Approximation Hardness of the Traveling Salesman Reoptimization Problem*

Richard Královič and Tobias Mömke

Department of Computer Science, ETH Zurich, Switzerland
{richard.kralovic, tobias.moemke}@inf.ethz.ch

Abstract. In TSP reoptimization, a complete weighted graph and an optimal Hamiltonian tour are given; the problem is to find an optimal solution for a locally modified input. In this paper, we consider the local modification where one edge of the graph becomes cheaper. We show that the best approximation ratio known so far that is achievable by polynomial-time algorithms heavily depends on various parameters. We show that for a wide range of parameters the approximation ratio is improved. Moreover, these parameters limit the class of hard instances, which might help to find better approximation algorithms also for the general case.

1 Introduction

Many practically important problems are proven to be hard. There are several theoretical concepts used for proving problem hardness, such as NP-completeness and APX-completeness. These classical complexity measures, however, assume that there is absolutely no auxiliary information about the problem instance available, which may not be the case in real applications. For example, solutions of similar problem instances may be known. Exploiting this knowledge can make the actual problem easier, as it can eliminate the need to recompute the solution from scratch.

The idea of making some auxiliary information about the considered problem instance available is not new. For example, the goal of the *restricted directed Hamiltonian circuit problem*, introduced in [8], is to find a Hamiltonian tour in a graph with given Hamiltonian path.

To deal with such a-priori knowledge in a more general way, the concept of *reoptimization problems* has been introduced in [1]. The algorithm solving the reoptimization problem is given an instance of the problem together with its optimal solution and a locally modified instance for which the new solution has to be computed. The exact meaning of *local modification* is defined in the reoptimization problem itself. For example, the reoptimization of the traveling salesman problem has been analyzed with respect to adding/removing a single

* This work was partially supported by SBF grant C 06.0108 as part of the COST 293 (GRAAL) project funded by the European Union.

vertex [1, 2] and with respect to changing the cost of a single edge [3, 4]; the reoptimization of the Steiner tree problem has been considered in [5].

Almost any reasonable reoptimization problem admits one to transform a trivial instance (e.g. the TSP in an empty graph or in a graph with unit edge costs) into any other instance using only a polynomial number of local modifications (e.g. adding vertices to the empty graph or changing edge weights in a complete graph with unit-cost edges). These examples show that reoptimization usually does not help to avoid NP-hardness. Nevertheless, the reoptimization approach can be used to obtain improved approximation algorithms. Hence, the study of reoptimization problems usually focuses on developing approximation algorithms with better approximation ratio than their best-known non-reoptimization counterparts, as it is the case in [1–5].

In our work, we focus on the reoptimization of the metric traveling salesman problem, i.e., on TSP instances where the edge costs satisfy the triangle inequality. We consider decreasing the cost of a single edge as the only allowed local modification. This problem has been analyzed in [3], where a reoptimization algorithm with an approximation ratio of 1.4 has been presented. This is an improvement over the best known non-reoptimization algorithm for metric TSP — the Christofides' algorithm — which achieves an approximation ratio of 1.5.

In this paper, we refine the analysis of the algorithm presented in [3] in the case of decreasing cost of a single edge by introducing several variables to parameterize the approximation ratio of the algorithm. Our analysis shows that, although the worst-case approximation ratio of the algorithm can be arbitrarily close to 1.4, very strict conditions are required to hold in that case. Through our analysis, we can guarantee a significantly better approximation ratio for a large class of input instances, and we obtain a characterization of the hard instances of the considered reoptimization problem[1]. We believe that this can help to develop better approximation algorithms for the reoptimization problem, as it allows us to focus on a very special subclass of input instances of the problem.

2 Preliminaries

A problem instance to the metric TSP is given by a complete graph $G = (V, E)$ with cost function $c : E \to \mathbb{Q}^+$ where all edge costs satisfy the triangle inequality, i.e.,

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

for all vertices u, v, and w in V.

Now we formulate the traveling salesman reoptimization problem with reduced edge cost.

Definition 1. *Let $G = (V, E)$ be a complete graph with a metric cost function $c_O : E \to \mathbb{Q}^+$ and let T_O be a minimum-cost Hamiltonian tour for (G, c_O). The* traveling salesman reoptimization problem with reduced edge cost *consists*

[1] The approximation ratio can be equal to 1.4 only if the new cost of the modified edge is equal to 0, which is not allowed by the problem definition.

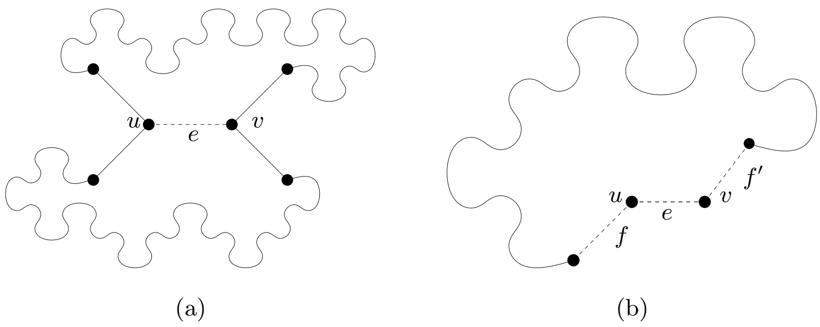


Fig. 1. (a) The old optimal solution T_O. We assume that the edge e is not part of T_O. (b) The solution of Algorithm 1. We assume that e,f, and f' are in T_N.

in finding an optimal Hamiltonian tour T_N for (G, c_N) where $c_N : E \to \mathbb{Q}^+$ is a metric cost function that coincides with c_O for all but one edge e, and $c_N(e) < c_O(e)$.

For simplicity, we simply write $c(e')$ for the cost of all edges $e' \neq e$. We denote the cost of an optimal solution T_O in (G, c_O) with Opt_O and the cost of an optimal solution T_N in (G, c_N) with Opt_N. We define $c(a_{\min})$ to be the minimum-cost edge that is adjacent to e.

If the old tour T_O contains e, then T_O is an optimal solution for (G, c_N) and we are done. Therefore, we assume in the following that T_O does not contain e, see Figure 1a; for the same reason, we assume that T_N contains e.

3 Parameterized Analysis of the Approximation Ratio

In our work, we refine the analysis of Algorithm 1, which has been introduced in [3]. The idea of the algorithm is to consider several Hamiltonian tours in the new graph and to choose the best among them. The old optimum T_O is one of the considered tours. Furthermore, for every pair of edges $\{f, f'\}$ incident to e, the Hamiltonian tour formed by e, f, f', and an approximation of the cheapest Hamiltonian path connecting the vertices of $(f \cup f') \setminus e$ in $G' = (V - e, E - \{e\})$ is considered, see Figure 1b.

We analyze the approximation ratio γ of the algorithm under the condition that $c(a_{\min})$, $c_N(e)$, or Opt_N is bounded. More precisely, we parameterize γ by the ratios $c_N(e)/\mathrm{Opt}_O$, $c_N(e)/c_O(e)$, $c(a_{\min})/\mathrm{Opt}_O$ and $\mathrm{Opt}_N/\mathrm{Opt}_O$. Given an input instance, we know $c(a_{\min})$, $c_N(e)$, and Opt_O. The reason for analyzing the ratio $\mathrm{Opt}_N/\mathrm{Opt}_O$ is to gain knowledge about Opt_N.

For our analysis, we need the following result from [3].

Lemma 1. *Given a graph $G = (V, E)$, let c_1 and c_2 be metric cost functions that coincide, except for one edge $e \in E$. Then, every edge adjacent to e has a cost of at least $\frac{1}{2}|c_1(e) - c_2(e)|$.* □

Algorithm 1 The 1.4-approximation algorithm for the TSP reoptimization.

Input: An instance (G, c_O, c_N, T_O) of the reoptimization problem for the TSP where $G = (V, E)$.

1: Let $e \in E$ be the edge where $c_O(e) \neq c_N(e)$.
   Let $\mathcal{E}$ be the set of all unordered pairs $\{f, f'\} \subseteq E$ where $f \neq f'$ are edges adjacent to e such that $f \cap f' = \emptyset$.

2: For all $\{f, f'\} \in \mathcal{E}$, compute a Hamiltonian path between the two vertices from $(f \cup f') \backslash e$ on the graph $G \backslash (e \cap (f \cup f'))$, using the $\frac{5}{3}$-approximation algorithm presented in [7]. Augment this path by the edges f, f', and e to obtain the Hamiltonian tour $C_{\{f,f'\}}$.

3: Let T_N be the least expensive of the Hamiltonian tours in the set $\{T_O\} \cup \{C_{\{f,f'\}} \mid \{f, f'\} \in \mathcal{E}\}$.

Output: The Hamiltonian tour T_N.

Next, we prove several inequalities that will be useful in the proofs of our results.

Lemma 2. *Let the cost of the solution computed by Algorithm 1 be $\gamma \cdot \mathrm{Opt}_N$. Then the following inequalities hold:*

$$c(a_{\min}) \geq \frac{c_O(e) - c_N(e)}{2} \tag{1}$$

$$\mathrm{Opt}_O - \mathrm{Opt}_N \leq c_O(e) - c_N(e) \tag{2}$$

$$\mathrm{Opt}_O \geq \gamma \cdot \mathrm{Opt}_N \tag{3}$$

$$\mathrm{Opt}_N \geq \frac{4c(a_{\min}) + 2c_N(e)}{5 - 3\gamma} \tag{4}$$

Proof. Inequality (1) follows directly from Lemma 1. Since T_O is optimal in the old graph and the cost of T_N in the old graph is $\mathrm{Opt}_N - c_N(e) + c_O(e)$, Inequality (2) holds.

The cost of the solution computed by Algorithm 1 is always at most Opt_O. If Opt_O is smaller that $\gamma \cdot \mathrm{Opt}_N$, the algorithm is better than γ-approximative, since T_O itself is a good approximation. Thus we can restrict further analysis only to the case of Inequality (3).

Algorithm 1 computes an approximation of a Hamiltonian path with two given endpoints in $G = (V - e, E - \{e\})$. According to [6, 7], this can be done with approximation ratio 5/3. Let f and f' be the edges adjacent to e in the new optimal solution T_N, then the cost of the corresponding solution calculated by the algorithm is at most $c_N(e) + c(f) + c(f') + \frac{5}{3}(\mathrm{Opt}_N - c(f) - c(f') - c_N(e)) = \frac{5}{3}\mathrm{Opt}_N - \frac{2}{3}c_N(e) - \frac{2}{3}c(f) - \frac{2}{3}c(f')$. Hence, we have an upper bound on the solution found by the algorithm,

$$\gamma \cdot \mathrm{Opt}_N \leq \frac{5}{3}\mathrm{Opt}_N - \frac{4}{3}c(a_{\min}) - \frac{2}{3}c_N(e),$$

which immediately implies Equation (4). $\square$

The approximation ratio γ depends strongly on the size of $c(a_{\min})$ relative to Opt_O. In the following theorem, we present the bounds of $c(a_{\min})$ for a given approximation ratio γ.

Theorem 1. *Let (G, c_O, c_N, T_O) be an input of Algorithm 1 and let the cost of the computed solution be $\gamma \cdot \mathrm{Opt}_N$. Then*

$$\frac{1 - 1/\gamma}{2}\mathrm{Opt}_O \leq c(a_{\min}) \leq \frac{(5/\gamma - 3)}{4}\mathrm{Opt}_O.$$

Proof. We determine a lower bound on $c(a_{\min})$ as follows.

$$\begin{aligned} c(a_{\min}) &\underset{(1)}{\geq} \frac{c_O(e) - c_N(e)}{2} \underset{(2)}{\geq} \frac{\mathrm{Opt}_O - \mathrm{Opt}_N}{2} \\ &\underset{(3)}{\geq} \frac{\mathrm{Opt}_O - \mathrm{Opt}_O/\gamma}{2} \geq \frac{1 - 1/\gamma}{2}\mathrm{Opt}_O. \end{aligned}$$

Furthermore, we get the following upper bound on $c(a_{\min})$:

$$\begin{aligned} c(a_{\min}) &\underset{(4)}{\leq} \frac{(5 - 3 \cdot \gamma)\mathrm{Opt}_N - 2c_N(e)}{4} \\ &\leq \frac{(5 - 3 \cdot \gamma)\mathrm{Opt}_N}{4} \underset{(3)}{\leq} \frac{(5/\gamma - 3)\,\mathrm{Opt}_O}{4}. \end{aligned}$$

□

Note that Theorem 1 implies $\lim_{\gamma \to 1.4} c(a_{\min}) = \mathrm{Opt}_O/7$. (As we will show later, $\gamma \neq 1.4$, unless edges of zero cost are allowed.) The result of Theorem 1 can be used to express the approximation ratio γ using the parameter $r_1 = \mathrm{Opt}_O/c(a_{\min})$; Figure 2a shows the graph of the obtained bound on γ.

Corollary 1. *Let $r_1 := \mathrm{Opt}_O/c(a_{\min})$. Then*

$$\gamma \leq \min\left(\frac{r_1}{r_1 - 2}, \frac{5r_1}{4 + 3r_1}\right).$$

The approximation ratio of Algorithm 1 depends also on the size of Opt_N, as shown by the following theorem.

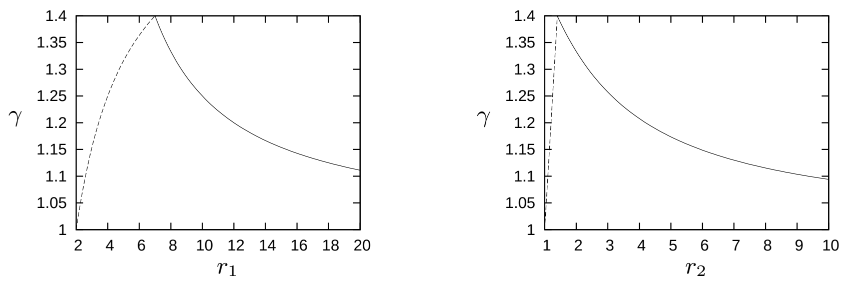
Theorem 2. *Let (G, c_O, c_N, T_O) be an input of Algorithm 1 and let the cost of the computed solution be $\gamma \cdot \mathrm{Opt}_N$. Then*

$$\frac{2 - 2/\gamma}{5 - 3\gamma}\mathrm{Opt}_O \leq \mathrm{Opt}_N \leq \mathrm{Opt}_O/\gamma.$$

Proof. The upper bound follows from simply applying (3). It remains to show the lower bound on Opt_N.

$$\begin{aligned} \mathrm{Opt}_N &\underset{(4)}{\geq} \frac{4c(a_{\min}) + 2c_N(e)}{5 - 3\gamma} \geq \frac{4c(a_{\min})}{5 - 3\gamma} \\ &\underset{(Theorem\ 1)}{\geq} \frac{4\frac{1-1/\gamma}{2}\mathrm{Opt}_O}{5 - 3\gamma} = \frac{2 - 2/\gamma}{5 - 3\gamma}\mathrm{Opt}_O. \end{aligned}$$

□



(a) (b)

Fig. 2. (a) The approximation ratio γ depending on the minimum-cost adjacent edge of e relative to the old optimum Opt_O, i.e., $c(a_{\min}) = \mathrm{Opt}_O/r$. (b) The approximation ratio γ depending on the cost of new optimum Opt_N relative to the old optimum Opt_O, i.e., $\mathrm{Opt}_N = \mathrm{Opt}_O/r$.

Thus, $\lim_{\gamma \to 1.4} \mathrm{Opt}_N = 5\mathrm{Opt}_O/7$ (compare Theorem 1). Again, we can use the result of Theorem 2 to obtain a parameterization of γ using the parameter $r_2 = \mathrm{Opt}_O/\mathrm{Opt}_N$; the obtained bound is shown in Figure 2b.

Corollary 2. *Let $r_2 := \mathrm{Opt}_O/\mathrm{Opt}_N$. Then*

$$\gamma \leq \min\left(r_2,\, 5/6 - r_2/3 + \sqrt{4r_2^2 + 4r_2 + 25/6}\right).$$

The main purpose of Theorem 2 and of Corollary 2 is to gain knowledge about Opt_N and thus also knowledge about the achieved approximation ratio for a given input instance.

Our next results deal with the relationship of γ and $c_N(e)$. At first, we consider the ratio of $c_N(e)$ and Opt_O as the parameter.

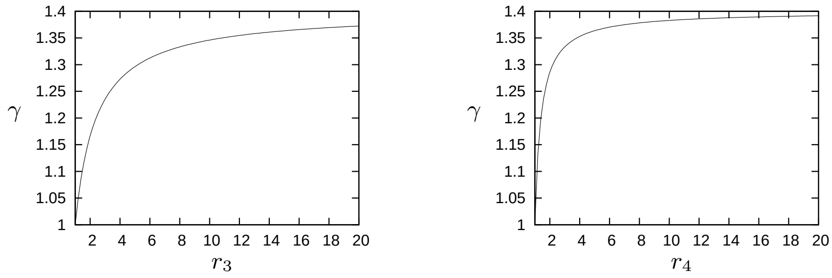
Theorem 3. *Let (G, c_O, c_N, T_O) be an input of Algorithm 1 and let the cost of the computed solution be $\gamma \cdot \mathrm{Opt}_N$. Then*

$$0 < c_N(e) \leq \left(\frac{7}{2\gamma} - \frac{5}{2}\right)\mathrm{Opt}_O.$$

Proof. The first inequality, $0 < c_N(e)$, holds trivially since all edge costs are positive. We determine an upper bound on $c_N(e)$ as follows.

$$\begin{aligned} c_N(e) \quad &\underset{(4)}{\leq} \quad \frac{(5-3\gamma)\mathrm{Opt}_N - 4c(a_{\min})}{2} \underset{(3)}{\leq} \left(\frac{5}{2\gamma} - \frac{3}{2}\right)\mathrm{Opt}_O - 2c(a_{\min}) \\ &\underset{(Theorem\ 1)}{\leq} \left(\frac{5}{2\gamma} - \frac{3}{2}\right)\mathrm{Opt}_O - 2 \cdot \frac{1-1/\gamma}{2}\mathrm{Opt}_O = \left(\frac{7}{2\gamma} - \frac{5}{2}\right)\mathrm{Opt}_O. \end{aligned}$$

□



(a) (b)

Fig. 3. The approximation ratio γ depending on the cost of e_n (a) relative to Opt_O, i.e., $c_N(e) = \text{Opt}_O/r_1$ and (b) relative to Opt_O, i.e., $c_N(e) = c_O(e)/r_2$.

Theorem 3 implies that Algorithm 1 is always better than 1.4-approximative, unless zero-cost edges are allowed. Nevertheless, the approximation ratio of the algorithm can be arbitrarily close to 1.4. As in the previous cases, we can bound γ by the parameter $r_3 = \text{Opt}_O/c_N(e)$; Figure 3a shows the graph of the obtained bound.

Corollary 3. *Let* $r_3 := \text{Opt}_O/c_N(e)$. *Then*

$$\gamma \leq \frac{7r_3}{5r_3 + 2}.$$

Corollary 3 shows that a large value for $c_N(e)$ implies a good approximation ratio.

The result of Theorem 3 can be also used to bound γ by the parameter $r_4 = c_O(e)/c_N(e)$, yielding the following corollary. The obtained bound is shown in Figure 3b.

Corollary 4. *Let* $r_4 := c_O(e)/c_N(e)$. *Then*

$$\gamma \leq \frac{7r_4 - 5}{5r_4 - 3}.$$

Proof. Since $c_O(e) \underset{(2)}{\geq} \text{Opt}_O - \text{Opt}_N + c_N(e)$, we know

$$c_N(e) \geq \frac{c_N(e)}{c_O(e)}(\text{Opt}_O - \text{Opt}_N + c_N(e)) = \frac{1}{r_4}(\text{Opt}_O - \text{Opt}_N + c_N(e)). \quad (5)$$

Hence,

$$c_N(e) \underset{(5)}{\geq} \frac{\text{Opt}_O - \text{Opt}_N}{r_4 - 1} \underset{(3)}{\geq} \frac{\text{Opt}_O - \frac{\text{Opt}_O}{\gamma}}{r_4 - 1} = \frac{\gamma - 1}{\gamma(r_4 - 1)}\text{Opt}_O.$$

Applying Theorem 3, we get

$$\frac{\gamma-1}{\gamma(r-1)}\mathrm{Opt}_O \le \left(\frac{7}{2\gamma}-\frac{5}{2}\right)\mathrm{Opt}_O,$$

and the claim of the corollary follows after cancelling Opt_O and solving the inequality. □

Corollary 4 shows that the difference of $c_N(e)$ and $c_O(e)$ directly influences the approximation ratio: a small difference implies a good approximation ratio. The absolute difference of the costs, however, is not important; only the factor r matters.

4 Conclusion

A detailed analysis of the approximation algorithm for the traveling salesman reoptimization problem from [3] reveals that (i) this algorithm's approximation ratio is always better than (although arbitrarily close to) 1.4 and that (ii) the achieved approximation ratio depends largely on the parameters $c(a_{\min})$, $c_N(e)$, and Opt_N. The remaining class of input instances is very limited. We therefore expect that it is possible to find an algorithm with better approximation ratio for all instances in future work.

References

1. Archetti, C., Bertazzi, L., Speranza, M.G.: Reoptimizing the traveling salesman problem. Networks **42**(3) (2003) 154–159
2. Ausiello, G., Escoffier, B., Monnot, J., Paschos, V.T.: Reoptimization of minimum and maximum traveling salesman's tours. In: Algorithm theory—SWAT 2006. Volume 4059 of Lecture Notes in Comput. Sci. Springer, Berlin (2006) 196–207
3. Böckenhauer, H.J., Forlizzi, L., Hromkovič, J., Kneis, J., Kupke, J., Proietti, G., Widmayer, P.: Reusing optimal TSP solutions for locally modified input instances (extended abstract). In: Fourth IFIP International Conference on Theoretical Computer Science—TCS 2006. Volume 209 of IFIP Int. Fed. Inf. Process. Springer, New York (2006) 251–270
4. Böckenhauer, H.J., Forlizzi, L., Hromkovič, J., Kneis, J., Kupke, J., Proietti, G., Widmayer, P.: On the approximability of TSP on local modifications of optimally solved instances. Algorithmic Operations Research (to appear)
5. Escoffier, B., Milanic, M., Paschos, V.T.: Simple and fast reoptimizations for the Steiner tree problem. Technical Report 2007-01, DIMACS (2007)
6. Guttmann-Beck, N., Hassin, R., Khuller, S., Raghavachari, B.: Approximation algorithms with bounded performance guarantees for the clustered traveling salesman problem. Algorithmica **28**(4) (2000) 422–437
7. Hoogeveen, J.A.: Analysis of Christofides' heuristic: some paths are more difficult than cycles. Oper. Res. Lett. **10**(5) (1991) 291–295
8. Papadimitriou, C.H., Steiglitz, K.: On the complexity of local search for the traveling salesman problem. SIAM J. Comput. **6**(1) (1977) 76–83