Reusing Optimal TSP Solutions for Locally Modified Input Instances*
(Extended Abstract)

Hans-Joachim Böckenhauer[1], Luca Forlizzi[2], Juraj Hromkovič[1], Joachim Kneis[3**], Joachim Kupke[1], Guido Proietti[2,4], and Peter Widmayer[1]

[1] Department of Computer Science, ETH Zurich, Switzerland,
{hjb,juraj.hromkovic,jkupke,widmayer}@inf.ethz.ch
[2] Department of Computer Science, Università di L'Aquila, Italy,
{forlizzi,proietti}@di.univaq.it
[3] Department of Computer Science, RWTH Aachen University, Germany,
joachim.kneis@cs.rwth-aachen.de
[4] Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti", CNR, Roma, Italy

Abstract. Given an instance of an optimization problem together with an optimal solution, we consider the scenario in which this instance is modified locally. In graph problems, e. g., a singular edge might be removed or added, or an edge weight might be varied, etc. For a problem U and such a local modification operation, let LM-U (local-modification-U) denote the resulting problem. The question is whether it is possible to exploit the additional knowledge of an optimal solution to the original instance or not, i. e., whether LM-U is computationally more tractable than U. Here, we give non-trivial examples both of problems where this is and problems where this is not the case. Our main results are these:

1. The local modification to change the cost of a singular edge turns the traveling salesperson problem (TSP) into a problem LM-TSP which is as hard as TSP itself, i. e., unless $P = NP$, there is no polynomial-time $p(n)$-approximation algorithm for LM-TSP for any polynomial p. Moreover, LM-TSP where inputs must satisfy the β-triangle inequality (LM-Δ_β-TSP) remains NP-hard for all $\beta > \frac{1}{2}$.
2. For LM-Δ-TSP (i. e., metric LM-TSP), an efficient 1.4-approximation algorithm is presented. In other words, the additional information enables us to do better than if we simply used Christofides' algorithm for the modified input.
3. Similarly, for all $1 < \beta < 3.34899$, we achieve a better approximation ratio for LM-Δ_β-TSP than for Δ_β-TSP.
4. Metric TSP with deadlines (time windows), if a single deadline or the cost of a single edge is modified, exhibits the same lower bounds on the approximability in these local-modification versions as those currently known for the original problem.

* This work was partially supported by SNF grant 200021-109252/1, by the research project GRID.IT, funded by the Italian Ministry of Education, University and Research, and by the COST 293 (GRAAL) project funded by the European Union.

** This author was staying at ETH Zurich when this work was done.

1 Introduction

Traditionally, optimization theory has been concerned with the task of finding good feasible solutions to (practically relevant) input instances, little or nothing about which is known in advance. Many applications, however, demand good, sometimes optimal, solutions to a limited set of input instances which reflect a supposedly-constant environment (imagine, e. g., an existing railway system or communications network). When this environment does change, maybe only slightly and maybe only locally, do we have no choice but to recompute some good feasible solution, effectively forgetting about the old one?

Here, we will analyze *local* modifications only. In a graph problem, for example, the cost of a single edge might change, an edge might be removed or added, or some other local parameter might be adjusted. Results related to this work pertain to the question by how much a given instance of an optimization problem may be varied if it is desired that optimal solutions to the original instance retain their optimality [12, 17, 18, 20, 21]. In contrast with this so-called "postoptimality analysis," our approach here is to ask, if we cannot avoid to lose the optimality of a given solution when an instance is varied arbitrarily, what can we do to *restore* the quality of a solution, maybe in an approximative sense?

Surely, for some problems, knowing an optimal solution to the original instance trivially makes their local-modification variants easy to solve because the given optimal solution is itself a very good solution to the modified instance. For example, adding an edge in the instance of a coloring problem will increase the cost of an optimal solution by at most the amount of one – an excellent approximation, but certainly not the object of our interest.

Our goal is to present non-trivial examples of problems, some where the knowledge of an optimal solution to an instance close to the input is helpful and some where it is not. To this end, we will study TSP, its restricted versions, and its generalizations such as TSP with deadlines (a special case of TSP with time windows).

Let Δ-TSP denote metric TSP, and, for all $\beta \geq \frac{1}{2}$, let Δ_β-TSP denote the special case of TSP where all instances satisfy the β-triangle inequality

$$c(\{x,z\}) \leq \beta \cdot \big(c(\{x,y\}) + c(\{y,z\})\big)$$

for all vertices x, y, and z. If $\frac{1}{2} \leq \beta < 1$, we call this the *strengthened* triangle inequality; and if $\beta > 1$, we call it the *relaxed* triangle inequality.

For an optimization problem U, we denote our local-modification variant of U by LM-U. For the aforementioned TSP-based problems, we regard it as a local modification to change the cost of exactly one edge. For TSP with deadlines, we also regard it as a local modification to shift one deadline by the amount of at least one time unit.

Our main results are as follows:

(i) It is well-known that TSP is not approximable in polynomial time with a polynomial approximation ratio (unless $P = NP$). We show that this

holds for LM-TSP, too. Thus, in terms of a worst-case analysis, LM-TSP is as hard as TSP, and we do not have anything to gain from knowing an optimal solution to a close problem instance. By parameterizing TSP with respect to the β-triangle inequality [1, 2, 3, 4, 5] and by introducing the concept of stability of approximation [15, 5], it was shown that TSP is not as hard as it may look like in the light of worst-case analyses. For any $\beta > \frac{1}{2}$, we have a constant polynomial-time approximation ratio, depending on β only. Böckenhauer and Seibert [8] proved that Δ_β-TSP is APX-hard for every $\beta > \frac{1}{2}$ (note that for $\beta = \frac{1}{2}$, the problem becomes trivially solvable in polynomial time). Here, we prove that LM-Δ_β-TSP is NP-hard for every $\beta > \frac{1}{2}$. This implies in particular that LM-Δ-TSP, too, is NP-hard. We conjecture that this problem is also APX-hard, which, so far, we have been unable to prove and thus leave as an open research problem.

(ii) For many years, Christofides' algorithm [9] with its approximation ratio of 1.5 has been the best known approximation algorithm for attacking Δ-TSP. It remains a grand challenge to improve on Christofides' algorithm. We will show that, intriguingly enough, LM-Δ-TSP admits an efficient 1.4-approximation algorithm. This result can be generalized to LM-Δ_β-TSP, and the resulting approximation guarantee beats all previously-known approximation algorithms for Δ_β-TSP for all $1 < \beta < 3.34899$, which includes the practically most relevant TSP instances.

(iii) TSP with time windows is one of the fundamental problems in operations research [10]. Usually, only heuristic algorithms are used to attack it although the question how hard it is w. r. t. approximability has only been resolved in [6, 7], where even an $\Omega(n)$ lower bound on the polynomial-time approximability of Δ-TSP with time windows was shown, in contrast to the constant approximability of Δ-TSP. This lower bound already holds for the special case of this problem where all time windows are immediately open, a special case of the problem which we will call TSP with deadlines, or Δ-DLTSP for short. Here, we consider local-modification versions of Δ-TSP with deadlines. We show that already if we only allow a single deadline to be changed, and only by an amount of one time unit, the resulting problem, LM-Δ-DLTSP, has the same lower bound of $\Omega(n)$ on the approximation ratio as Δ-DLTSP. Let us underscore the importance of this negative result: Not only does TSP with deadlines remain an intractable problem in its LM version, but the extra knowledge of an optimal solution to a related instance does not even help a single bit. Likewise, we will establish the lower bound of $(2 - \varepsilon)$, for any $\varepsilon > 0$, for LM-Δ-DLTSP with a constant number of deadlines, the same as is known for Δ-DLTSP with a constant number of deadlines [6, 7]. These results can also be obtained if, again, we modify the cost of an edge rather than a deadline.

So, on the one hand, additional information about an optimal solution to a related input instance may be useful to some extent, and on the other hand, the local-modification problem variant may remain exactly as hard as the original problem. Yet, the final aim of our paper is to call forth the investigation of

the hardness of local-modification optimization problems in order to develop approaches to handle situations where multiple (and, potentially, dynamically determined) local modifications may arise.

The paper is subdivided into two main sections. In Section 2, we will analyze TSP with local modifications and present hardness results as well as approximation algorithms for the metric and near-metric case. Section 3 is devoted to inapproximability results for the local-modification version of TSP with deadlines.

2 Results for TSP

In this section, we will analyze the local-modification version of TSP. In Subsection 2.1, we will present our hardness results. In Subsection 2.2, we will present a 1.4-approximation algorithm for the local-modification metric TSP, and Subsection 2.3 is devoted to approximability results for the case of the relaxed triangle inequality.

We start off with a formal definition of TSP and its local-modification variants.

Definition 1. *Let $G = (V, E, c)$ be a weighted complete graph, and let $\beta \geq \frac{1}{2}$ be a real value. We say that G obeys the Δ_β-inequality iff for all vertices x, y, $z \in V$, we have*

$$c(\{x,z\}) \leq \beta \cdot \big(c(\{x,y\}) + c(\{y,z\})\big) \quad . \qquad (\Delta_\beta)$$

By TSP*, we denote the following optimization problem. For a given weighted complete graph $G = (V, E, c)$, find a minimum cost Hamiltonian cycle, i. e., a tour on all vertices of cost*

$$OT_G := \min \left\{ \sum_{e \in C'} c(e) \;\middle|\; (V, C') \text{ is a Hamiltonian cycle} \right\} \quad .$$

*Restricting, for some value of β, the set of admissible input instances to those which obey the Δ_β-inequality yields the problem Δ_β-*TSP*. Besides, define Δ-*TSP* := Δ_1-*TSP*.*

Definition 2. *Let $U \in \{\text{TSP}, \Delta\text{-TSP}, \Delta_\beta\text{-TSP}\}$. The problem* LM-*$U$ is defined as follows.*
Input:

- *two complete weighted graphs $G_O = (V, E, c_O)$, $G_N = (V, E, c_N)$ such that G_O and G_N are both admissible inputs for U and such that c_O and c_N coincide, except for one edge;*
- *a Hamiltonian cycle $(V, \overline{C})$ such that $\sum_{e \in \overline{C}} c_O(e) = OT_{G_O}$.*

Problem: *Find a Hamiltonian cycle (V, C) that minimizes $\sum_{e \in C} c_N(e)$.*

2.1 Hardness Results

Before presenting approximation algorithms for LM-Δ-TSP, we start by proving some hardness results.

First, we will show that LM-TSP is as hard to approximate as "normal" (i. e., unaltered) TSP.

Theorem 1. *There is no polynomial-time $p(n)$-approximation algorithm for* LM-TSP *for any polynomial p (unless $P = NP$).*

Proof idea. We will give a reduction from the Hamiltonian cycle problem (HC): Given an undirected, unweighted graph G, decide whether G contains a Hamiltonian cycle or not. Let $G = (V, E)$ be an input instance for HC where $V = \{v_1, \ldots, v_n\}$.

In order to construct an input instance $(G_O, G_N, \overline{C})$ for LM-TSP, we employ a graph construction due to Papadimitriou and Steiglitz [19], who used the same construction in order to give examples of TSP instances which are hard for local search strategies: For each vertex v_i, we construct a so-called diamond graph D_i as shown in Figure 1 (a). These diamonds are connected as shown in Figure 1 (b).

The edge costs in G_O are set as follows. Let $M := n \cdot 2^n + 1$. All diamond edges shown in Figure 1 (a) and the connections from E_i to W_{i+1} and from E_n to W_1 as shown in Figure 1 (b) are assigned a cost of 1 each. Edges $\{N_i, S_j\}$ are assigned a cost of 1 whenever $\{v_i, v_j\} \in E$ and a cost of M otherwise. All other edges receive a cost of M each. In G_N, the cost of the edge $\{E_n, W_1\}$ is changed from 1 to M. The given optimal Hamiltonian cycle $\overline{C}$ is the one shown in Figure 1 (b). This optimal solution for G_O has a cost of $8n$.

It is easy to see that if there is a Hamiltonian cycle H' in G, a corresponding Hamiltonian cycle H in G can traverse all diamonds from N_i via W_i via E_i to S_i. Hence, $c_N(H) = 8n$. All Hamiltonian cycles in G_N that do not correspond (in this way) to Hamiltonian cycles in G cost at least $M + 8n - 1$. Thus, the approximation ratio of any non-optimal solution is at least as bad as $1 + 2^{n-3}$. For a more detailed description of diamond graph constructions, also see, for example, [16]. □

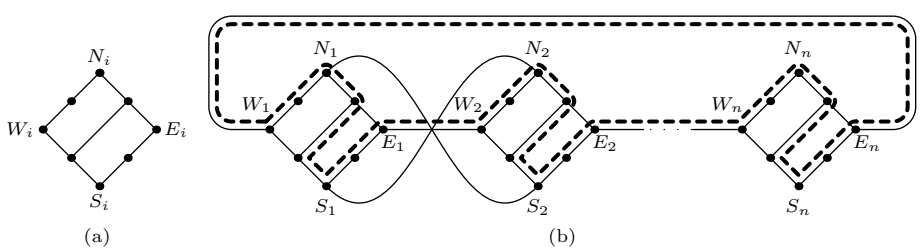

Fig. 1. The diamond construction in the proof of Theorem 1.

Now, we will show that LM-Δ-TSP remains a hard problem for any $\beta > \frac{1}{2}$.

Theorem 2. LM-Δ_β-TSP *is NP-hard for any* $\beta > \frac{1}{2}$.

Proof. We will use a reduction from the restricted Hamiltonian cycle problem (RHC). The objective in RHC is, given an unweighted, undirected graph G and a Hamiltonian path P in G which cannot be trivially extended to a Hamiltonian cycle by joining its end-points, to decide whether a Hamiltonian cycle in G exists. This problem is well-known to be NP-complete (see, for example, [16]).

The reduction uses an idea analogous to the standard reduction from the Hamiltonian cycle problem to TSP: Let (G, P) be an instance of RHC where $G = (V, E)$, $V = \{v_1, \ldots, v_n\}$, and $P = (v_1, \ldots, v_n)$. From this, we construct an instance $(G_O, G_N, \overline{C})$ of LM-Δ_β-TSP as follows: Let $G_O = (V, \tilde{E}, c_O)$ and $G_N = (V, \tilde{E}, c_N)$ where $(V, \tilde{E})$ is a complete graph, $c_O(e) = 1$ for all $e \in E \cup \{\{v_n, v_1\}\}$ and $c_O(e) = 2\beta$ otherwise, and $c_N(\{v_n, v_1\}) = 2\beta$. Let $\overline{C} = (v_1, v_2, \ldots, v_n, v_1)$. Clearly, this reduction can be done in polynomial time, and it is easy to see that there is a Hamiltonian cycle in G iff there is a Hamiltonian cycle of cost n in G_N. □

2.2 The Metric Case

In what follows, we will show that LM-Δ-TSP admits a $\frac{7}{5}$-approximation, which beats the naïve approach of using Christofides' algorithm (which would yield a $\frac{3}{2}$-approximation), whereby the input cycle $(V, \overline{C})$ would be ignored altogether.

Theorem 3. *There is a* 1.4*-approximation algorithm for* LM-Δ-TSP.

In order to prove Theorem 3, we will need the following few lemmas. Our crucial observation is that in a metric graph, all of the neighboring edges of short edges can only be modified by small amounts.

Lemma 1. *Let* $G_1 = (V, E, c_1)$ *and* $G_2 = (V, E, c_2)$ *be metric graphs such that* c_1 *and* c_2 *coincide, except for one edge* $e \in E$*. Then, every edge adjacent to* e *has a cost of at least* $\frac{1}{2}|c_1(e) - c_2(e)|$.

Proof. We set $\{a, a'\} := \{c_1(e), c_2(e)\}$ such that $a' > a$ and $\delta := a' - a$. Let $f \in E$ be any edge adjacent to e, and for any such f, let $f' \in E$ be the one edge that is adjacent to both e and f. Then, by the triangle inequality, we have:

$$a' \leq c(f) + c(f') \qquad\qquad c(f') \leq c(f) + a$$

and hence $a' - a \leq 2c(f)$. □

We will have to distinguish two cases. Either, an edge becomes more expensive, or it becomes less expensive. In either case, our strategy is to compare the input solution (to the old problem instance) with an approximate solution (to the new problem instance).

Let us start with the latter case.

Lemma 2. *Let* $(G_O, G_N, \overline{C})$ *be an admissible input for* LM-Δ-TSP *such that* $\delta := c_O(e) - c_N(e) > 0$ *for the edge* e*. If* $\frac{\delta}{OT_{G_N}} \leq \frac{2}{5}$*, it is a* $\frac{7}{5}$*-approximation to output the feasible solution* $C := \overline{C}$ *for* LM-Δ-TSP*.*

Proof.

$$\frac{c_N(\overline{C})}{OT_{G_N}} \leq \frac{c_O(\overline{C})}{OT_{G_N}} = \frac{OT_{G_O}}{OT_{G_N}} \leq \frac{OT_{G_N} + \delta}{OT_{G_N}} = 1 + \frac{\delta}{OT_{G_N}} \leq 1 + \frac{2}{5} = \frac{7}{5}$$

□

Lemma 3. *Let* $(G_O, G_N, \overline{C})$ *be an admissible input for* LM-Δ-TSP *such that* $\delta := c_O(e) - c_N(e) > 0$ *for the edge* e*. If* $\frac{\delta}{OT_{G_N}} \geq \frac{2}{5}$*, there is a* $\frac{7}{5}$*-approximation for* LM-Δ-TSP*.*

Proof. We may assume that optimal TSP tours in G_N use the edge e. For if they did not, $\overline{C}$ would already constitute an optimal solution. Fix one such optimal tour C_{OPT} in G_N. In C_{OPT}, e is adjacent to two edges f and f'. Let v be the vertex incident with f, but not with e, and let v' be the vertex incident with f', but not with e. By P, denote the path from v to v' in C_{OPT} that does *not* involve e.

Consider the following algorithm: For every pair $\tilde{f}$, $\tilde{f}'$ of disjoint edges, both of which are adjacent to e, compute an approximate solution to the TSP path problem on the subgraph of G_N induced by the vertex set $V \setminus e$ (i. e., without *two* vertices) with start vertex $\tilde{v}$ and end vertex $\tilde{v}'$ where $\{\tilde{v}\} = \tilde{f} \setminus e$ and $\{\tilde{v}'\} = \tilde{f}' \setminus e$. It is known [13, 14] that this can be done with an approximation guarantee of $\frac{5}{3}$. Each of these paths is augmented by $\tilde{f}$, e, and $\tilde{f}'$ so as to yield a TSP tour. The algorithm concludes by outputting the least expensive of all of these tours.

Note that since *all* pairs $\tilde{f}$, $\tilde{f}'$ are taken into account, one of the considered tours uses exactly those edges $\tilde{f} = f$, $\tilde{f}' = f'$ that C_{OPT} uses. This is why the algorithm outputs a tour of cost at most

$$c(f) + c(f') + c_N(e) + \frac{5}{3}c(P) = \big(OT_{G_N} - c(P)\big) + \frac{5}{3}c(P) = OT_{G_N} + \frac{2}{3}c(P)$$

(where c is short-hand notation for c_N wherever c_O and c_N coincide) and thus achieves an approximation guarantee of

$$1 + \frac{2}{3} \cdot \frac{c(P)}{OT_{G_N}} \quad .$$

Since by Lemma 1, $\min\{c(f), c(f')\} \geq \frac{\delta}{2}$ for $i \in \{1, 2\}$, we have $OT_{G_N} - c(P) \geq \delta$ and hence:

$$\frac{c(P)}{OT_{G_N}} \leq 1 - \frac{\delta}{OT_{G_N}} \leq \frac{3}{5} \quad .$$

So, we obtain an overall approximation guarantee of $1 + \frac{2}{5} = \frac{7}{5}$. □

Corollary 1. *There is a $\frac{7}{5}$-approximation algorithm for the subproblem of* LM-Δ-TSP *where edges may only become less expensive.*

Proof. Compute, as laid out in Lemma 3, an approximate solution to LM-Δ-TSP and compare it with the input solution $\overline{C}$. Output the less expensive of the two solutions. Depending on whether the value of $\frac{\delta}{OT_{G_N}}$ (where $\delta := c_O(e) - c_N(e) > 0$) is less or greater than $\frac{2}{5}$ (which we cannot necessarily tell), one of the considered two feasible solutions is a $\frac{7}{5}$-approximation. □

We will now turn to the case where an edge becomes more expensive. We can state a lemma akin to Lemma 2, but notice that by reusing a formerly optimal solution, we incur a certain extra cost.

Lemma 4. *Let $(G_O, G_N, \overline{C})$ be an admissible input for* LM-Δ-TSP *such that $\delta := c_N(e) - c_O(e) > 0$ for the edge e. If $\frac{\delta}{OT_{G_N}} \leq \frac{2}{5}$, it is a $\frac{7}{5}$-approximation to output the feasible solution $C := \overline{C}$ for* LM-Δ-TSP.

Proof.

$$\frac{c_N(\overline{C})}{OT_{G_N}} \leq \frac{c_O(\overline{C}) + \delta}{OT_{G_N}} = \frac{OT_{G_O} + \delta}{OT_{G_N}} \leq \frac{OT_{G_N} + \delta}{OT_{G_N}} = 1 + \frac{\delta}{OT_{G_N}} \leq 1 + \frac{2}{5} = \frac{7}{5}$$

□

When computing an approximate solution, things become slightly different from what they used to be like in Lemma 3: We may assume that e *used* to be a part of $\overline{C}$ and that a new solution should no longer use it. Instead, it will use two edges f and f' such that f and f' are non-disjoint and both incident with the same vertex of e. This pair may be chosen at either end-point of e, a choice which is completely *arbitrary.*

We conjecture that, if an improvement of the approximation guarantee is possible, this is precisely the point where to start at.

Lemma 5. *Let $(G_O, G_N, \overline{C})$ be an admissible input for* LM-Δ-TSP *such that $\delta := c_N(e) - c_O(e) > 0$ for the edge e. If $\frac{\delta}{OT_{G_N}} \geq \frac{2}{5}$, there is a $\frac{7}{5}$-approximation for* LM-Δ-TSP.

Proof. We may assume that optimal TSP tours in G_N do not use the edge e. For if they did, $\overline{C}$ would already constitute an optimal solution. Fix one such optimal tour C_{OPT}, and fix one vertex w incident with e. In C_{OPT}, w is incident with two edges f and f'. Let v be the vertex incident with f, but not with e, and let v' be the vertex incident with f', but not with e. By P, denote the path from v to v' in C_{OPT} that does *not* involve w.

Consider the following algorithm: For every pair $\tilde{f}$, $\tilde{f}'$ of edges incident with w, compute an approximate solution to the TSP path problem on the subgraph of G_2 induced by the vertex set $V \setminus \{w\}$ with start vertex $\tilde{v}$ and end vertex $\tilde{v}'$ where $\{\tilde{v}\} = \tilde{f} \setminus e$ and $\{\tilde{v}'\} = \tilde{f}' \setminus e$. It is known [13, 14] that this can be done

with an approximation guarantee of $\frac{5}{3}$. Each of these paths is augmented by $\tilde{f}$ and $\tilde{f}'$ so as to yield a TSP tour. The algorithm concludes by outputting the least expensive of all of these tours.

Note that since *all* pairs $\tilde{f}$, $\tilde{f}'$ are taken into account, one of the considered tours uses exactly those edges $\tilde{f} = f$, $\tilde{f}' = f'$ that C_{OPT} uses. This is why the algorithm outputs a tour of cost at most

$$c(f) + c(f') + \frac{5}{3}c(P) = \big(OT_{G_N} - c(P)\big) + \frac{5}{3}c(P) = OT_{G_N} + \frac{2}{3}c(P) \quad ,$$

just as in the proof of Lemma 3. □

Using the same arguments as in the proof of Corollary 1, the preceding lemma yields the following corollary.

Corollary 2. *There is a $\frac{7}{5}$-approximation algorithm for the subproblem of* LM-Δ-TSP *where edges may only become more expensive.* □

2.3 The Near-Metric Case

The algorithm outlined in Lemma 3 can be generalized to graphs which are not necessarily metric, but only near-metric, i. e., where the metricity constraint is relaxed by a factor of β. Since it will pay off later, let us pay extra attention to the fact that input instances for all the problems from Definition 2 contain two distinct graphs, potentially obeying relaxed triangle inequalities according to different values of β.

Notice that the parameter β need not be greater for the graph with the costlier edge. Under some circumstances, it might even decrease when we modify the cost of a single edge. In the following generalization of Lemma 1, the convention is therefore that c_1 is the cost function of the less expensive graph, c_2 that of the more expensive one, and both c_i obey the Δ_{β_i}-inequality, $i \in \{1, 2\}$.

Lemma 6. *Let $G_1 = (V, E, c_1)$ and $G_2 = (V, E, c_2)$ be graphs such that c_i obeys the Δ_{β_i}-inequality for $i \in \{1, 2\}$ and some values $\beta_1, \beta_2 \geq 1$ and such that c_1 and c_2 coincide, except for one edge $e \in E$. By convention, let $c_1(e) \leq c_2(e)$. Then, every edge adjacent to e has a cost of at least $\frac{c_2(e) - \beta_1\beta_2 c_1(e)}{\beta_1\beta_2 + \beta_2}$.*

Proof. Analogous to Lemma 1. □

Note that for relatively small changes, the value $c_2(e) - \beta_1\beta_2 c_1(e)$ may well be non-positive, rendering Lemma 6 trivial in such a case.

The algorithm from Lemmas 3 and 4 should be adjusted to accommodate for the relaxation of the triangle inequality. More precisely, in order to find a Hamiltonian path between a given pair of vertices in a β-metric graph, we will employ the algorithm by Forlizzi et al. [11], a variation of the path-matching Christofides algorithm (PMCA, see [5]) for the path version of near-metric TSP, which yields an approximation guarantee of $\frac{5}{3}\beta^2$. This gives us Algorithm 1.

Algorithm 1

Input: An instance $(G_O, G_N, \overline{C})$ of LM-Δ_β-TSP where $G_O = (V, E, c_O)$ and $G_N = (V, E, c_N)$.

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 if $c_O(e) < c_N(e)$: $f \cap f' \cap e$ is a singleton; and
   if $c_O(e) > c_N(e)$: $f \cap f' = \emptyset$.
2. For all $\{f, f'\} \in \mathcal{E}$, compute a Hamiltonian path between the two vertices from $(f \cup f') \setminus e$ on the graph $G \setminus (e \cap (f \cup f'))$, using the PMCA path variant by Forlizzi et al. [11]. Augment this path by edges f, f', and, if $c_O(e) > c_N(e)$, edge e to obtain the cycle $C_{\{f,f'\}}$.
3. Let C be the least expensive of the cycles in the set $\{\overline{C}\} \cup \{C_{\{f,f'\}} \mid \{f, f'\} \in \mathcal{E}\}$.

Output: The Hamiltonian cycle C.

Lemma 7. *Algorithm 1 achieves an approximation guarantee of*

$$\beta_{\mathsf{L}}\beta_{\mathsf{H}} \cdot \frac{15\beta_{\mathsf{L}}^2 + 5\beta_{\mathsf{L}} - 6}{10\beta_{\mathsf{L}}^2 + 3\beta_{\mathsf{L}}\beta_{\mathsf{H}} + 3\beta_{\mathsf{H}} - 6} \tag{1}$$

for input graph pairs (G_O, G_N) such that G_O obeys the Δ_{β_O}-inequality and G_N obeys the Δ_{β_N}-inequality and where $\beta_{\mathsf{L}} := \min\{\beta_O, \beta_N\}$ and $\beta_{\mathsf{H}} := \max\{\beta_O, \beta_N\}$.

Proof. Adhering to the convention of Lemma 6, set $\{c_1, c_2\} = \{c_O, c_N\}$ such that $c_1(e) \leq c_2(e)$ for all edges $e \in E$. In other words, we have $c_2 = c_N$ if an edge becomes more expensive and $c_1 = c_N$ otherwise.

We may assume that optimal TSP tours in $G_N = (V, E, c_N)$ use the edge e iff $c_N = c_1$; otherwise, $\overline{C}$ is an optimal solution, and we are done. Fix one such optimal tour C_{OPT} in G_N, and let $\{f, f'\} \in \mathcal{E}$ be such that C_{OPT} uses both f and f'. By P, denote the path that results from C_{OPT} by removing edges f, f', and, potentially, e. Set

$$\alpha := \frac{C(P)}{OT_{G_N}} \quad \text{and let, for brevity,} \quad \vartheta := \beta_{\mathsf{L}}\beta_{\mathsf{H}} \cdot \frac{15\beta_{\mathsf{L}}^2 + 5\beta_{\mathsf{L}} - 6}{10\beta_{\mathsf{L}}^2 + 3\beta_{\mathsf{L}}\beta_{\mathsf{H}} + 3\beta_{\mathsf{H}} - 6}$$

denote the approximation guarantee claimed in (1). In terms of α, Algorithm 1 always achieves an approximation guarantee of

$$\underbrace{1 - \alpha}_{\text{edges } f,\, f',\, \text{(potentially) } e \text{ are chosen optimally}} + \underbrace{\frac{5}{3}\beta_{\mathsf{L}}^2\alpha}_{P \text{ will be approximated}},$$

even if we did not have $\overline{C}$ at our disposal. (Note that the strategy to approximate P may rely on the $\Delta_{\beta_{\mathsf{L}}}$ inequality, i. e., the less relaxed one of the two because this strategy removes the edge e from the graph.) Hence, unless

$$\alpha > \frac{\vartheta - 1}{\frac{5}{3}\beta_{\mathsf{L}}^2 - 1} \quad , \tag{2}$$

we are done. Let use therefore assume that (2) holds. By Lemma 6, we have

$$\min\{c(f), c(f')\} \geq \frac{c_2(e) - \beta_1\beta_2 c_1(e)}{\beta_1\beta_2 + \beta_2} \geq \frac{c_2(e) - \beta_{\mathsf{L}}\beta_{\mathsf{H}} c_1(e)}{\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}}}$$

and hence

$$1 - \alpha \geq \frac{2 \cdot (c_2(e) - \beta_{\mathsf{L}}\beta_{\mathsf{H}} c_1(e))}{OT_{G_N} \cdot (\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}})} \quad .$$

Putting this together with (2), we know that

$$\frac{\vartheta - 1}{\frac{5}{3}\beta_{\mathsf{L}}^2 - 1} \leq 1 - \frac{2 \cdot (c_2(e) - \beta_{\mathsf{L}}\beta_{\mathsf{H}} c_1(e))}{OT_{G_N} \cdot (\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}})} \quad ,$$

which yields

$$\frac{c_2(e) - \beta_{\mathsf{L}}\beta_{\mathsf{H}} c_1(e)}{OT_{G_N}} \leq \frac{\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}}}{2} - \frac{(\vartheta - 1) \cdot (\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}})}{\frac{10}{3}\beta_{\mathsf{L}}^2 - 2} \quad .$$

By adding $(\beta_{\mathsf{L}}\beta_{\mathsf{H}} - 1)\frac{c_1(e)}{OT_{G_N}}$ to both sides, we are given:

$$\frac{c_2(e) - c_1(e)}{OT_{G_N}} \leq \frac{\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}}}{2} - \frac{(\vartheta - 1) \cdot (\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}})}{\frac{10}{3}\beta_{\mathsf{L}}^2 - 2} + (\beta_{\mathsf{L}}\beta_{\mathsf{H}} - 1) \cdot \underbrace{\frac{c_1(e)}{OT_{G_N}}}_{\leq 1}$$

and thus, substituting the value (1) for ϑ,

$$\begin{aligned}
\frac{c_2(e) - c_1(e)}{OT_{G_N}} &\leq \frac{3}{2}\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \frac{1}{2}\beta_{\mathsf{H}} - 1 - \frac{(\vartheta - 1) \cdot (\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}})}{\frac{10}{3}\beta_{\mathsf{L}}^2 - 2} \\
&= \frac{3}{2}\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \frac{1}{2}\beta_{\mathsf{H}} - 1 - \frac{(\beta_{\mathsf{L}}\beta_{\mathsf{H}} \cdot \frac{15\beta_{\mathsf{L}}^2 + 5\beta_{\mathsf{L}} - 6}{10\beta_{\mathsf{L}}^2 + 3\beta_{\mathsf{L}}\beta_{\mathsf{H}} + 3\beta_{\mathsf{H}} - 6} - 1)(\beta_{\mathsf{L}}\beta_{\mathsf{H}} + \beta_{\mathsf{H}})}{\frac{10}{3}\beta_{\mathsf{L}}^2 - 2} \\
\text{(tedious calculations)} \quad &= \cdots = \beta_{\mathsf{L}}\beta_{\mathsf{H}} \cdot \frac{15\beta_{\mathsf{L}}^2 + 5\beta_{\mathsf{L}} - 6}{10\beta_{\mathsf{L}}^2 + 3\beta_{\mathsf{L}}\beta_{\mathsf{H}} + 3\beta_{\mathsf{H}} - 6} - 1 = \vartheta - 1 \quad .
\end{aligned}$$

Since, by the same reasoning as that of Lemmas 2 and 4, reusing the input optimal solution $\overline{C}$ inflicts a deviation from the new optimum by at most $c_2(e) - c_1(e) \leq (\vartheta - 1) \cdot OT_{G_N}$, Algorithm 1 is a ϑ-approximation algorithm. □

Hence, whenever the β values of G_O and G_N coincide, we have Theorem 4.

Theorem 4. *There is a (polynomial-time)* $\beta^2 \cdot \dfrac{15\beta^2 + 5\beta - 6}{13\beta^2 + 3\beta - 6}$*-approximation algorithm for* LM-Δ_β-TSP.

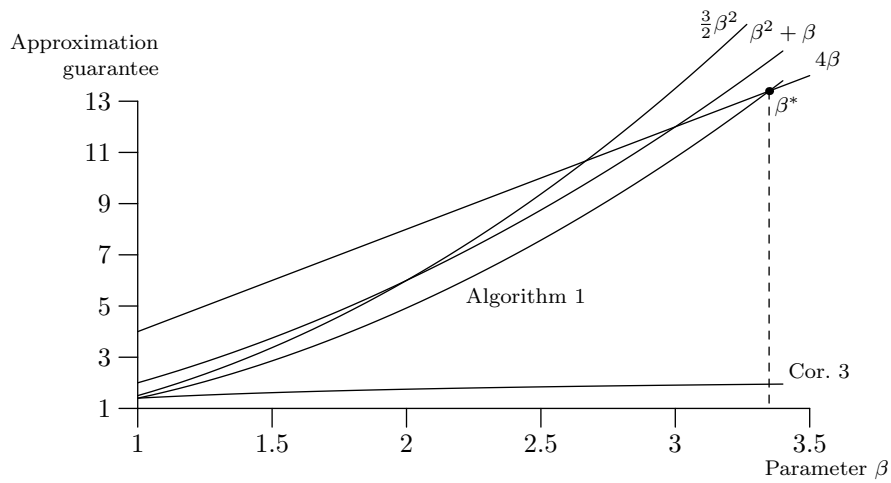

Fig. 2. Approximation guarantees of various algorithms, depending on β

Interestingly, Algorithm 1 achieves a better approximation guarantee not just than PMCA [5], but also than Bender's and Chekuri's 4β-approximation algorithm [3] for the most practically relevant values of β. The turning point is about at $\beta^* \approx 3.34899$. More to the point, Andreae's $(\beta^2+\beta)$-approximation [1], which performs better than 4β only when $\beta < 3$, always performs worse than Algorithm 1 in the interval $\beta \in (1, \beta^*)$. These observations are illustrated in Figure 2.

Another practical special case is that where $\beta_{\mathrm{L}} = 1$, i. e., where we start with a metric graph, but changing the cost of an edge will violate the Δ-inequality.

Corollary 3. LM-Δ_β-TSP*, restricted to those inputs where G_O is metric, admits a $\frac{7\beta}{2+3\beta}$-approximation.* □

3 Deadline TSP

In this section, we will analyze the approximability of local-modification variants of TSP with deadlines. To begin with, let us define this problem formally.

Definition 3. *Let $G = (V, E)$ be a complete graph weighted by $c\colon E \to \mathbb{N}^+$. We call (s, D, d) a* deadline set *for G if $s \in V, D \subseteq V \setminus \{s\}$ and $d\colon D \to \mathbb{N}^+$. A vertex $v \in D$ is called* deadline vertex. *A path $(v_0, v_1, \ldots, v_n)$* satisfies the deadlines *iff $s = v_0$ and, for all $v_i \in D$, we have $\sum_{j=1}^{i} c(\{v_{j-1}, v_j\}) \leq d(v_i)$.*

A cycle $(v_0, v_1, \ldots, v_n, v_0)$ satisfies the deadlines *iff it contains a path $(v_0, v_1, \ldots, v_n)$ satisfying the deadlines.*

Definition 4. *The problem Δ_β-*DLTSP *is defined as follows: For a given complete graph $G = (V, E)$ with edge weights $c\colon E \to \mathbb{N}^+$ satisfying the Δ_β-*

inequality, deadlines (s, D, d) *for* G*, and a Hamiltonian cycle satisfying the deadlines*[1]*, find a minimum-weight Hamiltonian cycle satisfying all deadlines.*

If $|D|$ *is a constant* k*, the resulting subproblem is* k-Δ_β-DLTSP*. We set* Δ-DLTSP $:= \Delta_1$-DLTSP *and* k-Δ-DLTSP $:= k$-Δ_1-DLTSP *for all* k*.*

In the case of TSP with deadlines, we will regard it as a local modification to change a single deadline although the LM operation from the previous section would let us obtain exactly the same results. The connection between these two LM operations will be presented in detail in the journal version of this paper.

Definition 5. *The optimization problem* LM-DLTSP *is defined as:*
Input: *A complete weighted graph* $G = (V, E, c)$*, deadlines* $O = (s, D, d_O)$ *for* G *with a minimal Hamiltonian cycle satisfying the deadlines* O*, new deadlines* $N = (s, D, d_N)$ *such that* d_O *and* d_N *differ in exactly one vertex, and a Hamiltonian cycle satisfying* N*.*
Problem: *Find a minimum-cost Hamiltonian cycle satisfying* N*.*

By LM-k-DLTSP*,* LM-Δ-DLTSP*,* LM-k-Δ-DLTSP*,* LM-Δ_β-DLTSP*,* LM-k-Δ_β-DLTSP*, we denote the canonical special cases of* LM-DLTSP*.*

For our proofs, we will need some reductions from the following problem, which can easily be shown to be NP-hard analogously to the proof of the NP-hardness of the *restricted Hamiltonian cycle problem,* as presented, e.g., in [16].

Definition 6. *For a given graph* $G = (V, E)$*,* s*,* $t \in V$ *and a given Hamiltonian path* P *from* s *to* t*, the problem* RHP *is to decide whether* G *contains a Hamiltonian path starting in* s*, but ending in some vertex* $v \neq t$*.*

3.1 Bounded Number of Deadline Vertices

We start with the case where only few deadline vertices occur. Note that k-Δ-DLTSP can be approximated within a ratio of 2.5 [6, 7]. Furthermore, a lower bound of $2 - \varepsilon$ on the approximability, for every $\varepsilon > 0$, can be proved [6, 7]. We will show that this lower bound also holds for LM-k-Δ-DLTSP.

Theorem 5. *Let* $\varepsilon > 0$*. There is no polynomial-time* $(2 - \varepsilon)$*-approximation algorithm for the subproblem of* LM-k-Δ-DLTSP *where one deadline is increased by* ξ *time units,* $\xi \geq 1$*, unless* $P = NP$*.*

Proof. By means of a reduction, we will show that such an approximation algorithm could be used to solve RHP. Let $\varepsilon > 0$.

Let (G', P) be an input instance for RHP where $G' = (V', E')$, $|V'| = n+1$, $s', t' \in V'$, and P is a Hamiltonian path from s' to t'. Pick a $\gamma > \frac{5n+3}{2\varepsilon}$ (which implies $\frac{4\gamma+n-1}{2\gamma+3n+1} > 2 - \varepsilon$).

[1] Requiring a feasible Hamiltonian cycle as part of the input ensures that the problem is in NPO. Otherwise, it would even be a hard problem to find a feasible solution. For details, see [6, 7].

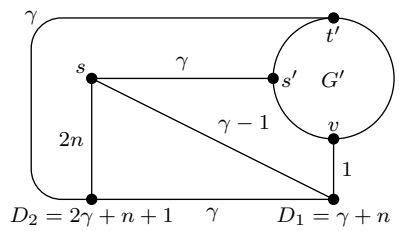

Fig. 3. Increasing a deadline. All vertices $v' \in V' \setminus \{s', t'\}$ are connected like v.

We construct a complete weighted graph $G = (V, E, c)$ as part of an input for LM-k-Δ-DLTSP as shown in Figure 3: We set $V := V' \dot\cup \{s, D_1, D_2\}$, and, for any edge e between two vertices $v_1, v_2 \in V'$, let $c(e) = 1$ if $e \in E'$ and $c(e) = 2$ otherwise. All edges depicted in Figure 3 have the indicated costs while non-depicted edges obtain maximal possible costs.

For these deadlines, one optimal solution $\overline{C}$ is the cycle $s, D_1, D_2, t', \ldots, s', s$, which uses the Hamiltonian path P from s' to t' in G'. It costs exactly $\gamma - 1 + \gamma + \gamma + n + \gamma = 4\gamma + n - 1$. All other feasible solutions visit some vertices in V' between s and D_1, but cost at least the amount of 1 more.

Now, we increase $d(D_1)$ by ξ. If G' contains a Hamiltonian path P from s' to some vertex $v \neq t'$, a new optimal solution is s, P, D_1, D_2, s, and it costs $\gamma + n + 1 + \gamma + 2n = 2\gamma + 3n + 1$. If G' does not contain such a path, it is not possible to visit all vertices in V' before reaching D_1 and D_2. As $c(\{t', D_1\}) \geq 2$, we cannot follow the given Hamiltonian path P because this would violate the deadline $d(D_2)$. Similar arguments hold for every other possibility. Hence, $\overline{C}$ remains an optimal solution in this case. Thus, we could use any approximation algorithm with an approximation guarantee better than

$$\frac{4\gamma + n - 1}{2\gamma + 3n + 1} > 2 - \varepsilon$$

to solve RHP. This is why approximating this subproblem of LM-k-Δ-DLTSP within $2 - \varepsilon$ is NP-hard for all $k \geq 2$. □

Theorem 6. *Let $\varepsilon > 0$. There is no polynomial-time $(2 - \varepsilon)$-approximation algorithm for the subproblem of* LM-k-Δ-DLTSP *where one deadline is decreased by ξ time units, $\xi \geq 1$, unless $P = NP$.*

Proof. Let $\varepsilon > 0$. Like in the preceding proof, we will use a reduction from RHP.

Let (G', P) be an input instance for RHP where $G' = (V', E')$, $|V'| = n+1$, $s', t' \in V'$, and P is a Hamiltonian path from s' to t'. Pick some γ such that $\frac{4\gamma}{2\gamma + 8n} > 2 - \varepsilon$.

We construct a complete weighted graph $G = (V, E, c)$ as part of an input for LM-k-Δ-DLTSP as shown in Figure 4: We set $V := V' \dot\cup \{s, D_1, D_2, D_3, D_4\}$, and, for any edge e between two vertices $v_1, v_2 \in V'$, let $c(e) = 1$ if $e \in E'$ and

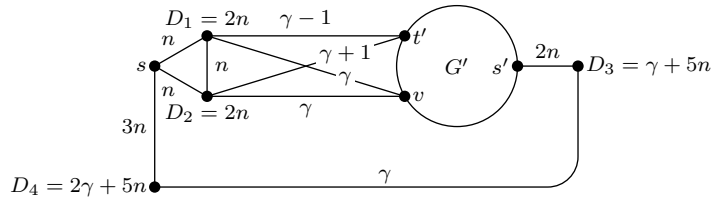

Fig. 4. Decreasing a deadline. All vertices $v' \in V' \setminus \{s', t'\}$ are connected like v.

$c(e) = 2$ otherwise. All edges depicted in Figure 4 have the indicated costs while non-depicted edges obtain maximal possible costs.

The initial deadlines are depicted in Figure 4. In this setting, an optimal solution is the cycle $s, D_2, D_1, t', \ldots, s', D_3, D_4, s$, which contains the Hamiltonian path from s' to t'. This path costs $2n + \gamma - 1$ on its way to G', spends n on the path from t' to s', and reaches s at time $2\gamma + 8n - 1$.

Now, we decrease the deadline $d(D_1)$ by ξ, whereby the old optimal solution becomes infeasible. Any new solution must visit D_1 before D_2. If we try to reuse the Hamiltonian path from t' to s', we have to spend $2n + \gamma + 1$ on the way to t'. Therefore, we cannot reach D_3 if we follow the complete Hamiltonian path. Furthermore, we cannot visit any vertex $v \in V'$ between visiting D_3 and D_4 because D_3 is not reached before $4n + \gamma$, going back to V' would cost another $2n$, and the cheapest path from V' to D_4 costs more than γ. This is why any solution using a Hamiltonian path between s' and t' violates one of the deadlines $d(D_3)$, $d(D_4)$.

If G' contains a Hamiltonian path P from s' to some $v \neq t'$, the new optimal solution contains this path in reverse on its way to D_3. The path s, D_1, D_2, P, D_3 visits all vertices in V' between v and s' and reaches D_3 at time $\gamma + 5n$. Therefore, this new optimal solution costs $2\gamma + 8n$.

If G' does not contain such a Hamiltonian path, the optimal solution cannot visit all vertices in V' before reaching D_3 or even D_4, and consequently, it is more expensive than 4γ. Thus, we could use an approximation algorithm with an approximation guarantee better than

$$\frac{4\gamma}{2\gamma + 8n} > 2 - \varepsilon$$

to solve RHP. Hence, approximating this subproblem of LM-k-Δ-DLTSP within $2 - \varepsilon$ is NP-hard. □

3.2 Unbounded Number of Deadline Vertices

When the number of deadline vertices is unbounded, we can show a linear lower bound on the approximability of LM-Δ-DLTSP. Our reduction from RHP involves two steps. A first construction will guarantee that an optimal path becomes shorter by a constant factor if a Hamiltonian path exists in the RHP

instance. A second construction inflates this advantage. Tours which start at time X, different from those that start between times $X + g$ and $X + \zeta g$, may spend some extra time to visit a group of vertices which, unless visited early, will cause belated tours to run k times zigzag across a huge distance γ.

The following lemma describes the construction in detail. See Figure 5 for an overview.

Lemma 8. *Let $X, g, k, \gamma, \zeta \in \mathbb{N}$ such that k is even, $\zeta \geq 1$ and $\gamma \geq g$. Let $G' = (V', E')$ be a graph with deadline set (s, D', d') such that any Hamiltonian path in G' respecting the deadlines ends in the same vertex t. Then, we can construct a complete graph $G \supset G'$ and deadlines (s, D, d) such that $D \supset D'$, $d_{|D'} = d'$ and any path that reaches t in time X can be extended to a Hamiltonian cycle which costs at most*

$$X + (k + 2\zeta - 4)g + 2\gamma \quad ,$$

while any path that reaches t after $X + g$, but before $X + \zeta g$ can only be extended to a Hamiltonian cycle which costs at least

$$X + \left(\frac{k-3}{2} + \zeta\right) g + k\gamma \quad .$$

Proof. We construct $G = (V, E)$ with $V = V' \cup \{E_1, \ldots E_k\}$ and edge costs as depicted in Figure 5, where $b := g(\zeta - \frac{1}{2})$. To all other edges, we assign maximal possible costs. Note that the edge $\{t, E_1\}$ costs exactly the same as the path $E_{k-1}, E_{k-3}, \ldots, E_1$.

We set the deadlines

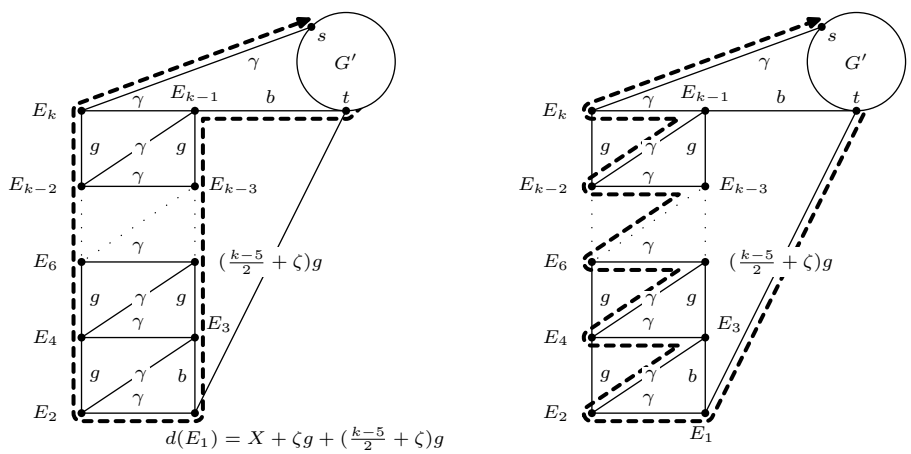

Fig. 5. The zigzag construction for the proof of Lemma 8. The left-hand side shows the optimal path if t is reached at time X. The right-hand side shows the optimal solution if t is reached after $X + g$. We set $b := g(\zeta - \frac{1}{2})$ and $d(D_{i+1}) := d(D_i) + \gamma$.

$$d(E_1) := X + \zeta g + \left(\frac{k-5}{2} + \zeta\right) g \quad \text{and}$$
$$d(E_{i+1}) := d(E_i) + \gamma \qquad \text{for all } i \in \{1, \ldots, k-1\} \quad .$$

If a path reaches t after $X+g$, it must proceed immediately to E_1. Note that it cannot use any other edge since it would have to use an edge of an additional cost of at least $b = g(\zeta - \frac{1}{2}) > g(\zeta - 1)$, then. Together with even the shortest path to E_1, this would violate this deadline. But then, it is forced to follow the sequence $E_2, E_3, \ldots, E_k$ to reach every deadline since even if we visited E_3 before E_2, we would incur an extra cost of b, and this would violate the deadline of E_2. Hence, the Hamiltonian cycle costs at least $X + g + (\frac{k-5}{2} + \zeta)g + k\gamma$.

A path that visits t before X can visit $E_{k-1}, E_{k-3}, \ldots, E_3$ before E_1 because this path to E_1 costs at most

$$X + b + (\frac{k}{2} - 2)g + b = X + \zeta g + \left(\frac{k-6}{2} + \zeta\right) g \leq d(E_1) \quad .$$

Closing the cycle to s, we obtain a cost of at most

$$X + \zeta g + \left(\frac{k-6}{2} + \zeta\right) g + \left(\frac{k}{2} - 1\right) g + 2\gamma = X + (k + 2\zeta - 4)g + 2\gamma \quad .$$

□

We will now employ Lemma 8 to prove the desired lower bound.

Theorem 7. *Let $\varepsilon > 0$. There is no polynomial-time $\left(\left(\frac{1}{2} - \varepsilon\right) \cdot |V|\right)$-approximation algorithm for the subproblem of* LM-Δ-DLTSP *where one deadline is increased by $\xi \geq 1$, unless $P = NP$.*

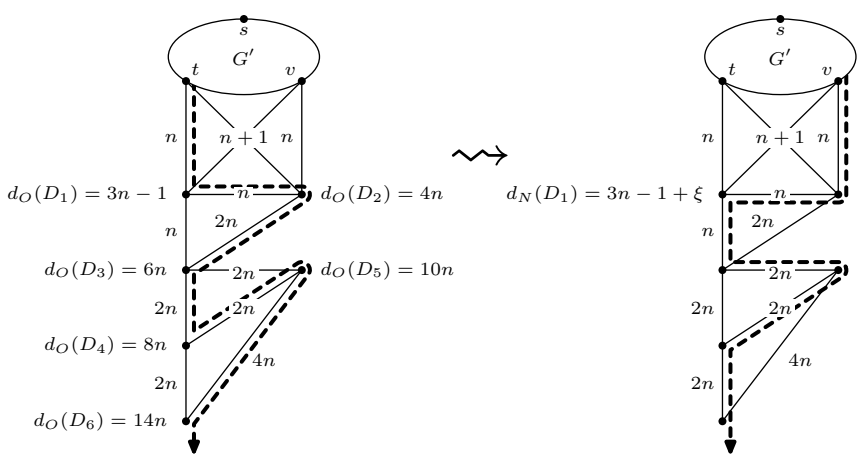

Fig. 6. Increasing a deadline: If the deadline for the vertex D_1 is increased, using a Hamiltonian path from s to v leads to a new optimal solution.

Proof. By means of a reduction, we will show that such an approximation algorithm could be used to solve RHP.

Let (G', P) be an input instance for RHP, where $G' = (V', E')$, $|V'| = n+1$, $s, t \in V'$, and P is a Hamiltonian path from s to t. We construct a complete weighted graph $G = (V, E, c)$ as part of an input for the LM-Δ-DLTSP as shown in Figure 6: We set $V = V' \cup \{D_1, \ldots, D_6\}$ and, for any edge e between two vertices $v_1, v_2 \in V'$, $c(e) = 1$, if $e \in E'$, and $c(e) = 2$ otherwise. To the other edges, assign costs as depicted in Figure 6, and maximal possible costs to the non-depicted edges, and set the deadlines $d_O(D_i)$ according to Figure 6.

Pick some suitable $0 < \delta < 1$ and $0 < \alpha < 1$ such that $\frac{\alpha}{2+\delta} \geq \frac{1}{2} - \varepsilon$. We use the zigzag construction defined in Lemma 8 with parameters $X = 10n$, $g = 2n$, $\zeta = 2$, $k \geq (n+7)\frac{\alpha}{1-\alpha}$, and $\gamma \geq \frac{2kn+10n}{\delta}$ to obtain the graph G_O of our input instance. This guarantees $2kn + 10n \leq \delta\gamma$ and $k \geq \alpha(k+n+6)$.

The given optimal Hamiltonian tour $\overline{C}$ in G_O starts in s, uses the given Hamiltonian path in G' to t, and afterwards follows the sequence D_1, D_2, D_3, D_4, D_5, D_6. Hence, it reaches D_6 in time $13n$. Following the zigzag construction, this leads to a cost of at least $10n + \left(\frac{k-3}{2} + \zeta\right) g + k\gamma$. In G_N, we change the deadline for D_1 to $d_N(D_1) = 3n - 1 + \xi$ for some $\xi \geq 1$. $\overline{C}$ remains a feasible solution. If G' contains a Hamiltonian path from s to some vertex $v \neq t$, an optimal solution uses this path and follows the sequence $D_2, D_1, D_3, D_5, D_4, D_6$. This solution reaches D_6 in time $10n$. By Lemma 8, this cycle costs $10n + (k + 2\zeta - 4)g + 2\gamma$.

If G' does not contain any Hamiltonian path to such a vertex v, $\overline{C}$ remains the optimal solution in the case where $\xi = 1$. If $\xi \geq 2$, an optimal solution follows P to t and afterwards uses the sequence $D_2, D_1, D_3, D_4, D_5, D_6$. This solution reaches D_6 in time $12n + 1 > X + g$. By Lemma 8, we obtain a cost of $10n + (\frac{k-3}{2} + \zeta)g + k\gamma$. This leads to a ratio of at least

$$\frac{10n + (\frac{k-3}{2} - 2)2n + k\gamma}{10n + (k+4-4)2n + 2\gamma} > \frac{k\gamma}{2kn + 10n + 2\gamma}$$
$$> \frac{k\gamma}{(2+\delta)\gamma} = \frac{k}{(2+\delta)} \geq \frac{\alpha}{2+\delta}(k+n+7) \geq (\frac{1}{2} - \varepsilon)|V| \quad .$$

Hence, a polynomial-time $(\frac{1}{2} - \varepsilon)|V|$-approximation algorithm could be used to solve RHP. □

Theorem 8. *Let $\varepsilon > 0$. There is no polynomial-time $\left((\frac{1}{2} - \varepsilon)\,|V|\right)$-approximation algorithm for the subproblem of* LM-Δ-DLTSP *where one deadline is decreased by $\xi \geq 1$ unless $P = NP$.*

Proof idea. The proof can be done in a way similar to the proof of Theorem 7. The relevant construction is illustrated in Figure 7. Details will be given in a journal version of this paper. □

Corollary 4. *Let $\varepsilon > 0$. There is no polynomial-time $\left((\frac{1}{2} - \varepsilon)|V|\right)$-approximation algorithm for* LM-Δ-DLTSP *unless $P = NP$.* □

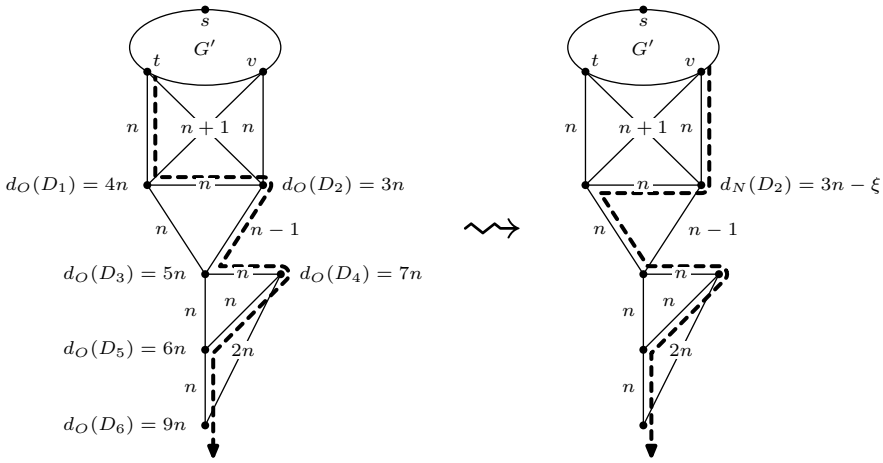

Fig. 7. Decreasing a deadline: If the deadline for the vertex D_2 is decreased, the old optimal solution (depicted on the left-hand side) becomes infeasible. If G' contains a Hamiltonian path from s to v, we obtain the depicted new optimal solution. If no such Hamiltonian path exists, the new optimal solution must follow $D_2, D_1, D_3, D_5, D_4, D_6$.

4 Conclusion

In this work, we have introduced and successfully applied the concept of reusing optimal solutions when input instances are locally modified. In the case of metric TSP, we are able to improve on the previously-known upper bound of 1.5, as achieved by Christofides' algorithm (applied to the new instance, ignoring the given optimal solution), with non-trivial extensions to the near-metric case. As for TSP with deadlines, which is remarkably hard [6], we have been able to reestablish almost all known lower bounds on the approximability of its variants in the setting of local modifications.

As an open problem, we state the question how hard it is to approximate LM-k-Δ_β-DLTSP. Another open problem is whether the NP-hard LM-Δ-TSP is also APX-hard.

References

1. T. Andreae: On the traveling salesman problem restricted to inputs satisfying a relaxed triangle inequality. *Networks* 38, 2001, pp. 59–67.
2. T. Andreae, H.-J. Bandelt: Performance guarantees for approximation algorithms depending on parameterized triangle inequalities. *SIAM Journal on Discrete Mathematics* 8, 1995, pp. 1–16.
3. M. Bender, C. Chekuri: Performance guarantees for TSP with a parameterized triangle inequality. *Information Processing Letters* 73, 2000, pp. 17–21.

4. H.-J. Böckenhauer, J. Hromkovič, R. Klasing, S. Seibert, W. Unger: Approximation algorithms for TSP with sharpened triangle inequality. *Information Processing Letters* 75, 2000, pp. 133–138.
5. H.-J. Böckenhauer, J. Hromkovič, R. Klasing, S. Seibert, W. Unger: Towards the notion of stability of approximation for hard optimization tasks and the traveling salesman problem. *Theoretical Computer Science* 285, 2002, pp. 3–24.
6. H.-J. Böckenhauer, J. Hromkovič, J. Kneis, J. Kupke: On the parameterized approximability of TSP with deadlines. *Theory of Computing Systems*, to appear.
7. H.-J. Böckenhauer, J. Hromkovič, J. Kneis, J. Kupke: On the approximation hardness of some generalizations of TSP. *Proc. SWAT 2006*, to appear.
8. H.-J. Böckenhauer, S. Seibert: Improved lower bounds on the approximability of the traveling salesman problem. *RAIRO Theoretical Informatics and Applications* 34, 2000, pp. 213–255.
9. N. Christofides: Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, 1976.
10. J.-F. Cordeau, G. Desaulniers, J. Desrosiers, M. M. Solomon, F. Soumis: VRP with time windows. In: P. Toth, D. Vigo (eds.): *The Vehicle Routing Problem*, SIAM 2001, pp. 157–193.
11. L. Forlizzi, J. Hromkovič, G. Proietti, S. Seibert: On the stability of approximation for Hamiltonian path problems. *Algorithmic Operations Research* 1(1), 2006, pp. 31–45.
12. H. Greenberg: An annotated bibliography for post-solution analysis in mixed integer and combinatorial optimization. In: D. L. Woodruff (ed.): *Advances in Computational and Stochastic Optimization, Logic Programming, and Heuristic Search*, Kluwer Academic Publishers, 1998, pp. 97–148.
13. N. Guttmann-Beck, R. Hassin, S. Khuller, B. Raghavachari: Approximation algorithms with bounded performance guarantees for the clustered traveling salesman problem. *Algorithmica* 28, 2000, pp. 422–437.
14. J. A. Hoogeveen: Analysis of Christofides' heuristic: Some paths are more difficult than cycles. *Operations Research Letters* 10, 1978, pp. 178–193.
15. J. Hromkovič: Stability of approximation algorithms for hard optimization problems. *Proc. SOFSEM'99*, Springer LNCS 1725, 1999, pp. 29–47.
16. J. Hromkovič: *Algorithmics for Hard Problems. Introduction to Combinatorial Optimization, Randomization, Approximation, and Heuristics*. Springer 2003.
17. M. Libura: Sensitivity analysis for minimum Hamiltonian path and traveling salesman problems. *Discrete Applied Mathematics* 30, 1991, pp. 197–211.
18. M. Libura, E. S. van der Poort, G. Sierksma, J. A. A. van der Veen: Stability aspects of the traveling salesman problem based on k-best solutions. *Discrete Applied Mathematics* 87, 1998, pp. 159–185.
19. Ch. Papadimitriou, K. Steiglitz: Some examples of difficult traveling salesman problems. *Operations Research* 26, 1978, pp. 434–443.
20. Y. N. Sotskov, V. K. Leontev, E. N. Gordeev: Some concepts of stability analysis in combinatorial optimization. *Discrete Appl. Math.* 58, 1995, pp. 169–190.
21. S. Van Hoesel, A. Wagelmans: On the complexity of postoptimality analysis of 0/1 programs. *Discrete Applied Mathematics* 91, 1999, pp. 251–263.