Stability of Reapproximation Algorithms for the β -Metric Traveling Salesman (Path) Problem

Annalisa D'Andrea¹, Luca Forlizzi¹, and Guido Proietti^{1,2(✉)}

¹ Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica,
Università degli Studi dell'Aquila, Via Vetoio, 67100 L'Aquila, Italy
{annalisa.dandrea, luca.forlizzi, guido.proietti}@univaq.it

² Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti",
Consiglio Nazionale delle Ricerche, Via dei Taurini 19, 00185 Rome, Italy

Abstract. Inspired by the concept of *stability of approximation*, we consider the following (re)optimization problem: Given a *minimum-cost Hamiltonian cycle* of a complete non-negatively real weighted graph $G = (V, E, c)$ obeying the *strengthened* triangle inequality (i.e., for some *strength factor* $1/2 \leq \beta < 1$, we have that $\forall u, v, z \in V, c(u, z) \leq \beta(c(u, v) + c(v, z))$), and given a vertex v whose removal from G (resp., addition to G), along with all its incident edges, produces a new weighted graph still obeying the strengthened triangle inequality, find a minimum-cost Hamiltonian cycle of the modified graph. This problem is known to be NP-hard, but we show that it admits a PTAS, which just consists of either returning the old optimal cycle (after having by-passed the removed node), or instead computing (for finitely many inputs) a new optimal solution from scratch – depending on the required accuracy in the approximation. Then, we turn our attention to the case in which a *minimum-cost Hamiltonian path* is given instead, and the underlying graph obeys the *relaxed* triangle inequality. Here, if one edge weight is increased, and $\beta_o, \beta_m \geq 1$ denotes the relaxation factor of the original and the modified graph, respectively, then we show how to obtain an approximation of $1 + 2 \min\{\beta_o, \beta_m\}$, which improves over existing solutions as soon as $\min\{\beta_o, \beta_m\} \geq \frac{3+2\sqrt{6}}{5} \approx 1.58$.

1 Introduction

The hardness of optimization problems is commonly defined according to the computational complexity of worst-case input instances. However, in many cases one can find significant sets of input instances that behave substantially better in terms of tractability. In order to promote a broader notion of tractability, less tied to the tyranny of worst-case complexity, Hromkovič introduced the concept of *stability of approximation* [26, 27]. This provides a formal tool to partition the set of input instances of optimization problems into possibly infinitely many subclasses according to the hardness of their approximability, and allows us to

have an efficient algorithm for deciding the membership of any problem instance to one of the subclasses considered. The idea behind this concept is to find a distance measure between input instances that captures their difference from a computational complexity perspective. An algorithm that offers a good approximation for some set $\mathcal{I}$ of input instances is then called *stable* with respect to this measure if its approximation ratio grows with the distance of an input instance from $\mathcal{I}$ but not with the size of the input instances.

As a paradigmatic case study, the literature offers many examples of optimization problems which are computationally hard when defined on general weighted graphs, but become easier to solve/approximate when the input graph happens to be *metric* (i.e., its cost function obeys the *triangle inequality*). Following the concept of stability, it is then natural to assume, as distance measures from “good” instances, a parametrization of the metricity of the input graph. More precisely, we say that a non-negatively real weighted graph $G = (V, E, c)$ satisfies the β -triangle inequality, where $\beta \geq 1/2$, if $c(u, v) \leq \beta(c(u, z) + c(z, v))$, $\forall u, v, z \in V$. If $\beta < 1$, we speak about the *strengthened* triangle inequality, while if $\beta > 1$, we speak about the *relaxed* triangle inequality. Then the parameter β can be used to partition the set of all input instances of a hard optimization problem (from $\beta = 1/2$, i.e., unweighted graphs, up to $\beta = \infty$, i.e., general weighted graphs), and to classify them according to their respective computational properties. This idea proved to be very fruitful, giving rise to numerous results for different hard optimization problems, such as k -connectivity [11, 12], Steiner tree [8, 9, 14, 19], Traveling Salesman Problem (TSP for short) [1, 2, 5, 15–17, 22], and its variants [7, 10, 13, 18, 20, 21, 24, 25].

In this paper we further explore the usefulness of the notion of stability of approximation in order to attack the *reoptimization version* of the TSP (and a Hamiltonian-path variant of it) when the input graph is β -metric.

1.1 The Reoptimization Version of the TSP

The TSP is that of finding a minimum-cost cycle spanning all the nodes of a complete real weighted graph, say $G = (V, E, c)$. It is well-known to be not only NP-hard, but also not approximable in polynomial time within a $2^{p(|V|)}$ -factor, where p is any polynomial, unless $P = NP$. On the other hand, in the metric case, although it remains APX-hard, it can be approximated within a factor of $3/2$ by means of the classic Christofides algorithm [23]. For an excellent survey on the TSP, we refer the reader to [32].

To get closer to the subject of this paper, imagine now a scenario in which, given an instance of the general TSP, along with an optimal solution, i.e., a minimum-cost Hamiltonian cycle of G , this instance is subject to a small *modification*, consisting in the insertion or the deletion of a new node into or from the graph. Then, two natural arising questions are the following: Is it still NP-hard to find an optimal solution for this problem? And if so, what about its approximability? This was exactly the pioneering formalization of a *reoptimization* version of the TSP given in [3]. More precisely, the authors focused on the metric TSP, and showed that both problems (i.e., insertion or deletion of a new node) remain

NP-hard. Moreover, they showed that for the former, the adaptation of the classic *cheapest insertion* heuristic guarantees a *linear-time* $3/2$ -approximation ratio, as opposed to the $O(|V|^3)$ time required by the Christofides algorithm, while for the latter, they showed that simply deleting the edges of the solution incident to the node that we have to eliminate, and then linking the other endpoints of such edges, we obtain a $3/2$ -approximation ratio.

Inspired by this work, in [13] the authors addressed another natural reoptimization variant of the metric TSP, namely that in which the instance undergoes the alteration of the weight of a single edge. Once again, the authors proved the NP-hardness of the problem, and therefore provided an extensive comparative analysis with the canonical metric TSP as far as the approximability of the problem was concerned. To this end, the authors developed their study by classifying the approximability of TSP depending on the metricity of both the input and the modified instance. More formally, they proved that: (i) if the input and the modified instance obey the *strengthened* triangle inequality, the problem admits a PTAS (this compares favorably with the APX-hardness of the counterpart); (ii) if the input and the modified instance are both metric, the problem can be approximated within a factor of $7/5$, which compares favorably with the approximation ratio guaranteed by the Christofides algorithm; this result was further improved in [6], where the authors proved the problem can be approximated within a factor of $4/3$; and finally, (iii) if the input and the modified instance obey the *relaxed* triangle inequality with $1 < \beta < 3.34899$, the problem admits a $\beta^2 \frac{15\beta^2+5\beta-6}{13\beta^2+3\beta-6}$ -approximation algorithm, which is better than its counterparts given in [5, 17]. In other words, for the considered range of values of β , all these reapproximation algorithms are stable.

Further reoptimization versions of the metric TSP were developed for both the strengthened and the relaxed case [4, 13, 21].

1.2 Our Results

In this paper we focus on two reoptimization variants of the TSP, and we show that they admit stable reapproximation algorithms.

First, we concentrate on a reoptimization version of the *strengthened metric* TSP ($\Delta_{\beta < 1}$ -TSP for short). We apply the same kind of perturbations introduced in [3], namely insertion and deletion of a node, trying to exploit both the optimal solution and the strengthened metric properties of the input instances, and we show that also in this case the problem on the perturbed instances admits a PTAS. Notice that, for the case of node deletion, our approach has a natural application counterpart in the so-called *transient failure* setting, namely that in which the malfunctioning of a network component (in our case, a node) is supposed to be promptly repaired. In such a scenario, it makes sense to exploit the knowledge that can be gathered from the optimal solution of the old instance (which was presumably found with much effort and time), in order to compute quickly an approximation to the new instance, and we will show that this is actually possible.

Then, we turn our attention to a more general version of the TSP, called the *Traveling Salesman Path Problem* (TSPP for short), in which we are also given two vertices $s, t \in V$ and the goal is to find a path from s to t visiting each vertex in V exactly once. More precisely, we address the *relaxed metric TSPP* ($\Delta_{\beta>1}$ -TSPP for short), whose input instances obey the relaxed triangle inequality, and whose goal is the same as for the TSPP, i.e., finding a minimum-cost Hamiltonian path between the given vertices s and t . The best current approximation algorithm for this problem is the so-called *Path Matching Christofides Algorithm-TSPP* [25], with an approximation ratio of $\frac{5}{3}\beta^2$. Also in this case, we focus on the reoptimization version of the relaxed metric TSPP. More precisely, if one edge weight is (transiently) increased, and we assume that the original and the modified graph obey the triangle inequality within a relaxation factor of $\beta_o, \beta_m \geq 1$, respectively, then we show how to obtain an approximation of $1 + 2\beta_l$, where $\beta_l = \min\{\beta_o, \beta_m\}$, which becomes the best approximation ratio for the $\Delta_{\beta>1}$ -TSPP when $\beta \geq \frac{3+2\sqrt{6}}{5} \approx 1.58$.

2 Reoptimizing the Strengthened Metric TSP on Single Node Modifications

In this section we study the reoptimization of TSP subject to a (transient) single node modification. For the remaining part of this section, for an input graph G , let OPT_G denote the cost of a minimum-cost Hamiltonian cycle of G . Formally, we define two problems, to cover both the insertion and the deletion of a single node of G .

Definition 1. *The node-inserted TSP problem (TSP^+) is defined as follows: Given*

- *two complete weighted graphs $G_o = (V_o, E_o, c_o)$ and $G_m = (V_m, E_m, c_m)$ such that $V_o = \{v_1, \dots, v_n\}$, $V_m = V_o \cup \{v_{n+1}\}$, and G_o coincides with the subgraph of G_m induced by V_o ;*
- *a Hamiltonian cycle $\bar{C}$ of G_o such that $\sum_{e \in \bar{C}} c(e) = \text{OPT}_{G_o}$;*

find a Hamiltonian cycle C of G_m such that $\sum_{e \in C} c(e) = \text{OPT}_{G_m}$.

Definition 2. *The node-deleted TSP problem (TSP^-) is defined as follows: Given*

- *two complete weighted graphs $G_o = (V_o, E_o, c_o)$ and $G_m = (V_m, E_m, c_m)$ such that $V_o = \{v_1, \dots, v_{n+1}\}$, $V_m = V_o \setminus \{v_i\}$, for any $i = 1, \dots, n+1$, and G_m coincides with the subgraph of G_o induced by V_m ;*
- *a Hamiltonian cycle $\bar{C}$ of G_o such that $\sum_{e \in \bar{C}} c(e) = \text{OPT}_{G_o}$;*

find a Hamiltonian cycle C of G_m such that $\sum_{e \in C} c(e) = \text{OPT}_{G_m}$.

These problems are studied in [3, 4], where it is shown that:

1. the problems are NP-hard;
2. in the case of an arbitrary distance between the vertices, the problems are not $2^{p(n)}$ -approximable, if $P \neq NP$, for any polynomial p ;
3. in the case where the distances between the vertices obey the triangle inequality, TSP^+ is approximable within a factor of $4/3$ in $O(n^3)$ time [4], and within a factor of $3/2$ in $O(n)$ time [3]; while TSP^- is approximable, in constant time, within a factor of $3/2$ [3].

In this paper we further improve these results for input instances where both G_o and G_m obey the strengthened triangle inequality up to a strength factor of β_o and β_m , respectively, with $\beta_o, \beta_m < 1$. We denote the restriction of TSP^+ (resp., TSP^-) to instances of such kind as $\Delta_{\beta < 1}$ - TSP^+ (resp., $\Delta_{\beta < 1}$ - TSP^-).

In contrast to the edge-perturbed TSP [24], where we know that changing the cost of even a single edge can either increase or decrease the minimum β -value such that the graph satisfies the β -triangle inequality, it is easy to see that by adding (resp., removing) a vertex to (resp., from) an instance, the corresponding β -value does not decrease (resp., increase), as shown by the following lemma:

Lemma 1. *Let G_{n+1} be a complete weighted graph of $n+1$ nodes that satisfies the $\bar{\beta}$ -triangle inequality, for some $1/2 \leq \bar{\beta} < 1$, and let G_n be the subgraph of G_{n+1} induced by $V(G_n) = V(G_{n+1}) \setminus \{v\}$, for some $v \in V(G_{n+1})$. Then G_n satisfies the $\bar{\beta}$ -triangle inequality.*

Proof. For any $e_1, e_2, e_3 \in E(G_n)$ that form a triangle, the edge costs in G_n are the same as those in G_{n+1} , hence they obey the $\bar{\beta}$ -triangle inequality. $\square$

A natural way to reoptimize a TSP^+ instance is to add the vertex v_{n+1} to the optimal Hamiltonian cycle $\bar{C}$ of G_o , by means of a *vertex-insertion* operation that removes from $\bar{C}$ an edge (x, y) , selected according to certain criteria, and replaces it with the edges (x, v_{n+1}) and (v_{n+1}, y) . Concerning TSP^- , w.l.o.g. let us assume in the following that the removed vertex is v_{n+1} . Then, one can reoptimize a TSP^- instance by using a *vertex-deletion* operation that removes from the optimal Hamiltonian cycle $\bar{C}$ of G_o the two edges incident to v_{n+1} , say (u, v_{n+1}) and (v_{n+1}, w) , and then adds (u, w) to $\bar{C}$.

Such kind of operations are the foundations of the reoptimization algorithms for the metric case presented in [3, 4], as well as of well-known approximation algorithms for the classic TSP [28]. Notice that for the TSP^+ one can specify different rules to select the edge to be removed from $\bar{C}$. The algorithms in [3, 4] use, in particular, the *cheapest insertion* rule, that selects, among the edges in $\bar{C}$, any one that minimizes the cost increase of the Hamiltonian cycle C built by the operation. For our results, it is sufficient to employ the slightly simpler *cheapest-removal insertion* rule that just removes from $\bar{C}$ an edge of minimum cost.

Formally, our algorithms are based on the following operations:

Definition 3 (Insertion Operation). *Given a cycle C and a node v not in C , select an edge (x^*, y^*) of minimum cost in C , and build a new cycle C' from C , by removing (x^*, y^*) and then adding edges (x^*, v) and (v, y^*) .*

Definition 4 (Deletion Operation). *Given a cycle C and a node v such that (x, v) and (v, y) are the edges incident to v in C , build a new cycle C' from C by removing edges (x, v) and (v, y) , and then adding edge (x, y) .*

The strengthened triangle inequality induces a strong relationship between edge costs, that can be used to bound the increase (resp., decrease) of the cost of the cycle modified by an insertion (resp., deletion) operation. We show this in Lemma 3, that in turn uses the following technical lemma proven in [16]:

Lemma 2 (Böckenhauer et al. [16]). *Given a complete weighted graph $G = (V, E, c)$ that satisfies the β -triangle inequality, for some $1/2 \leq \beta < 1$:*

- *For any two edges $e_1, e_2 \in E$ with a common endpoint:*

$$c(e_1) \leq \frac{\beta}{1-\beta} c(e_2). \quad (1)$$

- *Let $c_{\max}$ and $c_{\min}$ denote the maximum and the minimum, respectively, among all edge costs. Then,*

$$\frac{c_{\max}}{c_{\min}} \leq \frac{2\beta^2}{1-\beta}. \quad (2)$$

Lemma 3. *Let G_{n+1} be a complete weighted graph that satisfies the β -triangle inequality, for some $1/2 \leq \beta < 1$, such that $|V(G_{n+1})| = n + 1$, and let G_n be a subgraph of G_{n+1} such that $V(G_{n+1}) = V(G_n) \cup \{v_{n+1}\}$. Let C_{n+1} and C_n be a minimum-cost Hamiltonian cycle for G_{n+1} and for G_n , respectively, of cost $\text{OPT}_{G_{n+1}}$ and OPT_{G_n} , respectively. Then we have*

$$\text{OPT}_{G_n} < \text{OPT}_{G_{n+1}} \leq \left(1 + \left(\frac{2\beta}{1-\beta} - 1\right) \frac{1}{n}\right) \text{OPT}_{G_n}. \quad (3)$$

Proof. Let s and t be the vertices of G_{n+1} that are adjacent to v_{n+1} in C_{n+1} . From C_{n+1} we can construct a Hamiltonian cycle C_a of G_n by removing v_{n+1} with a deletion operation. Since C_n is a minimum-cost Hamiltonian cycle for G_n , we have $\text{OPT}_{G_n} \leq c(C_a)$. On the other hand, the strengthened triangle inequality rules out the existence of edges of weight 0 (but for the degenerate case in which all the edges have weight 0), which implies $c(C_a) < \text{OPT}_{G_{n+1}}$, and so we have that

$$\text{OPT}_{G_n} < \text{OPT}_{G_{n+1}}.$$

From C_n we can build a Hamiltonian cycle C_b of G_{n+1} by inserting v_{n+1} with an insertion operation; let (x^*, y^*) be the edge removed by the operation. Since C_{n+1} is a minimum-cost Hamiltonian cycle for G_{n+1} , we have $\text{OPT}_{G_{n+1}} \leq c(C_b)$.

On the other hand, $c(C_b) = \text{OPT}_{G_n} + c(x^*, v_{n+1}) + c(v_{n+1}, y^*) - c(x^*, y^*)$. Since (x^*, v_{n+1}) and (v_{n+1}, y^*) have an endpoint in common with (x^*, y^*) , from (1) we have that the cost of each of them is bounded from above by $\frac{\beta}{1-\beta} c(x^*, y^*)$. Since (x^*, y^*) has the minimum cost among the edges in C_n , we have $c(x^*, y^*) \leq \frac{1}{n} \text{OPT}_{G_n}$. Therefore it follows that

$$\text{OPT}_{G_{n+1}} \leq c(C_b) \leq \left(1 + \left(\frac{2\beta}{1-\beta} - 1\right) \frac{1}{n}\right) \text{OPT}_{G_n}. \quad \square$$

The previous lemma immediately suggests that it is possible to achieve a very good approximation for the $\Delta_{\beta < 1}$ -TSP⁺ and for the $\Delta_{\beta < 1}$ -TSP⁻, and allows us to provide our main result. More precisely, we show that returning the cycle obtained by using an insertion or deletion operation, respectively, guarantees an approximation which can be made arbitrarily close to 1 for sufficiently large input graphs.

Theorem 1. *Let $(G_o, G_m, \overline{C})$ be an input instance of $\Delta_{\beta < 1}$ -TSP⁺ (resp., $\Delta_{\beta < 1}$ -TSP⁻) such that G_o and G_m obey the triangle inequality up to a strength factor of β_o and β_m , respectively, with $\beta_o, \beta_m < 1$. Let G_o and G_m agree except for a single node v_i along with all its incident edges. Let $\beta_{\max} := \max\{\beta_o, \beta_m\} < 1$ and $n = \min\{|V(G_o)|, |V(G_m)|\}$. Then, it is a $\left(1 + \left(\frac{2\beta_{\max}}{1-\beta_{\max}} - 1\right) \frac{1}{n}\right)$ -approximation to simply output the cycle obtained by using the insertion (resp., deletion) operation.*

Proof. Let us start with the $\Delta_{\beta < 1}$ -TSP⁺: let $n = |V(G_o)|$ and let v_{n+1} be the vertex of G_m that does not belong to G_o . From Lemma 1, we know that G_o satisfies the β_m -triangle inequality, hence $\beta_{\max} = \beta_m$. Using the insertion operation, we obtain, from $\overline{C}$, a Hamiltonian cycle C of G_m that, as shown in Lemma 3, has a cost bounded from above by $\left(1 + \left(\frac{2\beta_{\max}}{1-\beta_{\max}} - 1\right) \frac{1}{n}\right) \text{OPT}_{G_o}$. From Lemma 3, we also have that $\text{OPT}_{G_o} \leq \text{OPT}_{G_m}$, therefore

$$c(C) \leq \left(1 + \left(\frac{2\beta_{\max}}{1-\beta_{\max}} - 1\right) \frac{1}{n}\right) \text{OPT}_{G_m}.$$

For the $\Delta_{\beta < 1}$ -TSP⁻, let $n = |V(G_m)|$ and let v be the vertex of G_o that does not belong to G_m . From Lemma 1, again, we know that $\beta_{\max} = \beta_o$. Now, from $\overline{C}$, using the deletion operation, we obtain a Hamiltonian cycle C of G_m that, by the β_o -triangle inequality, has a cost bounded from above by OPT_{G_o} . Since, by Lemma 3, OPT_{G_o} is bounded from above by $\left(1 + \left(\frac{2\beta_{\max}}{1-\beta_{\max}} - 1\right) \frac{1}{n}\right) \text{OPT}_{G_m}$, we have

$$c(C) \leq \left(1 + \left(\frac{2\beta_{\max}}{1-\beta_{\max}} - 1\right) \frac{1}{n}\right) \text{OPT}_{G_m}. \quad \square$$

Notice that for the $\Delta_{\beta < 1}$ -TSP⁺, the above result holds true also if one uses, instead of the insertion operation, a similar one that selects the edge to be removed from the original cycle according to the *cheapest insertion* rule, since

Algorithm 1. $\Delta_{\beta < 1}$ -TSP^{+/-}

```

1 For the given input instance compute  $\beta_{\max}$  and  $n$ ;
2 By Theorem 1, compute the approximation  $\alpha$  which can be guaranteed by
  simply returning the cycle  $C$  obtained by using the insertion (resp., deletion)
  operation;
3 if  $\alpha \leq \varepsilon$  then
4   output  $C$ .
5 else
6   Perform exhaustive search for an optimal solution.
7 end

```

the cost of the cycle constructed by this operation is not greater than that of the cycle constructed by our insertion operation.

The algorithms for the $\Delta_{\beta < 1}$ -TSP⁺ and for the $\Delta_{\beta < 1}$ -TSP⁻ obtained from Theorem 1 can be easily adapted to provide a $(1 + \varepsilon)$ -approximation guarantee, for a parameter $\varepsilon > 0$ (see Algorithm 1).

Theorem 2. *Algorithm 1 is a PTAS.*

Proof. Given an instance of $\Delta_{\beta < 1}$ -TSP⁺ (resp., $\Delta_{\beta < 1}$ -TSP⁻) with $n = |V(G_o)|$ (resp., $n = |V(G_m)|$), one can compute

$$c = 1 + \left(\frac{2\beta_{\max}}{1 - \beta_{\max}} - 1 \right)$$

in $O(n^3)$ time. Then, for any $\varepsilon > 0$, we proceed as follows:

1. If $n \geq \frac{c}{\varepsilon}$, we know from Theorem 1 that returning the cycle C obtained by using the insertion (resp., deletion) operation guarantees an approximation ratio of $1 + \frac{c}{n} \leq 1 + \varepsilon$, and thus this case costs $O(1)$ time;
2. otherwise, we perform an exhaustive search for a new optimal solution, and this costs $O(n!) = O(n^{\frac{c}{\varepsilon} + 1})$ time.

Since Case 2 happens for finitely many inputs only, the claim follows. $\square$

3 Reoptimizing the Relaxed Metric TSP on Single Edge Weight Increases

In this section we study the reoptimization version of $\Delta_{\beta > 1}$ -TSP subject to a (transient) single edge weight increase. The best current approximation algorithm for the $\Delta_{\beta > 1}$ -TSP is the PMCA-TSP [25] algorithm, which has an approximation ratio of $\frac{5}{3}\beta^2$. We analyze the impact of an edge weight increase in an instance for which we have an optimal solution, trying to exploit the relaxed metric properties of the instance. In the following, for a given graph G , let $\bar{P}$ denote a minimum-cost Hamiltonian path in G with predetermined endpoints s and t , and let OPT_G be the cost of $\bar{P}$.

Now we can give a formal definition of our problem:

Definition 5. The edge-increased $\Delta_{\beta>1}$ -TSPP (ei- $\Delta_{\beta>1}$ -TSPP for short) is defined as follows: Given

- two complete non-negatively real weighted undirected graphs $G_o = (V, E, c_o)$ and $G_m = (V, E, c_m)$ such that G_o and G_m obey the triangle inequality up to a relaxation factor of β_o and β_m , respectively, with $\beta_o, \beta_m \geq 1$, and such that c_o and c_m coincide, except that, for an edge e_i , we have $c_o(e_i) < c_m(e_i)$;
- two predetermined vertices s and t ;
- a Hamiltonian path $\bar{P}$ of G_o between s and t such that $\sum_{e \in \bar{P}} c_o(e) = \text{OPT}_{G_o}$;

find a Hamiltonian path P of G_m between s and t such that $\sum_{e \in P} c_m(e) = \text{OPT}_{G_m}$.

In the following, we describe our approximation algorithm for the ei- $\Delta_{\beta>1}$ -TSPP. Let $e = (i, j)$ be the edge of G whose weight is increased (i.e., $c_o(e) < c_m(e)$). If $e \notin \bar{P}$ then, trivially, $\bar{P}$ is an optimal path in G_m . Thus we assume in the following $e \in \bar{P}$. Also it is worth noting that, if the number of nodes of the graph is at most 3, then the edge $e = (i, j)$ is always an edge of the optimal solution of the modified instance, and therefore $\text{OPT}_{G_m} = \text{OPT}_{G_o} - c_o(e) + c_m(e)$. Consequently, w.l.o.g., we assume that the instances have at least 4 nodes.

The idea is simple: if an edge $e = (i, j)$ of the optimal path $\bar{P}$ has increased its weight by a quantity c , either we keep $\bar{P}$ as a solution or we find an alternative s - t path that does not include e . The algorithm, in the general case, finds two alternative subpaths P_i and P_j of $\bar{P}$ (see Fig. 1), and then selects the cheaper of the two, say P' . Let P^* be the unique subpath of $\bar{P}$ that has the same endpoints as P' . Then the algorithm returns the cheaper (with respect to the increase in the cost of e) between $\bar{P}$ and the alternative path $\tilde{P} = \bar{P} - P^* + P'$.

Notice that there is only one case in which both $i < 3$ and $j > t - 2$ holds, namely when $i = 2$ and $j = 3$ and the input instance has 4 nodes, but in this case, since $s = 1$ and $t = 4$ are fixed, the edge $e = (2, 3)$ is always an edge of the optimal solution of the modified instance. For all other instances with at least 5 nodes, it is not possible that both $i < 3$ and $j > t - 2$ holds. Indeed, if we consider the extreme case in which the instance has exactly 5 nodes we have the following:

- The vertices of $\bar{P}$ are numbered in non-decreasing order from 1 to 5, where $s = 1$ and $t = 5$ are the endpoints of $\bar{P}$;
- $e = (i, j)$, with $i < j$, is the edge of $\bar{P}$ whose weight is increased;
- thus, since $e \in \bar{P}$ and the vertices of $\bar{P}$ are numbered in nondecreasing order, then $i = j - 1$ and therefore it is not possible that $i < 3$ and $j > t - 2 = 3$.

This ensures that if L is empty after line 12, the algorithm returns $\bar{P}$ which is optimal for the modified instance, otherwise L is not empty.

We are now ready to give our main result, that is we show that Algorithm 2 guarantees an approximation of $1 + 2\beta_l$ for the ei- $\Delta_{\beta>1}$ -TSPP, which outperforms the $\frac{5}{3}\beta^2$ approximation ratio of the PMCA-TSPP algorithm for $\beta \geq \frac{3+2\sqrt{6}}{5} \approx 1.58$.

Fig. 1. From the top to the bottom, we have: $\overline{P}$ with the vertices numbered in non decreasing order and with $s = 1, t = 12$, the bold edge $e = (6, 7)$ is the edge whose weight is increased; P_i^* is the unique path in $\overline{P}$ having as endpoints $i - 2 = 4$ and $j = 7$; P_i is a path having the same endpoints as P_i^* and not including $e = (6, 7)$; P_j^* is the unique path in $\overline{P}$ having as endpoints $i = 6$ and $j + 2 = 9$; P_j is a path having the same endpoints as P_j^* and not including $e = (6, 7)$

Theorem 3. Let $(G_o, G_m, \overline{P})$ be an input instance of $ei - \Delta_{\beta > 1}$ -TSPP such that G_o and G_m obey the triangle inequality up to, respectively, β_o and β_m , with $\beta_o, \beta_m \geq 1$. Let G_o and G_m agree except for an edge e such that $c_m(e) > c_o(e)$. Let $\beta_L = \min\{\beta_o, \beta_m\}$ and let $s, t \in V(G_o)$ be the endpoints of $\overline{P}$. Then, Algorithm 2 provides a $(1 + 2\beta_L)$ -approximation with respect to the cost of P .

Proof. It is easy to verify that, if the number of nodes of the graph is 4, our algorithm always returns an optimal solution, so, w.l.o.g., we assume that the number of nodes is at least 5 and L is not empty. Let P' and P^* be the paths found by Algorithm 2 at lines 17 and 18, respectively. Since $e = (i, j) \notin P'$, we have $c_o(P') = c_m(P')$, while, on the other hand, since $e \in P^*$ and $c_m(e) > c_o(e)$, we have $c_o(P^*) < c_m(P^*)$. Now we show that

$$c_m(P') - c_o(P^*) \leq 2\beta_o \text{OPT}_{G_o} \quad (4)$$

and

$$c_m(P') - c_o(P^*) \leq 2\beta_m \text{OPT}_{G_m}. \quad (5)$$

By definition, we have

$$c_o(P_i) = c_o(i - 2, i) + c_o(i, i - 1) + c_o(i - 1, j)$$

Algorithm 2. ei- $\Delta_{\beta>1}$ -TSPP

```

1  Let  $s$  and  $t$  be the endpoints of  $\overline{P}$ ;
2  Number the vertices of  $\overline{P}$  in non decreasing order from 1 to  $n$ , with  $s = 1$  and
    $t = n$  and let  $e = (i, j)$ , with  $i < j$ , be the edge whose weight is increased;
3  Let  $L$  be a list of paths, initially empty;
4  if  $i \geq 3$  then
5      Let  $P_i^*$  be the unique path in  $\overline{P}$  having as endpoints  $i - 2$  and  $j$ ;
6      Let  $P_i = (i - 2, i), (i, i - 1), (i - 1, j)$  be a path having the same endpoints
       as  $P_i^*$ ;
7      Add  $P_i$  to  $L$  and set  $\text{key}(P_i)$  to the value of  $c(P_i)$ ;
8  end
9  if  $j \leq t - 2$  then
10     Let  $P_j^*$  be the unique path in  $\overline{P}$  having as endpoints  $i$  and  $j + 2$ ;
11     Let  $P_j = (i, j + 1), (j + 1, j), (j, j + 2)$  be a path having the same endpoints
       as  $P_j^*$ ;
12     Add  $P_j$  to  $L$  and set  $\text{key}(P_j)$  to the value of  $c(P_j)$ ;
13  end
14  if  $L$  is empty then
15      output  $\overline{P}$ .
16  else
17      Let  $P'$  be the path in  $L$  having minimum cost;
18      Let  $P^*$  be the unique subpath of  $\overline{P}$  having the same endpoints as  $P'$ ;
19      if  $c_M(P') \geq c_M(P^*)$  then
20          output  $\overline{P}$ .
21      else
22          output  $\tilde{P} = \overline{P} - P^* + P'$ .
23      end
24  end

```

and

$$c_o(P_i^*) = c_o(i - 2, i - 1) + c_o(i - 1, i) + c_o(i, j).$$

Hence

$$c_o(P_i) - c_o(P_i^*) = c_o(i - 2, i) + c_o(i - 1, j) - c_o(i - 2, i - 1) - c_o(i, j). \quad (6)$$

Moreover, since the β -triangle inequality holds, we have that

$$c_o(i - 2, i) \leq \beta_o (c_o(i - 2, i - 1) + c_o(i - 1, i))$$

and

$$c_o(i - 1, j) \leq \beta_o (c_o(i - 1, i) + c_o(i, j)).$$

Hence

$$c_o(i - 2, i) + c_o(i - 1, j) \leq 2\beta_o c_o(P_i^*), \quad (7)$$

and from (6) and (7) we have

$$c_M(P_i) - c_O(P_i^*) = c_O(P_i) - c_O(P_i^*) \leq 2\beta_O c_O(P_i^*) \leq 2\beta_O \text{OPT}_{G_O}. \quad (8)$$

Moreover we have that

$$c_O(i-2, i) = c_M(i-2, i) \leq \beta_M (c_M(i-2, i-1) + c_M(i-1, i))$$

and

$$c_O(i-1, j) = c_M(i-1, j) \leq \beta_M (c_M(i-1, i) + c_M(i, j)).$$

Therefore

$$c_M(i-2, i) + c_M(i-1, j) \leq 2\beta_M c_M(P_i^*), \quad (9)$$

and from (6) and (9) we have

$$c_M(P_i) - c_O(P_i^*) \leq 2\beta_M c_M(P_i^*) \leq 2\beta_M \text{OPT}_{G_M}. \quad (10)$$

With a similar argument, one can prove that

$$c_M(P_j) - c_O(P_j^*) \leq 2\beta_O c_O(P_j^*) \leq 2\beta_O \text{OPT}_{G_O}$$

and

$$c_M(P_j) - c_O(P_j^*) \leq 2\beta_M c_M(P_j^*) \leq 2\beta_M \text{OPT}_{G_M}.$$

Hence (4) and (5) hold true. The rest is proven by a case distinction.

Case 1: If $c_M(P^*) \leq c_M(P')$, we return $\bar{P}$. Indeed:

$$\begin{aligned} \frac{c_M(\bar{P})}{\text{OPT}_{G_M}} &= \frac{\text{OPT}_{G_O} - c_O(e) + c_M(e)}{\text{OPT}_{G_M}} \\ &= \frac{\text{OPT}_{G_O} - c_O(P^*) + c_M(P^*)}{\text{OPT}_{G_M}} \\ &\leq \frac{\text{OPT}_{G_O} - c_O(P^*) + c_M(P')}{\text{OPT}_{G_M}} \\ &\stackrel{(4)}{\leq} \frac{\text{OPT}_{G_O} + 2\beta_O \text{OPT}_{G_O}}{\text{OPT}_{G_O}} \\ &= 1 + 2\beta_O. \end{aligned} \quad (11)$$

But we also have

$$\begin{aligned} \frac{c_M(\bar{P})}{\text{OPT}_{G_M}} &= \frac{\text{OPT}_{G_O} - c_O(e) + c_M(e)}{\text{OPT}_{G_M}} \\ &= \frac{\text{OPT}_{G_O} - c_O(P^*) + c_M(P^*)}{\text{OPT}_{G_M}} \\ &\leq \frac{\text{OPT}_{G_O} - c_O(P^*) + c_M(P')}{\text{OPT}_{G_M}} \\ &\stackrel{(5)}{\leq} \frac{\text{OPT}_{G_M} + 2\beta_M \text{OPT}_{G_M}}{\text{OPT}_{G_M}} \\ &= 1 + 2\beta_M. \end{aligned} \quad (12)$$

Thus, the combination of (11) and (12) yields

$$\frac{c_M(\bar{P})}{\text{OPT}_{G_M}} \leq 1 + 2\beta_L. \quad (13)$$

Case 2: otherwise, if $c_M(P^*) > c_M(P')$, we return $\tilde{P} = \bar{P} - P^* + P'$. Indeed:

$$\begin{aligned} \frac{c_M(\tilde{P})}{\text{OPT}_{G_M}} &\leq \frac{\text{OPT}_{G_O} - c_O(P^*) + c_M(P')}{\text{OPT}_{G_M}} \\ &\stackrel{(4)}{\leq} \frac{\text{OPT}_{G_O} + 2\beta_O \text{OPT}_{G_O}}{\text{OPT}_{G_O}} \\ &= 1 + 2\beta_O. \end{aligned} \quad (14)$$

But we also have

$$\begin{aligned} \frac{c_M(\tilde{P})}{\text{OPT}_{G_M}} &\leq \frac{\text{OPT}_{G_O} - c_O(P^*) + c_M(P')}{\text{OPT}_{G_M}} \\ &\stackrel{(5)}{\leq} \frac{\text{OPT}_{G_M} + 2\beta_M \text{OPT}_{G_M}}{\text{OPT}_{G_M}} \\ &= 1 + 2\beta_M. \end{aligned} \quad (15)$$

Thus, the combination of (14) and (15) yields

$$\frac{c_M(\tilde{P})}{\text{OPT}_{G_M}} \leq 1 + 2\beta_L, \quad (16)$$

and so the claim follows by (13) and (16). $\square$

4 Conclusion

In this paper, we reviewed the influential concept of stability of approximation, introduced by Hromkovič [26,27], that has inspired many advances in the investigation of hard optimization problems. In turn, we used this idea to study two reoptimization variants of β -metric TSP and TSPP for which no results were known in the literature. We obtained a PTAS for the reoptimization of the strengthened metric TSP subject to a single node modification, and a constant approximation algorithm for the reoptimization of the relaxed metric TSPP subject to a single edge weight increase.

Although there are already many applications of stability of approximation, we think that this concept can still bring many interesting results. For example, we envision possible new results on TSPP, by combining such a powerful idea with recent developments in the study of this problem on metric input instances [29,30]. Furthermore, another interesting direction is that of analyzing under the stability perspective the *asymmetric* metric TSP, for which a recent breakthrough showed the existence of a constant-factor approximation algorithm [31].

Dedication

The authors would like to thank Juraj Hromkovič for having inspired their work on the topic discussed in this paper, as well as on several other research areas of interest.

References

1. Andreae, T.: On the traveling salesman problem restricted to inputs satisfying a relaxed triangle inequality. *Netw. Int. J.* **38**(2), 59–67 (2001)
2. Andreae, T., Bandelt, H.-J.: Performance guarantees for approximation algorithms depending on parametrized triangle inequalities. *SIAM J. Discrete Math.* **8**(1), 1–16 (1995)
3. Archetti, C., Bertazzi, L., Speranza, M.G.: Reoptimizing the traveling salesman problem. *Networks* **42**(3), 154–159 (2003)
4. Ausiello, G., Escoffier, B., Monnot, J., Paschos, V.T.: Reoptimization of minimum and maximum traveling salesman's tours. In: Arge, L., Freivalds, R. (eds.) *SWAT 2006*. LNCS, vol. 4059, pp. 196–207. Springer, Heidelberg (2006). https://doi.org/10.1007/11785293_20
5. Bender, M.A., Chekuri, C.: Performance guarantees for the TSP with a parameterized triangle inequality. In: Dehne, F., Sack, J.-R., Gupta, A., Tamassia, R. (eds.) *WADS 1999*. LNCS, vol. 1663, pp. 80–85. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48447-7_10
6. Berg, T., Hempel, H.: Reoptimization of traveling salesperson problems: changing single edge-weights. In: Dediu, A.H., Ionescu, A.M., Martín-Vide, C. (eds.) *LATA 2009*. LNCS, vol. 5457, pp. 141–151. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00982-2_12
7. Bilò, D., Forlizzi, L., Proietti, G.: Approximating the metric TSP in linear time. *Theory Comput. Syst.* **49**(3), 615–631 (2011)
8. Bilò, D., Zych, A.: New advances in reoptimizing the minimum Steiner tree problem. In: Rován, B., Sassone, V., Widmayer, P. (eds.) *MFCS 2012*. LNCS, vol. 7464, pp. 184–197. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32589-2_19
9. Bilò, D., et al.: Reoptimization of Steiner trees. In: Gudmundsson, J. (ed.) *SWAT 2008*. LNCS, vol. 5124, pp. 258–269. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-69903-3_24
10. Böckenhauer, H.-J., Hromkovič, J., Kneis, J., Kupke, J.: On the approximation hardness of some generalizations of TSP. In: Arge, L., Freivalds, R. (eds.) *SWAT 2006*. LNCS, vol. 4059, pp. 184–195. Springer, Heidelberg (2006). https://doi.org/10.1007/11785293_19
11. Böckenhauer, H.-J., et al.: On the hardness of constructing minimal 2-connected spanning subgraphs in complete graphs with sharpened triangle inequality. *Theor. Comput. Sci.* **326**(1–3), 137–153 (2004)
12. Böckenhauer, H.-J., et al.: On k -connectivity problems with sharpened triangle inequality. *J. Discrete Algorithms* **6**(4), 605–617 (2008)
13. Böckenhauer, H.-J., et al.: On the approximability of TSP on local modifications of optimally solved instances. *Algorith. Oper. Res.* **2**, 83–93 (2007)
14. Böckenhauer, H.-J., Freiermuth, K., Hromkovič, J., Mömke, T., Sprock, A., Steffen, B.: Steiner tree reoptimization in graphs with sharpened triangle inequality. *J. Discrete Algorithms* **11**, 73–86 (2012)

15. Böckenhauer, H.-J., Hromkovič, J., Klasing, R., Seibert, S., Unger, W.: Approximation algorithms for the TSP with sharpened triangle inequality. *Inf. Process. Lett.* **75**, 133–138 (2000)
16. Böckenhauer, H.-J., Hromkovič, J., Klasing, R., Seibert, S., Unger, W.: An improved lower bound on the approximability of metric TSP and approximation algorithms for the TSP with sharpened triangle inequality. In: Reichel, H., Tison, S. (eds.) *STACS 2000. LNCS*, vol. 1770, pp. 382–394. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-46541-3_32
17. Böckenhauer, H.-J., Hromkovič, J., Klasing, R., Seibert, S., Unger, W.: Towards the notion of stability of approximation for hard optimization tasks and the traveling salesman problem. *Theor. Comput. Sci.* **285**(1), 3–24 (2002)
18. Böckenhauer, H.-J., Hromkovič, J., Kneis, J., Kupke, J.: The parameterized approximability of TSP with deadlines. *Theory Comput. Syst.* **41**(3), 431–444 (2007)
19. Böckenhauer, H.-J., Hromkovič, J., Královic, R., Mömke, T., Rossmann, P.: Reoptimization of Steiner trees: changing the terminal set. *Theor. Comput. Sci.* **410**(36), 3428–3435 (2009)
20. Böckenhauer, H.-J., Kneis, J., Kupke, J.: Approximation hardness of deadline-TSP reoptimization. *Theor. Comput. Sci.* **410**(21–23), 2241–2249 (2009)
21. Böckenhauer, H.-J., Komm, D.: Reoptimization of the metric deadline TSP. *J. Discrete Algorithms* **8**(1), 87–100 (2010)
22. Böckenhauer, H.-J., Seibert, S.: Improved lower bounds on the approximability of the traveling salesman problem. *ITA* **34**(3), 213–255 (2000)
23. Christofides, N.: Worst-case analysis of a new heuristic for the traveling salesman problem. Technical report, Graduate School of Industrial Administration, Carnegie-Mellon University (1976)
24. D'Andrea, A., Proietti, G.: Reoptimizing the strengthened metric TSP on multiple edge weight modifications. In: Klasing, R. (ed.) *SEA 2012. LNCS*, vol. 7276, pp. 111–122. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-30850-5_11
25. Forlizzi, L., Hromkovič, J., Proietti, G., Seibert, S.: On the stability of approximation for Hamiltonian path problems. *Algorithm. Oper. Res.* **1**(1), 31–45 (2006)
26. Hromkovič, J.: Stability of approximation algorithms and the knapsack problem. In: Karhumäki, J., Maurer, H., Păun, G., Rozenberg, G. (eds.) *Jewels are Forever, Contributions on Theoretical Computer Science in Honor of Arto Salomaa*, pp. 238–249. Springer, Heidelberg (1999). https://doi.org/10.1007/978-3-642-60207-8_21
27. Hromkovič, J.: Stability of approximation algorithms for hard optimization problems. In: Pavelka, J., Tel, G., Bartošek, M. (eds.) *SOFSEM 1999. LNCS*, vol. 1725, pp. 29–47. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-47849-3_2
28. Rosenkrantz, D.J., Stearns, R.E., Lewis II, P.M.: An analysis of several heuristics for the traveling salesman problem. *SIAM J. Comput.* **6**(3), 563–581 (1977)
29. Sebö, A.: Eight-fifth approximation for the path TSP. In: Goemans, M., Correa, J. (eds.) *IPCO 2013. LNCS*, vol. 7801, pp. 362–374. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36694-9_31
30. Sebö, A., van Zuylen, A.: The salesman's improved paths: A $3/2+1/34$ approximation. In: *IEEE 57th Annual Symposium on Foundations of Computer Science. FOCS 2016*, 9–11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA, pp. 118–127 (2016)

31. Svensson, O., Tarnawski, J., Végh, L.A.: A constant-factor approximation algorithm for the asymmetric traveling salesman problem. CoRR, abs/1708.04215 (2017)
32. Vygen, J.: New approximation algorithms for the TSP. OPTIMA **90**, 1–12 (2012)