The Parameterized Approximability of TSP with Deadlines[*]

Hans-Joachim Böckenhauer,[1] Juraj Hromkovič,[1] Joachim Kneis,[2] and Joachim Kupke[3]

[1]Department of Computer Science, ETH Zurich,
CH-8092 Zürich, Switzerland
{hjb,juraj.hromkovic}@inf.ethz.ch

[2]Department of Computer Science, RWTH Aachen University,
D-52056 Aachen, Germany
joachim.kneis@cs.rwth-aachen.de

[3]Google, Inc.
Mountain View, CA 94043, USA
joachimk@google.com

Abstract. Modern algorithm theory includes numerous techniques to attack hard problems, such as approximation algorithms on the one hand and parameterized complexity on the other hand. However, it is still uncommon to use these two techniques simultaneously, which is unfortunate, as there are natural problems that cannot be solved using either technique alone, but rather well if we combine them.

The problem to be studied here is not only natural, but also practical: Consider TSP, generalized as follows. We impose deadlines on some of the vertices, effectively constraining them to be visited prior to a given point of time. The resulting problem DLTSP (a special case of the well-known *TSP with time windows*) inherits its hardness from classical TSP, which is both well known from practice and renowned to be one of the hardest problems with respect to approximability: Within polynomial time, not even a polynomial approximation ratio (let alone a constant one) can be achieved (unless $P = NP$).

We will show that DLTSP is even harder than classical TSP in the following sense. Classical TSP, despite its hardness, admits good approximation algorithms if restricted to metric (or near-metric) inputs. Not so DLTSP (and hence, neither TSP with time windows): We will prove that even for metric inputs, no constant approximation ratio can ever be achieved (unless $P = NP$).

This is where parameterization becomes crucial: By combining methods from the field of approximation algorithms with ideas from the theory of parameterized complexity, we apply the concept of parameterized approximation. Thereby, we

obtain a 2.5-approximation algorithm for DLTSP with a running time of $k! \cdot poly(|G|)$, where k denotes the number of deadlines. Furthermore, we prove that there is no fpt-algorithm with an approximation guarantee of $2 - \varepsilon$ for any $\varepsilon > 0$, unless $P = NP$.

Finally, we show that, unlike TSP, DLTSP becomes much harder when relaxing the triangle inequality. More precisely, for an arbitrary small violation of the triangle inequality, DLTSP does not admit an fpt-algorithm with approximation guarantee $((1 - \varepsilon)/2)|V|$ for any $\varepsilon > 0$, unless $P = NP$.

1. Introduction

The traveling salesperson problem (TSP) is one of the most prominent optimization problems with numerous practical applications. Worst-case analyses show that it is indeed one of the hardest problems with respect to approximability because, provided that $P \neq NP$, there is no polynomial-time approximation algorithm for TSP with an approximation ratio bounded by a polynomial in the problem instance size [H2]. This holds true even if the distances between vertices are represented in unary.[1]

An important generalization of TSP which naturally and frequently appears in a number of applications is TSP with time windows. Here, for some of the vertices of the input graph, an opening and a closing point of time for a window is given, and the respective vertex must be visited while the window is open.

The importance of TSP with time windows has been recognized in operations research, and a multitude of both exact (yet exponential-time) algorithms and heuristics (yet without performance guarantee) have been proposed; for a survey, see [CDD+]. It is due to the hardness of TSP with time windows that so far, it has not been possible to establish (reasonable) performance guarantees. (Unaltered) TSP is already one of the hardest known problems with respect to its worst-case approximability, so there is no hope for (decent) approximability results regarding TSP with time windows.

However, it is a somewhat surprising fact that TSP is sometimes not quite as hard as it looks, not only from a practical point of view, but also in worst-case analyses. Indeed, metric TSP (ΔTSP) can be solved efficiently with an approximation guarantee of 1.5 [C] and, using the concept of approximation stability [H1], [BHK+], [FHPS], quite a few papers have shown that even for a relaxation of the metricity constraint by relaxing the triangle inequality to the so-called β-triangle inequality for some $\beta > 1$, i.e.,

$$c(\{v_i, v_j\}) \leq \beta \cdot \left(c(\{v_i, v_z\}) + c(\{v_z, v_j\}) \right)$$

for any three vertices v_i, v_j, v_z, a constant approximation ratio can be achieved in polynomial time [AB], [A], [BC], [BHK+]. More precisely, TSP on input instances satisfying a relaxed β-triangle inequality, Δ_βTSP for short, can be approximated in polynomial time within $\min\{\beta^2 + \beta, \frac{3}{2}\beta^2, 4\beta\}$.

---

[1] Efficient algorithms on inputs represented in unary would translate to efficient algorithms for the special case where the distances between vertices are "natural" in that they are bounded by a polynomial in the number of vertices. All of our results, while naturally expressed with no regard to the arity of the input representation, carry over to the unary case.

Now, not only can TSP be restricted to metric and near-metric inputs, but also TSP with time windows can be similarly restricted. In this paper we focus on the special case of TSP with time windows, where the opening time is set to zero for all vertices. We call the closing time the *deadline* and the resulting problem *TSP with deadlines* or DLTSP for short.[2]

Naturally, DLTSP is at least as hard as TSP. We will show that it is even harder by proving that there is no polynomial-time $o(|V|)$-approximation algorithm for metric DLTSP, while metric TSP, as mentioned above, admits a constant approximation algorithm. Attacking a problem that is even harder than TSP calls for strategies more sophisticated than those commonly employed.

Our strategy of choice will be to combine two well-known strategies, namely approximation and parameterization. These are rarely used in conjunction, although the idea can be found in the literature since the early nineties [CC]. Most of the publications combining these approaches use the approximation guarantee as the parameter, thereby establishing efficient polynomial-time approximation schemes (EPTAS) [CT]. Likewise, counting problems have been attacked using randomized parameterized approximation algorithms [AR].

Here, we use a parameter that is completely independent from the approximation guarantee. Therefore, we can apply this idea to problems that are APX-hard or do not admit a polynomial-time algorithm with a constant approximation ratio in the non-parameterized case. Throughout this paper, input instances will be parameterized in the most natural way, i.e., by the number of vertices that have a deadline assigned to them.

The good news is that there exists a parameterized 2.5-approximation algorithm for metric DLTSP. The bad news is that there is no parameterized approximation algorithm for metric DLTSP with approximation ratio $< 2 - \varepsilon$ unless $P = NP$. (This implies that, as long as we insist on exact solutions, metric TSP with deadlines is not fixed-parameter tractable. In other words, neither approximation nor parameterization alone would yield tractability.) This lower bound even holds for metric DLTSP with as few as two deadlines.

As another lower bound, we will establish that there is no parameterized $o(|V|)$-approximation algorithm for TSP with deadlines in the near-metric case. This bound already holds for an arbitrary small deviation from metricity and already in the case of a single deadline vertex, rendering any conceivable parameterized approach useless. In other words (see [BHK+]): While TSP admits a *stable* approximation algorithm with respect to the parameter β, we show that TSP with a constant number of deadlines does not.

All these inapproximability results directly carry over to the respective variations of TSP with time windows.

The paper is organized as follows: In Section 2 we introduce the concept of parameterized approximation and formally define TSP with deadlines. Section 3 contains a parameterized approximation algorithm for metric TSP with deadlines. We prove inapproximability results for parameterized approximation in Section 4 and for non-parameterized approximation in Section 5.

---

[2] A variation on TSP with deadlines was investigated in [BBCM]. Here, the goal was to find a tour which obeys a maximum number of deadlines. In contrast, we only look for solutions that obey every given deadline.

2. Definitions

We assume that the reader is familiar with the concepts of parameterized complexity and approximation algorithms. Detailed introductions can for instance be found in [DF] and [H1].

Definition 1. A parameterized optimization (or decision, resp.) problem is a pair (U, κ) such that U is an optimization (or decision, resp.) problem and κ is a parameterization of U, i.e., a function which maps all admissible inputs of U to $\mathbb{N}$. By k-U, we denote the (non-parameterized) restriction of U where only those inputs x are admissible that have $\kappa(x) \leq k$.

We define fixed-parameter algorithms as follows:

Definition 2. An algorithm A for a parameterized problem (U, κ) is an *fpt-algorithm* iff

- A solves U; and
- there exist a function f and a polynomial p such that for all input instances x of U, the running time $Time_A(x)$ of A on x can be bounded by

    $Time_A(x) \leq f(\kappa(x)) \cdot p(|x|).$

Those problems that admit an fpt-algorithm form the class of fixed-parameter tractable problems, denoted by FPT.

We will show that we cannot guarantee a good approximation ratio for metric TSP with deadlines if we restrict ourselves to polynomial-time approximation algorithms. Besides, we prove that there is no fpt-algorithm solving this problem exactly (unless $P = NP$). We therefore combine both approaches to design a practical algorithm. This combination gives rise to the notion of fpt-approximation algorithms.

Definition 3. Let (U, κ) be a parameterized optimization problem. An algorithm A is an fpt-α-approximation algorithm for (U, κ) iff

- A is an α-approximation algorithm for U; and
- there exist a function f and a polynomial p such that for all input instances x of U, the running time $Time_A(x)$ of A on x can be bounded by

    $Time_A(x) \leq f(\kappa(x)) \cdot p(|x|).$

We now define formally *TSP with deadlines*, which the remainder of this paper is concerned with.

Definition 4. Let $G = (V, E)$ be a complete graph with edge weights $c: E \rightarrow \mathbb{Q}^+$. We call (s, D, d) a *deadline set* for G if $s \in V$, $D \subseteq V \setminus \{s\}$ and $d: D \rightarrow \mathbb{Q}^+$. A vertex $v \in D$ is called a *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 5. The problem $\Delta_\beta\text{DLTSP}$ is defined as follows.

Instance: A complete weighted graph $G = (V, E, c)$ satisfying the Δ_β-inequality, deadlines (s, D, d) for G.
Goal: Find a minimum-weight Hamiltonian cycle satisfying all deadlines.
Parameter: $|D|$.

For the most common case of $\beta = 1$, we use the abbreviation $\Delta\text{DLTSP} := \Delta_1\text{DLTSP}$.

In Section 5 we will show that ΔDLTSP is not approximable with an approximation guarantee in $o(|V|)$. Therefore, we use the concept of parameterized approximation to obtain a useful algorithm for ΔDLTSP.

We start with a lemma describing the change in the cost of a path, in a graph satisfying the relaxed triangle inequality, when a subpath is replaced by the direct edge between its endpoints.

Lemma 1. *Let $G = (V, E)$ be a complete graph with edge weights $c \colon E \rightarrow \mathbb{Q}^+$ satisfying the Δ_β-inequality. Let $P := (v_0, \ldots, v_k)$ be a (simple) path in G. Then $c(\{v_0, v_k\}) \leq \beta^{\log_2 k} \cdot \text{cost}(P)$, where $\text{cost}(P)$ denotes the sum of edge weights along the path. Furthermore, there exists a graph such that this estimation is tight.*

Proof idea. The inequality can easily be proven by induction. For a tight example, see Figure 1. $\square$
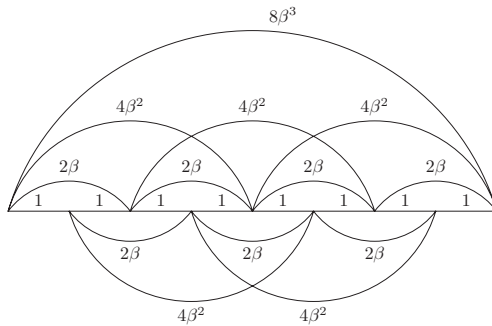

Fig. 1. Maximal edge weights for a given path (Lemma 1).

3. A Parameterized Approximation Algorithm for ΔDLTSP

If only we knew the order in which the deadlines are visited in an optimal solution we could try to start with this order and somehow insert the remaining vertices. As k is a constant, we can exhaustively try every permutation of deadlines to obtain this information. Unfortunately, inserting the remaining vertices into this sequence in an optimal way is still a hard problem. Therefore, Algorithm 1 just inserts them after the last deadline. This results in a 2.5-approximation.

Algorithm 1

Input: A complete weighted graph $G = (V, E, c)$ with n vertices and deadline set (s, D, d) for G with $|D| = k$.

   Compute a Hamiltonian cycle $X = (s, x_1, \ldots, x_{n-k-1}, s)$ on $V \setminus D$ using Christofides' algorithm.
   for every linear order $\pi = (s, p_1, p_2, \ldots, p_k)$ of all vertices in $D \cup \{s\}$ **do**
      if π satisfies the deadlines **then**
         Concatenate the linear order of the deadline vertices and the Hamiltonian tour X and obtain an Eulerian tour $U_\pi = (s, p_1, p_2, \ldots, p_k, s, x_1, x_2, \ldots, x_{n-k-1}, s)$. Shorten U_π to a Hamiltonian cycle $H_\pi = (s, p_1, p_2, \ldots, p_k, x_1, x_2, \ldots, x_{n-k-1}, s)$ for G.

Output: The cheapest of all computed Hamiltonian cycles H_π satisfying all deadlines, if one exists; an error message, otherwise.

Theorem 1. *Algorithm* 1 *solves* ΔDLTSP *with an approximation ratio* ≤ 2.5 *in time* $\mathcal{O}(n^3 + k! \cdot k)$ *where n is the number of vertices in the input graph and k denotes the number of deadline vertices*.

Proof.   Let I be an input instance for ΔDLTSP, consisting of a complete graph $G = (V, E)$ with edge weights $c\colon E \to \mathbb{Q}^+$ and a set of deadlines (s, D, d). Let n denote the number of vertices of G and let $k = |D|$ denote the number of deadline vertices.

   Algorithm 1 first computes the Hamiltonian tour X using Christofides' algorithm. This can be done in time $\mathcal{O}(n^3)$ [PS]. Afterwards, the algorithm considers all $k!$ possible linear orderings of the vertices in D. For each of these orderings, it checks its feasibility, which can be done in time $\mathcal{O}(k)$, and constructs the Hamiltonian tour H_π for each feasible ordering π, a construction which can be done in constant time for any feasible ordering. This establishes the claimed bound on the running time.

   Algorithm 1 cannot output a solution that misses a deadline because it checks this condition for every candidate. Let us therefore show that it finds a feasible solution for ΔDLTSP if such a solution exists.

   Let F be a feasible solution for the given instance. Let π be the order of deadline vertices given by F. Then Algorithm 1, too, considers this order π of the deadline vertices and computes a corresponding Hamiltonian tour H_π. This tour starts in s and visits the vertices of D in the order described by π directly after s. F also starts in s and uses the same order on D, but may visit some other vertices in between. Hence, H_π reaches

every deadline vertex at the time when it would be visited by F or even earlier. Thus, H_π is a feasible solution, and Algorithm 1 outputs a feasible solution, too.

Let us now prove the upper bound on the approximation ratio. Let Opt be an optimal solution for I and let Sol be the solution computed by Algorithm 1. Let π be the order of the deadline vertices in Opt. Then Algorithm 1 considers the tour H_π and we have $\text{cost}(Sol) \le \text{cost}(H_\pi)$.

The tour H_π is a shortening of the Eulerian tour U_π which consists of two cycles, first by a cycle C through the vertices in $D \cup \{s\}$ ordered according to π, and second by a Hamiltonian tour X through all vertices in $V \setminus D$.

We have $\text{cost}(C) \le \text{cost}(Opt)$ because C is just a shortening of Opt. Moreover, $\text{cost}(X) \le \frac{3}{2} \text{cost}(Opt)$ since X is a $\frac{3}{2}$-approximation of the shortest Hamiltonian tour of some subgraph G' of G, and Opt is a Hamiltonian cycle in G. Therefore, we obtain

$$\text{cost}(Sol) \le \text{cost}(C) + \text{cost}(X) \le \tfrac{5}{2} \text{cost}(Opt). \qquad \square$$

Corollary 1. *Algorithm 1 is an fpt-2.5-approximation algorithm for ΔDLTSP.*

Note that this algorithm only works if the Δ-inequality holds. If just the Δ_β-inequality holds for some $\beta > 1$, visiting the deadline vertices first does not guarantee that all of the deadlines are reached in time since direct edges between deadline vertices may be much more expensive than longer paths (see Lemma 1).

4. Lower Bounds for k-Δ_βDLTSP

At first sight, Algorithm 1 seems to have three major problems. First, it has running time exponential in the number of deadlines. Second, it only works if the Δ-inequality holds, i.e., for $\beta > 1$, it does not necessarily find a feasible solution, let alone a good one. Finally, its approximation ratio is only 2.5. We discuss whether polynomial time is possible in Section 5; the other problems are both handled here. First, let us show that for the near-metric case, no algorithm with a constant approximation ratio exists which would solve TSP with a single deadline vertex only (i.e., 1-Δ_βDLTSP), unless $P = NP$. In a second step, we will show that there is no $(2 - \varepsilon)$-approximation algorithm (again, unless $P = NP$) for metric TSP with as few as two deadline vertices, i.e., $2 - \varepsilon$ is a lower bound on the approximability of 2-ΔDLTSP.

Theorem 2. *Let $\beta > 1$ and $0 < h < 1$. There is no polynomial-time algorithm for 1-Δ_βDLTSP with approximation ratio $\frac{1}{4}\beta^{\log_2(h|V|)-2}$ unless $P = NP$.*

Proof. By means of a reduction, we will show that such an approximation algorithm could be used to solve the HAMILTONIAN PATH problem, i.e., the problem of deciding whether a given undirected graph contains a Hamiltonian path or not, which is *NP*-complete [GJ].

Let $\beta > 1$ and $0 < h < 1$, and let $G' = (V', E')$ be an input instance for the HAMILTONIAN PATH problem with $|V'| = n$. We construct a complete graph $G = (V, E, c)$ for the Δ_βDLTSP as follows:

Let $p > h(|V'| + 1)/(1 - h)4$ and $V := V' \cup \{v_1, \ldots, v_p\} \cup \{s, v'_1, \ldots, v'_{p-1}\} \cup$

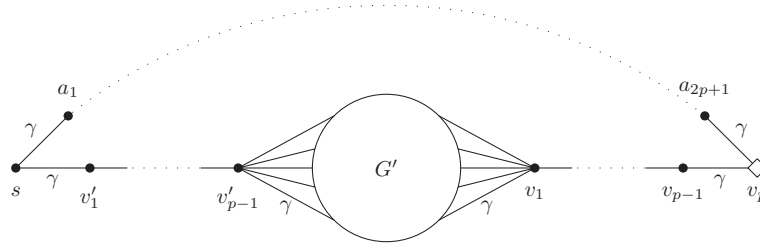**Fig. 2.**   Construction for 1-Δ_βDLTSP (Theorem 2).

$\{a_1, \ldots, a_{2p+1}\}$ and assign to every $e \in E$ the cost

$$
c(e) := \begin{cases} 1 & \text{if } e \in E', \\ 2 & \text{if } e \in (E')^{\complement}, \\ \gamma & \text{if } e \in \{\{s, v_1'\}\} \cup \{\{v_i', v_{i+1}'\} \mid i = 1, \ldots, p - 2\} \\ & \quad \cup \{\{v_{p-1}', v\} \mid v \in V'\} \\ & \quad \cup \{\{v, v_1\} \mid v \in V'\} \cup \{\{v_i, v_{i+1}\} \mid i = 1, \ldots, p - 1\} \\ & \quad \cup \{\{s, a_1\}, \{v_p, a_{2p+1}\}\} \cup \{\{a_i, a_{i+1}\} \mid i = 1, \ldots, 2p\}. \end{cases}
$$

All other edges have maximal possible costs such that the Δ_β-inequality is satisfied (see Figure 2). Here, γ may be chosen arbitrarily, provided

$$
\gamma > \max\left\{\frac{n}{2(\beta - 1)}, \frac{n}{2}\right\}.
$$

Note that $\gamma > n/2$ implies $4p\gamma > \gamma + n - 1$, which will be useful later in this proof. This graph satisfies the Δ_β-inequality because $\gamma > n/2(\beta - 1)$ implies $\gamma > 1/2\beta$. Here, we set $d(v_p) := 2p\gamma + n - 1$.

   Let W' be some path in V' and $W := (v_1', \ldots, v_{p-1}', W', v_1, \ldots, v_p)$. Obviously, W reaches v_p in time iff it spends at most time $n - 1$ in V', thus $\text{cost}(W') \leq n - 1$. The shortest path from s to v_p is $(s, v_1', v_2', \ldots, v_{p-1}', v, v_1, v_2, \ldots, v_p)$ for some $v \in V'$ and costs exactly $2p\gamma$.

   A path that visits some v_i' or v_j after v_p cannot reach this deadline because it causes an additional cost of at least $(\beta - 1)2\gamma \geq (\beta - 1)2(n/2(\beta - 1)) > n - 1$ on the way from s to v_p compared with the shortest path from s to v_p. If a path costs n or more in V' before visiting any vertex in $\{v_1, \ldots, v_p\}$, the deadline will also be missed, regardless of the path to v_p. Finally, a path that visits some a_i before v_p will also violate this deadline because among all those paths from s to v_p which contain one of the vertices a_i, the path from s via *all* of the a_i to v_p is the least expensive, and this one already costs

$$
(2p + 2)\gamma = 2p\gamma + 2\gamma > 2p\gamma + 2 \cdot \frac{n}{2} > d(v_p).
$$

   Assume G' contains a Hamiltonian path P. Then an optimal solution for k-Δ_βDLTSP is $(s, v_1', \ldots, v_{p-1}', P, v_1, \ldots, v_p, a_{2p+1}, \ldots, a_1, s)$. This cycle costs exactly $4p\gamma + \gamma + n - 1$.

Otherwise, an optimal solution cannot visit all vertices in V' before reaching v_p. Furthermore, it must visit all vertices v_i, v_i' for $i = 1, \ldots, p-1$ before v_p. Therefore, it must visit some vertex in V' after v_p. To do so, it must use some edge from a vertex in $\{v_p\} \cup \{a_i \mid i = 1, \ldots, 2p+1\}$ to some vertex $v \in V'$. By Lemma 1, such an edge costs at least $\beta^{\log_2(p)} p\gamma$. In order to leave V' again, another expensive edge must be used. Thus, an optimal solution costs at least $2\beta^{\log_2(p)} p\gamma + 4p\gamma + \gamma + n - 2$ if G does not contain a Hamiltonian path.

This leads to the ratio

$$
\frac{2\beta^{\log_2(p)} p\gamma + 4p\gamma + \gamma + n - 2}{4p\gamma + \gamma + n - 1}
$$

$$
= \frac{2\beta^{\log_2(p)} p\gamma}{4p\gamma + \gamma + n - 1} + 1 - \frac{1}{4p\gamma + \gamma + n - 1} > \frac{2\beta^{\log_2(p)} p\gamma}{8p\gamma}
$$

$$
= \frac{\beta^{\log_2(p)}}{4}.
$$

Our choice of p ensures $p > \frac{1}{4}(h(|V|))$ and hence

$$
\frac{\beta^{\log_2(p)}}{4} > \frac{\beta^{\log_2((1/4)h|V|)}}{4}.
$$

Therefore, a polynomial-time algorithm for the 1-Δ_βDLTSP with approximation ratio $\frac{1}{4}\beta^{\log_2(h|V|)-2}$ could be used to solve the HAMILTONIAN PATH problem in polynomial time. Because of $p < |V|/4$, increasing p does not improve this bound. $\square$

Corollary 2. *Let $\beta > 1$. There is no fpt-approximation algorithm for Δ_βDLTSP with approximation ratio $\frac{1}{4}(\beta^{\log_2(h)} + |V|^{\log_2(\beta)})$ for any $0 < h < 1$ and any $k \in \mathbb{N}$, unless $P = NP$.*

We can use a similar construction to obtain a lower bound of $\frac{3}{2} - \varepsilon$ on the approximation ratio for 1-ΔDLTSP. Compared with the ratio $\frac{5}{2}$ of Algorithm 1, this leaves a rather large gap. The next theorem shows that for 2-ΔDLTSP we can raise the lower bound to $2 - \varepsilon$. Note that any bound larger than $2 + \frac{1}{219} + \varepsilon$, would directly imply a lower bound of $\frac{220}{219} + \varepsilon$ for ΔTSP, improving the currently best known lower bound from [PV].

Theorem 3. *There is no polynomial-time algorithm for 2-ΔDLTSP with approximation ratio $2 - \varepsilon$ for any $\varepsilon > 0$ unless $P = NP$.*

Proof. Let $\varepsilon > 0$. We show that such an approximation algorithm for 2-ΔDLTSP could be used to solve the HAMILTONIAN PATH problem. Let $G' = (V', E')$ be an input instance for the HAMILTONIAN PATH problem with $|V'| = n$. We construct a complete graph $G = (V, E, c)$ for Δ_βDLTSP as follows:
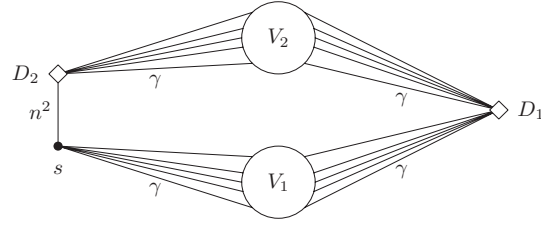
Fig. 3. Construction for 2-ΔDLTSP.

Let $V' = \{v_1, \ldots, v_n\}$. We construct two disjoint copies $V_1 := \{v_1^1, \ldots, v_n^1\}$, $V_2 := \{v_1^2, \ldots, v_n^2\}$ of V'. Then we set $V := V_1 \cup V_2 \cup \{s, D_1, D_2\}$ and

$$c(\{v_i^1, v_j^1\}) := \begin{cases} 1 & \text{if } \{v_i, v_j\} \in E', \\ 2 & \text{if } \{v_i, v_j\} \notin E', \end{cases} \qquad c(\{v_i^2, v_j^2\}) := \begin{cases} n & \text{if } \{v_i, v_j\} \in E', \\ 2n & \text{if } \{v_i, v_j\} \notin E'. \end{cases}$$

Furthermore, we set

$$c(e) := \begin{cases} \gamma & \text{if } e \in \{\{s, v^1\} \mid v^1 \in V_1\} \cup \{\{v^1, D_1\} \mid v^1 \in V_1\} \\ & \qquad \cup \{\{D_1, v^2\} \mid v^2 \in V_2\} \cup \{\{v^2, D_2\} \mid v^2 \in V_2\}, \\ n^2 & \text{if } e = \{s, D_2\}, \end{cases}$$

where γ may be chosen arbitrarily, provided

$$\gamma > \frac{(4n^2 - 2)/\varepsilon - 2n^2 + 1}{4}.$$

This easily yields $(4n^2 - 2)/(4\gamma + 2n^2 - 1) < \varepsilon$. Note that this graph satisfies the Δ-inequality. Finally, we assign the maximal possible costs satisfying the Δ-inequality to all other edges. Then the distance between $v^1 \in V_1$ and $v^2 \in V_2$ is exactly 2γ. An edge between $v^1 \in V_1$ and D_2 costs $\gamma + n^2$. We set $d(D_1) = 2\gamma + n - 1$ and $d(D_2) = 4\gamma + n^2 - 1$.

Now, assume G' contains a Hamiltonian path P. An optimal solution for 2-ΔDLTSP starts in s, follows P in V_1, and reaches D_1 in time $2\gamma + n - 1$. After this, it follows P in V_2, visits D_2 at $4\gamma + n^2 - 1$, and returns to s. This cycle costs $4\gamma + 2n^2 - 1$.

If G' contains no Hamiltonian path, it is not possible to visit all vertices in V_1 before reaching D_1. Hence, an optimal solution can reach D_1 in time $2\gamma + l$ by visiting $l + 1$ vertices in V_1, but $l + 1 < n$. Thus, this solution must visit some vertex v^1 in V_1 later, but it cannot visit v^1 directly after D_1 because that would violate the second deadline D_2, as it would reach D_2 no earlier than $2\gamma + l + 2\gamma + n^2 > 4\gamma + n^2 - 1$.

Therefore, this solution must go straight on to D_2. On its way, it can visit some vertices in V_2, but not all of them, as this would cost at least $\gamma + (n - 2)n + 2n + \gamma = 2\gamma + n^2$. After visiting D_2, this solution must go back to V_1 and V_2 and return to s in the end. D_2 is reached no sooner than at time 4γ, and the way back costs at least 4γ.

We obtain the ratio

$$\frac{8\gamma}{4\gamma + 2n^2 - 1} = \frac{8\gamma + 4n^2 - 2}{4\gamma + 2n^2 - 1} - \frac{4n^2 - 2}{4\gamma + 2n^2 - 1} > 2 - \varepsilon.$$

Therefore, a $(2 - \varepsilon)$-approximation algorithm for 2-ΔDLTSP could be used to solve the HAMILTONIAN PATH problem. □

Corollary 3. *There is no fpt-approximation-algorithm for* ΔDLTSP *with approximation ratio* $2 - \varepsilon$ *for any* $\varepsilon > 0$ *unless* $P = NP$.

5. Lower Bounds for ΔDLTSP

In the previous sections we have seen that ΔDLTSP allows for a parameterized approximation algorithm with approximation ratio 2.5 and we established a lower bound of $2 - \varepsilon$ on the parameterized approximation ratio. Now, we will show that this parameterized approach is crucial to obtaining a constant approximation ratio.

In fact, if we assign to all the vertices of a (metric) graph the same deadline, namely the minimum length (which we now interpret as duration) of a Hamiltonian path, finding feasible solutions *means* solving the HAMILTONIAN PATH problem. Yet beyond, we would like to prove that ΔDLTSP is hard, irrespective of how hard it might be to find feasible solutions. In order to accomplish this, let us define formally a slight variation of ΔDLTSP. The idea is to augment the input to any algorithm for the problem by one feasible solution.

Definition 6. The problem Δ_βDLTSP* is defined as follows.

Instance: A complete weighted graph $G = (V, E, c)$ satisfying the Δ_β-inequality, deadlines (s, D, d) for G, and a Hamiltonian cycle C satisfying all deadlines.
Goal: Find a minimum-weight Hamiltonian cycle satisfying all deadlines.
Parameter: $|D|$.

Again, we use the abbreviation ΔDLTSP* := Δ_1DLTSP*.

As a consequence, finding a feasible solution becomes trivial for ΔDLTSP*. We are now ready to show why ΔDLTSP* is nevertheless hard.

Theorem 4. *There is no polynomial-time algorithm for* ΔDLTSP* *with approximation ratio* $((1 - \varepsilon)/2)|V|$, *for any* $0 < \varepsilon < 1$ *unless* $P = NP$.

Proof. We show that such an approximation algorithm could be used to solve the HAMILTONIAN PATH problem. Let $G' = (V', E')$ be a graph with $|V'| = n$ as in the proof to Theorem 3. We construct a complete weighted graph $G = (V, E, c)$ for ΔDLTSP as follows. Let k be an even number such that

$$k > \frac{1 - \varepsilon}{\varepsilon}(n + 4),$$

$$V := V' \cup \{s, D_1, D_2, D_3, F_1, \dots, F_k\}, \quad \text{and}$$

$$c(e) := \begin{cases} 1 & \text{if} \quad e \in E'; \\ 2 & \text{if} \quad e \in (E')^{\complement}; \\ n & \text{if} \quad e \in \{\{s, t\} \mid t \in V'\} \cup \{\{t, D_1\} \mid t \in V'\} \\ & \qquad \cup \{\{t, D_2\} \mid t \in V'\} \\ & \qquad \cup \{\{D_1, D_2\}, \{D_1, D_3\}, \{D_3, F_{k-1}\}\} \\ & \qquad \cup \{\{F_i, F_{i+2}\} \mid i = 1, \dots, k-2\}; \\ 2n & \text{if} \quad e = \{D_2, D_3\}; \\ (k/2 - 1)n & \text{if} \quad e = \{D_3, F_1\}; \\ \gamma & \text{if} \quad e \in \{\{s, F_k\}\} \cup \{\{F_i, F_{i+1}\} \mid i = 1, \dots, k-1\}, \end{cases}$$

where

$$\gamma > \max\left\{\frac{((1-\varepsilon)/2)(k+n+4)(kn+4n)}{k - (1-\varepsilon)(k+n+4)}, \frac{n}{2}\right\}$$

is a huge number. (By the choice of k, we have $k > (1-\varepsilon)(n+k+4)$, which makes the denominator of the first fraction in this expression positive.) Note that these edge weights satisfy the Δ-inequality. We assign to all other edges the maximal possible costs such that the Δ-inequality remains satisfied. Figure 4 shows the resulting graph. For these values of γ and k, we obtain

$$\frac{k\gamma}{kn + 4n + 2\gamma} > \frac{(1-\varepsilon)}{2}(k+n+4)$$

by a short calculation.

We assign the deadlines as follows:

$$3n \quad \text{to any vertex in } V', \qquad 4n \quad \text{to } D_1, \qquad\qquad 5n \quad \text{to } D_2,$$

$$7n \quad \text{to } D_3, \qquad 7n + \left(\frac{k}{2} - 1\right)n \quad \text{to } F_1.$$
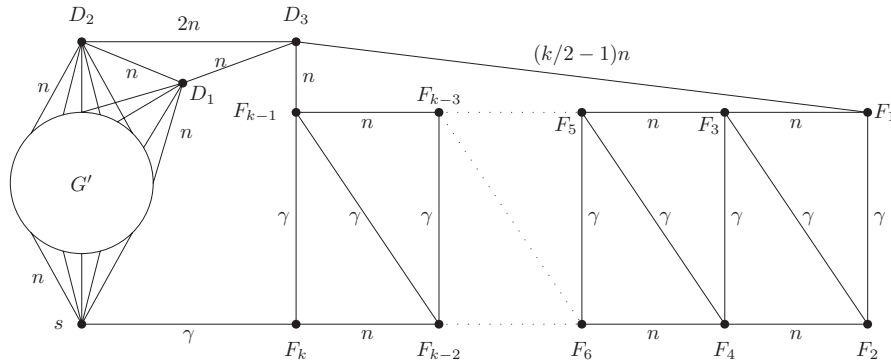

Fig. 4. Construction for ΔDLTSP.

This guarantees that only a solution that finds a Hamiltonian path in G' can visit D_2 before D_1. Hence, it can reach D_3 within time $5n$ whereas every alternative solution reaches D_3 no earlier than $6n + 1$. Furthermore, it is only possible to visit $F_{k-1}, F_{k-3}, \ldots, F_5, F_3$ before F_1 if D_3 is not visited after $6n$. Now, we set the additional deadlines $d(F_{i+1}) = d(F_i) + \gamma$ for $i = 1, \ldots, k-1$.

If G' has a Hamiltonian path, any optimal solution on our input instance visits the vertices $F_{k-1}, F_{k-3}, \ldots, F_5, F_3$ before visiting F_1 and follows the path $F_2, F_4, \ldots$ to F_k, resulting in a cost of

$$5n + n + \left(\tfrac{k}{2} - 1\right) n + \gamma + \left(\tfrac{k}{2} - 1\right) n + \gamma = kn + 4n + 2\gamma. \tag{$\star$}$$

Conversely, if there is no Hamiltonian path in G', any solution is forced to visit F_{k-1}, $F_{k-3}, \ldots, F_5, F_3$ after F_1, which, in any case, is reached between times $6n+1+(k/2-1)n$ and $7n + (k/2 - 1)n$. Thus, it must follow the path $F_2, F_3, \ldots, F_{k-1}, F_k$ in order to meet every deadline. Finally, it must travel back to s. Altogether, this leads to costs of at least

$$6n + 1 + \left(\frac{k}{2} - 1\right) n + k\gamma. \tag{$\star\star$}$$

It is straightforward to divide ($\star\star$) by ($\star$):

$$\frac{6n + 1 + (k/2 - 1)n + k\gamma}{kn + 4n + 2\gamma} > \frac{k\gamma}{kn + 4n + 2\gamma} > \frac{(1 - \varepsilon)}{2}(k + n + 4) = \frac{(1 - \varepsilon)}{2}|V|.$$

(Note that G contains exactly $k + n + 4$ vertices.)

Since algorithms for ΔDLTSP* need to be given one feasible tour as part of their input, let, for some Hamiltonian path P' of $G|_{V'}$, C be the tour $(s, P', D_1, D_2, D_3, F_1, F_2, \ldots, F_k)$. This tour satisfies all of the deadlines and obeys the analysis at ($\star\star$).

Hence, a $((1 - \varepsilon)/2)|V|$-approximation algorithm would need to find a Hamiltonian path in G' in order to guarantee its approximation ratio and could thus be used to solve the HAMILTONIAN PATH problem. $\square$

6. Conclusions

We have successfully outlined a novel technique, i.e., using two established approaches in algorithmics, namely approximation algorithms and parameterized complexity in order to characterize broadly the hardness of a practical problem, TSP with deadlines. In the case of this problem, we have seen that either approach alone falls short of providing us with comparably satisfactory results. It is an interesting question how these methods can be applied to other problems. As to metric TSP with deadlines, it remains an open problem to narrow the gap between the presented lower and upper bounds.

References

[A] T. Andreae: On the traveling salesman problem restricted to inputs satisfying a relaxed triangle inequality. *Networks* **38** (2001), 59–67.

[AB] T. Andreae, H.-J. Bandelt: Performance guarantees for approximation algorithms depending on parameterized triangle inequalities. *SIAM Journal on Discrete Mathematics* **8** (1995), 1–16.

[AR] V. Arvind, V. Raman: Approximation Algorithms for some Parameterized Counting Problems. ECCC TR02-031.

[BBCM] N. Bansal, A. Blum, S. Chawla, A. Meyerson: Approximation algorithms for deadline-TSP and vehicle routing with time windows. *Proc.* 36*th ACM Symposium on the Theory of Computing* (*STOC* '04), pp. 166–174, 2004.

[BC] M. Bender, C. Chekuri: Performance guarantees for TSP with a parametrized triangle inequality. *Information Processing Letters* **73** (2000), 17–21.

[BHK+] 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), 3–24.

[C] 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, PA, 1976.

[CC] L. Cai, J. Chen: On fixed-parameter tractability and approximability of NP optimization problems. *Proc.* 2*nd Israel Symposium Theory of Computing and Systems*, pp. 118–126, 1993.

[CDD+] 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*, pp. 157–193. SIAM Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, PA, 2001.

[CT] M. Cesati and L. Trevisan: On the efficiency of polynomial time approximation schemes. ECCC TR97-001.

[DF] R.G. Downey, M.R. Fellows: *Parameterized Complexity*. Monographs in Computer Science. Springer-Verlag, New York, first edition, 1999.

[FHPS] L. Forlizzi, J. Hromkovič, G. Proietti, S. Seibert: On the stability of approximation for Hamiltonian path problems. *Proc. SOFSEM* 2005: *Theory and Practice of Computer Science*, pp. 147–156. LNCS 3381. Springer, Berlin, 2005.

[GJ] M. R. Garey, D. S. Johnson: *Computers and Intractability—A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA, 1979.

[H1] J. Hromkovič: Stability of approximation algorithms for hard optimization problems. *Proc. SOFSEM* '99, pp. 29–47. LNCS 1725. Springer, Berlin, 1999.

[H2] J. Hromkovič: *Algorithmics for Hard Problems*. *Introduction to Combinatorial Optimization*, *Randomization*, *Approximation*, *and Heuristics*. Springer, Berlin, 2003.

[PS] Ch. Papadimitriou, K. Steiglitz: *Combinatorial Optimization*: *Algorithms and Complexity*. Prentice-Hall, Englewood Cliffs, NJ, 1982.

[PV] Ch. Papadimitriou, S. Vempala: On the approximability of the traveling salesman problem. *Proc.* 32*nd Annual Symposium on Theory of Computing* (*STOC* '00), ACM, New York, 2000. Corrected full version available at http://www.cs.berkeley.edu/∼christos/