it comes with no surprise that the Weibull distribution has been used to model $X_i, i = 1, \ldots, n$, and, interestingly, several authors have use it with astonishing results.

### 6.3 The three-parameter Weibull distribution

The three-parameter Weibull distribution is especially useful in life testing and reliability problems where the life may be bounded from below. This distribution is, in some sense, a generalization of the exponential distribution. A random variable $X$ has a three-parameter Weibull distribution if its probability density function is given by

$$f_X(x) = \frac{c}{b} \times \left(\frac{x-a}{b}\right)^{c-1} e^{-\left(\frac{x-a}{b}\right)^c} \times I_{[a, +\infty)}(x)$$

where $-\infty < a < +\infty$, $b > 0$ and $c > 0$ represent the location, scale and shape parameters, respectively. In this case, $X$ has distribution function equal to

$$F_X(x) = \left[1 - e^{-\left(\frac{x-a}{b}\right)^c}\right] \times I_{[a, +\infty)}(x).$$

![Figure 6.1: Probability density functions of the three-parameter Weibull with $a = 0$, $b = 1$ and $c = 0.5, 1, 2$.](image)
Note that since we assume that the costs are non-negative, then the range of possible values of the location parameter, $a$, will be restricted to $[0, +\infty)$. We should also point out the importance of the shape parameter: it is responsible for various appearances of the probability density function, as depicted by Figure 6.1.
Chapter 7

Point estimation of the optimal cost

Assuming that the three-parameter Weibull distribution characterizes the costs of the approximate solutions of the TSP, our goal is to estimate its location parameter \( a \) since it corresponds to the cost of the optimal tour, by making use of the results of \( n \) runs of an approximate algorithm.

As we shall see, the maximum likelihood estimate (MLE) of \( a \) is not as trivial as one might think, thus the need to introduce alternative estimates.

7.1 Maximum Likelihood estimates

For the three-parameter Weibull distribution the likelihood function is given by

\[
L(a, b, c|x) = \left( \frac{c}{b} \right)^n \times \left( \prod_{i=1}^{n} \frac{x_i - a}{b} \right)^{c-1} \times \exp \left[ -n \sum_{i=1}^{n} \left( \frac{x_i - a}{b} \right)^c \right], \tag{7.1}
\]

where \( x = (x_{(1)}, \ldots, x_{(n)}) \), \( a \in [0, x_{(1)}] \) and \( b, c \in (0, +\infty) \).

The associated log-likelihood function is equal to

\[
\log L(a, b, c|x) = n \log(c) - nc \log(b) + (c - 1) \sum_{i=1}^{n} \log(x_i - a) - \sum_{i=1}^{n} \left( \frac{x_i - a}{b} \right)^c. \tag{7.2}
\]

From now on, the triple \((\hat{a}, \hat{b}, \hat{c})\) will represent the MLE of \((a, b, c)\). Some difficulties may arise in the computation of these estimates. According to Rockette et al. (1974), the
estimates are not always obtained by solving the following system of likelihood equations:

\[
\begin{align*}
\frac{\partial \log[L(a,b,c|x)]}{\partial a} \bigg|_{(a,b,c)=(\hat{a},\hat{b},\hat{c})} &= 0 \\
\frac{\partial \log[L(a,b,c|x)]}{\partial b} \bigg|_{(a,b,c)=(\hat{a},\hat{b},\hat{c})} &= 0 \\
\frac{\partial \log[L(a,b,c|x)]}{\partial c} \bigg|_{(a,b,c)=(\hat{a},\hat{b},\hat{c})} &= 0 \\
\end{align*}
\]

\[
\begin{align*}
-(\hat{c} - 1) \sum_{i=1}^{n} (x_i - \hat{a})^{-1} + \frac{\hat{c}}{b} \sum_{i=1}^{n} (x_i - \hat{a})^{\hat{c}-1} &= 0 \\
\frac{n \hat{c}}{b} + \frac{\hat{c}}{b^{\hat{c}+1}} \sum_{i=1}^{n} (x_i - \hat{a})^{\hat{c}} &= 0 \\
\frac{n \hat{c}}{c} - n \log(\hat{b}) + \sum_{i=1}^{n} \log(x_i - \hat{a}) - \sum_{i=1}^{n} \left( \frac{x_i - \hat{a}}{b} \right)^{\hat{c}} \log \left( \frac{x_i - \hat{a}}{b} \right) &= 0.
\end{align*}
\]  

(7.3)

In fact, the obtention of the MLE of \((a,b,c)\) will depend on the true value of the shape parameter \(c\).

If \(c \in (0,1)\), \(\frac{\partial \log[L(a,b,c|x)]}{\partial a}\) is a non-negative function regardless of the values of \(b\) and \(c\). Moreover, the log-likelihood function verifies

\[
\lim_{a \to x^{(1)}} \log[L(a,b,c|x)] = +\infty.
\]  

(7.4)

Hence, \(\hat{a} = x^{(1)}\) and the estimates \(\hat{b}\) and \(\hat{c}\) are obtained by solving the last two equations of (7.3).

As stated by Rockette et al. (1974), when \(c \in [1, +\infty)\) and the system (7.3) has no solution, the log-likelihood has a maximum at

\[
(\hat{a}, \hat{b}, \hat{c}) = \left( x^{(1)}, \frac{1}{n} \sum_{i=1}^{n} (x_i - x^{(1)}), 1 \right).
\]  

(7.5)

The same author, based on numerous numerical investigations, believes that only in special cases there is a unique solution to the likelihood equations. For this reason, some authors proposed alternatives to the MLE, such as the ones in the following sections.
7.2 Zanakis estimates

Zanakis (1979) presents a study on several analytic estimates for the three-parameter Weibull distribution and suggests the use of

\[
\bar{a} = \frac{x_{(1)} \times x_{(n)} - x_{(2)}^2}{x_{(1)} + x_{(n)} - 2 \times x_{(2)}}
\]

(7.6)

\[
\bar{b} = -\bar{a} + x_{(\lceil 0.63n \rceil)}
\]

(7.7)

\[
\bar{c} = \frac{\log(\log(1 - p_k)) - \log(1 - p_i)}{\log\left(\frac{x_{(\lceil np_k \rceil)} - \bar{a}}{x_{(\lceil np_i \rceil)} - \bar{a}}\right)}
\]

(7.8)

where \(p_i = 0.16731\) and \(p_k = 0.97366\).

The author refers that these estimates have proved to be more accurate than the MLE for all three parameters (especially the shape parameter) when \(c < 2\), in particular, when \(n\) is small.

Another advantage of the use of the Zanakis’ estimates is the fact that they only depend on a few entries of the ordered sample and have closed expressions. This simplicity is a tremendous advantage over the MLE, especially when we have limited computational resources.

Note that this estimates have been used by Cerdeira (1986) in a different combinatorial optimization problem with satisfying results according to the author.

7.3 Wyckoff-Bain-Engelhardt estimates

Wyckoff et al. (1980) propose estimates that result from previous work by Dubey (1967) and Engelhardt and Bain (1977) for the two-parameter Weibull distribution whose scale and shape parameters are unknown and whose location parameter is assumed to be 0.

The three estimates depend on crude estimate of the shape parameter based on the ordered sample

\[
\bar{c}_0 = \frac{\log(\log(1 - p_k)) - \log(1 - p_i)}{\log\left(\frac{x_{(\lceil np_k \rceil)} - x_{(1)}}{x_{(\lceil np_i \rceil)} - x_{(1)}}\right)}
\]

(7.9)

Observe that this estimate is a particular case of (7.8) if we replace \(\bar{a}\) by \(x_{(1)}\).
The Wyckoff-Bain-Engelhardt estimates have the following expressions:

\[
\hat{a} = \frac{x_{(1)} - \bar{x}}{1 - \frac{1}{n^{1/\gamma_0}}}
\]

\[\hat{b} = \exp\left\{\frac{\gamma}{c_0} + \frac{1}{n} \sum_{i=1}^{n} \log(x_{(i)} - \hat{a})\right\}
\]

\[
\hat{c} = \frac{n \times k_n}{-\sum_{i=1}^{s} \log(x_{(i)} - \hat{a}) + \frac{s}{n-s} \sum_{i=s+1}^{n} \log(x_{(i)} - \hat{a})},
\]

where \(\bar{x}\) is the arithmetic mean, \(\gamma\) is the Euler constant, \(s = \lceil 0.84n \rceil\) and \(k_n\) is a constant whose value depends on the dimension of the sample (for exact values of \(k_n\) see Table 6 in Engelhardt and Bain, 1977).

It is worth noting that there are some differences between the Zanakis and Wyckoff-Bain-Engelhardt estimates. Unlike the Zanakis estimates of \(b\), the estimate of the scale parameter in (7.11) not only depends on the ordered sample but also on a crude estimate of the shape parameter.

Finally, we should point out that the minimum of the sample plays an important part in the estimation of the location parameter. We can compare \(x_{(1)}\) with the three estimates of \(a\) referred in this chapter, and if the minimum is considerably larger than \(\hat{a}\), \(\bar{a}\) or \(\tilde{a}\) it is wise to try to improve the approximate solutions.
Chapter 8

Interval estimation for the optimal cost

It is usual to present point estimates along with interval estimates. For this reason, we introduce the reader to different confidence intervals proposed by Golden and Alt (1979), Los Lardinois (1982) and Morais (1994).

8.1 Golden and Alt confidence interval

The confidence interval proposed by Golden and Alt (1979) is based on the fact that $X_{(1)} \sim \text{Weibull} \left(a, \frac{b}{n^{1/c}}, c\right)$ and that

$$P[a < X_{(1)} < a + b] = P[X_{(1)} - b < a < X_{(1)}] = 1 - e^{-n}.$$ 

Hence, a $(1 - e^{-n}) \times 100\%$ confidence interval for the location parameter is given by

$$CI_{(1-e^{-n})\times100\%}(a) = [x_{(1)} - b; x_{(1)}].$$ (8.1)

When the scale parameter is unknown, Golden and Alt (1979) suggest that $b$ is replaced by its MLE, $\hat{b}$, yielding an approximate confidence interval

$$CI_{GA}(a) = [x_{(1)} - \hat{b}; x_{(1)}].$$ (8.2)

Even though replacing $b$ by its MLE $\hat{b}$ naturally changes the confidence level, it is expected to be close to $(1 - e^{-n}) \times 100\%$. In order to emphasize this fact, when the confidence level is unknown we just use the authors initials to represent the interval. Also note that we do not know the exact confidence level.
8.2 Los and Lardinois confidence interval

Los and Lardinois (1982) generalized the interval estimate (8.1) in order to obtain an interval with a predetermined confidence level $(1 - \alpha) \times 100\%$, independent of the sample size:

$$CI_{(1-\alpha)\times100\%}(a) = \left[ x_{(1)} - \frac{b}{(-n/\log[\alpha])^{1/\epsilon}}; x_{(1)} \right].$$

(8.3)

As in the previous section, the unknown parameters are replaced by their MLE, leading to the following approximate confidence interval:

$$CI_{LL}(a) = \left[ x_{(1)} - \frac{\hat{b}}{(-n/\log[\alpha])^{1/\epsilon}}; x_{(1)} \right].$$

(8.4)

8.3 Other confidence intervals

As proposed by Morais (1994), we can replace the scale and shape parameters, not only by the MLE, but also by the Zanakis and Wyckoff-Bain-Engelhardt estimates. Hence, we can generalize formula (8.1) in two different ways:

- the Golden-Alt-Zanakis confidence interval
  $$CI_{GAZ}(a) = \left[ x_{(1)} - \bar{b}; x_{(1)} \right].$$
  (8.5)

- the Golden-Alt-Wyckoff-Bain-Engelhardt confidence interval
  $$CI_{GAWBE}(a) = \left[ x_{(1)} - \tilde{b}; x_{(1)} \right].$$
  (8.6)

We can proceed similarly with formula (8.3) and obtain:

- the Los-Lardinois-Zanakis confidence interval
  $$CI_{LLZ}(a) = \left[ x_{(1)} - \bar{b} \left( -n/\log[\alpha] \right)^{1/\epsilon}; x_{(1)} \right];$$
  (8.7)

- the Los-Lardinois-Wyckoff-Bain-Engelhardt confidence interval
  $$CI_{LLWBE}(a) = \left[ x_{(1)} - \tilde{b} \left( -n/\log[\alpha] \right)^{1/\epsilon}; x_{(1)} \right].$$
  (8.8)
Chapter 9

Statistical analysis of the results of the \( \lambda \)-optimal and \( \lambda \)-optimal greedy algorithms

This report focuses now on the statistical analysis of the approximate solutions obtained by applying the \( \lambda \)-optimal and \( \lambda \)-optimal greedy algorithms (\( \lambda = 2, 3 \)) to four different problems.

- **Dantzig42**
  Dantzig *et al.* (1954) introduced and illustrated an approximation method by means of a 49-city instance of the TSP. It resulted from picking one city from each of the 48 states in the U.S.A. at the time and adding Washington, D.C. In fact, the authors only worked with a 42-city instance since they realized that the optimal tour through the 42 cities uses roads that pass through the 7 cities that were excluded. This problem is part of the TSPLIB (comopt.ifi.uni-heidelberg.de/software/TSPLIB95).

- **Krolak**
  Problem 24 of Krolak *et al.* (1971) consists of 100 cities with Cartesian coordinates in Table II of this reference. It has already been studied in a statistical perspective by Golden and Alt (1979) and Morais (1994).
• **Random**

We randomly generated 100 points with integer coordinates between 0 and 5000, using the routine `RandomInteger` in the program *Mathematica* (www.wolfram.com), and made use of the Euclidean distances.

• **Gr120**

In this case the data set consists of the travel distances between 120 cities in and around Germany. This problem is also part of the TSPLIB and has been a standard test instance for the TSP since 1977, when it was proposed by Martin Grötschel.

The approximate solutions were obtained by using existing programs written in *Mathematica* (demonstrations.wolfram.com/ComparingAlgorithmsForTheTravelingSalesman Problem) modified to control the initial tour. The input of any program is a random tour; and to allow the comparison of results (for a given problem), any run with the same number starts with the same randomly generated tour. The output of each run is the minimum cost and the associated tour. Then after 100 runs of each algorithm, we get a data set of minimum costs and compute the MLE of \((a, b, c)\), using the `NMaximize` routine in *Mathematica*, as well as the Zanakis and Wyckoff-Bain-Engelhardt estimates. We also obtain approximate confidence intervals for the optimal cost, as described in Chapter 8.

In order to compare the different algorithms and to determine which one yields better solutions, we perform the Mann-Whitney test, as described in Conover (1971, pp. 224–226), using the routine `wilcox.test` in *R* (www.r-project.org). We present the observed values of the Kolmogorov-Smirnov goodness-of-fit test statistic (see Conover, 1971, pp. 295–296) and the associated \(p\)-value, computed using the `ks.test` routine of *R*; the three conjectured distributions are three-parameter Weibull distributions with \((a, b, c)\) equal to ML, Zanakis and Wyckoff-Bain-Engelhardt estimates.

We used the program *Concorde TSP Solver* (www.tsp.gatech.edu/concorde/index.html) in order to obtain the optimal cost (and the associated tour for each problem) and to confront it with its point and interval estimates.
9.1 Preliminary analysis

In this section we shall investigate the differences between the algorithms, namely in terms of their impact on the location and dispersion of the data set of minimum costs. We start with some general remarks about the performance of the algorithms for each problem.

Probably due to its size, we were able to obtain the optimal solution for the Dantzig42 problem with all four algorithms, namely the 3-optimal greedy algorithm yielded the optimal solution 23 out of 100 runs.

For the Krolak problem only the 3-optimal algorithms led to the optimal solution; this happened just a single time for each of these two algorithms.

For the remaining instances of the TSP, none of the algorithms were able to identify the optimal solution.

It is apparent from Figure 9.1 that the results of the 2-optimal algorithms have a wider range than the results of the 3-optimal algorithms; in addition the 3-optimal algorithms are associated to minimum costs closer to the optimal one. Moreover, in the 3-optimal algorithms there is a significant difference between the four problems: the mode of the minimum costs for the Dantzig42 and Krolak problems appear to be closer to the optimal cost. This difference can be explained by the fact that, for these two problems, the 3-optimal algorithms yielded approximate solutions closer to the optimal cost more frequently.

Clearly, Figure 9.2 confirms the wider range of the results of the 2-optimal algorithms and the closeness to the optimal cost of the minimum costs attained by the 3-optimal algorithms. Indeed, the Box-and-whisker plots show that the 3-optimal algorithms tend to perform remarkably better than their 2-optimal counterparts.

Moreover, the results for the 2-optimal algorithm are more or less symmetrical whereas the results of the 3-optimal algorithms are clearly skew to the right. In addition, the 3rd quartile of the data set associated to the 3-optimal algorithm is always smaller than the 1st quartile of the data of the 2-optimal algorithm. This means that 75% of the smallest results of the 3-optimal algorithm are smaller than 75% of the largest results yielded by the 2-optimal algorithm. In addition, for the Random and Gr120 problems, the sample median obtained via the 3-optimal algorithm is actually smaller than the sample minimum associated to the 2-optimal algorithm, meaning that in at least more than a half of the 100 runs, the 3-optimal algorithm provided better approximate solutions.
Figure 9.1: Histograms of the minimum costs for four instances of the TSP — $\lambda$-optimal and $\lambda$-optimal greedy ($\lambda = 2, 3$).
Figure 9.2: Box-and-whisker plots of the minimum costs for four instances of the TSP — $\lambda$-optimal and $\lambda$-optimal greedy ($\lambda = 2, 3$).
The confrontation between the results of the 2- and 3-optimal greedy algorithms leads to similar conclusions. However, we should point out that the maximum cost attained by the 3-optimal greedy algorithm is smaller than the sample median of the one obtained by the 2-optimal greedy algorithm, for all four instances of the TSP with the exception of the Dantzig42 problem.

When we apply the 2-optimal greedy algorithm instead of the 2-optimal algorithm, the median sample is smaller than the 1st quartile with the exception of the Dantzig42 problem. This means that at least 50% of the minimum costs when we run the 2-optimal greedy algorithm are smaller than 75% of the largest results of the 2-optimal algorithm. Moreover, the results yielded by the 2-optimal greedy algorithm are more skew to the right, with the exception of the Dantzig42 problem.

For the Dantzig42 and the Krolak problems, it is important to notice that not only the interquartile distance gets smaller when we replace the 3-optimal algorithm by its greedy version, but also the results are more condensed and closer to the optimal cost.

9.2 Point estimates of \((a, b, c)\)

As for the point estimates of the optimal cost for the Dantzig42 problem, the Zanakis estimates of \(a\) are equal to this optimal cost for all algorithms. This is easily explained: if in (7.6) we take into account that \(x_{(1)} = x_{(2)}\). As the WBE estimate of \(a\) is concerned, it is, in general, the smallest one, hence one might think that it overly underestimates the optimal cost. However, when we compare it to the optimal solution, we see that this is not the case for the 2-optimal and 2-optimal greedy algorithms; in fact, for these algorithms it is the most reasonable estimate of the location parameter.

The more “powerful” the algorithm the smaller the estimate of the scale parameter. This fact confirms something already apparent in the histograms and Box-Whisker plots: ranges and interquartile ranges get smaller as we progress from the 2-optimal to the 3-optimal greedy algorithm.

We could not compute the Zanakis estimate of the shape parameter for the 3-optimal greedy algorithm because \(x_{(\lceil np_{1} \rceil)} = x_{(16)} = x_{(1)}\) and therefore the denominator of \(\bar{c}\), defined by (7.8), cannot be calculated. We could neither compute the WBE estimates of all three parameters since we had a similar problem while obtaining the crude estimate \(\bar{c}_{0}\) thus preventing the evaluation of the estimates \(\bar{a}, \bar{b}\) and \(\bar{c}\).
<table>
<thead>
<tr>
<th>Estimates</th>
<th>Dantzig42</th>
<th>Krolak</th>
<th>Random</th>
<th>Gr120</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Location</td>
<td>Scale</td>
<td>Shape</td>
<td>Location</td>
</tr>
<tr>
<td>ML</td>
<td>671.545</td>
<td>44.419</td>
<td>2.472</td>
<td>21385.322</td>
</tr>
<tr>
<td></td>
<td>677.457</td>
<td>36.472</td>
<td>1.852</td>
<td>21282.262</td>
</tr>
<tr>
<td></td>
<td>679.202</td>
<td>0.332</td>
<td>0.159</td>
<td>21282.565</td>
</tr>
<tr>
<td></td>
<td>679.202</td>
<td>0.371</td>
<td>0.236</td>
<td>21285.443</td>
</tr>
<tr>
<td></td>
<td>679.202</td>
<td>37.012</td>
<td>1.782</td>
<td>21349.383</td>
</tr>
<tr>
<td></td>
<td>679.202</td>
<td>9.108</td>
<td>-</td>
<td>21285.443</td>
</tr>
<tr>
<td>WBE</td>
<td>676.441</td>
<td>37.699</td>
<td>2.127</td>
<td>21315.823</td>
</tr>
<tr>
<td></td>
<td>679.202</td>
<td>37.644</td>
<td>1.889</td>
<td>21268.367</td>
</tr>
<tr>
<td></td>
<td>679.202</td>
<td>-</td>
<td>-</td>
<td>21278.599</td>
</tr>
<tr>
<td>Minimum</td>
<td>679.202</td>
<td>-</td>
<td>-</td>
<td>21285.139</td>
</tr>
<tr>
<td>Optimal solution</td>
<td>679.202</td>
<td>21285.443</td>
<td>39486.840</td>
<td>1596.347</td>
</tr>
</tbody>
</table>

Table 9.1: Estimates of the location, scale and shape parameters for four instances of the TSP — listed in order corresponding to the 2-optimal, 2-optimal greedy, 3-optimal and 3-optimal greedy algorithms.
Apart from the Gr120, there is not a substantial difference in the estimates of the scale and shape parameters when we replace a \( \lambda \)-optimal algorithm by its greedy version. However, when the 2-optimal and 2-optimal greedy algorithms are replaced by the 3-optimal counterparts: the estimates of the scale parameters are much smaller reflecting the dispersion reduction depicted by Figure 9.2.

Unlike the results in Morais (1994) that refer to the Krolak problem and the 3-optimal algorithm, most instances of the TSP and algorithms are associated to estimates of the shape parameter larger than one (exponential distribution). Interestingly enough, when we make use of the 3-optimal greedy algorithm to solve the Krolak problem, the estimates of the shape are smaller than the unit.

### 9.3 Interval estimates for the optimal cost

In Table 9.2 we can find the relative range of the 6 different confidence intervals and an indicator of whether or not the optimal cost belongs to the confidence interval, for four instances of the TSP — listed in order corresponding to the 2-optimal, 2-optimal greedy, 3-optimal and 3-optimal greedy algorithms.

Table 9.2 confirms the results obtained by Morais (1994) and predicted by Golden and Alt (1979): when we replace the 2-optimal and 2-optimal greedy algorithms by the 3-optimal and 3-optimal greedy algorithms (respectively) there is a clear reduction of the relative ranges of the approximate confidence intervals for the optimal cost. This is essentially due to the fact that the range is an increasing function of the estimates of the scale parameter which are smaller when we use the 3-optimal algorithms.

The 3 Golden-Alt approximate confidence intervals always contain the optimal cost as opposed to the 3 Los-Lardinois approximate confidence intervals. This fact can be explained by the approximate confidence level which is very close to one for the 3 Golden-Alt intervals. A possible way out is to choose a larger confidence level for the 3 Los-Lardinois intervals.

Furthermore, we observe that for the 3 Golden-Alt intervals with the 3-optimal algorithms in the Random and Gr120 problems, the relative range is between 2% and 3%. In these cases, none of the algorithms yielded the optimal cost. Hence, a relative range smaller than 1% might be a good criteria to decide whether or not we should improve our solution.
Table 9.2: Relative range of the 6 different confidence intervals and whether or not the optimal solution belongs to the confidence interval for four instances of the TSP — listed in order corresponding to the 2-optimal, 2-optimal greedy, 3-optimal and 3-optimal greedy algorithms.
9.4 Kolmogorov-Smirnov goodness-of-fit test

Table 9.3 condenses the results of the Kolmogorov-Smirnov goodness-of-fit test for four instances of the TSP — listed in order corresponding to the 2-optimal, 2-optimal greedy, 3-optimal and 3-optimal greedy algorithms.

Most p-values are larger than the usual significance levels (α = 0.01, 0.05, 0.10), thus the Weibull distribution seems to be a reasonable model for the results of most instances of the TSP and algorithms.

In general, the ML estimates seem to be the ones that yield the best results (i.e., the larger p-values). Although, the estimates for the location parameter are more accurate when we use the 3-optimal and 3-optimal greedy algorithms, they have smaller p-values associated.

Finally, we should also point out that for the Krolak problem, we must rejected all of the conjectured distributions for the 3-optimal greedy algorithm. A possible reason can be the fact that all estimates of the shape parameter belong to (0, 1) and in this case our point estimates might not be accurate and should be used with care.

9.5 Comparison of (average) performances of the λ-optimal and λ-optimal greedy algorithms

In this section we shall confront the λ-optimal algorithms (λ = 2, 3) with their greedy versions and try to determine which algorithms perform better in average thus complementing the comparative analysis previously done.

First of all, let us have a look at the plots of the minimum costs (100 runs) for four instances of the TSP in Figure 9.3. Looking at them we have reasons to believe that the greedy version of the λ-optimal (λ = 2, 3) algorithms perform better in general. This fact has already been confirmed by the smaller sample ranges and the skewness to the right associated to the greedy algorithms for most of the four TSP instances (see Figure 9.2).

In order to statistically confirm that the λ-optimal greedy algorithms outperform in average the λ-optimal counterparts (λ = 2, 3), we perform the Mann-Whitney test. To do so, let X be the minimum cost attained by the λ-optimal algorithm and Y the minimum cost obtained by the greedy version of this algorithm, and let us confront the following null and alternative hypotheses:
<table>
<thead>
<tr>
<th>Conjectured distribution</th>
<th>Dantzig42</th>
<th>Krolak</th>
<th>Random</th>
<th>Gr120</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Obs. value</td>
<td>p-value</td>
<td>Obs. value</td>
<td>p-value</td>
</tr>
<tr>
<td>Weibull-ML</td>
<td>0.0814</td>
<td>0.5222</td>
<td>0.0496</td>
<td>0.9663</td>
</tr>
<tr>
<td></td>
<td>0.0623</td>
<td>0.8318</td>
<td>0.0765</td>
<td>0.6020</td>
</tr>
<tr>
<td></td>
<td>0.4375</td>
<td>0.0000</td>
<td>0.1263</td>
<td>0.0824</td>
</tr>
<tr>
<td></td>
<td>0.5306</td>
<td>0.0000</td>
<td>0.2596</td>
<td>0.0000</td>
</tr>
<tr>
<td>Weibull-Zanakis</td>
<td>0.1007</td>
<td>0.2624</td>
<td>0.0616</td>
<td>0.8430</td>
</tr>
<tr>
<td></td>
<td>0.0990</td>
<td>0.2807</td>
<td>0.1068</td>
<td>0.2043</td>
</tr>
<tr>
<td></td>
<td>0.3700</td>
<td>0.0000</td>
<td>0.0947</td>
<td>0.3307</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-</td>
<td>0.2239</td>
<td>0.0000</td>
</tr>
<tr>
<td>Weibull-WBE</td>
<td>0.1175</td>
<td>0.1266</td>
<td>0.5930</td>
<td>0.8736</td>
</tr>
<tr>
<td></td>
<td>0.1210</td>
<td>0.1071</td>
<td>0.0723</td>
<td>0.6726</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-</td>
<td>0.1438</td>
<td>0.0320</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>-</td>
<td>0.2140</td>
<td>0.0002</td>
</tr>
</tbody>
</table>

Table 9.3: Results of the goodness-of-fit tests for four instances of the TSP — listed in order corresponding to the 2-optimal, 2-optimal greedy, 3-optimal and 3-optimal greedy algorithms.
<table>
<thead>
<tr>
<th>Problem</th>
<th>Dantzig42</th>
<th>Krolak</th>
<th>Random</th>
<th>Gr120</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><img src="image1.png" alt="Graph" /></td>
<td><img src="image2.png" alt="Graph" /></td>
<td><img src="image3.png" alt="Graph" /></td>
<td><img src="image4.png" alt="Graph" /></td>
</tr>
<tr>
<td></td>
<td><img src="image5.png" alt="Graph" /></td>
<td><img src="image6.png" alt="Graph" /></td>
<td><img src="image7.png" alt="Graph" /></td>
<td><img src="image8.png" alt="Graph" /></td>
</tr>
</tbody>
</table>

Figure 9.3: Minimum costs (100 runs) for four instances of the TSP — listed in order corresponding to 2-optimal vs. 2-optimal greedy and 3-optimal vs. 3-optimal greedy algorithms.
\[ H_0 : E[Y] \leq E[X] \]
\[ H_1 : E[Y] > E[X]. \]

That is, we are conjecturing in \( H_0 \) that the greedy version of \( \lambda \)-optimal algorithm does not yield (in average) a larger minimum cost than the \( \lambda \)-optimal algorithm.

The observed values of the Mann-Whitney test statistic and the associated \( p \)-values are summarized in Table 9.4, for the four instances of the TSP.

<table>
<thead>
<tr>
<th>Problem</th>
<th>Dantzig42</th>
<th>Krolak</th>
<th>Random</th>
<th>Gr120</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obs. value</td>
<td>( p )-value</td>
<td>Obs. value</td>
<td>( p )-value</td>
<td>Obs. value</td>
</tr>
<tr>
<td>2-opt vs. 2-optG</td>
<td>5315.0</td>
<td>0.7796</td>
<td>7346.0</td>
<td>1.0000</td>
</tr>
<tr>
<td>3-opt vs. 3-optG</td>
<td>5281.0</td>
<td>0.7542</td>
<td>7307.5</td>
<td>1.0000</td>
</tr>
</tbody>
</table>

Table 9.4: Results of the Mann-Whitney test for four instances of the TSP — listed in order corresponding to the confrontation of 2-optimal (2-opt) vs. 2-optimal greedy (2-optG) and 3-optimal (3-opt) vs. 3-optimal greedy (3-optG) algorithms.

All \( p \)-values are very large and lead us to conclude that we should not reject \( H_0 \) for all the usual significance levels. However, we do not recommend the exclusive use of the greedy versions of the \( \lambda \)-optimal algorithms since for the Random problem both \( \lambda \)-optimal algorithms yielded better approximate solutions than their greedy counterparts (see Table 9.1).
Chapter 10

Final remarks

This report focuses on the statistical approach to the TSP, as described by Golden (1977) and Morais (1994).

As far as additions to previous statistical analysis of the TSP, we have:

- used greedy versions of the $\lambda$-optimal algorithm ($\lambda = 2, 3$) to obtain approximate costs;
- compared the approximate costs with the optimal cost obtained by the Concorde TSP Solver;
- also consider more instances of the TSP with slightly more cities or with randomly generated coordinates.

We firmly believe that the main results of this report are:

- the 3-optimal and 3-optimal greedy algorithms yield minimum costs that are more skew to the right than the ones obtained by the 2-optimal counterparts;
- the WBE estimates tend to be smaller than the remaining estimates of the optimal cost;
- the optimal cost does not always belong to the 3 Los-Lardinois approximate confidence intervals for this location parameter when an approximate confidence level of 95% is at use, thus we recommend considering larger confidence levels;
- the Weibull distribution seems to be a reasonable model to characterize the behavior of the heuristic solutions for most instances of the TSP and algorithms considered;
- the greedy versions of the \( \lambda \)-optimal algorithms have given better results in average.

A possibility of further work, that certainly deserves some consideration, is to investigate the benefits of using the 4-optimal algorithms since the computational power has largely increased in the last few years. Another one, is to propose different versions of the step 4 of the \( \lambda \)-optimal algorithm, as described in Chapter 5, in a similar fashion to the greedy version and apply the resulting algorithm to the TSP. Finally, we shoudl apply the statistical approach to large instances of the TSP.

We strongly hope that this report gives a stimulus and contributes to fill the gap between the performance analysis of approximate algorithms and Statistics.
Bibliography


