Aspects of sensitivity analysis for the traveling salesman problem

Poort, Edo Stephan van der

Chapter 5

Using stability information for solving the k-best TSP

In this chapter it is investigated to what extent information concerning the stability of optimal solutions can be used to solve the k-best problem. Since k-best problems are at least as hard as their corresponding optimization problems, it is especially of interest to see whether the k-best problems for $\mathcal{NP}$-hard combinatorial optimization problems can be solved more efficiently when having stability information. In this chapter the k-best Traveling Salesman Problem (TSP) is studied for the case that an optimal solution together with all its tolerances (the maximum perturbations of single edge lengths preserving the optimality of the given tour) are given. We focus on the following three issues. The first issue concerns the determination of a partial ordering of the tours based on the optimal solution of the TSP and its tolerances. The second one deals with the determination of polynomial algorithms for solving the 2-best TSP. Finally, it is shown that the k-best TSP is $\mathcal{NP}$-hard for $k \geq 3$ even if an optimal tour and its tolerances are known.

5.1 Introduction

In this chapter the relationship between the stability and k-best problems is studied for the TSP. The TSP is the problem of determining a round trip on which a number of cities is visited exactly once, and such that its length is minimal with respect to a given length vector. The *stability problem* for the TSP is the problem of determining the extent to which the length vector can be changed while preserving the optimality of a given optimal round trip. In the case that the lengths of single edges are subject to change, we are interested in the so-called *tolerances*, these being the maximum perturbations preserving the optimality of a given optimal solution. The *k-best TSP* is the problem of determining a set of round trips of cardinality k such that any round trip not in this set is not shorter than the longest round trip in the set.

Obviously, the stability and k-best problems are related in the sense that the solution to one of the problems also contains information about the solution to the other problem. For instance, if stability information is available one might be able to deduct information on the second-best solution, third-best solution, et cetera. Conversely, if the k-best solutions to the problem are known, one might be able to conclude some stability information on the optimal solution from that. Therefore, rather than solving the two problems separately from scratch, it makes sense to use knowledge of the solution to one of the two problems to solve the other one, at least partially.

In this chapter we investigate whether information on the stability of optimal solutions for the TSP can be used to solve the k-best TSP more efficiently. The opposite question, i.e. how to use a set of k-best solutions in order to determine stability information for a given optimal solution of the TSP, is considered in Chapter 6. The main question addressed in this chapter is the following. Assume that an optimal tour and its tolerances are known, what can we say about the set of k-best tours? First, we will show that, based on a given optimal tour and its tolerances, a partial ordering of the tours can be determined (see Section 5.3). Next, we will show that the length of a second-best tour can be determined in polynomial time, and that the 2-best TSP given an optimal tour and its tolerances is polynomially solvable when the set of 2-best tours is unique. Unfortunately, the 2-best TSP with given optimal tour and tolerances may take exponential time in the general case (see Section 5.4). Furthermore, it will be shown that the k-best TSP is $\mathcal{NP}$-hard for $k \geq 3$, even if an optimal tour and its tolerances are known (see Section 5.5). Finally, we investigate the possibilities of solving the k-best TSP by using tolerances with respect to a transformed length vector (see Section 5.6).

5.2 Definitions and basic results

For $n \geq 3$ and $1 \leq m \leq \binom{n}{2}$, consider the graph $G = (V, E)$ with the set of vertices $V = \{1, \ldots, n\}$ and the set of edges $E = \{e_1, \ldots, e_m\} \subseteq \{\{i, j\} : i, j \in V, i \neq j\}$. The length of edge e is a real number denoted by $d(e)$. The vector $d = [d(e_1), \ldots, d(e_m)]^T \in \mathbb{R}^m$ is called the *length vector* of the graph G and the pair (G, d) a *weighted graph*. The length of an edge set S with respect to d is given by $L_d(S) := \sum_{e \in S} d(e)$. A *Hamiltonian tour* (a *tour* for short) in the graph G is a subset of E that forms a cycle containing each vertex in V exactly once. By $\mathcal{H}$ we denote the set of all tours in G. The TSP is defined as the problem of finding a tour in $\arg\min \{L_d(H) : H \in \mathcal{H}\}$.

Let $1 \leq k \leq |\mathcal{H}|$. A set $\mathcal{H}(k) = \{H_{(1)}, \ldots, H_{(k)}\}$ of different tours in $\mathcal{H}$ satisfying

$$L_d(H_{(1)}) \leq L_d(H_{(2)}) \leq \ldots \leq L_d(H_{(k)}) \leq L_d(H) \text{ for all } H \in \mathcal{H} \backslash \mathcal{H}(k)$$

is called a *set of k-best tours*. The *k-best TSP* is defined as the problem of finding a set $\mathcal{H}(k)$ in (G, d). Throughout this chapter we assume, without loss of generality, that G contains at least k tours. This is no restriction as we may assume that all edges not in E have a sufficiently large length. Obviously, $\mathcal{H}(k)$ is in general not uniquely determined. In the extreme case, the so-called *constant TSP* (see e.g. Gilmore *et al.* [26] and Chapter 3 of this thesis), all tours have the same length, so that any subset of $\mathcal{H}$ of cardinality k is a set of k-best tours. The difference in length between a longest and a shortest tour in $\mathcal{H}(k)$ is denoted by L_k, i.e. $L_k := L_d(H_{(k)}) - L_d(H_{(1)})$. Note that, unlike $\mathcal{H}(k)$, L_k is uniquely determined by (G, d). Any ordered pair of tours $T_1, T_2 \in \mathcal{H}$ will be called *consecutive* when there is no tour T in $\mathcal{H}$ such that $L_d(T_1) < L_d(T) < L_d(T_2)$. For any $\mathcal{S} \subseteq \mathcal{H}$, define $\cup\mathcal{S} := \cup\{H : H \in \mathcal{S}\}$ and $\cap\mathcal{S} := \cap\{H : H \in \mathcal{S}\}$. Hence, $\cap\mathcal{H}(k)$ is the set of edges that are in all tours of $\mathcal{H}(k)$, and $\cup\mathcal{H}(k)$ is the set of edges that are in at least one of the tours in $\mathcal{H}(k)$.

Throughout this chapter we use the following example.

Example. Figure 5.1 shows a weighted graph (G, d) with all its tours listed in order of length. Note that all tours have different length. This implies that the set $\mathcal{H}(k)$ is unique, for $k = 1, \ldots, 5$. For instance, $\mathcal{H}(3) = \{H_{(1)}, H_{(2)}, H_{(3)}\}$ with $L_3 = 2$, $\cap\mathcal{H}(3) = \{\{2, 3\}, \{5, 6\}\}$, and $\cup\mathcal{H}(3) = E$. $\square$

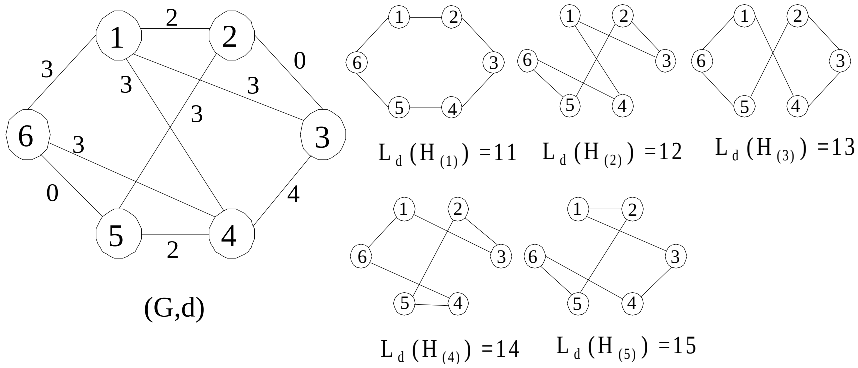

Figure 5.1: A weighted graph (G, d) with all its tours.

The *tolerance problem* for the TSP is to determine by how much the length of a single edge can be changed while preserving the optimality of the tour $H_{(1)}$. In other words, the tolerance problem is the problem of finding, for each $e \in E$, the maximum increase $u(e)$ and the maximum decrease $l(e)$ in the edge length $d(e)$ while preserving the optimality of $H_{(1)}$, under the assumption that the lengths of all other edges remain unchanged (cf. Libura [53]). For each $e \in E$, $u(e)$ and $l(e)$ are called the *upper* and *lower tolerances* of edge e with respect to $H_{(1)}$ and d. Note that $u(e)$ and $l(e)$ may be infinite. Also note that $u(e)$ and $l(e)$ depend on $H_{(1)}$, i.e. if multiple optimal solutions exist then $u(e)$ and $l(e)$ are only valid for the $H_{(1)}$ selected. In Chapter 2 it has been shown that edge tolerances are "hard" to compute, even if an optimal solution is given.

In Libura [53] it is shown that for each edge one of its tolerances is infinite and the other one can be calculated by determining the optimal value of an auxiliary instance of the TSP defined on a restricted set of tours. Note that this result is an instance of Theorem 2.2. This fact is restated in the following theorem.

Theorem 5.1 *(Libura [53]).*
For $e \in H_{(1)}$ it holds that $l(e) = \infty$ and that

$$u(e) = \min\{L_d(H) \ : H \in \mathcal{H}, e \notin H\} - L_d(H_{(1)}),$$

For $e \in E \backslash H_{(1)}$ it holds that $u(e) = \infty$ and that

$$l(e) = \min\{L_d(H) : H \in \mathcal{H} \ , \ e \in H\} - L_d(H_{(1)}). \qquad \square$$

The following corollary identifies the set of edges for which only one of the tolerances is finite as the set of edges that occur in at least one tour but not in all of them.

Corollary 5.2 $\{e \in E : u(e) < \infty \text{ } or \text{ } l(e) < \infty\} = \cup\mathcal{H} \backslash \cap \mathcal{H}$.

Proof. It follows from Theorem 5.1 that $\{e \in E : u(e) < \infty \text{ or } l(e) < \infty\} = \{e \in H_{(1)} : u(e) < \infty\} \cup \{e \in E\backslash H_{(1)} : l(e) < \infty\}$. We have that $\{e \in H_{(1)} : u(e) < \infty\} = \{e \in H_{(1)} : \{H \in \mathcal{H} : e \notin H\} \neq \emptyset\} = \{e \in H_{(1)} : e \notin \cap\mathcal{H}\} = H_{(1)} \backslash \cap \mathcal{H}$. It can be shown in a similar way that $\{e \in E\backslash H_{(1)} : l(e) < \infty\} = \cup\mathcal{H}\backslash H_{(1)}$. Hence, we have that $\{e \in H_{(1)} : u(e) < \infty\} \cup \{e \in E\backslash H_{(1)} : l(e) < \infty\} = (H_{(1)} \backslash \cap \mathcal{H}) \cup (\cup\mathcal{H}\backslash H_{(1)}) = \cup\mathcal{H} \backslash \cap \mathcal{H}$. □

Example (continued). Figure 5.2 gives the lower and upper tolerances with respect to $H_{(1)}$ and (G, d) in Figure 5.1. For instance, $l(\{1, 4\}) = 1$ and $u(\{2, 3\}) = 4$. □

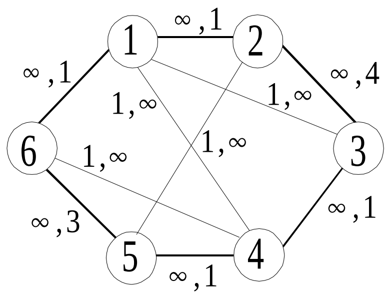

Figure 5.2: Lower (first number) and upper tolerances (second number).

As mentioned earlier, we assume throughout this chapter that for a given weighted graph (G, d), with the set of tours denoted by $\mathcal{H}$, an optimal tour $H_{(1)}$ and all its tolerances are given. For convenience, we will use a shorthand notation for the various problems that will be discussed in this chapter. We therefore denote the k-best TSP with known optimal tour and tolerances by k-best TSP$[H_{(1)}$, tolerances], where the "[·]-part" denotes the known information. Similar problems are denoted under the same convention.

5.3 A partial ordering of the tours based on the optimal tour and its tolerances

In this section a partial ordering of the tours in (G, d) will be determined based on the knowledge of an optimal solution and all its tolerances.

Let $\mathcal{T}$ denote the set of finite different tolerance values with respect to (G, d) and $H_{(1)}$, i.e.

$$\mathcal{T} := \{u(e) : e \in H_{(1)}, u(e) < \infty\} \cup \{l(e) : e \in E \backslash H_{(1)}, l(e) < \infty\}.$$

Note that $|\mathcal{T}| \leq |\cup \mathcal{H} \backslash \cap \mathcal{H}|$. In the following theorem it is shown that the set $\mathcal{T}$ is empty if and only if G contains exactly one tour.

Theorem 5.3 *$\mathcal{T} = \emptyset$ if and only if $|\mathcal{H}| = 1$.*

Proof. We first prove the -*if*- part. Let $\mathcal{H} = \{H_{(1)}\}$. Then, $(\cap \mathcal{H}) \cup (E \backslash \cup \mathcal{H}) = H_{(1)} \cup (E \backslash H_{(1)}) = E$. Recall from Corollary 5.2 that the set of edges for which one of the tolerances is finite is equal to $\cup \mathcal{H} \backslash \cap \mathcal{H}$. Consequently, the lower and upper tolerances for all edges in E are infinite. Hence $\mathcal{T} = \emptyset$. Consider now the -*only if*- part. Suppose, to the contrary, that $\mathcal{T} = \emptyset$ and $|\mathcal{H}| \neq 1$. Since, by assumption, G is Hamiltonian, it follows that $|\mathcal{H}| \geq 2$. Let $\{H_{(1)}, H_{(2)}\}$ be a set of 2-best tours in (G, d). Clearly, $H_{(2)} \backslash H_{(1)} \neq \emptyset$. Take any $e \in H_{(2)} \backslash H_{(1)}$. Then, obviously, $\min\{L_d(H) : H \in \mathcal{H}, e \in H\} = \min\{L_d(H) : H \in \mathcal{H} \backslash \{H_{(1)}\}, e \in H\} = L_d(H_{(2)})$. We therefore obtain, from Theorem 5.1, that $l(e) = L_d(H_{(2)}) - L_d(H_{(1)}) = L_2 < \infty$. Hence, $\mathcal{T} \neq \emptyset$, which gives a contradiction. □

Let the elements in $\mathcal{T}$ be ordered in such a way that $\mathcal{T} = \{t_1, \ldots, t_{|\mathcal{T}|}\}$ and $0 \leq t_1 < t_2 < \ldots < t_{|\mathcal{T}|}$, i.e. t_j denotes the j-th smallest tolerance value with respect to $H_{(1)}$ and (G, d). In the following theorem it is shown that t_j is at least equal to the difference in length between a longest and a shortest tour in a set of $(j+1)$-best tours.

Theorem 5.4 *For all $j = 1, \ldots, |\mathcal{T}|$, $t_j \geq L_{j+1}$.*

Proof. The proof is by induction. We first prove that $t_1 \geq L_2$. Take any $e \in H_{(1)}$. Since $\min\{L_d(H) : H \in \mathcal{H}, e \notin H\} = \min\{L_d(H) : H \in \mathcal{H} \backslash \{H_{(1)}\}, e \notin H\} \geq \min\{L_d(H) : H \in \mathcal{H} \backslash \{H_{(1)}\}\} = L_d(H_{(2)})$, it follows from Theorem 5.1 that $u(e) \geq L_2$. Similarly, for each $e \in E \backslash H_{(1)}$, it holds that $l(e) \geq L_2$. Since t_1 is the smallest tolerance value, we have that $t_1 \geq L_2$.

Next, assume that $t_j \geq L_{j+1}$ for $j = 1, \ldots, i-1$ and $i \in \{2, \ldots, |\mathcal{T}|\}$. It has to be shown that $t_i \geq L_{i+1}$. Since $t_i > t_{i-1}$, we have that $t_i > L_i$. Assume, to the contrary, that $t_i < L_{i+1}$. Recall from the definition of t_i that there is some $e \in E$ such that either $t_i = u(e)$ or $t_i = l(e)$. Moreover, Theorem 5.1 implies that there is some $H \in \mathcal{H}$ such that $t_i = L_d(H) - L_d(H_{(1)})$. Then, $L_d(H_{(i)}) - L_d(H_{(1)}) = L_i < t_i = L_d(H) - L_d(H_{(1)}) < L_{i+1} = L_d(H_{(i+1)}) - L_d(H_{(1)})$. Consequently, $L_d(H_{(i)}) < L_d(H) < L_d(H_{(i+1)})$ which is impossible since $H_{(i)}$ and $H_{(i+1)}$ are consecutive tours in the ordering of the tours in $\mathcal{H}$. This proves that $t_i \geq L_{i+1}$, which completes the proof. □

Example (continued). Consider (G, d) in Figure 5.1. The tolerances with respect to $H_{(1)}$ and d are given in Figure 5.2. We have that $\mathcal{T} = \{1, 3, 4\}$, i.e. $t_1 = 1$, $t_2 = 3$, and $t_3 = 4$. Note that $t_1 = L_2 = 1$, $t_2 > L_3 = 2$, and $t_3 > L_4 = 3$. □

For each tour H in $\mathcal{H}$ satisfying $L_d(H) = L_d(H_{(1)}) + t_j$, we say that H *corresponds to* t_j. In the following theorem it is shown that there is at least one tour corresponding to each finite tolerance value.

Theorem 5.5 *For each $j = 1, \ldots, |\mathcal{T}|$, there is at least one tour H in (G, d) that corresponds to t_j.*

Proof. By definition, for each $j = 1, \ldots, |\mathcal{T}|$, there is an $e \in E$ such that either $t_j = u(e)$ or $t_j = l(e)$. Consider the case that $t_j = u(e)$. From Theorem 5.1, it follows that $t_j = \min\{L_d(H) : H \in \mathcal{H}, e \notin H\} - L_d(H_{(1)})$. Let $H_j \in \arg\min\{L_d(H) : H \in \mathcal{H}, e \notin H\}$. Then, clearly $L_d(H_j) = \min\{L_d(H) : H \in \mathcal{H}, e \notin H\} = L_d(H_{(1)}) + t_j$, i.e. H_j corresponds to t_j. The case that $t_j = l(e)$ can be proven similarly. □

For $j = 1, \ldots, |\mathcal{T}|$, let H_j denote a tour that corresponds to t_j. Since $0 \leq t_1 < t_2 < \ldots < t_{|\mathcal{T}|}$, we obtain the following partial ordering of the tours in (G, d)

$$L_d(H_{(1)}) \leq L_d(H_1) < L_d(H_2) < \ldots < L_d(H_{|\mathcal{T}|}).$$

Note that the tours $H_{(1)}, H_1, H_2, \ldots, H_{|\mathcal{T}|}$ need not be (pairwise) consecutive in (G, d), even if all tours in $\mathcal{H}$ have different length. This is demonstrated in the following example.

Example (continued). Consider (G, d) in Figure 5.1 with $|\cup\mathcal{H} \setminus \cap\mathcal{H}| = 10$ and $|\mathcal{T}| = 3$. It holds that $t_1 = L_d(H_{(2)}) - L_d(H_{(1)})$, $t_2 = L_d(H_{(4)}) - L_d(H_{(1)})$, and $t_3 = L_d(H_{(5)}) - L_d(H_{(1)})$. Hence, $H_{(2)}$, $H_{(4)}$, and $H_{(5)}$

correspond to t_1, t_2 and t_3, respectively. Hence, from the finite tolerances the partial ordering $L_d(H_{(1)}) < L_d(H_{(2)}) < L_d(H_{(4)}) < L_d(H_{(5)})$ can be determined. Clearly, the tours H_1 and H_2 are not consecutive in (G, d) since $L_d(H_1) < L_d(H_{(3)}) < L_d(H_2)$. □

In Theorem 5.5, we have shown the existence of tours corresponding to the finite tolerance values with respect to (G, d) and $H_{(1)}$. We now turn to the question whether it is possible to actually construct tours H_j, still assuming that $H_{(1)}$ and its tolerances are known. In the following theorem, it is shown that tours H_j can be determined partly from $H_{(1)}$ and its tolerances.

Theorem 5.6 *For each $j = 1, \ldots, |\mathcal{T}|$, let H_j be a tour that corresponds to t_j. It holds that*

$$\{e \in H_{(1)} : u(e) > t_j\} \subseteq H_j \text{ and } \{e \in E \backslash H_{(1)} : l(e) > t_j\} \cap H_j = \emptyset,$$

Moreover, if H_j is the unique tour corresponding to t_j then also

$$\{e \in E \backslash H_{(1)} : l(e) = t_j\} \subseteq H_j \text{ and } \{e \in H_{(1)} : u(e) = t_j\} \cap H_j = \emptyset.$$

Proof. Let $j \in \{1, \ldots, |\mathcal{T}|\}$, and take any $H_j \in \mathcal{H}$ satisfying $L_d(H_j) = L_d(H_{(1)}) + t_j$. We first show that $\{e \in H_{(1)} : u(e) > t_j\} \subseteq H_j$. Take any $e \in H_{(1)}$ such that $u(e) > t_j$. Suppose, to the contrary, that $e \notin H_j$. Then, $\min\{L_d(H) : H \in \mathcal{H}, e \notin H\} \leq L_d(H_j)$, and consequently applying Theorem 5.1 gives that $u(e) \leq L_d(H_j) - L_d(H_{(1)}) = t_j$, which is a contradiction. The proof that $\{e \in E \backslash H_{(1)} : l(e) > t_j\} \cap H_j = \emptyset$ is similar and therefore omitted here.

Now, assume that H_j is determined uniquely in (G, d). We first show that $\{e \in E \backslash H_{(1)} : l(e) = t_j\} \subseteq H_j$. Take any $e \in E \backslash H_{(1)}$ such that $l(e) = t_j$. Recall, from Theorem 5.5, that there is a $H \in \mathcal{H}$ such that $e \in H$ and $l(e) = L_d(H) - L_d(H_{(1)})$. Since $l(e) = t_j$, we have that $L_d(H) - L_d(H_{(1)}) = L_d(H_j) - L_d(H_{(1)})$. However, H_j is unique in (G, d), meaning that $H = H_j$, and hence, $e \in H_j$. Finally, we show that $\{e \in H_{(1)} : u(e) = t_j\} \cap H_j = \emptyset$. Take any $e \in H_{(1)}$ such that $u(e) = t_j$. Again, there is a $H \in \mathcal{H}$ such that $e \notin H$ and $u(e) = L_d(H) - L_d(H_{(1)})$. Since $u(e) = t_j$, it follows that $L_d(H) = L_d(H_{(1)}) + t_j = L_d(H_j)$. Then, from the assumption that H_j is unique in (G, d), we obtain that $H = H_j$. Hence, $e \notin H_j$, which completes the proof. □

Example (continued). Consider (G, d) in Figure 5.1. Recall that the tours $H_{(2)}$, $H_{(4)}$, and $H_{(5)}$ correspond uniquely to $t_1 = 1$, $t_2 = 3$, and

$t_3 = 4$, respectively. The sets of edges that are to be included in the tours H_j and the sets of edges that are to be excluded from the tours H_j are given in Table 5.1. □

j	t_j	$\{e \in H_{(1)} : u(e) > t_j\} \cup$ $\{e \in E \backslash H_{(1)} : l(e) = t_j\}$	$\{e \in E \backslash H_{(1)} : l(e) > t_j\} \cup$ $\{e \in H_{(1)} : u(e) = t_j\}$
1	1	$\{\{2,3\},\{5,6\}\} \cup$ $\{\{1,3\},\{2,5\},\{1,4\},\{4,6\}\}$	$\emptyset \cup \{\{1,2\},\{3,4\},\{6,1\},\{4,5\}\}$
2	3	$\{\{2,3\}\} \cup \emptyset$	$\emptyset \cup \{\{5,6\}\}$
3	5	$\emptyset \cup \emptyset$	$\emptyset \cup \{\{2,3\}\}$

Table 5.1: Subsets of edges (not) being included in tour H_j.

The example shows that, in general, the tours H_j can only be determined partly from $H_{(1)}$ and its tolerances by using Theorem 5.6. Using Theorem 5.6 it is undecided whether edges e with $u(e) < t_j$ or $l(e) < t_j$ are in H_j or not, and if H_j is not unique the same holds for edges with $u(e) = t_j$ or $l(e) = t_j$. However, in the case that H_1 is the unique tour corresponding to t_1, we have the following result.

Corollary 5.7 *If $|\mathcal{T}| \geq 1$ and H_1 is the unique tour corresponding to t_1 then*

$$H_1 = \{e \in H_{(1)} : u(e) > t_1\} \cup \{e \in E \backslash H_{(1)} : l(e) = t_1\}.$$

Proof. It follows from Theorem 5.6 that

$$\{e \in H_{(1)} : u(e) > t_1\} \cup \{e \in E \backslash H_{(1)} : l(e) = t_1\} \subseteq H_1$$

and that

$$\begin{aligned} H_1 &\subseteq E \backslash (\{e \in H_{(1)} : u(e) = t_1\} \cup \{e \in E \backslash H_{(1)} : l(e) > t_1\}) \\ &= \{e \in H_{(1)} : u(e) > t_1\} \cup \{e \in E \backslash H_{(1)} : l(e) = t_1\}. \end{aligned}$$

Hence, $H_1 = \{e \in H_{(1)} : u(e) > t_1\} \cup \{e \in E \backslash H_{(1)} : l(e) = t_1\}$. □

Note that the opposite of Corollary 5.7 does not hold, i.e. the fact that $H := \{e \in H_{(1)} : u(e) > t_1\} \cup \{e \in E \backslash H_{(1)} : l(e) = t_1\}$ is a tour in (G, d) does not imply that H corresponds to t_1. For example, consider (G, d) in Figure 5.3(a) with $H_{(1)}$ indicated by the bold lines.

The tolerances as well as the tour H, indicated by the bold lines, are given in Figure 5.3(b). Note that $L_d(H) - L_d(H_{(1)}) = 2 > t_1 = 1$, meaning that H does not correspond to t_1.

This completes the discussion on the partial ordering of the tours that can be obtained from a given optimal tour and its tolerances. In the following section we will show that the smallest tolerance value t_1 corresponds to the difference in length between an optimal and a second-best tour and that H_1 is a second-best tour in the case that the set of 2-best tours is unique.

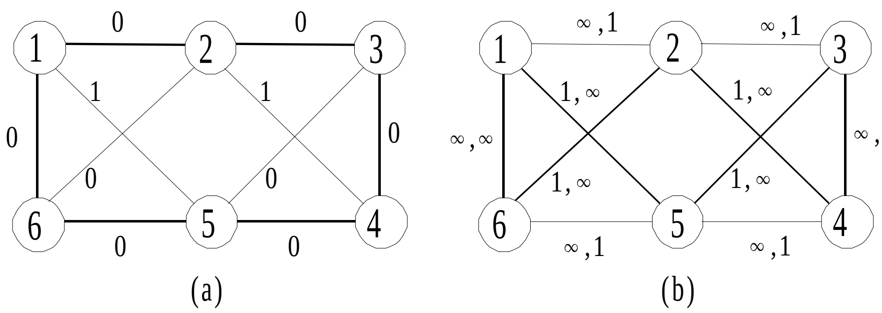

Figure 5.3: (a) (G, d) with $H_{(1)}$ indicated by the bold lines and (b) the lower and upper tolerances.

5.4 The 2-best TSP with given optimal tour and tolerances

In this section we consider the 2-best TSP for the case that an optimal tour and its tolerances are given, i.e. the 2-best TSP[$H_{(1)}$,tolerances]. It will be shown that the length of a second-best tour can be determined in polynomial time and that if $\mathcal{H}(2)$ is unique in (G, d) then also the 2-best TSP[$H_{(1)}$,tolerances] can be solved in polynomial time. Furthermore, it will be argued that, in general, solving the 2-best TSP[$H_{(1)}$,tolerances] may take exponential time.

We start this section by showing that the difference in length between $H_{(1)}$ and a second-best tour is equal to the smallest value of both the upper and the lower tolerances.

Theorem 5.8 *If* $|\mathcal{T}| \geq 1$ *then*

$$L_2 = \min\{u(e) : e \in H_{(1)}\} = \min\{l(e) : e \in E \backslash H_{(1)}\} = t_1.$$

Proof. Recall from Theorem 5.3 that $|\mathcal{T}| \geq 1$ implies that $|\mathcal{H}| \geq 2$. Hence, L_2 is well defined. We first show that $L_2 = \min\{u(e) : e \in H_{(1)}\}$. Recall from Theorem 5.1 that, for each $e \in H_{(1)}$, $u(e) = \min\{L_d(H) : H \in \mathcal{H}, e \notin H\} - L_d(H_{(1)})$. Since $\mathcal{H}\backslash\{H_{(1)}\} = \cup\{\{H \in \mathcal{H} : e \notin H\} : e \in H_{(1)}\}$, it follows that

$$\begin{aligned}
&\min\{u(e) : e \in H_{(1)}\} \\
&\quad = \min\{\min\{L_d(H) : H \in \mathcal{H}, e \notin H\} : e \in H_{(1)}\} - L_d(H_{(1)}) \\
&\quad = \min\{L_d(H) : H \in \mathcal{H}\backslash\{H_{(1)}\}\} - L_d(H_{(1)}),
\end{aligned}$$

which, according to the definition of $L_d(H_{(2)})$, is equal to $L_d(H_{(2)}) - L_d(H_{(1)}) = L_2$. It can be shown in a similar way that $L_2 = \min\{l(e) : e \in E\backslash H_{(1)}\}$. Finally, note that t_1 is the smallest tolerance value in (G, d) with respect to $H_{(1)}$. Hence, $\min\{u(e) : e \in H_{(1)}\} = \min\{l(e) : e \in E\backslash H_{(1)}\} = t_1$. □

Let H be any tour that corresponds to t_1. As a corollary of Theorem 5.8, it follows that H is a second-best tour.

Corollary 5.9 *For each* $H \in \mathcal{H}$*, it holds that* $\{H_{(1)}, H\}$ *is a set of* 2*-best tours in* (G, d) *if and only if* $L_d(H) = L_d(H_{(1)}) + t_1$.

Proof. The *–only if–* part is straightforward. Now consider the *–if–* part. Take any $H_1 \in \mathcal{H}$ satisfying $L_d(H_1) = L_d(H_{(1)}) + t_1$. Then, it follows from Theorem 5.8 that $L_d(H_1) = L_d(H_{(1)}) + L_2 = \min\{L_d(H) : H \in \mathcal{H}\backslash\{H_{(1)}\}\}$, which proves that $L_d(H_1) \leq L_d(H)$ for all $H \in \mathcal{H}\backslash\{H_{(1)}\}$. Hence, $\{H_{(1)}, H_1\}$ is a set of 2-best tours in (G, d). □

Note that, by Corollary 5.9, in order to find a set of 2-best tours, it is sufficient to find a tour corresponding to t_1. In the remainder of this section, we distinguish between the cases that $\mathcal{H}(2)$ is unique and that $\mathcal{H}(2)$ is not unique.

First consider the case that $\mathcal{H}(2)$ is unique. As a corollary of Theorem 5.8, we have the following result.

Corollary 5.10 *If* $\mathcal{H}(2)$ *is unique then*

$$H_{(2)} = \{e \in H_{(1)} : u(e) > t_1\} \cup \{e \in E\backslash H_{(1)} : l(e) = t_1\}.$$

Proof. Follows directly from Corollaries 5.7 and 5.9. □

Example (continued). Consider (G, d) in Figure 5.1 with the tolerances given in Figure 5.2. Since all tours in (G, d) have different length, $\mathcal{H}(2)$ is clearly uniquely determined in (G, d). We have that $t_1 = 1$, so that $H_{(2)} = \{e \in H_{(1)} : u(e) > 1\} \cup \{e \in E \backslash H_{(1)} : l(e) = 1\} = \{\{2,3\},\{5,6\}\} \cup \{\{1,3\},\{1,4\},\{2,5\},\{4,6\}\}$. □

Now consider the case that $\mathcal{H}(2)$ is not unique, i.e. there are several tours corresponding to t_1 in G. Our aim is to construct at least one such a tour. Unfortunately, the fact that there is more than one second-best tour does not make it easier to find one. As far as we know for these cases there is no polynomial algorithm for finding a second-best tour (still assuming that an optimal tour and its tolerances are known). In order to make the situation clear, let us summarize. According to Corollary 5.9, starting from an optimal tour and its tolerances, the length of a second-best tour $H_{(2)}$ can be computed by $L_d(H_{(2)}) = L_d(H_{(1)}) + t_1$. By using Theorem 5.6, we also know that if $e \in H_{(1)}$ and $u(e) > t_1$ then $e \in H_{(2)}$ and that if $e \notin H_{(1)}$ and $l(e) > t_1$ then $e \notin H_{(2)}$. So, we can always determine the length of a second-best tour, but the more edges have tolerance value equal to t_1 the less we know about the tour $H_{(2)}$ itself. Obviously, the more tours there are that together with $H_{(1)}$ constitute a set of 2-best tours, the more edges have tolerance value equal to t_1. This implies that, roughly speaking, the more second-best tours there are, the less we can say about them. For example, in the special case that all tours have the same length, i.e. all edges have tolerance value equal to zero, then no information about a second-best tour can be derived form Theorem 5.6. Another, less trivial, example for which a similar result holds is given in Papadimitriou & Steiglitz [68] where a graph (G, d) is presented on $n = 8k$ vertices with one optimal tour and $2^{k-1}(k-1)!$ second-best tours.

With the above remarks in mind we formulate the following conjecture.

Conjecture 5.11 *There is no polynomial time algorithm for solving the 2-best TSP[$H_{(1)}$,tolerances] unless $\mathcal{P} = \mathcal{NP}$.*

Although we are not able to prove the above statement, we have several reasons to believe that it is true. In the remainder of this section, we state several facts that support this conjecture.

First of all, note that the 2-best TSP with known optimal tour, i.e. the 2-best TSP[$H_{(1)}$], is $\mathcal{NP}$-hard. This follows immediately from the fact that the *Second Hamiltonian Cycle Problem*, i.e. the problem of determining whether a given graph with a tour contains a second tour, is $\mathcal{NP}$-complete; see Johnson & Papadimitriou [40].

As we discussed before, by knowing an optimal tour and its tolerances, we are able to derive two additional pieces of information:

1. The length of a second-best tour, i.e. $L_d(H_{(2)})$, and
2. Some information about which edges are and which edges are not contained in a second-best tour (by using Theorem 5.6).

It will be shown that having this information is not enough to solve the 2-best TSP[$H_{(1)}$,tolerances] in polynomial time.

First, we consider the complexity of the 2-best TSP[$H_{(1)}$,$L_d(H_{(2)})$], i.e. the 2-best TSP with given optimal tour and length of a second-best tour. Clearly, the problem here is only to construct the second-best tour, i.e. there is no corresponding decision problem (the existence of a second-best tour is given). In Papadimitriou [67] such problems are called *total functions*. In the following theorem it is shown that the existence of a polynomial time algorithm for solving the 2-best TSP[$H_{(1)}$,$L_d(H_{(2)})$] is very unlikely.

Theorem 5.12 *The 2-best TSP[$H_{(1)}$,$L_d(H_{(2)})$] cannot be solved in polynomial time unless $\mathcal{P} = \mathcal{NP}$.*

Proof. We will show that the existence of a polynomial time algorithm for solving the 2-best TSP[$H_{(1)}$,$L_d(H_{(2)})$] would imply that $\mathcal{P} = \mathcal{NP}$. Suppose that there exists such a polynomial time algorithm, say A, whose running time is bounded by some polynomial $p(n)$. Take an instance of the Second Hamiltonian Cycle Problem, i.e. a graph $G = (V, E)$ with a tour $H \in \mathcal{H}$, and reduce it to the 2-best TSP with a given optimal tour in the standard way, i.e. every vertex in V becomes a city, every edge in E gets length 1, and every edge not in E gets length 2. Let A' denote the modification of algorithm A such that it counts each elementary operation and stops when either A' finds a solution or the number of elementary operations exceeds $p(n)$. Apply algorithm A' to this instance of the 2-best TSP with the length of the second-best tour being equal to n.

Now there are two possibilities. Firstly, assume that the graph G has a second tour. Then, algorithm A' receives a feasible input with the correct length n. Consequently, algorithm A' will come up with a set of 2-best tours in (G, d) with $L_2 = 0$ using at most $p(n)$ operations. Secondly, assume that the graph does not have a second tour. Then, it can either happen that algorithm A' stops because the number of elementary operations exceeds $p(n)$ or algorithm A' produces some (unspecified) output. In the latter case, it can be easily checked in polynomial time whether the output is a set of 2-best tours in (G, d) with $L_2 = 0$. So, algorithm A' either finds a second tour with length n, or we know from the behavior of algorithm A' that G does not contain a second tour with length n. Since the Second Hamiltonian Cycle Problem is $\mathcal{NP}$-complete, the existence of such an algorithm would imply that $\mathcal{P} = \mathcal{NP}$. □

In Theorem 5.12 it has been shown that in order to solve the 2-best TSP[$H_{(1)}$,tolerances] in polynomial time, the knowledge of the length of a second-best tour is not sufficient. This implies that more information is required, i.e. also the second piece of information (the knowledge about which edges to include and which edges to exclude from $H_{(2)}$) has to be used in order to be able to find a polynomial time algorithm. However, as we have already discussed, it might turn out that this piece of information is very limited, and consequently does not provide additional clues. This gives us yet another reason to believe that Conjecture 5.11 is true.

We close this section by presenting some results from looking at the 2-best TSP[$H_{(1)}$,tolerances] from a different angle. Again, recall from Theorem 5.6 that some edges are to be included and some edges are to be excluded from $H_{(2)}$. In fact, the crux of the problem is that we do not know which of the edges having tolerance value equal to t_1 are to be included in $H_{(2)}$ and which not. However, for a given set S in $\{e \in E \backslash H_{(1)} : l(e) = t_1\}$, the set of tours on the edge set $H_{(1)} \cup S$ that contain S can be determined in polynomial time. Moreover, since the length of a second-best tour is known, it can also be checked in polynomial time whether this set of tours contains a second-best tour. We have the following result.

Theorem 5.13 *Assume that an optimal tour $H_{(1)}$, its tolerances, and the length of a second-best tour $L_d(H_{(2)})$ are known. For each $S \subseteq \{e \in E \backslash H_{(1)} : l(e) = t_1\}$, the problem of determining the set $\{H \in \mathcal{H} : H \backslash H_{(1)} = S, L_d(H) = L_d(H_{(1)}) + t_1\}$ is solvable in $O(m)$ time.*

Proof. See Appendix 5.A. □

Note that if $\{H \in \mathcal{H} : H\backslash H_{(1)} = S, L_d(H) = L_d(H_{(1)}) + t_1\}$ is nonempty, each tour contained in it is a second-best tour. Furthermore, note that Theorem 5.13 implies an algorithm for solving the 2-best TSP[$H_{(1)}$,tolerances]; namely the following: First determine $L_d(H_{(2)})$ and $\{e \in E\backslash H_{(1)} : l(e) = t_1\}$. Then pick a subset $S \subseteq \{e \in E\backslash H_{(1)} : l(e) = t_1\}$ and determine $\{H \in \mathcal{H} : H\backslash H_{(1)} = S, L_d(H) = L_d(H_{(1)})+t_1\}$. If this set is nonempty then a second-best tour has been found. Otherwise, select another subset S and continue doing so until the set $\{H \in \mathcal{H} : H\backslash H_{(1)} = S, L_d(H) = L_d(H_{(1)}) + t_1\}$ is nonempty. Unfortunately, this algorithm is, in general, not polynomial because in the worst case all possible subsets S in $\{e \in E\backslash H_{(1)} : l(e) = t_1\}$ are to be considered. Actually, this is in line with Conjecture 5.11. In conclusion we can say that having the set S is a vital piece of information in order to solve the 2-best TSP[$H_{(1)}$,tolerances] in polynomial time. However, there is no way to obtain S from the tolerances, so that the above discussion provides another reason to believe that Conjecture 5.11 is true.

5.5 The k-best TSP with given optimal tour and tolerances is $\mathcal{NP}$-hard for k ≥ 3

In this section we will show that the problem of solving the k-best TSP is $\mathcal{NP}$-hard for $k \geq 3$ even if an optimal tour and its tolerances are given.

Throughout this section it will be assumed that $\mathcal{H}(2)$ is uniquely determined. Obviously, if this assumption does not hold then the problem of finding a third-best tour is more or less the same problem as finding a second-best tour.

Unfortunately, the length of a third-best tour cannot be determined in a similar way as the length of a second-best tour. The reason is that, in general, the equality $L_d(H_{(3)}) = L_d(H_{(1)})+t_2$ does not hold even if all tours in $\mathcal{H}$ have different length. This is demonstrated in the following example.

Example (continued). Consider (G, d) in Figure 5.1. Recall that all tours have different length. We have that $t_2 = L_d(H_{(4)}) - L_d(H_{(1)}) = 3$. Consequently, $H_{(4)}$ in (G, d) corresponds to t_2, so that $L_d(H_{(3)}) - L_d(H_{(1)}) \neq t_2$. □

The above example shows that there may exist tours in (G, d) for which the tolerances do not provide any information about their length.

Further analysis shows that this is the case for all tours H in $\mathcal{H}$ for which all edges that are both in $H_{(1)}$ and $H_{(2)}$ are also in H and that all other edges in H are either in $H_{(1)}$ or in $H_{(2)}$. The set of all such tours will be denoted by $\mathcal{H}_3$, i.e.

$$\mathcal{H}_3 := \{H \in \mathcal{H} \setminus \mathcal{H}(2) : \cap\mathcal{H}(2) \subseteq H \subseteq \cup\mathcal{H}(2)\}.$$

The following theorem gives an expression for the length of a third-best tour.

Theorem 5.14 *Let $\mathcal{H}(2)$ be the unique set of 2-best tours in (G,d). If $|\mathcal{T}| \geq 2$ then*

$$L_d(H_{(3)}) = \min\left\{L_d(H_{(1)}) + t_2, \min\{L_d(H) : H \in \mathcal{H}_3\}\right\}.$$

Otherwise, $L_d(H_{(3)}) = \min\{L_d(H) : H \in \mathcal{H}_3\}$.

Proof. We first consider the case that $|\mathcal{T}| \geq 2$. By the assumption that $\mathcal{H}(2)$ is unique, it follows from Theorem 5.1 that $u(e) > t_1$ for all $e \in \cap\mathcal{H}(2)$ and $l(e) > t_1$ for each $E\setminus \cup\mathcal{H}(2)$. Hence, we have that

$$t_2 = \min\{\min\{u(e) : e \in \cap\mathcal{H}(2)\}, \min\{l(e) : e \in E\setminus \cup\mathcal{H}(2)\}\}.$$

It follows from Theorem 5.1 that

$$\begin{aligned} &\min\{u(e) : e \in \cap\mathcal{H}(2)\} \\ &\quad= \min\{\min\{L_d(H) : H \in \mathcal{H}, e \notin H\} : e \in \cap\mathcal{H}(2)\} - L_d(H_{(1)}) \\ &\quad= \min\{L_d(H) : H \in \mathcal{H}\setminus\mathcal{H}(2), \cap\mathcal{H}(2) \nsubseteq H\} - L_d(H_{(1)}). \end{aligned} \tag{5.1}$$

It can be shown in a similar way that

$$\begin{aligned} &\min\{l(e) : e \in E\setminus \cup\mathcal{H}(2)\} \\ &\quad= \min\{L_d(H) : H \in \mathcal{H}\setminus\mathcal{H}(2), H \nsubseteq \cup\mathcal{H}(2)\} - L_d(H_{(1)}). \end{aligned} \tag{5.2}$$

Combining (5.1) and (5.2) gives that

$$t_2 = \min\{L_d(H) : H \in \mathcal{H}\setminus\mathcal{H}(2), \cap\mathcal{H}(2) \nsubseteq H \text{ or } H \nsubseteq \cup\mathcal{H}(2)\} - L_d(H_{(1)}).$$

So, it holds that $\min\left\{L_d(H_{(1)}) + t_2, \min\{L_d(H) : H \in \mathcal{H}_3\}\right\} = L_d(H_{(3)})$. If $|\mathcal{T}| < 2$ then $u(e) > \infty$ for all $e \in \cap\mathcal{H}(2)$ and $l(e) > \infty$ for each $E\setminus \cup\mathcal{H}(2)$. Consequently, $L_d(H_{(3)}) = \min\{L_d(H) : H \in \mathcal{H} \setminus \mathcal{H}(2), \cap\mathcal{H}(2) \subseteq H \subseteq \cup\mathcal{H}(2)\}\} = \min\{L_d(H) : H \in \mathcal{H}_3\}$. □

Example (continued). Consider (G, d) in Figure 5.1, we have that $\cap\mathcal{H}(2) = \{\{2,3\},\{5,6\}\}$, $\cup\mathcal{H}(2) = E$, and $\mathcal{H}_3 = \{H_{(3)}\}$. Furthermore, $t_2 = L_d(H_{(4)}) - L_d(H_{(1)})$. Hence, $L_d(H_{(3)}) = \min\{L_d(H) : H \in \mathcal{H}_3\}$. □

It follows from Theorem 5.14 that in order to find the length of the third-best tour, we need to solve the problem $\min\{L_d(H) : H \in \mathcal{H}_3\}$. Before discussing the computational complexity of this problem, we will first develop some intuition with respect to the set $\mathcal{H}_3$, and therefore of the sets $\cap\mathcal{H}(2)$ and $\cup\mathcal{H}(2)$.

We first show that $\cap\mathcal{H}(2)$ is nonempty.

Theorem 5.15 *If $\mathcal{H}(2) = \{H_{(1)}, H_{(2)}\}$ is uniquely determined in (G, d) then $H_{(1)}$ and $H_{(2)}$ are not edge-disjoint (i.e. $\cap\mathcal{H}(2) \neq \emptyset$).*

Proof. Suppose, to the contrary, that $\cap\mathcal{H}(2) = \emptyset$. Let $x, y \in E$ be arbitrary edges such that $x \in H_{(1)}$ and $y \in H_{(2)}$. In Thomason [95] it is proven that the graph formed by the edge-disjoint tours $H_{(1)}$ and $H_{(2)}$ contains an even number of pairs of edge disjoint tours with x and y in the same tour. Let H, H' be one such a pair, and assume that $\{x, y\} \subseteq H$. Hence, the tours H and H' are pairwise different with $H_{(1)}$ and $H_{(2)}$, because neither $H_{(1)}$ or $H_{(2)}$ contains both the edges x and y. Since $\mathcal{H}(2)$ is uniquely determined in (G, d), it holds that $L_d(H_{(1)}) + L_d(H_{(2)}) < L_d(H) + L_d(H')$. Furthermore, it follows from $H \cup H' = H_{(1)} \cup H_{(2)}$ that $L_d(H_{(1)}) + L_d(H_{(2)}) = L_d(H) + L_d(H')$. Hence, we have a contradiction. □

In the following theorem a sufficient condition is given for $\mathcal{H}_3$ being the empty set. In contrast to Theorem 5.15, the fact that $\{H_{(1)}, H_{(2)}\}$ is set of 2-best tours will not be used in this result.

Theorem 5.16 *If $|\cap\mathcal{H}(2)| \geq n - 3$ then $\mathcal{H}_3 = \emptyset$.*

Proof. The proof is based on Sierksma [83]. Note that we only have to consider the cases that $|\cap\mathcal{H}(2)| = n-3, n-2$. First consider the case that $|H_{(1)} \cap H_{(2)}| = n-2$, i.e. $H_{(2)}$ is obtained from $H_{(1)}$ as the result of a 2-edge interchange. In a 2-edge interchange two edges, say $\{a, b\}$ and $\{c, d\}$, are replaced by two new edges, $\{a, c\}$ and $\{b, d\}$, such that $H_{(2)}$ is again a tour in G, i.e. $H_{(2)} = (H_{(1)} \setminus \{\{a, b\}, \{c, d\}\}) \cup \{\{a, c\}, \{b, d\}\}$. We will show that $\mathcal{H}_3$ is empty. Suppose, to the contrary, that $\mathcal{H}_3 \neq \emptyset$, say $H \in \mathcal{H}_3$. If $\{a, b\} \in H$ then $\{a, c\}, \{b, d\} \notin H$. Hence, $\{c, d\} \in H$, and consequently $H = H_{(1)}$, which is a contradiction. If $\{a, c\} \in H$ then $\{a, b\}, \{c, d\} \notin H$. Hence, $\{b, d\} \in H$, and consequently $H = H_{(2)}$, which again gives a contradiction. This proves that $\mathcal{H}_3 = \emptyset$ for the case $|H_{(1)} \cap H_{(2)}| = n - 2$.

Now consider the case that $|H_{(1)} \cap H_{(2)}| = n - 3$, i.e. $H_{(2)}$ is obtained from $H_{(1)}$ by applying a 3-edge interchange. In a 3-edge interchange three edges, say $\{a, b\}$, $\{c, d\}$ and $\{e, f\}$, are replaced by three new edges such that $H_{(2)}$ is again a tour in G. If no two edges of $\{a, b\}$, $\{c, d\}$ and $\{e, f\}$ are adjacent, then the edge set $H_{(1)} \setminus \{\{a, b\}, \{c, d\}, \{e, f\}\}$ can be extended into a tour in four ways, schematically depicted in Figure 5.4, labeled from $(1), \ldots, (4)$. If two of the edges $\{a, b\}$, $\{c, d\}$ and $\{e, f\}$ are adjacent, then the edge set $H_{(1)} \setminus \{\{a, b\}, \{c, d\}, \{e, f\}\}$ can be extended into a tour in three ways, shown in Figure 5.4, labeled from $(5), \ldots, (7)$. For each form of $H_{(2)}$, one can easy check that $\mathcal{H}_3 = \emptyset$. Consider for instance (1), i.e. $H_{(2)} = H_{(1)} \setminus \{\{a, b\}, \{c, d\}, \{e, f\}\} \cup \{\{a, c\}, \{b, e\}, \{d, f\}\}$. Suppose, to the contrary, that $\mathcal{H}_3 \neq \emptyset$, say $H \in \mathcal{H}_3$. If $\{a, b\} \in H$ then $\{a, c\}, \{b, e\} \notin H$. Hence, $\{c, d\}, \{e, f\} \in H$, so that $\{d, f\} \notin H$. Consequently, $H = H_{(1)}$, which yields a contradiction. If $\{a, b\} \notin H$ then $\{a, c\}, \{b, e\} \in H$. Hence, $\{c, d\}, \{e, f\} \notin H$, so that $\{d, f\} \in H$. So, $H = H_{(2)}$, which yields a contradiction and proofs that $\mathcal{H}_3 = \emptyset$. □

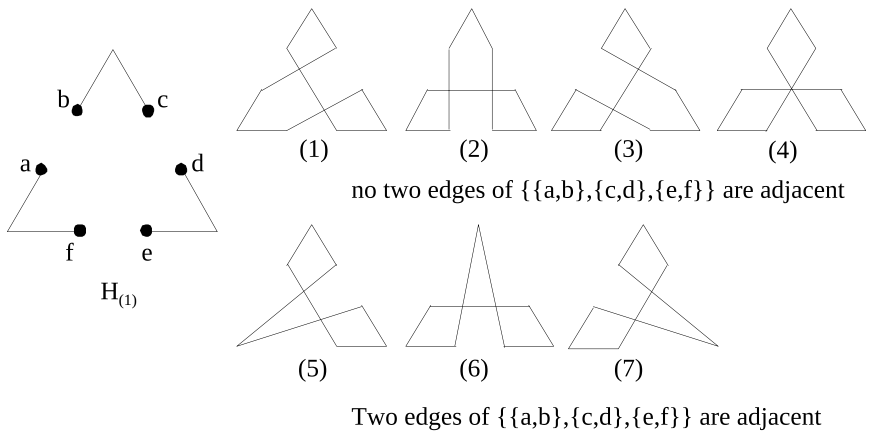

Figure 5.4: All possible 3-edge interchanges.

Note that according to the Theorems 5.14 and 5.16, it follows that if $H_{(1)}$ and $H_{(2)}$ differ in either a 2 or 3–edge interchange, then $H_{(3)} = H_2$. However, if $H_{(1)}$ and $H_{(2)}$ differ by a k-edge interchange with $4 \leq k \leq n - 2$ then $\mathcal{H}_3$ need not be the empty set, and consequently the problem of finding a third-best tour might be more complex. Figure 5.1 shows an example with a nonempty set $\mathcal{H}_3$ for the case that $H_{(1)}$ and $H_{(2)}$ differ in a 4-edge interchange.

From Theorem 5.14 it is clear that the complexity of finding a third-best tour is determined by the complexity of finding a shortest tour in $\mathcal{H}_3$. It will be shown that this problem is in fact $\mathcal{NP}$-hard. In order to do so, it will first be shown that the following problem is $\mathcal{NP}$-complete.

Restricted 3^{rd} tour problem

Instance: A pair of tours T_1 and T_2 in a graph on $n \geq 3$ vertices.
Question: Is there a tour H such that $H \notin \{T_1, T_2\}$ and $T_1 \cap T_2 \subseteq H \subseteq T_1 \cup T_2$?

We establish a reduction from the

3^{rd} tour problem

Instance: A pair of tours T_1 and T_2 in a graph on $n \geq 3$ vertices.
Question: Is there a tour H such that $H \notin \{T_1, T_2\}$ and $H \subseteq T_1 \cup T_2$?

which was proven to be $\mathcal{NP}$-complete in Papadimitriou [66]. In that paper the problem was formulated as: Does a given graph that is restricted to the union of two tours contain a third tour?

Note that the 3^{rd} *tour problem* and the *restricted* 3^{rd} *tour problem* are really different. For instance, the graph of Figure 5.5, which is formed by the tours T_1 and T_2, contains one more tour denoted by T_3. It holds that $T_3 \subseteq T_1 \cup T_2$, but $T_1 \cap T_2 \nsubseteq T_3$.

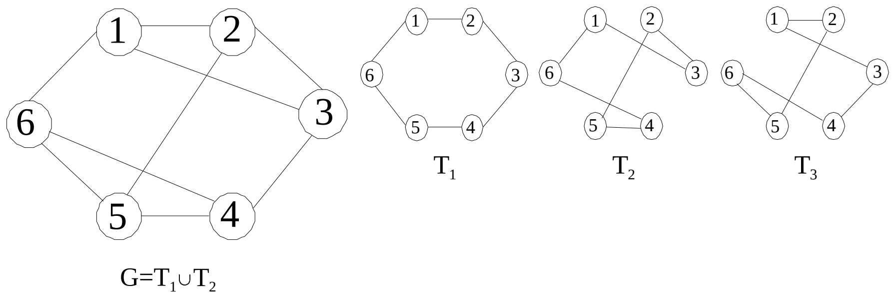

Figure 5.5: A graph that is both a "yes-instance" of the 3^{rd} *tour problem* and a "no-instance" of the *restricted* 3^{rd} *tour problem.*

In the following theorem it is shown that the *restricted* 3^{rd} *tour problem* is as hard as the 3^{rd} *tour problem.*

Theorem 5.17 *The restricted* 3^{rd} *tour problem is* $\mathcal{NP}$*-complete.*

Proof. We give a reduction from the 3^{rd} *tour problem* to the *restricted* 3^{rd} *tour problem.* Take an instance of the 3^{rd} *tour problem*, i.e. tours T_1 and T_2 such that $T_1 \neq T_2$ and $T_1 \cap T_2 \neq \emptyset$. The set of edges that are in both T_1 and T_2 can be partitioned in a number of paths. Clearly, every path in $T_1 \cap T_2$ of length larger than 1 is contained in all tours of $G = T_1 \cup T_2$, because such a path contains vertices that only can be visited by traversing it. Now consider the paths in $T_1 \cap T_2$ of length 1, i.e. edges $\{i, j\}$ that are both contained in T_1 and T_2 which are *not* adjacent to other edges in $T_1 \cap T_2$. Let x, y, z, w be vertices such that the edges $\{w, i\}, \{j, z\}, \{x, i\}, \{j, y\}, \{i, j\}$ are pairwise different, with $\{w, i\}, \{i, j\}, \{j, z\} \in T_1$, and $\{x, i\}, \{i, j\}, \{j, y\} \in T_2$. Construct a graph G' from G as follows. Remove the edge $\{j, y\}$ and add a new vertex ij together with the edges $\{i, ij\}, \{ij, y\}, \{ij, j\}$; see Figure 5.6 where G and G' are shown as multi-graphs.

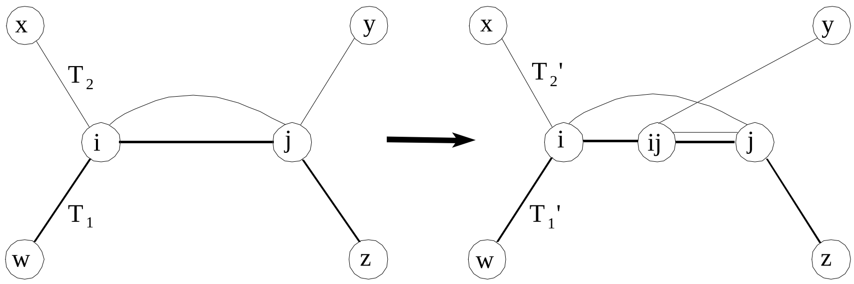

Figure 5.6: Reducing the 3^{rd} *tour problem* to the *restricted* 3^{rd} *tour problem.*

Note that the graph G' is formed by the tours $T_1' = (\ldots, w, i, ij, j, z, \ldots)$ and $T_2' = (\ldots, x, i, j, ij, y, \ldots)$. Also note that, apart from the paths of length larger than 1, $\{ij, j\}$ is the only edge that is contained in both tours. It is now easy to see that there is a tour T in $\mathcal{H}$ if and only if there is a tour T' in G'. The corresponding tours are as follows:

$$T = (\ldots, w, i, j, z, \ldots) \;\leftrightarrow\; T' = (\ldots, w, i, ij, j, z, \ldots)$$
$$T = (\ldots, x, i, j, y, \ldots) \;\leftrightarrow\; T' = (\ldots, x, i, j, ij, y, \ldots)$$

$$
\begin{aligned}
T = (\ldots, w, i, j, y, \ldots) \quad &\leftrightarrow \quad T' = (\ldots, w, i, j, ij, y, \ldots) \\
T = (\ldots, x, i, j, z, \ldots) \quad &\leftrightarrow \quad T' = (\ldots, x, i, ij, j, z, \ldots) \\
T = (\ldots, x, i, w, \ldots, y, j, z, \ldots) \quad &\leftrightarrow \quad T' = (\ldots, x, i, w, \ldots, y, ij, j, z, \ldots).
\end{aligned}
$$

The left side shows the ways a tour T in $\mathcal{H}$ can traverse the vertices x, y, i, j, w, z, and the right side shows the ways a tour T' in G', containing the edge $\{ij, j\}$, can traverse the vertices x, y, i, j, ij, w, z. For each form of T, the $\leftrightarrow$ symbol indicates the corresponding from of T', and vice versa. Note that the edges $\{i, ij\}, \{ij, y\}, \{ij, j\}$ in G' can be traversed in one more way, namely by $(\ldots, y, ij, i, j, z, \ldots)$. However, this is not a "yes-instance" of the *restricted* 3^{rd} *tour problem*, because this tour does not contain the edge $\{ij, j\}$.

So, in conclusion, the given instance of the 3^{rd} *tour problem* is a "yes-instance" if and only if the corresponding instance of the *restricted* 3^{rd} *tour problem* is a "yes-instance". Since the3^{rd} *tour problem* is $\mathcal{NP}$-complete, it follows that the *restricted* 3^{rd} *tour problem* is $\mathcal{NP}$-complete as well. □

We now turn our attention to the complexity of the 3-best TSP with given optimal tour and tolerances. It will be shown that the decision version of the k-best TSP[$H_{(1)}$,tolerances], namely

k-best TSP decision[$H_{(1)}$,tolerances] ($k > 1$)

Instance:	A weighted graph (G, d) with a given optimal tour $H_{(1)}$ and its tolerances. A number $B \in \mathbb{R}$.
Question:	Is there a set $\mathcal{H}(k)$ in (G, d) such that $L_k \leq B$?

is $\mathcal{NP}$-complete for $k = 3$.

Theorem 5.18 *The k-best TSP decision[$H_{(1)}$,tolerances] is $\mathcal{NP}$-complete for $k = 3$.*

Proof. We give a reduction from the *restricted* 3^{rd} *tour problem* to the 3-*best TSP decision*[$H_{(1)}$,tolerances]. Take an instance of the *restricted* 3^{rd} *tour problem*, i.e. tours T_1 and T_2 such that $T_1 \neq T_2$. Assume, without loss of generality, that $T_1 \cap T_2 \neq \emptyset$. Recall that the set of edges that are both in T_1 and T_2 can be partitioned in a number of paths. Let p denote the number of paths in $T_1 \cap T_2$, i.e. $p \geq 1$. For $i = 1, \ldots, p$, let s_i and f_i denote the endpoints of path i. The transformation is as follows.

Step 1. Let $G' = (V', E')$ with $V' = V$ and $E' := T_1 \cup T_2$.

Step 2. Remove the vertices and edges of path 1, except for the vertices s_1 and f_1. Add new vertices s_1' and f_1' together with the edges $\{s_1, s_1'\}, \{s_1', f_1'\}, \{f_1', f_1\}, \{s_1, f_1'\}, \{s_1', f_1\}$. See Figure 5.7.
Step 3. For $i = 2, \ldots, p$, remove the vertices and edges of path i, except for the vertices s_i and f_i. Add the new vertex v_i and the new edges $\{s_i, v_i\}, \{v_i, f_i\}$.
Step 4. Let $d'(s_1, f_1') := 1$, and $d'(e) := 0$ for all $e \in E' \backslash \{\{s_1, f_1'\}\}$.

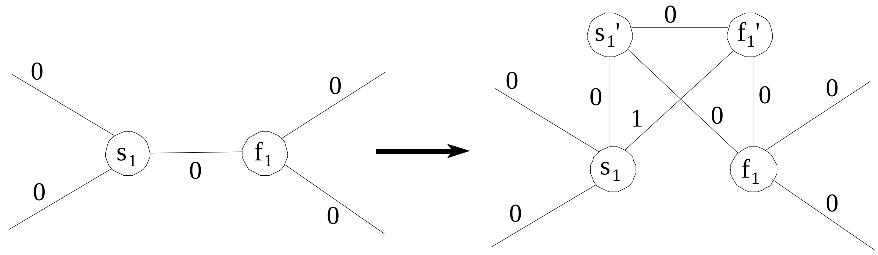

Figure 5.7: Reducing the *restricted* 3^{rd} *tour problem* to the 3-*best TSP decision*$[H_{(1)}$,tolerances].

Note that for each tour in $G = T_1 \cup T_2$ there are precisely two corresponding tours in G', one with length 0 containing the edges $\{s_1, s_1'\}$, $\{s_1', f_1'\}, \{f_1', f_1\}$, and an other with length 1 containing the edges $\{s_1, f_1'\}$, $\{f_1', s_1'\}$, $\{f_1', s_1\}$. Let T_1' and T_2' denote the tours in (G', d') with length 0 that correspond to the tours T_1 and T_2, respectively. Note that T_1' and T_2' contain the edges $\{s_1, s_1'\}, \{s_1', f_1'\}, \{f_1', f_1\}$ and are both optimal tours in (G', d'). Furthermore, it is easy to see (by applying Theorem 5.1) that the finite tolerances with respect to T_1' and d' are given by $l(e) = 0$ for each $e \in T_2' \backslash T_1'$, $u(e) = 0$ for each $e \in T_1' \backslash T_2'$, and $u(\{s_1, s_1'\}) = u(\{f_1, f_1'\}) = l(\{s_1, f_1'\}) = l(\{s_1', f_1'\}) = 1$. Hence, $t_1 = 0$ and $t_2 = 1$. Note that $T_2' \backslash T_1' = \{e \in E \backslash T_1' : l(e) = t_1\}$ and $T_2' \cap T_1' = \{e \in T_1' : u(e) > t_1\}$.Let $B := 0$. We will show that there is a third tour H in $\mathcal{H}$ such that $T_1 \cap T_2 \subseteq H \subseteq T_1 \cup T_2$ if and only if there is a set $\mathcal{H}(3)$ in (G', d') such that $L_3 \leq 0$. (*–only if–*) Let H be a third tour in $G = T_1 \cup T_2$ such that $T_1 \cap T_2 \subseteq H \subseteq T_1 \cup T_2$. Let H' denote the corresponding tour in (G', d') containing the edges $\{s_1, s_1'\}, \{s_1', f_1'\}, \{f_1', f_1\}$. Hence, $L_{d'}(H') = 0$. Consequently, $\{T_1', T_2', H'\}$ is a set of 3 best tours with $L_3 = 0$. (*–if–*) Assume that there is a set $\mathcal{H}(3)$ in (G', d') such that $L_3 \leq 0$. Then, there is tour H in $G = T_1 \cup T_2$ such that the corresponding tour H' in G' with length 0 is not equal to T_1' or T_2'. Since

$L_{d'}(H') = 0$, it follows from Theorem 5.14 that $T_1' \cap T_2' \subseteq H' \subseteq T_1' \cup T_2'$. Hence, we have that also $T_1 \cap T_2 \subseteq H \subseteq T_1 \cup T_2$. So, there is a third tour H in $G = T_1 \cup T_2$ such that $T_1 \cap T_2 \subseteq H \subseteq T_1 \cup T_2$.

So, in conclusion, the given instance of the *restricted* 3^{rd} *tour problem* is a "yes-instance" if and only if the corresponding instance of the 3-*best TSP decision*$[H_{(1)}$,tolerances] is a "yes-instance". Since the *restricted* 3^{rd} *tour problem* is $\mathcal{NP}$-complete, it follows that the 3-*best TSP decision*$[H_{(1)}$,tolerances] is $\mathcal{NP}$-complete as well. □

Note that if the k-*best TSP decision*$[H_{(1)}$,tolerances] is polynomially solvable for $k \geq 3$, then the 3-*best TSP decision*$[H_{(1)}$,tolerances] is polynomially solvable as well. Hence, we have the following corollary of Theorem 5.18.

Corollary 5.19 *The* 3*-best TSP decision*$[H_{(1)}$*,tolerances] is* $\mathcal{NP}$*-complete.*

Following the same reasoning as for the TSP and its decision version (see e.g. Johnson & Papadimitriou [40]), there is a polynomial-time algorithm for the k-*best TSP*$[H_{(1)}$,tolerances] if and only if there is a polynomial time algorithm for the k-*best TSP decision*$[H_{(1)}$,tolerances]. Hence, as a corollary of Theorem 5.18, we obtain the main result in this section.

Corollary 5.20 *The k-best TSP is* $\mathcal{NP}$*-hard for* $k \geq 3$*, even if an optimal tour and its tolerances are given.*

5.6 Tolerances with respect to a transformed length vector

In this section we present a transformation of the length vector such that all tours get a different length and a tour can be easily constructed from its length. It will be shown that if the transformation is used, then stronger results can be derived than the ones given in the previous sections. In particular, it will be shown that if an optimal tour and its tolerances are given with respect to the transformed length vector, then all tours H_j can be determined and the 2-best TSP is polynomially solvable.

We assume throughout this section that all edge lengths $d(e)$ are integer-valued and that the edges in E are labeled from e_1 up to e_m. Define the vector $d' \in \mathbb{Q}^m$, for $i = 1, \ldots, m$, by

$$d'(e_i) := d(e_i) + 10^{-i} \tag{5.3}$$

Note that each subset of edges S in E can be determined uniquely from the decimal part of the length $L_{d'}(S)$.

In the following theorem it is shown that d' preserves the partial ordering in (G, d), that all tours in (G, d') have different lengths, and that each tour can be determined from its length.

Theorem 5.21 *For all tours H and H' in (G, d') with $H \neq H'$, the following assertions hold.*

1. $L_{d'}(H) \neq L_{d'}(H')$.
2. *If $L_d(H) < L_d(H')$ then $L_{d'}(H) < L_{d'}(H')$.*
3. *If L is the length of a tour in (G, d'), then a tour H with $L_{d'}(H) = L$ can be determined in $\mathcal{O}(m)$ time.*

Proof. In the proof we will use the following claim.

Claim: For all subsets S, S' in (G, d'), it holds that

1. $L_{d'-d}(S) = L_{d'-d}(S')$ if and only if $S = S'$,

2. $-1 < L_{d'-d}(S) - L_{d'-d}(S') < 1$.

The proof of the claim is straightforward and therefore omitted here.

(1) Note that $L_{d'}(H) = L_{d'}(H')$ if and only if both the integer and the decimal parts of $L_{d'}(H)$ and $L_{d'}(H')$ are equal. We have that $L_{d'}(H) = L_d(H) + L_{d'-d}(H)$ and $L_{d'}(H') = L_d(H') + L_{d'-d}(H')$. It follows, from the assumption that all edge lengths $d(e)$ are integer-valued, that $L_d(H), L_d(H') \in \mathbb{Z}$. Hence, $L_{d'-d}(H)$ and $L_{d'-d}(H')$ are the decimal parts of $L_{d'}(H)$ and $L_{d'}(H')$, respectively. Now recall from Claim 1. that $L_{d'-d}(H) = L_{d'-d}(H')$ if and only if $H = H'$. So, $H \neq H'$ implies that $L_{d'}(H) \neq L_{d'}(H')$.

(2) We have that $L_{d'}(H) - L_{d'}(H') = (L_d(H) + L_{d'-d}(H)) - (L_d(H') + L_{d'-d}(H')) = L_d(H) - L_d(H') - (L_{d'-d}(H') - L_{d'-d}(H))$. Since, by assumption, $L_d(H)$ and $L_d(H')$ are integer-valued, it follows from $L_d(H) < L_d(H')$ that $L_d(H) - L_d(H') \leq -1$. Furthermore, recall from Claim 2. that $(L_{d'-d}(H') - L_{d'-d}(H)) > -1$. Hence, $(L_d(H) - L_d(H')) -$

$(L_{d'-d}(H') - L_{d'-d}(H)) \leq -1 - (L_{d'-d}(H') - L_{d'-d}(H)) < 0$.

(3) Given $L_{d'}(H)$ for some $H \in \mathcal{H}$, the edges of H can be determined by considering the decimal part of the tour length. For $i = 1, \ldots, m$, edge e_i is contained in H if and only if the i-th decimal in $L_{d'}(H)$ is equal to one. Hence, in the worst case all decimals in $L_{d'}(H)$ have to be scanned in order to determine H from its length, which takes $\mathcal{O}(m)$ time. □

Example (continued). Consider again (G, d) of Figure 5.1. The labeling of the edges is shown in Figure 5.8. For instance, $L_{d'}(H_{(1)}) = 11.0010101111$. □

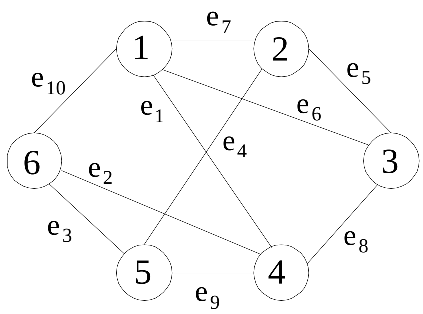

Figure 5.8: Labeling of the edges.

Now, we will investigate how the tolerances with respect to the transformed length vector can be used for solving the k-best TSP. Here we assume that the optimal tour and its tolerances are determined with respect to the length vector d'. We like to stress that knowing an optimal tour and its tolerances with respect to (G, d) does not imply that we are able to determine the tolerances with respect to (G, d'). Furthermore, note that the number of finite tolerances does not change. In other words, it is necessary to first apply the transformation and then determine the optimal tour and its tolerances.

As a corollary of the Theorems 5.5 and 5.21, we have that the tours H_j in (G, d') can be determined in polynomial time.

Corollary 5.22 *For $j = 1, \ldots, |\mathcal{T}|$, each tour H_j in (G, d') is unique and can be determined in $\mathcal{O}(m)$ time.*

Proof. Since all tours in (G, d') have different length (cf. Theorem 5.21), it follows from Theorem 5.5 that, for each $j = 1, \ldots, |\mathcal{T}|$, there is an unique tour $H_j \in \mathcal{H}$ such that $L_d(H_j) = L_d(H_{(1)}) + t_j$. Applying Theorem 5.21 gives the corresponding set of edges. □

Furthermore, as a corollary of the Theorems 5.8 and 5.21, we have that the 2-best TSP in (G, d') with given optimal tour and tolerances is polynomially solvable.

Corollary 5.23 *For (G, d') with $|\mathcal{T}| \geq 1$, the 2-best TSP with given optimal tour and tolerances is solvable in $\mathcal{O}(m)$ time.*

Proof. Recall from Theorem 5.8 that $L_2 = t_1$. Applying Theorem 5.21 to $L_d(H_{(1)}) + t_1$ gives the corresponding set of edges. □

In contrast to this "good" news, the next Corollary of Theorem 5.18 shows that the k-best TSP for (G, d') with given optimal tour and tolerances remains $\mathcal{NP}$-hard for $k \geq 3$.

Corollary 5.24 *For (G, d'), the k-best TSP[$H_{(1)}$,tolerances] is $\mathcal{NP}$-hard for $k \geq 3$.*

Proof. Again, we first show that the 3-best TSP decision[$H_{(1)}$, tolerances] for (G, d') with given optimal tour and tolerances is $\mathcal{NP}$-complete. We therefore again use the reduction given in the proof of Theorem 5.18. We only have to modify the tolerance values used in the reduction. This is straightforward and therefore omitted here. Consequently, the k-best TSP decision[$H_{(1)}$, tolerances] is $\mathcal{NP}$-complete for $k \geq 3$, so that the k-best TSP is $\mathcal{NP}$-hard for $k \geq 3$. □

Example (continued). Consider (G, d) in Figure 5.1 with the labeling of the edges given in Figure 5.8. The tolerances with respect to $H_{(1)}$ and d' are given in Figure 5.9. We have that $L_d(H_{(1)}) = 11.0010101111$, $t_1 = 1.1101008889$, $t_2 = 3.00910089$, and $t_3 = 4.0100909989$. Hence, $L_d(H_1) = L_d(H_{(1)}) + t_1 = 12.1111110000$, $L_d(H_2) = L_d(H_{(1)}) + t_2 = 14.0101110011$, and $L_d(H_3) = L_d(H_{(1)}) + t_3 = 15.0111011100$. Applying Theorem 5.21 gives that $H_1 = \{e_1, e_2, e_3, e_4, e_5, e_6\} = H_{(2)}$, $H_2 = \{e_2, e_4, e_5, e_6, e_9, e_{10}\} = H_{(4)}$, and $H_3 = \{e_2, e_3, e_4, e_6, e_7, e_8\} = H_{(5)}$. □

5.7 Conclusion

In this chapter the k-best TSP ($k \geq 2$) is considered for the case that an optimal tour and all its tolerances are given. Firstly, the extra information is used to determine a partial ordering of the tours in the weighted graph (G,d). As a second point of interest, the complexity of the k-best TSP is studied for the case that an optimal tour and all its tolerances are given. It is shown that the length of a second-best tour can be determined in polynomial time and that the 2-best TSP is polynomially solvable in the case that the set of 2-best tours in (G,d) is unique. It is conjectured that the 2-best TSP is $\mathcal{NP}$-hard in the other case. Furthermore, it is shown the k-best TSP with given optimal tour and tolerances is $\mathcal{NP}$-hard for $k \geq 3$. Finally, a transformation of the length vector is presented such that, given an optimal tour and the tolerances with respect to the transformed length vector, all tours in the partial ordering can be determined exactly and the 2-best TSP is polynomially solvable. In conclusion, the knowledge of an optimal tour and its tolerances does, in general, not provide sufficient information for solving the k-best TSP.

An obvious point for further research is to study the computational complexity of the 2-best TSP for the case that an optimal tour and all its tolerances are given and the set of 2-best tours in (G,d) is not unique. Furthermore, it is of interest to apply the idea of using stability information for solving the k-best problem to other combinatorial optimization problems.

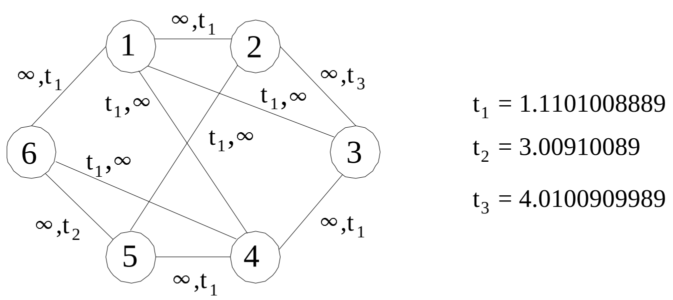

Figure 5.9: Lower (first number) and upper (second number) tolerances.

Appendix 5.A: The proof of Theorem 5.13

In this appendix we give the proof of Theorem 5.13. We first discuss two "elementary" results. In the following lemma it will be shown for each tour H in $\mathcal{H}$ and subset S in $E\backslash H$ that there are at most two tours on the edge set $H \cup S$ each containing the edge set S.

Lemma 5.25 *For each $H \in \mathcal{H}$ and $S \subseteq E\backslash H$, $|\{T \in \mathcal{H}: T\backslash H = S\}| \leq 2$.*

Proof. Take any $H \in \mathcal{H}$ and $S \subseteq E\backslash H$. We first show that it is sufficient to consider only graphs with each vertex being incident to one edge in S. Clearly, there is no tour T in $\mathcal{H}$ such that $T\backslash H = S$ when there is some vertex being incident to more than two edges in S. Consider therefore graphs with vertices being incident to 0, 1, or 2 edges in S. Assume that the vertices of G are ordered according to H. First, consider a vertex u in G being incident to two edges in S, say $\{u, v\}$ and $\{u, w\}$. Hence, each tour $T \in \mathcal{H}$ such that $T\backslash H = S$ does not contain the edges $\{u-1, u\}, \{u, u+1\}$, so that by removing the edges $\{u, w\}, \{u, u+1\}$ and by introducing a new vertex u' and new edges $\{u, u'\}, \{u', w\}, \{u', u+1\}$ the number of tours T' in $\mathcal{H}'$ such that $T'\backslash H' = S'$ will not decrease, where T', H', $\mathcal{H}'$, and S' denote the corresponding sets in the new graph. Now, consider vertices u, v in G such that $u + 1 < v$, u and v being incident to at least one edge in S, and all vertices w, with $u < w < v$, not being incident to S. Again, by removing the vertices $u+1, \ldots, v-1$ and edges $\{u, u+1\}, \ldots, \{v-1, v\}$, and adding the new edge $\{u, v\}$, the number of tours T' in $\mathcal{H}'$ such that $T'\backslash H' = S'$ will not decrease, where T', H', $\mathcal{H}'$, and S' denote the corresponding sets in the new graph. This proves that we only need to consider graphs with each vertex being incident to one edge in S.

In the case that each vertex in G is incident to one edge in S, we have that G is cubic and that $n = 2|S|$. Let $H := \{e_1, \ldots, e_n\}$ with $e_i = \{i, (i \bmod n) + 1\}$ for $i = 1, \ldots, n$. Define $H_{odd} := \{e_i : i = 1, \ldots, n; i \text{ is odd}\} \cup S$ and $H_{even} := \{e_i : i = 1, \ldots, n; i \text{ is even}\} \cup S$. Note that H_{odd} and H_{even} are perfect 2-matchings in G. (Recall that any set of edges M in G is called a *perfect 2-matching* when each vertex in G is incident to precisely two edges of M.) We will prove that H_{odd} and H_{even} are the only perfect 2-matchings T in G such that $T\backslash H = S$. Suppose, to the contrary, that there is another perfect 2-matching T in G such that $T\backslash H = S$. Since $T \neq H_{odd}$ and $T \neq H_{even}$, it follows that there is some $i \in \{1, \ldots, n\}$ such that $e_{((i+n-2) \bmod n)+1}, e_i \in T$. But, then the assumption that each vertex of G is incident to precisely one edge in S implies that vertex i is incident to three edges in T. Hence, T is no perfect 2-matching, and consequently we have a contradiction. Hence, H_{odd} and H_{even} are the only perfect 2-matchings T in G such that $T \backslash H = S$. Since both H_{odd} and H_{even} can be tours, it follows that $|\{T \in \mathcal{H} : T \backslash H = S\}| \leq 2$. $\square$

Figure 5.10 shows a cubic graph $G = (V, E)$, with a tour H indicated by the bold lines and $S = E \backslash H$, for which there are two tours in G containing S. Similarly, Figure 5.11 shows a cubic graph G for which there is only one tour containing S.

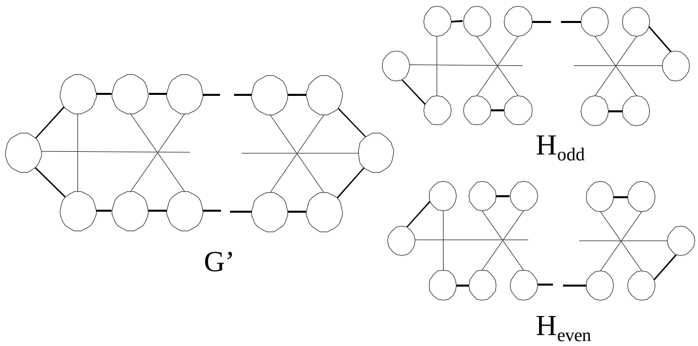

Figure 5.10: Graph G with two tours containing S.

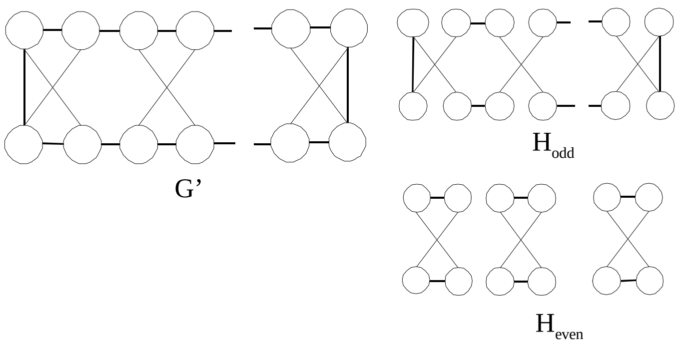

Figure 5.11: Graph G with only one tour containing S.

For each $H \in \mathcal{H}$ and $S \subseteq E \backslash H$, the set of all tours T in $\mathcal{H}$ such that $T \backslash H = S$ can be determined by applying Algorithm 5.1 below.

Algorithm 5.1: Determining tours T in G such that $T\backslash H = S$.

Input: The graph $G = (V, E)$ where $\mathcal{H}$ denotes the set of tours in G. A tour H in G and a set $S \subseteq E\backslash H$.
Output: The set $\{T \in \mathcal{H} : T\backslash H = S\}$.

```
‘ Label the vertices in such a way that vertex 1 is incident to S ’;
u := 1; i := 0;
while u ≤ |V| do
begin
   if ‘ vertex u is incident to two edges in S ’
   then begin i := i + 1; E_i := ∅ end;
   ‘ Let v be the next vertex that is incident to S ’;
   u := v; i := i + 1;
   E_i := {{u, u + 1}, ..., {v − 1, v}};
end;
H_odd := ∪{E_j : j = 1, ..., i; j is odd} ∪ S;
H_even := ∪{E_j : j = 1, ..., i; j is even} ∪ S;
{T ∈ H : T\H = S} := H ∩ {H_odd, H_even}.
```

The correctness of Algorithm 5.1 will be established in the following lemma.

Lemma 5.26 *For each $H \in \mathcal{H}$ and $S \subseteq E\backslash H$, Algorithm 5.1 determines the set $\{T \in \mathcal{H} : T\backslash H = S\}$ in $\mathcal{O}(m)$ time.*

Proof. Take any $H \in \mathcal{H}$ and $S \subseteq E\backslash H$. Assume that the vertices are labeled as in Algorithm 5.1, and let E_j, for $j = 1, \ldots, i$, be the sets determined by Algorithm 5.1. Determine the corresponding cubic graph $G' = (V', E')$ by applying the constructions described in the proof of Lemma 5.25. Let S' denote the set of edges in G' that corresponds to S in G. The tour H in G corresponds to the tour $H' = \{e_1, \ldots, e_j\}$ in G', with $e_j = \{j, (j \bmod i) + 1\}$ for $j = 1, \ldots, i$. Note that edge e_j in G' corresponds to the edge set E_j in G. Furthermore, each tour in G corresponds to a tour in G', but not the other way around.

Now, recall from Lemma 5.25 that there are at most two tours T' in G' such that $T'\backslash H' = S'$, defined by $H'_{odd} := \cup\{e_j : j = 1, \ldots, i;\ j$ is odd$\} \cup S'$ and $H'_{even} := \cup\{e_j : j = 1, \ldots, i; j$ is even$\} \cup S'$. The corresponding perfect 2-matchings in G are defined by $H_{odd} := \cup\{E_j : j = 1, \ldots, i; j \text{ is odd}\} \cup S$ and $H_{even} := \cup\{E_j : j = 1, \ldots, i; j \text{ is even}\} \cup S$. Since each tour T in $\mathcal{H}$ such that $T\backslash H = S$ corresponds to either H'_{odd} or

H'_{even}, only the corresponding perfect 2-matchings H_{odd} and H_{even} can be tours T in $\mathcal{H}$ such that $T\backslash H = S$. Hence, the set $\{T \in \mathcal{H} : T\backslash H = S\}$ can be determined by taking the intersection of $\{H_{odd}, H_{even}\}$ and $\mathcal{H}$.

In the worst case, all vertices and edges in G need to be considered by Algorithm 5.1 taking $\mathcal{O}(m)$ time. Furthermore, checking whether H_{odd} and H_{even} are tours in G can also be done in $\mathcal{O}(m)$ time, so that $\{T \in \mathcal{H} : T\backslash H = S\}$ can be determined in $\mathcal{O}(m)$ time. □

Now we are ready to prove Theorem 5.13.

Proof of Theorem 5.13. First, apply Algorithm 5.1 with $H = H_{(1)}$ in order to determine $\{H \in \mathcal{H} : H\backslash H_{(1)} = S\}$. Next, for each tour in $\{H \in \mathcal{H} : H\backslash H_{(1)} = S\}$ check whether it has length $L_d(H_{(1)}) + t_1$. Since Algorithm 5.1 runs in $\mathcal{O}(m)$ time, the set $\{H \in \mathcal{H} : H\backslash H_{(1)} = S, L_d(H) = L_d(H_{(1)}) + t_1\}$ can be determined in $\mathcal{O}(m)$ time. □

Example (continued). Consider (G, d) in Figure 5.1 with the tolerances given in Figure 5.2. Applying Algorithm 5.1 to $H = H_{(1)}$ and $S = \{e \in E\backslash H_{(1)} : l(e) = t_1\} = \{\{1,3\},\{1,4\},\{2,5\},\{4,6\}\}$ gives that $E_1 = \emptyset$, $E_2 = \{\{1,2\}\}$, $E_3 = \{\{2,3\}\}$, $E_4 = \{\{3,4\}\}$, $E_5 = \emptyset$, $E_6 = \{\{4,5\}\}$, $E_7 = \{\{5,6\}\}$, and $E_8 = \{\{1,6\}\}$. The sets H_{odd} and H_{even} are shown in Figure 5.12. Note that only H_{odd} is a tour in G. Since $L_d(H_{odd}) = L_d(H_{(1)}) + t_1$, it follows that H_{odd} is a second-best tour in (G, d). □

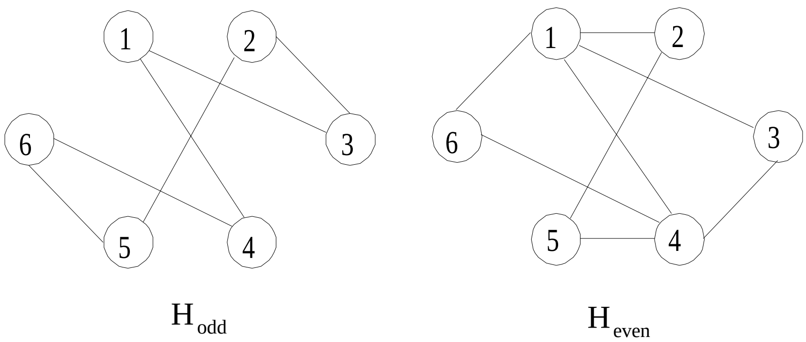

Figure 5.12: H_{odd} and H_{even} in (G, d).