RAIRO-Theor. Inf. Appl. 47 (2013) 293–314
DOI: 10.1051/ita/2013040

Available online at:
www.rairo-ita.org

ANALYSIS OF A NEAR-METRIC TSP APPROXIMATION ALGORITHM*

Sacha Krug

Abstract. The traveling salesman problem (TSP) is one of the most fundamental optimization problems. We consider the β-metric traveling salesman problem (Δ_β-TSP), *i.e.*, the TSP restricted to graphs satisfying the β-triangle inequality $c(\{v,w\}) \leq \beta(c(\{v,u\}) + c(\{u,w\}))$, for some cost function c and any three vertices u, v, w. The well-known path matching Christofides algorithm (PMCA) guarantees an approximation ratio of $3\beta^2/2$ and is the best known algorithm for the Δ_β-TSP, for $1 \leq \beta \leq 2$. We provide a complete analysis of the algorithm. First, we correct an error in the original implementation that may produce an invalid solution. Using a worst-case example, we then show that the algorithm cannot guarantee a better approximation ratio. The example can also be used for the PMCA variants for the Hamiltonian path problem with zero and one prespecified endpoints. For two prespecified endpoints, we cannot reuse the example, but we construct another worst-case example to show the optimality of the analysis also in this case.

Mathematics Subject Classification. 90C27, 68W25.

1. Introduction

The traveling salesman problem (TSP) is one of the most studied optimization problems. In its most general form, it is not polynomially approximable.

Keywords and phrases. traveling salesman problem, combinatorial optimization, approximation algorithms, graph theory.

* *This work was partially supported by SNF grant No. 200021-132510/1.*

[1] Department of Computer Science, ETH Zurich, Switzerland. sacha.krug@inf.ethz.ch

Article published by EDP Sciences

© EDP Sciences 2013

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

Certain subsets of the TSP, however, allow for a constant-factor approximation. One such subset is the metric TSP (Δ-TSP), *i.e.*, the TSP restricted to input graphs satisfying the triangle inequality. Christofides' algorithm [7] provides a 3/2-approximation for the Δ-TSP. A natural idea is to apply this algorithm to a larger set of input instances. This idea is captured by the concept of stability of approximation [4–6, 12], which provides a formalism to express the changes of the approximation ratio of an algorithm when a different set of input instances is considered.

A natural generalization of the metric TSP is the β-metric TSP (Δ_β-TSP), *i.e.*, the TSP restricted to graphs (V, E) satisfying the β-triangle inequality $c(\{v, w\}) \leq \beta \cdot (c(\{v, u\}) + c(\{u, w\}))$, for some cost function $c\colon E \to \mathbb{Q}^+$ and any three vertices $u, v, w \in V$. Böckenhauer et al. [5] showed that Christofides' algorithm, when applied to Δ_β-TSP instances, for any $\beta > 1$, no longer provides a constant approximation ratio of 3/2, but an approximation ratio that depends on the size of the graph. Therefore, the authors devised the path matching Christofides algorithm (PMCA) that provides an approximation ratio of $3\beta^2/2$, for any $\beta \geq 1$. Other algorithms for the Δ_β-TSP, for $\beta \geq 1$, are due to Andreae [2] and Bender and Chekuri [3] and provide approximation ratios of $\beta^2 + \beta$ and 4β, respectively. Consider the problem of finding a Hamiltonian path with $l \in \{0, 1, 2\}$ prespecified endpoints in a β-metric graph, Δ_β-HPP$_l$ for short. Forlizzi et al. [9] combined the path matching Christofides algorithm and Hoogeveen's approximation algorithm for the Hamiltonian path problem [11] and obtained an approximation algorithm for Δ_β-HPP$_l$ (PMCA-HPP$_l$) that provides approximation ratios of $3\beta^2/2$, for $l = 0, 1$, and of $5\beta^2/3$, for $l = 2$.

We show that the four PMCA variants cannot provide better approximation ratios. In Section 2, we construct a Δ_β-TSP instance on which the PMCA cannot achieve an approximation ratio of $3\beta^2/2-\varepsilon$, for any $\varepsilon > 0$. This instance can also be used to establish tight lower bounds for the PMCA-HPP$_0$ and the PMCA-HPP$_1$. As we shall see, it is not possible to reuse the example for the PMCA-HPP$_2$. We therefore construct another worst-case instance in Section 3 to prove the optimality of the analysis also in this case.

We use the notation from [5]. Formally, the *β-metric traveling salesman problem* (Δ_β-TSP) is the following optimization problem: Given a complete graph G with edge costs that satisfy the β-triangle inequality, find a cycle in G that visits every vertex exactly once and has minimum cost. A *path matching* for a vertex set V of even size is a set of $|V|/2$ edge-disjoint paths having the vertices in V as their disjoint endpoints. For simplicity, we address a set of paths and, in particular, a path matching forming a tree (forest) as a tree (forest). Let $p = (v_1, v_2, \ldots, v_k)$ be a path. A vertex v is internal to p if $v = v_i$, for $1 < i < k$. A *bypass* in p is an edge $\{u, v\}$ replacing a sub-path $(u = v_i, v_{i+1}, \ldots, v_j = v)$, for $1 \leq i < i + 1 < j \leq k$. We say that the vertices $v_{i+1}, v_{i+2}, \ldots, v_{j-1}$ are *bypassed*. A *conflict* in a set of paths is a vertex that occurs in more than one path. A *conflict* in an Eulerian cycle is a vertex that is visited more than once.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

2. The PMCA for the traveling salesman problem

In this section, we first briefly explain the PMCA (Algorithm 1) and then construct a worst-case example to prove that the approximation ratio of the algorithm cannot be improved.

The implementations of steps 1, 2, and 4 are well-known [5, 8, 10]. In step 4, we need to ensure that each path q in M' occurs unchanged as a subpath in E. In other words, it must not happen in E that there are a few vertices of q, then some path p from T, and then the remaining vertices of q. We can achieve this by regarding each such path q as a single edge when computing E.

Algorithm 2 shows the implementation of step 3. It consists roughly of two parts. First, a path with only one conflict is searched[2], and then this conflict is resolved.

Algorithm 1 Path Matching Christofides Algorithm [5]

Require: A complete β-metric graph G, for some $\beta \geq 1$.

1: Find a minimum spanning tree T in G. Let U be all odd-degree vertices in T.
2: Construct a minimum path matching M for U. {The matching is edge-disjoint due to its minimality.}
3: Resolve conflicts in M and obtain a vertex-disjoint path matching M'.
4: Construct an Eulerian cycle $E := (p_1, q_1, p_2, q_2, \dots)$ on T and M' such that $p_1, p_2, \dots$ are paths in T and $q_1, q_2, \dots$ are paths in M'.
5: Transform $p_1, p_2, \dots$ into $p'_1, p'_2, \dots$ such that the forest T_f formed by $p'_1, p'_2, \dots$ has maximum degree 3. Let $E' := (p'_1, q_1, p'_2, q_2, \dots)$.
6: Resolve all remaining conflicts in E' and obtain a Hamiltonian cycle H.

Ensure: H.

In step 5, an arbitrary root vertex r is chosen. In every path p_i, the vertex v closest to r in T is bypassed if v is internal to p_i and if v is incident to at least four edges from T. This last condition was not stated in [5], but is necessary, as otherwise the algorithm might drop certain vertices, *i.e.*, they might not appear in the end result, which would then by definition no longer be a Hamiltonian cycle.

The last step is implemented as follows. First, bypass an arbitrary conflict x. If neighbors of x are conflicts, bypass one of them. Else, bypass an arbitrary conflict. Repeat this until no conflicts are left.

Theorem 2.1. *For every $\beta \geq 1$, the PMCA provides an approximation ratio of $3\beta^2/2$, and it cannot achieve a better approximation ratio on an infinite family of graphs satisfying the β-triangle inequality.*

The upper bound was shown in [5]. To prove the lower bound, we introduce a graph that contains a Hamiltonian cycle of a certain cost and present one possible implementation of the PMCA on this graph to obtain the desired lower bound.

[2] There is always such a path because the graph formed by M is cycle-free due to the minimality of M.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

Algorithm 2

Require: An edge-disjoint, cycle-free path matching M for U in G.
while M has conflicts **do**
pick an arbitrary path p that has at least two conflicts or at least one internal conflict
while p has more than one conflict **do**
let v, w be the first and the last conflict in p
let p_v, p_w be paths that contain v respectively w
pick as new p one of p_v, p_w that was formerly not picked
end while
let v be the only conflict in the finally chosen path p
if v is internal to p **then**
bypass v in p
else
bypass the unique edge incident to v in p together with one edge of the previously picked path as shown in Figure 1
end if
end while
Ensure: A vertex-disjoint path matching $M' = (q_1, q_2, \dots)$.

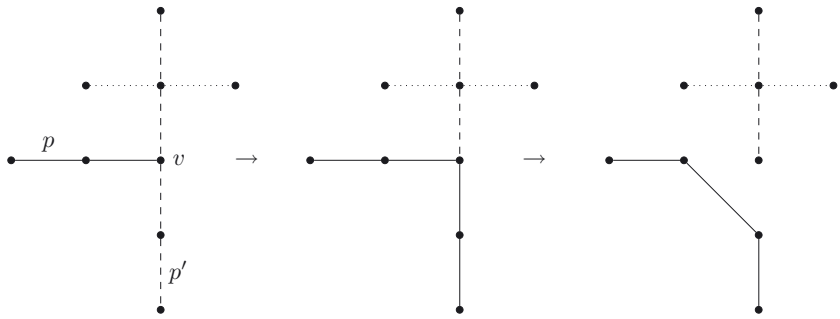


FIGURE 1. Conflict resolution between the finally chosen path p and another path p' that has at least one more conflict.

Let $G_{10,k}(\beta)$ be the complete graph with vertex set $\{v_{i,j} \mid 1 \leq i \leq 10, 1 \leq j \leq k\}$, for $k \in \mathbb{N}$, with edge costs

$$c(v_{1,i}, v_{2,i}) = c(v_{2,i}, v_{5,i}) = c(v_{3,i}, v_{6,i}) = c(v_{4,i}, v_{5,i}) =$$
$$c(v_{5,i}, v_{9,i}) = c(v_{6,i}, v_{7,i}) = c(v_{8,i}, v_{9,i}) = c(v_{9,i}, v_{10,i}) := 1/k,$$
$$c(v_{2,i}, v_{9,j}) = c(v_{5,i}, v_{6,i}) := 1, \qquad c(v_{1,i}, v_{7,j}) := 1 + 2/k,$$

for $i \neq j$, and maximum possible cost for all other edges such that the β-triangle inequality is satisfied. Figure 2 shows the basic structure of the graph. (Only some edges are shown.) Observe that the graph consists of k clusters, each consisting of ten vertices. We denote these clusters by C_i, *i.e.*, $C_i := \{v_{1,i}, v_{2,i}, \dots, v_{10,i}\}$, for $1 \leq i \leq k$. Furthermore, we call the set $\{v_{1,i}, v_{2,i}, v_{4,i}, v_{5,i}, v_{8,i}, v_{9,i}, v_{10,i}\}$ the *upper subcluster* i, denoted USC_i, and the vertex set $\{v_{3,i}, v_{6,i}, v_{7,i}\}$ the *lower*

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

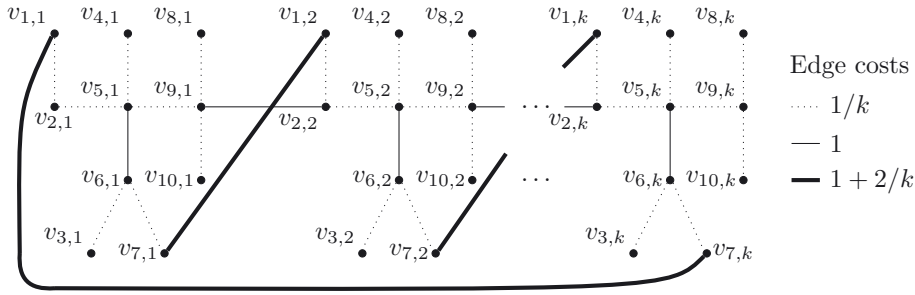


FIGURE 2. The graph $G_{10,k}(\beta)$.

subcluster i, denoted LSC_i. The vertices are called USC vertices and LSC vertices, respectively.

Lemma 2.2. *The graph $G_{10,k}(\beta)$ satisfies the β-triangle inequality.*

Proof. It suffices to show that $G_{10,k}(1)$ satisfies the triangle inequality; $G_{10,k}(\beta)$ then clearly satisfies the β-triangle inequality.

Let $\{u, v, w\}$ be a triangle, and let $e = \{u, v\}$ be the edge in it with highest cost, *i.e.*, the one that could be responsible for a violation. We show that at least one other edge in the triangle has high enough cost such that e does not violate the triangle inequality.

By construction, e can only be one of those edges shown in Figure 2. Furthermore, all edges not shown in Figure 2 have maximum possible cost such that the β-triangle inequality is satisfied; therefore, they all have cost at least $2/k$. All edges in the entire graph thus have cost at least $1/k$, and we have to prove the statement only in the following three cases.

Case 1: $e = \{v_{2,i}, v_{9,j}\}$. If $w \in C_i$, then the shortest path from w to $v_{9,j}$ has cost at least $1 + 1/k$, and thus also the edge $\{w, v_{9,j}\}$. Therefore, $c(v_{2,i}, v_{9,j}) = 1 \leq 1 + 2/k \leq c(v_{2,i}, w) + c(w, v_{9,j})$. If $w \notin C_i$, then the shortest path from $v_{2,i}$ to w has cost at least 1, and thus also the edge $\{v_{2,i}, w\}$. Therefore, $c(v_{2,i}, v_{9,j}) = 1 \leq 1 + 1/k \leq c(v_{2,i}, w) + c(w, v_{9,j})$.

Case 2: $e = \{v_{5,i}, v_{6,i}\}$. If $w \in USC_i$, then the shortest path from w to $v_{6,i}$ has cost at least $1 + 1/k$, and thus also the edge $\{w, v_{6,i}\}$. Therefore, $c(v_{5,i}, v_{6,i}) = 1 \leq 1 + 2/k \leq c(v_{5,i}, w) + c(w, v_{6,i})$. If $w \notin USC_i$, then the shortest path from $v_{5,i}$ to w has cost at least $1 + 1/k$, and thus also the edge $\{v_{5,i}, w\}$. Therefore, $c(v_{5,i}, v_{6,i}) = 1 \leq 1 + 2/k \leq c(v_{5,i}, w) + c(w, v_{6,i})$.

Case 3: $e = \{v_{1,i}, v_{7,j}\}$. If $w \in C_i$, then the shortest path from w to $v_{7,j}$ has cost at least $1 + 3/k$, and thus also the edge $\{w, v_{7,j}\}$. Therefore, $c(v_{1,i}, v_{7,j}) = 1 + 2/k \leq 1 + 4/k \leq c(v_{1,i}, w) + c(w, v_{7,j})$. If $w \notin C_i$, then the shortest path from $v_{1,i}$ to w has cost at least $1 + 1/k$, and thus also the edge $\{v_{1,i}, w\}$. Therefore, $c(v_{1,i}, v_{7,j}) = 1 + 2/k \leq 1 + 2/k) \leq c(v_{1,i}, w) + c(w, v_{7,j})$. □

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

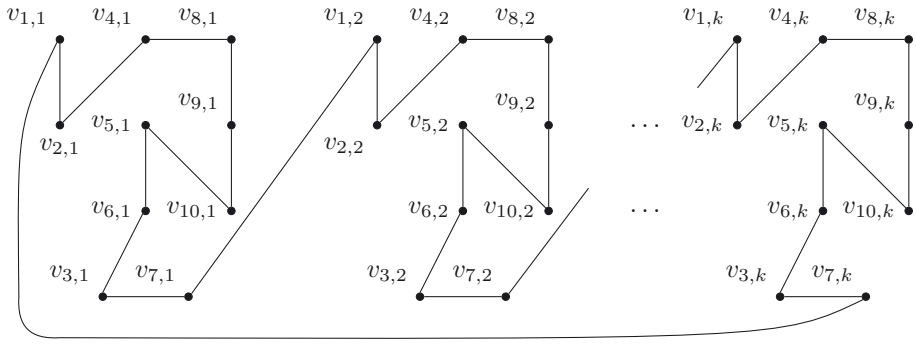


FIGURE 3. A Hamiltonian cycle of length $2k + 2\beta^2 + 7\beta + 6$ in $G_{10,k}(\beta)$.

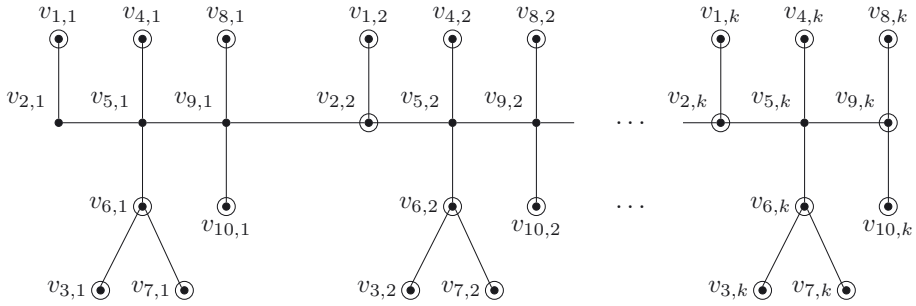


FIGURE 4. A minimum spanning tree in $G_{10,k}(\beta)$. The odd-degree vertices are circled.

Figure 3 shows a Hamiltonian cycle in $G_{10,k}(\beta)$ of cost $2k + 2\beta^2 + 7\beta + 6$. We now show one possible implementation of the PMCA that returns a Hamiltonian cycle of cost at least $3(k-1)\beta^2$ on input $G_{10,k}(\beta)$.

The PMCA first computes the minimum spanning tree T shown in Figure 4. The vertices in U are circled. It is easy to see that T is indeed a minimum spanning tree. Every edge in the graph has cost at least $1/k$, therefore the edges $\{v_{1,i}, v_{2,i}\}, \{v_{2,i}, v_{5,i}\}$, $\{v_{4,i}, v_{5,i}\}, \{v_{5,i}, v_{9,i}\}, \{v_{8,i}, v_{9,i}\}, \{v_{9,i}, v_{10,i}\}$, for $1 \leq i \leq k$, form minimum spanning trees for the respective upper subclusters. On the other hand, the edges $\{v_{3,i}, v_{6,i}\}$ and $\{v_{6,i}, v_{7,i}\}$, for $1 \leq i \leq k$, form minimum spanning trees for the respective lower subclusters. We only need to add an edge for every component to construct a minimum spanning tree for the whole graph. All available edges have cost at least 1, so we can just take the edges $\{v_{5,i}, v_{6,i}\}$, for $1 \leq i \leq k$, and $\{v_{9,i}, v_{2,i+1}\}$, for $1 \leq i \leq k-1$.

In the second step, the PMCA computes the minimum path matching M for U shown in Figure 5.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

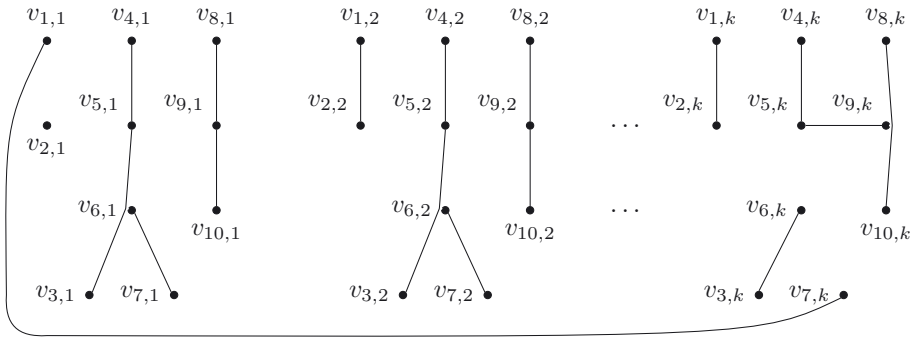


FIGURE 5. A minimum path matching for U in $G_{10,k}(\beta)$.

Theorem 2.3. *M is a minimum path matching for U.*

We need three lemmata to prove this result.

Lemma 2.4. *In a minimum path matching, exactly one vertex per lower subcluster is matched with a vertex that is not in this lower subcluster.*

Proof. The "at least" part holds because $|LSC_i \cap U|$ is odd.

Assume for contradiction that the "at most" part is wrong. Then, some minimum path matching contains the paths $(v_{3,i}, \ldots, u), (v_{6,i}, \ldots, v), (v_{7,i}, \ldots, w)$ for some vertices $u, v, w \notin LSC_i$. We replace them with shortest paths $(v_{3,i}, \ldots, u)$, $(v_{6,i}, v_{7,i}), (v, \ldots, w)$ and obtain a shorter path matching for sufficiently large k.
Case 1: v and w are USC vertices. Then, $c(v_{6,i}, \ldots, v) + c(v_{7,i}, \ldots, w) \geq 2 + 3/k$. The distance from v to w is at most $1+4/k$.[3] Therefore, $c(v_{6,i}, v_{7,i}) + c(v, \ldots, w) \leq 1 + 5/k$.
Case 2: v is a USC vertex and w is an LSC vertex. Then, $c(v_{6,i}, \ldots, v) + c(v_{7,i}, \ldots, w) \geq 3 + 5/k$. The distance from v to w is at most $1 + 8/k$.[4] Therefore, $c(v_{6,i}, v_{7,i}) + c(v, \ldots, w) \leq 1 + 9/k$. The inverse case is analogous.
Case 3: v and w are LSC vertices. Then, $c(v_{6,i}, \ldots, v) + c(v_{7,i}, \ldots, w) \geq 4 + 9/k$. The distance from v to w is at most $2+7/k$.[5] Therefore, $c(v_{6,i}, v_{7,i}) + c(v, \ldots, w) \leq 2 + 8/k$. □

Lemma 2.5. *In a minimum path matching, at most one of the vertices $v_{8,i}$ and $v_{10,i}$ per cluster is matched with an LSC vertex.*

Proof. Assume for contradiction that there are paths $(v_{8,i}, \ldots, v), (v_{10,i}, \ldots, w)$ for some LSC vertices v and w. We replace these two paths with $(v_{8,i}, v_{9,i}, v_{10,i})$ and a

[3] Remember that any two upper subclusters are connected by an edge of cost 1.

[4] Remember that any upper and lower subcluster not belonging to the same cluster are connected by an edge of cost $1 + 2/k$.

[5] In this case, the shortest path from v to w is via some vertex $v_{1,i}$ and contains one edge of cost $1 + 2/k$ and one edge of cost 1.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

shortest path $(v, \ldots, w)$ and obtain a shorter path matching for sufficiently large k. We know from Lemma 2.4 that v and w are not in the same lower subcluster. In particular, at least one of them is not in cluster C_i, which gives us the desired bounds.

Case 1: $v = v_{3,m}, w = v_{3,n}$.

Let us first consider the case $m = i$. Then the shortest path from $v_{8,i}$ to v is of the form

$$(v_{8,i}, v_{9,i}, v_{5,i}, v_{6,i}, v_{3,i})$$

and has cost $1 + 3/k$. The shortest path from $v_{10,i}$ to w is of the form

$$(v_{10,i}, v_{9,i}, v_{5,i}, v_{2,i}, v_{1,i}, v_{7,m}, v_{6,m}, v_{3,m})$$

and has cost $1 + 8/k$. The inverse case is analogous.

If $m \neq i$ and $n \neq i$, both paths have the latter form and thus cost $1 + 8/k$. Therefore, the minimum cost of the two shortest paths $(v_{8,i}, \ldots, v_{3,m})$ and $(v_{10,i}, \ldots, v_{3,n})$ is $2 + 11/k$. We can replace them with the path

$$(v_{3,m}, v_{6,m}, v_{5,m}, v_{2,m}, v_{1,m}, v_{7,n}, v_{6,n}, v_{3,n})$$

of cost $2+7/k$ and $(v_{8,i}, v_{9,i}, v_{10,i})$ of cost $2/k$, and obtain a shorter path matching.

Case 2: $v = v_{3,m}, w = v_{6,n}$. Then, $c(v_{8,i}, \ldots, v) + c(v_{10,i}, \ldots, w) \geq 2 + 10/k$ but $c(v_{8,i}, v_{9,i}, v_{10,i}) + c(v, \ldots, w) \leq 2 + 8/k$. The inverse case is analogous.

Case 3: $v = v_{3,m}, w = v_{7,n}$. Then, $c(v_{8,i}, \ldots, v) + c(v_{10,i}, \ldots, w) \geq 2 + 9/k$ but $c(v_{8,i}, v_{9,i}, v_{10,i}) + c(v, \ldots, w) \leq 2 + 7/k$. The inverse case is analogous.

Case 4: $v = v_{6,m}, w = v_{6,n}$. Then, $c(v_{8,i}, \ldots, v) + c(v_{10,i}, \ldots, w) \geq 2 + 9/k$ but $c(v_{8,i}, v_{9,i}, v_{10,i}) + c(v, \ldots, w) \leq 2 + 7/k$.

Case 5: $v = v_{6,m}, w = v_{7,n}$. Then, $c(v_{8,i}, \ldots, v) + c(v_{10,i}, \ldots, w) \geq 2 + 8/k$ but $c(v_{8,i}, v_{9,i}, v_{10,i}) + c(v, \ldots, w) \leq 2 + 6/k$. The inverse case is analogous.

Case 6: $v = v_{7,m}, w = v_{7,n}$. Then, $c(v_{8,i}, \ldots, v) + c(v_{10,i}, \ldots, w) \geq 2 + 9/k$ but $c(v_{8,i}, v_{9,i}, v_{10,i}) + c(v, \ldots, w) \leq 2 + 6/k$. □

Lemma 2.6. *Every minimum path matching can be transformed into a minimum path matching in which (i) no $v_{8,i}$ is matched with an LSC vertex and (ii) $v_{9,k}$ is matched neither with $v_{8,k}$ nor with $v_{10,k}$.*

Proof. (i) If the matching satisfies condition (i), go to step (ii). Otherwise, let $(v_{8,i}, \ldots, l)$ be one such path, for some LSC vertex l. We show a transformation that does not increase the cost of the matching. We know from Lemma 2.5 that there is a path $(v_{10,i}, \ldots, u)$ for some USC vertex u. These two paths have, without loss of generality, the forms $(v_{8,i}, v_{9,i}, \ldots, l)$ and $(v_{10,i}, v_{9,i}, \ldots, u)$, respectively. We can replace them with shortest paths $(v_{8,i}, v_{9,i}, \ldots, u)$ and $(v_{10,i}, v_{9,i}, \ldots, l)$, *i.e.*, we just "flipped" $v_{8,i}$ and $v_{10,i}$.

(ii) If the modified matching satisfies condition (ii), we are done. Otherwise it contains, without loss of generality, the path $(v_{8,k}, v_{9,k})$ because if it contains $(v_{9,k}, v_{10,k})$, we can just "flip" $v_{8,k}$ and $v_{10,k}$ as above. We replace $(v_{8,k}, v_{9,k})$

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

and $(v_{10,k}, \ldots, v)$ with $(v_{8,k}, v_{9,k}, v_{10,k})$ and a shortest path $(v_{9,k}, \ldots, v)$. We know from the construction of our graph that these two paths have at most the same cost. This transformation does obviously not violate condition (i). □

Now we are able to prove Theorem 2.3.

Proof. Let us first compute the cost of M. It contains $k-1$ paths $(v_{4,i}, v_{5,i}, v_{6,i}, v_{3,i})$ and $(v_{1,j}, v_{2,j})$, for $1 \leq i \leq k-1$ and $2 \leq j \leq k$, of cost $1+2/k$ and $1/k$, respectively, summing up to $(k-1) \cdot (1 + 2/k + 1/k) = k + 2 - 3/k$. It further contains k paths $(v_{8,i}, v_{9,i}, v_{10,i})$ of cost $2/k$ as well as k paths of cost $1/k$ connecting two LSC vertices, summing up to 3. The remaining two paths are $(v_{4,k}, v_{5,k}, v_{9,k})$ and $(v_{1,1}, v_{7,k})$ of cost $2/k$ and $1 + 2/k$, respectively. Thus the cost of M is $k + 2 - 3/k + 3 + 2/k + 1 + 2/k = k + 6 + 1/k$.

Assume thus for contradiction that some minimum path matching M'' has cost less than $k + 6 + 1/k$ and, without loss of generality, satisfies conditions (i) and (ii) of Lemma 2.6. We now compute the cost of M''.

First consider all paths having an LSC vertex as an endpoint. We know from Lemma 2.4 that, for every LSC_i, this means two paths $(l_1, \ldots, l_2)$ and $(l_3, \ldots, v)$, where l_1, l_2, l_3 denote the three vertices in LSC_i. It is possible that $v \in LSC_j$. Let therefore c be the number of paths matching two LSC vertices from different clusters. No matter what vertex we choose as l_3, the two paths always have cost at least $1 + 3/k$. On the other hand, a path connecting two LSC vertices from different clusters C_i and C_j has cost at least $2 + 4/k$, but we still have to consider the paths inside LSC_i and LSC_j, respectively. Summing up, the minimum cost of all paths having one or possibly two LSC vertices as endpoints amounts to

$$k \cdot 1/k + (k - 2c) \cdot (1 + 2/k) + c \cdot (2 + 4/k) = k + 3.$$

Now we compute the cost of all paths having some vertex $v_{8,i}$ as an endpoint. A path of the form $(v_{8,i}, \ldots, v)$, for some vertex $v \neq v_{8,j}$, has minimum cost $2/k$. But we again have to consider the possibility of paths $(v_{8,i}, \ldots, v_{8,j})$. Such a path has minimum cost $1+4/k$. Let d be the number of such paths. Then, the minimum cost of all paths having one or possibly two vertices $v_{8,i}, v_{8,j}$ as endpoints amounts to

$$(k - 2d) \cdot 2/k + d \cdot (1 + 4/k) = 2 + d \geq 2.$$

Observe that the two path sets considered above are disjoint due to condition (i) of Lemma 2.6.

Now we consider all remaining paths. The two path sets above contain together at most $3k$ paths, so we are left with at least k paths. Every path has cost at least $1/k$. But there is more to it. All possible paths of cost $1/k$ have the form $(v_{3,i}, v_{6,i}), (v_{6,i}, v_{7,i})$, for $1 \leq i \leq k$, or $(v_{1,j}, v_{2,j})$, for $2 \leq j \leq k$, or $(v_{8,k}, v_{9,k})$ or $(v_{9,k}, v_{10,k})$. We already considered above all paths of the first two forms, and the paths of the last two forms cannot occur in M'' due to condition (ii) of Lemma 2.6. Therefore, only the $k-1$ paths $(v_{1,j}, v_{2,j})$, for $2 \leq j \leq k$, are left. Since every other

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

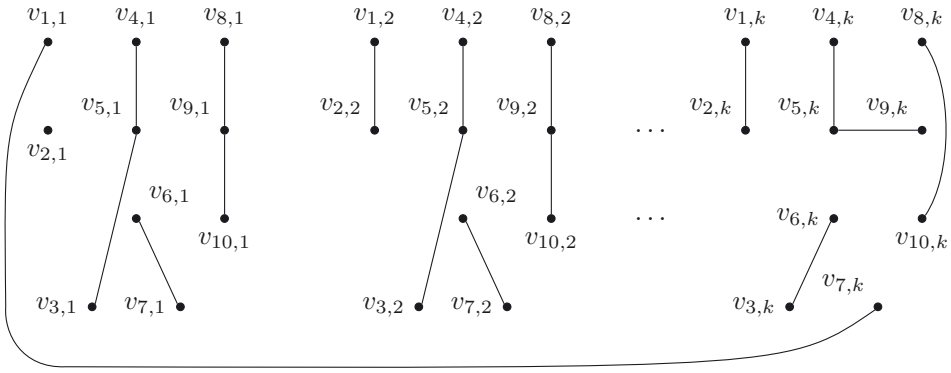


FIGURE 6. A vertex-disjoint path matching for U in $G_{10,k}(\beta)$.

path has cost at least $2/k$, the minimum cost of all remaining paths amounts to $1 + 1/k$.

We have shown that M'' has minimum cost

$$k + 3 + 2 + 1 + 1/k = k + 6 + 1/k,$$

which contradicts our assumption. □

The goal of the third step of Algorithm 1 is to resolve all conflicts in the minimum path matching M and obtain a vertex-disjoint path matching M'. The problematic paths in M are $(v_{4,i}, v_{5,i}, v_{6,i}, v_{3,i})$ and $(v_{6,i}, v_{7,i})$, for $1 \leq i \leq k-1$, as well as $(v_{4,k}, v_{5,k}, v_{9,k})$ and $(v_{8,k}, v_{9,k}, v_{10,k})$. For each component in the first set, the PMCA may choose the path $(v_{4,i}, v_{5,i}, v_{6,i}, v_{3,i})$ as p and thus bypass $v_{6,i}$ in this path. For the second set, the PMCA may choose the path $(v_{8,k}, v_{9,k}, v_{10,k})$ as p and thus bypass $v_{9,k}$ in this path. In this step, the PMCA thus computes the vertex-disjoint path matching M' shown in Figure 6.

Alternating between paths from T and paths from M', the PMCA computes in the fourth step the Eulerian cycle E shown in Figure 7.

The goal of the fifth step is that every vertex is incident to at most three edges from T. The problematic vertices are thus all $v_{5,i}$ and all $v_{9,i}$ except $v_{9,k}$. Let $r := v_{1,1}$. The PMCA bypasses the vertices $v_{5,i}$ between $v_{6,i}$ and $v_{9,i}$, for $1 \leq i \leq k-1$, the vertex $v_{5,k}$ between $v_{9,k}$ and $v_{6,k}$, and the vertices $v_{9,i}$ between $v_{8,i}$ and $v_{2,i+1}$, for $1 \leq i \leq k-1$, and obtains the modified Eulerian cycle E' shown in Figure 8.

The goal of the last step is that every vertex has degree 2. The problematic vertices are all vertices $v_{2,i}$ except $v_{2,1}$ and all vertices $v_{5,i}, v_{6,i}, v_{9,i}$. The PMCA obtains the Hamiltonian cycle H shown in Figure 9 by bypassing

- $v_{2,i}$ between $v_{8,i-1}$ and $v_{1,i}$, for $2 \leq i \leq k$;

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

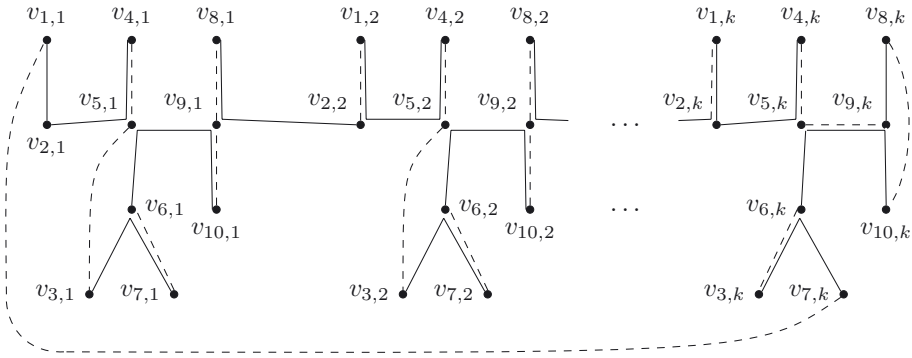


FIGURE 7. The Eulerian cycle E. The paths of M' are dashed.

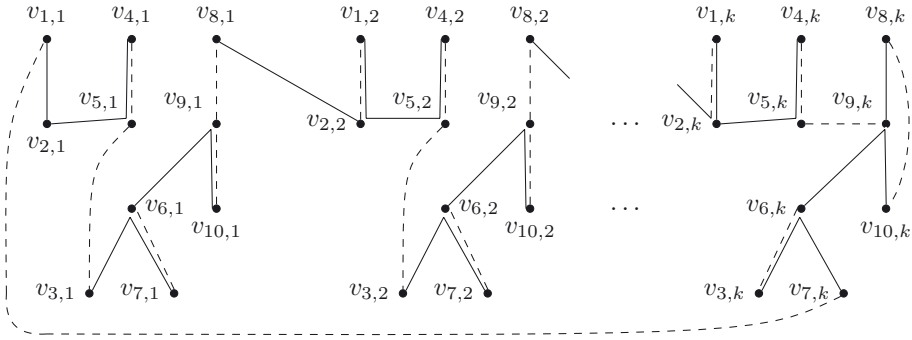


FIGURE 8. The modified Eulerian cycle E'. The paths of M' are dashed.

- $v_{5,i}$ between $v_{4,i}$ and $v_{3,i}$, for $1 \leq i \leq k-1$, and $v_{5,k}$ between $v_{4,k}$ and $v_{9,k}$;
- $v_{6,i}$ between $v_{7,i}$ and $v_{9,i}$, for $1 \leq i \leq k-1$, and $v_{6,k}$ between $v_{3,k}$ and $v_{7,k}$;
- $v_{9,i}$ between $v_{10,i}$ and $v_{8,i}$, for $1 \leq i \leq k-1$, and $v_{9,k}$ between $v_{10,k}$ and $v_{6,k}$.

Considering only the edges $\{v_{4,i}, v_{3,i}\}, \{v_{7,i}, v_{9,i}\}, \{v_{8,i}, v_{1,i+1}\}$, for $1 \leq i \leq k-1$, we obtain $\text{cost}(H) \geq 3(k-1)\beta^2$. For every $\beta \geq 1$ and arbitrarily small $\epsilon > 0$, we have thus shown that there is an implementation I of the PMCA such that

$$\frac{\text{cost}(I(G_{10,k}(\beta)))}{\text{Opt}_{\Delta_\beta-\text{TSP}}(G_{10,k}(\beta))} \geq \frac{3(k-1)\beta^2}{2k+2\beta^2+7\beta+6} \geq \frac{3}{2}\beta^2 - \varepsilon, \tag{2.1}$$

for sufficiently large k, *i.e.*, we have shown that the upper bound of $3\beta^2/2$ on the approximation ratio of the PMCA is tight.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

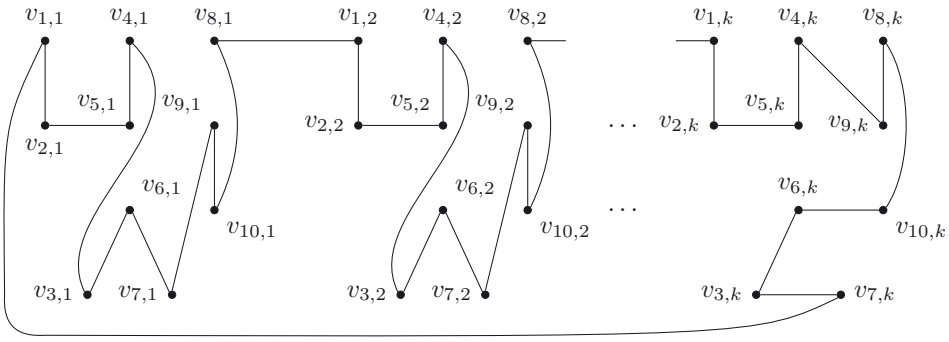


FIGURE 9. The Hamiltonian cycle H.

3. The PMCA for the Hamiltonian path problem

In this section, we analyze the PMCA variant for the β-metric Hamiltonian path problem with $l \in \{0, 1, 2\}$ prespecified endpoints (Δ_β-HPP$_l$) devised by Forlizzi et al. [9]. Formally, Δ_β-HPP$_l$ is the following optimization problem: Given a complete graph G with edge costs that satisfy the β-triangle inequality, find a path in G that visits every vertex exactly once, has minimum cost and starts and ends in the prespecified endpoints, if any.

We first briefly explain the PMCA-HPP$_l$ (Algorithm 3) and then construct a worst-case example to prove that the approximation ratio of the algorithm cannot be improved for $l = 2$.

The implementation of step 1 is well-known. Step 2 is implemented using the distance graph $d(G)$ of G, *i.e.*, the complete weighted graph with vertex set $V(G)$ in which the cost of an edge $\{u, v\}$ is the cost of a shortest path from u to v in G. The algorithm adds $2 - l$ dummy vertices to $d(G)$ with all edges incident to them having cost 0 except the edge connecting them having cost ∞. The algorithm computes a minimum perfect matching for U and the dummy vertices in $d(G)$, removes the edges incident to the dummy vertices from the matching and maps it back to G by connecting two vertices with a shortest path if they are matched in $d(G)$. After potentially removing an additional edge, the graph contains exactly two odd-degree vertices w and z.

The implementation of step 3 can be found in [9]. Essentially, it resolves the conflicts in M for each tree in M separately. If the tree contains z, the algorithm ensures that z is contained in the resulting forest.

In step 4, we distinguish two cases. Let y be the unique neighbor of w towards z in T. If there is a path $p = (z, \ldots, w)$ in M', we construct an Eulerian cycle $E = (w, u_1, \ldots, u_{h-1}, w)$ on T and $M' - \{p\}$, concatenate p and E and obtain the Eulerian path $(z, \ldots, w, u_1, \ldots, u_{h-1}, w)$. If there is no path $(z, \ldots, w)$, we essentially look if there are (unique) paths $p = (z, \ldots, u)$ and $q = (u', \ldots, w)$, and

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

if so, search for an Eulerian path P from u to u' and concatenate p, P, and q. The detailed implementation can be found in [9].

Step 5 is implemented in the same way as in the PMCA, except that we do not choose r arbitrarily, but set $r := z$. Step 6 is also implemented in the same way as in the PMCA, except that we first bypass w if it is a conflict.

Theorem 3.1. *For every $\beta \geq 1$, both the PMCA-HPP$_0$ and the PMCA-HPP$_1$ provide an approximation ratio of $3\beta^2/2$, and they cannot achieve a better approximation ratio on an infinite family of graphs satisfying the β-triangle inequality.*

Theorem 3.2. *For every $\beta \geq 1$, the PMCA-HPP$_2$ provides an approximation ratio of $5\beta^2/3$, and it cannot achieve a better approximation ratio on an infinite family of graphs satisfying the β-triangle inequality.*

Algorithm 3 PMCA-HPP$_l$

Require: A complete β-metric graph $G = (V, E)$, for some $\beta \geq 1$, and a set $A \subseteq V$ of size l.

1: Find a minimum spanning tree T in G. Let U be the vertices in $V - A$ having odd degree in T plus the vertices in A having even degree in T.
2: Construct a minimum path matching M for U. If necessary, remove an edge from T such that the multigraph formed by T and M has two odd-degree vertices w, z.
3: Resolve conflicts in M and obtain a vertex-disjoint path matching M' that does not contain z as an inner vertex.
4: Construct an Eulerian path $P = (p_1, q_1, p_2, q_2, \ldots)$ on T and M' from w to z such that $p_1, p_2, \ldots$ are paths in T and $q_1, q_2, \ldots$ are paths in M'.
5: Transform $p_1, p_2, \ldots$ into $p'_1, p'_2, \ldots$ such that the forest T_f formed by $p'_1, p'_2, \ldots$ has degree at most 3, w and z are the endpoints of $P' := (p'_1, q_1, p'_2, q_2, \ldots)$ and z is not a conflict in P'.
6: Resolve all remaining conflicts in P' and obtain a Hamiltonian path P''.

Ensure: P''.

The upper bounds were established in [9]. To prove the lower bounds for $l = 0, 1$, the graph $G_{10,k}(\beta)$ can be reused. The proofs are for the most part quite similar to the one of Theorem 2.1. For the PMCA-HPP$_1$, the prespecified endpoint is $v_{1,1}$. Both proofs can be found in [13]. To prove the lower bound for $l = 2$, however, we cannot reuse the graph $G_{10,k}(\beta)$, as we shall see.

For $\beta = 1$, observe that some implementation of the PMCA is an implementation of the Christofides algorithm. This implementation does not construct a path matching in the second step, but a normal matching.[6] For $\beta = 1$, observe further that some implementation of the PMCA-HPP$_2$ is an implementation of Hoogeveen's algorithm. As above, it is necessary always to construct a matching instead of a path matching.

[6] The condition $\beta = 1$ ensures no path matching is shorter than a minimum matching.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

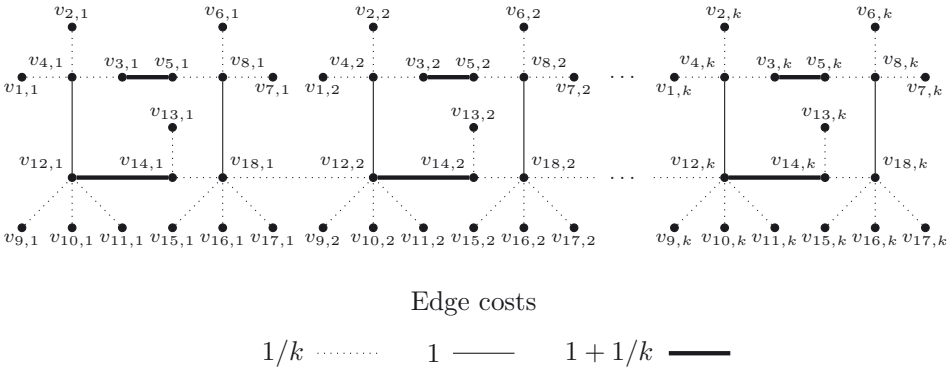


FIGURE 10. The graph $G_{18,k}(\beta)$.

Assume that we could use the graph $G_{10,k}(\beta)$ to establish a lower bound of $5\beta^2/3-\varepsilon$ on the approximation ratio of the PMCA-HPP$_2$. In particular, we could show that some implementation achieves an approximation ratio of $5/3$ on the graph $G_{10,k}(1)$. This would contradict the fact that the sets of worst-case instances for the metric TSP and the metric HPP$_2$ are disjoint [14].

To prove the lower bound of $5\beta^2/3-\varepsilon$, we thus introduce a new graph. Let $G_{18,k}(\beta)$ be the complete graph with vertex set $\{v_{i,j} \mid 1 \le i \le 18, 1 \le j \le k\}$, for $k \in \mathbb{N}$, with edge costs

$$\begin{aligned}
c(v_{1,i},v_{4,i}) = c(v_{2,i},v_{4,i}) = c(v_{3,i},v_{4,i}) = c(v_{5,i},v_{8,i}) = \\
c(v_{6,i},v_{8,i}) = c(v_{7,i},v_{8,i}) = c(v_{9,i},v_{12,i}) = c(v_{10,i},v_{12,i}) = \\
c(v_{11,i},v_{12,i}) = c(v_{13,i},v_{14,i}) = c(v_{14,i},v_{18,i}) = c(v_{15,i},v_{18,i}) = \\
c(v_{16,i},v_{18,i} = c(v_{17,i},v_{18,i}) = c(v_{18,j},v_{12,j+1}) := 1/k, \\
c(v_{4,i},v_{12,i}) = c(v_{8,i},v_{18,i}) := 1, \\
c(v_{3,i},v_{5,i}) = c(v_{12,i},v_{14,i}) := 1+1/k,
\end{aligned}$$

for $1 \le i \le k, 1 \le j \le k-1$, and maximum possible cost for all other edges such that the β-triangle inequality is satisfied. Figure 10 shows the basic structure of the graph. (Only some edges are shown). Observe that the graph consists of k clusters, each consisting of 18 vertices. We denote these clusters by C_i, *i.e.*, $C_i := \{v_{1,i}, v_{2,i}, \ldots, v_{18,i}\}$, for $1 \le i \le k$. Furthermore, we call the set $\{v_{1,i}, v_{2,i}, v_{3,i}, v_{4,i}\}$ the *upper left subcluster* i, denoted $ULSC_i$, the vertex set $\{v_{5,i}, v_{6,i}, v_{7,i}, v_{8,i}\}$ the *upper right subcluster* i, denoted $URSC_i$, and the vertex set

$$\{v_{13,j}, v_{14,j}, v_{15,j}, v_{16,j}, v_{17,j}, v_{18,j}, v_{9,j+1}, v_{10,j+1}, v_{11,j+1}, v_{12,j+1}\}$$

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

the *lower subcluster j,* denoted LSC_j, for $1 \leq i \leq k$, $1 \leq j \leq k-1$. Furthermore, let $v_{13,k}, v_{14,k}, v_{15,k}, v_{16,k}, v_{17,k}, v_{18,k}$ be the lower subcluster k, denoted LSC_k.[7]

Lemma 3.3. *The graph $G_{18,k}(\beta)$ satisfies the β-triangle inequality.*

Proof. As above, we only show that $G_{18,k}(1)$ satisfies the triangle inequality.

Let $\{u, v, w\}$ be a triangle, and let $e = \{u, v\}$ be the edge with in it with highest cost, *i.e.*, the one that could be responsible for a violation. We show that at least one other edge in the triangle has high enough cost such that e does not violate the triangle inequality.

By construction, e can one of those edges shwon in Figure 10. Furthermore, all other edges have cost at least $2/k$. All edges in the entire graph thus have cost at least $1/k$, and we have to prove the statement only in the following four cases.

Case 1: $e = \{v_{3,i}, v_{5,i}\}$. If $w \in ULSC_i$, then the shortest path from w to $v_{5,i}$ has cost at least 1, and thus also the edge $\{w, v_{5,i}\}$. Therefore, $c(v_{3,i}, v_{5,i}) = 1 + 1/k \leq 1/k + 1 \leq c(v_{3,i}, w) + c(w, v_{5,i})$. Otherwise, the shortest path from $v_{3,i}$ to w has cost at least 1, and thus also the edge $\{v_{3,i}, w\}$. Therefore, $c(v_{3,i}, v_{5,i}) = 1 + 1/k \leq 1 + 1/k \leq c(v_{3,i}, w) + c(w, v_{5,i})$.

Case 2: $e = \{v_{4,i}, v_{12,i}\}$. If $w \in ULSC_i$, then the shortest path from w to $v_{12,i}$ has cost at least 1, and thus also the edge $\{w, v_{12,i}\}$. Therefore, $c(v_{4,i}, v_{12,i}) = 1 \leq 1/k + 1 \leq c(v_{4,i}, w) + c(w, v_{12,i})$. Otherwise, the shortest path from $v_{4,i}$ to w has cost at least 1, and thus also the edge $\{v_{4,i}, w\}$. Therefore, $c(v_{4,i}, v_{12,i}) = 1 \leq 1 + 1/k \leq c(v_{4,i}, w) + c(w, v_{12,i})$.

Case 3: $e = \{v_{8,i}, v_{18,i}\}$. If $w \in URSC_i$, then the shortest path from w to $v_{18,i}$ has cost at least 1, and thus also the edge $\{w, v_{18,i}\}$. Therefore, $c(v_{8,i}, v_{18,i}) = 1 \leq 1/k + 1 \leq c(v_{8,i}, w) + c(w, v_{18,i})$. Otherwise, the shortest path from $v_{8,i}$ to w has cost at least 1, and thus also the edge $\{v_{8,i}, w\}$. Therefore, $c(v_{8,i}, v_{18,i}) = 1 \leq 1 + 1/k \leq c(v_{8,i}, w) + c(w, v_{18,i})$.

Case 4: $e = \{v_{12,i}, v_{14,i}\}$. If $w \in LSC_i$, then the shortest path from $v_{12,i}$ to w has cost at least 1, and thus also the edge $\{v_{12,i}, w\}$. Therefore, $c(v_{12,i}, v_{14,i}) = 1 + 1/k \leq 1 + 1/k \leq c(v_{12,i}, w) + c(w, v_{14,i})$. Otherwise, the shortest path from w to $v_{14,i}$ has cost at least 1, and thus also the edge $\{w, v_{14,i}\}$. Therefore, $c(v_{12,i}, v_{14,i}) = 1 + 1/k \leq 1/k + 1 \leq c(v_{12,i}, w) + c(w, v_{14,i})$. □

Figure 11 shows a Hamiltonian path from $v_{9,1}$ to $v_{17,k}$ in $G_{18,k}(\beta)$ of cost $3k + 2\beta^2 + 21\beta + 5 - (2\beta^2 + \beta)/k$. We show one possible implementation of the PMCA-HPP$_2$ that returns a Hamiltonian path of cost at least $5(k-1)\beta^2$ from $v_{9,1}$ to $v_{17,k}$ on inputs $G_{18,k}(\beta)$ and $A := \{v_{9,1}, v_{17,k}\}$.

The PMCA first computes the minimum spanning tree shown in Figure 12. The vertices in U are circled. It is easy to see that T is indeed a minimum spanning tree. Every edge in the graph has cost at least $1/k$, therefore the edges of cost $1/k$ incident to $v_{4,i}$, $v_{8,i}$, for $1 \leq i \leq k$, $v_{13,i}, v_{18,i}, v_{12,i+1}$, for $1 \leq i \leq k-1$, and $v_{12,1}$, form minimum spanning trees for the upper left, upper right, and lower

[7] Note that, for $1 \leq j \leq k-1$, the vertex set LSC_j contains vertices from both C_j and C_{j+1}, so in particular $LSC_j \not\subseteq C_j$.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

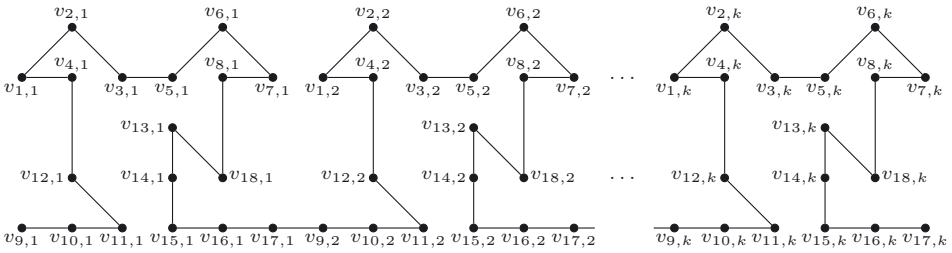


FIGURE 11. A Hamiltonian path in $G_{18,k}(\beta)$ from $v_{9,1}$ to $v_{17,k}$ of length $3k + 2\beta^2 + 21\beta + 5 - \frac{2\beta^2+\beta}{k}$.

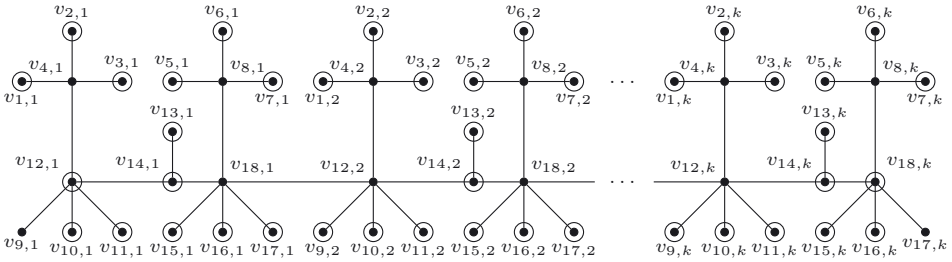


FIGURE 12. A minimum spanning tree in $G_{18,k}(\beta)$. The vertices in U are circled.

subcluster i, respectively.[8] We only have to add an edge for every component such that it is connected to the rest in the end. All available edges have cost at least 1, so we can just take the edges $\{v_{4,i}, v_{12,i}\}$ and $\{v_{8,i}, v_{18,i}\}$, for $1 \leq i \leq k$. After that, all available edges to connect the different components have cost at least $1 + 1/k$, so we can just take the edges $\{v_{12,i}, v_{14,i}\}$, for $1 \leq i \leq k$.

In the second step, the PMCA computes the minimum path matching M for U shown in Figure 13 and sets $w := v_{17,k}, z := v_{9,1}$.

Theorem 3.4. *M is a minimum path matching for U.*

Assume for contradiction that some minimum path matching M'' for U has cost less than $2k + 13 - 2/k$.

Lemma 3.5. *Without loss of generality, M'' contains the paths $(v_{13,i}, v_{14,i})$, for $1 \leq i \leq k$.*

Proof. Assume that two vertices $v_{13,i}, v_{14,i}$ are not matched with each other in M''. The shortest path with $v_{13,i}$ as an endpoint has, without loss of generality, the form

[8] The same holds of course for the edges $\{v_{9,1}, v_{12,1}\}$, $\{v_{10,1}, v_{12,1}\}$, $\{v_{11,1}, v_{12,1}\}$, and the vertex set $\{v_{9,1}, v_{10,1}, v_{11,1}, v_{12,1}\}$.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

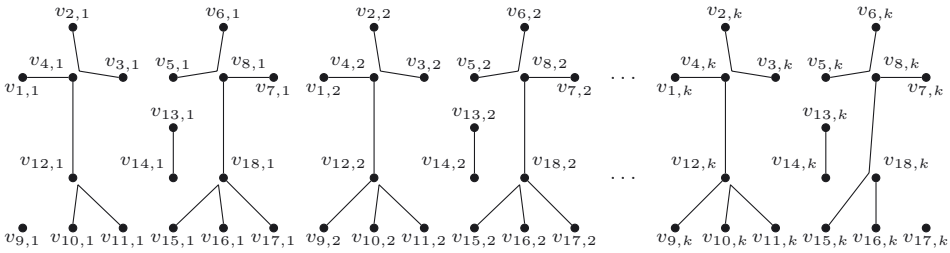


FIGURE 13. A minimum path matching for U in $G_{18,k}(\beta)$.

$(v_{13,i}, v_{14,i}, \ldots, u)$. Let v be the vertex matched with $v_{14,i}$. Replace the two paths with $(v_{13,i}, v_{14,i})$ and $(u, \ldots, v_{14,i}, \ldots, v)$ of the same cost. □

Lemma 3.6. *Without loss of generality, $v_{10,i}$ is matched with $v_{11,i}$ and $v_{15,i}$ with $v_{16,i}$ in M'', for $1 \leq i \leq k$.*

Proof. We first show that two of the three vertices $v_{9,i}, v_{10,i}, v_{11,i}$ are matched with each other, for $2 \leq i \leq k$.

Assume for contradiction that three different paths have one of the three vertices as an endpoint. These three paths have, without loss of generality, the form $(v_{9,i}, v_{12,i}, \ldots, v), (v_{10,i}, v_{12,i}, \ldots, v'), (v_{11,i}, v_{12,i}, \ldots, v'')$. We can replace the first two paths with the paths $(v_{9,i}, v_{12,i}, v_{10,i})$ and $(v, \ldots, v_{12,i}, \ldots, v')$ of the same cost. A similar argument shows that two of the three vertices $v_{10,1}, v_{11,1}, v_{12,1}$ are matched with each other.

Assume now for contradiction that $v_{10,1}$ or $v_{11,1}$ is matched with $v_{12,1}$. As in the proof of Lemma 2.6, we can flip the endpoints of the two paths having $v_{10,1}$ and $v_{11,1}$, respectively, as endpoints and obtain a path matching that contains the path $(v_{10,1}, v_{12,1}, v_{11,1})$.

For $i > 1$, assume for contradiction that $v_{9,i}$ is matched with $v_{10,i}$ or $v_{11,i}$ in M'', *i.e.*, M'' contains, without loss of generality, the path $(v_{9,i}, v_{12,i}, v_{10,i})$ or $(v_{9,i}, v_{12,i}, v_{11,i})$ and the path $(v_{11,i}, v_{12,i}, \ldots, v)$ or $(v_{10,i}, v_{12,i}, \ldots, v)$, respectively. We can again flip the endpoints and obtain the paths $(v_{10,i}, v_{12,i}, v_{11,i})$ and $(v_{9,i}, v_{12,i}, \ldots, v)$ of the same cost.

The proof is analogous for $v_{15,i}$ and $v_{16,i}$. □

Lemma 3.7. *Without loss of generality, $v_{1,i}$ is matched with $v_{2,i}$ and $v_{6,i}$ with $v_{7,i}$ in M'', for $1 \leq i \leq k$.*

Proof. The proof is similar to the previous one. As above, we prove the statement only for $v_{1,i}$ and $v_{2,i}$. The proof is analogous for $v_{6,i}$ and $v_{7,i}$.

Again, we first show that two of the three vertices $v_{1,i}, v_{2,i}, v_{3,i}$ are matched with each other.

Assume for contradiction three different paths have one of the three vertices as an endpoint. These paths have, without loss of generality, the form

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

$(v_{1,i}, v_{4,i}, \ldots, u)$, $(v_{2,i}, v_{4,i}, \ldots, v)$, and $(v_{3,i}, \ldots, w)$. (Observe that we cannot assume, without loss of generality, that the third path contains $v_{4,i}$, as *e.g.* the shortest path from $v_{3,i}$ to $v_{5,i}$ only consists of the edge $\{v_{3,i}, v_{5,i}\}$.) We can replace the first two paths with the paths $(v_{1,i}, v_{4,i}, v_{2,i})$ and $(u, \ldots, v_{4,i}, \ldots, v)$ of the same cost.

Assume for contradiction that $v_{1,i}$ or $v_{2,i}$ is matched with $v_{3,i}$ in M'', *i.e.*, M'' contains, without loss of generality, the path $(v_{1,i}, v_{4,i}, v_{3,i})$ or $(v_{2,i}, v_{4,i}, v_{3,i})$ and the path $(v_{2,i}, v_{4,i}, \ldots, v)$ or $(v_{1,i}, v_{4,i}, \ldots, v)$, respectively. We can replace these two paths with the paths $(v_{1,i}, v_{4,i}, v_{2,i}), (v_{3,i}, v_{4,i}, \ldots, v)$ of the same cost. □

The three proofs above do not interfere with each other, *i.e.*, we can indeed obtain a minimum path matching that satisfies the conditions of all three lemmata by applying the transformations described in the respective proofs. In other words, M'' contains, without loss of generality, all the paths $(v_{1,i}, v_{4,i}, v_{2,i}), (v_{6,i}, v_{8,i}, v_{7,i})$, $(v_{10,i}, v_{12,i}, v_{11,i})$, $(v_{13,i}, v_{14,i}), (v_{15,i}, v_{18,i}, v_{16,i})$.

The remaining unmatched vertices in U are

$$\{v_{3,i}, v_{5,i} \mid 1 \le i \le k\} \cup \{v_{12,1}, v_{18,k}\} \cup \{v_{9,i} \mid 2 \le i \le k\} \cup \{v_{17,i} \mid 1 \le i \le k-1\},$$

i.e., exactly four vertices per cluster remain unmatched.

Lemma 3.8. *M'' contains no path with endpoints in different clusters.*

Proof. We use induction. Assume for contradiction that two vertices in C_1 are matched with vertices outside C_1. Without loss of generality, they both use the edge $\{v_{18,1}, v_{12,2}\}$. Therefore, M'' is not edge-disjoint and thus not minimal. The same argument can be applied inductively to every cluster. □

So we only have to find a minimum path matching for each cluster, *i.e.*, for the vertices $v_{3,1}, v_{5,1}, v_{12,1}, v_{17,1}$ in C_1, for the vertices $v_{3,i}, v_{5,i}, v_{9,i}, v_{17,i}$ in the clusters 2 to $k-1$, and for the vertices $v_{3,k}, v_{5,k}, v_{9,k}, v_{18,k}$ in C_k. One can easily see that the path matching

$$\{(v_{3,1}, v_{4,1}, v_{12,1}), (v_{5,1}, v_{8,1}, v_{18,1}, v_{17,1})\} \cup \{(v_{3,i}, v_{4,i}, v_{12,i}, v_{9,i}) \mid i \ge 2\} \cup \{(v_{5,i}, v_{8,i}, v_{18,i}, v_{17,i}) \mid 2 \le i \le k-1\} \cup \{(v_{5,k}, v_{8,k}, v_{18,k})\}$$

is such a minimum path matching, and therefore M'' contains, without loss of generality, exactly these paths. They have cost

$$1 + 1/k + 1 + 2/k + (k-1) \cdot (1 + 2/k) + (k-2) \cdot (1 + 2/k) + 1 + 1/k = 2k + 4 - 2/k.$$

Let us now compute the cost of M''. It contains k paths $(v_{13,i}, v_{14,i})$ of cost $1/k$, $2k$ paths $(v_{10,i}, v_{12,i}, v_{11,i})$ and $(v_{15,i}, v_{18,i}, v_{16,i})$ of cost $2/k$ each, another $2k$ paths

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

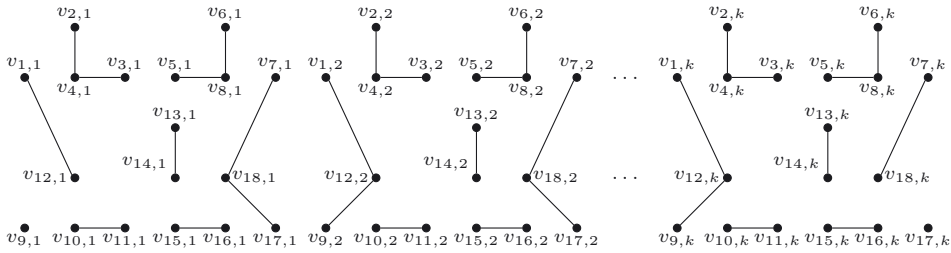


FIGURE 14. A vertex-disjoint path matching for U in $G_{18,k}(\beta)$.

$(v_{1,i}, v_{4,i}, v_{2,i})$ and $(v_{6,i}, v_{8,i}, v_{7,i})$ of cost $2/k$ each, and the path matching described in the previous paragraph of cost $2k + 4 - 2/k$. In total, we get

$$k \cdot 1/k + 2k \cdot 2/k + 2k \cdot 2/k + 2k + 4 - 2/k = 2k + 13 - 2/k,$$

which contradicts our assumption. This concludes the proof of Theorem 3.4.

The goal of the third step is to resolve all conflicts in the minimum path matching M in such a way that $z = v_{9,1}$ is still contained in the resulting vertex-disjoint path matching M'. The PMCA-HPP$_2$ does this for every connected component of M separately. The paths $\{(v_{13,i}, v_{14,i}) \mid 1 \leq i \leq k\}$ contain no conflicts. Let us therefore now look at the problematic paths of M, *i.e.*,

$$\begin{aligned}
&\{(v_{1,1}, v_{4,1}, v_{12,1}), (v_{2,1}, v_{4,1}, v_{3,1}), (v_{10,1}, v_{12,1}, v_{11,1})\} \cup \\
&\{(v_{5,i}, v_{8,i}, v_{6,i}), (v_{7,i}, v_{8,i}, v_{18,i}, v_{17,i}), (v_{15,i}, v_{18,i}, v_{16,i}) \mid 1 \leq i \leq k-1\} \cup \\
&\{(v_{1,i}, v_{4,i}, v_{12,i}, v_{9,i}), (v_{2,i}, v_{4,i}, v_{3,i}), (v_{10,i}, v_{12,i}, v_{11,i}) \mid 2 \leq i \leq k\} \cup \\
&\{(v_{5,k}, v_{8,k}, v_{6,k}), (v_{7,k}, v_{8,k}, v_{18,k}, v_{15,k}), (v_{16,k}, v_{18,k})\}. \qquad (3.1)
\end{aligned}$$

In the first set, the PMCA-HPP$_2$ bypasses $v_{2,1}$ in the path $(v_{10,1}, v_{12,1}, v_{11,1})$ and $v_{4,1}$ in the path $(v_{1,1}, v_{4,1}, v_{12,1})$. In the second set, it bypasses $v_{18,i}$ in the path $(v_{15,i}, v_{18,i}, v_{16,i})$ and $v_{8,i}$ in the path $(v_{7,i}, v_{8,i}, v_{18,i}, v_{17,i})$. In the third set, it bypasses $v_{12,i}$ in the path $(v_{10,i}, v_{12,i}, v_{11,i})$ and $v_{4,i}$ in the path $(v_{1,i}, v_{4,i}, v_{12,i}, v_{9,i})$. In the fourth set, it transform the two paths $(v_{7,k}, v_{8,k}, v_{18,k}, v_{15,k}), (v_{16,k}, v_{18,k})$ into the paths $(v_{15,k}, v_{16,k}), (v_{7,k}, v_{8,k}, v_{18,k})$ and bypasses $v_{8,k}$ in the latter path. This results in the vertex-disjoint path matching M' shown in Figure 14.

Now the PMCA-HPP$_2$ computes an Eulerian path from w to z. Because M' contains no paths with w or z as an endpoint, the algorithm computes an Eulerian path from z to y in $T - \{y, w\}$ and M', where y is the neighbor of w towards z in T, *i.e.*, $y = v_{18,k}$. Then, the PMCA-HPP$_2$ appends $\{y, w\}$ to this path and obtains the Eulerian path P from $v_{9,1}$ to $v_{17,k}$ shown in Figure 15.

The goal of the fifth step is that every vertex is incident to at most three edges from T. The PMCA-HPP$_2$ achieves this by considering every path p in P consisting only of edges from T separately. If the vertex closest to $r := z = v_{9,1}$ in p is internal

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

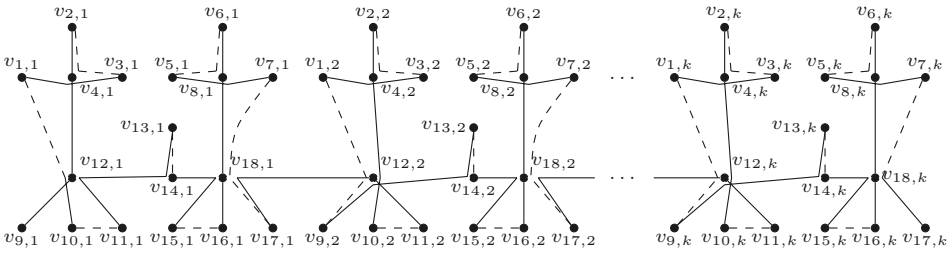


FIGURE 15. The Eulerian path P. The paths of M' are dashed.

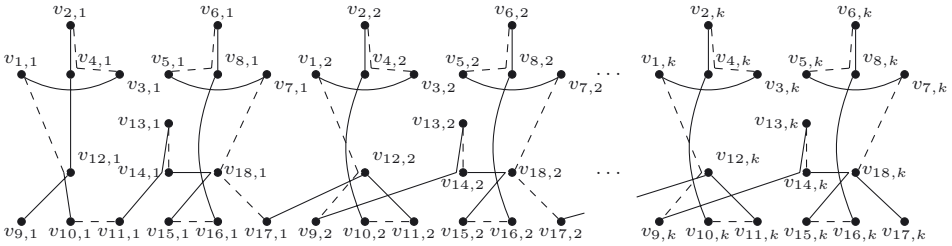


FIGURE 16. The modified Eulerian path P'. The paths of M' are dashed.

to p, it is bypassed. Therefore, it bypasses $v_{4,i}$ between $v_{3,i}$ and $v_{1,i}$ and $v_{8,i}$ between $v_{5,i}$ and $v_{7,i}$, for $1 \leq i \leq k$, $v_{12,1}$ between $v_{11,1}$ and $v_{14,1}$, $v_{12,i}$ between $v_{9,i}$ and $v_{14,i}$ and between $v_{10,i}$ and $v_{4,i}$, for $2 \leq i \leq k$, $v_{18,i}$ between $v_{16,i}$ and $v_{8,i}$ and between $v_{17,i}$ and $v_{12,i+1}$, for $1 \leq i \leq k-1$, and finally $v_{18,k}$ between $v_{16,k}$ and $v_{8,k}$. This results in the Eulerian path P' shown in Figure 16.

The goal of the last step is that every vertex except $v_{9,1}$ and $v_{17,k}$ has degree 2. The problematic vertices are thus $v_{4,i}, v_{8,i}, v_{12,i}, v_{14,i}, v_{18,i}$, for $1 \leq i \leq k$. Because $w = v_{17,k}$ is not a conflict in P', the PMCA-HPP$_2$ starts with the resolution of an arbitrary conflict. It obtains the Hamiltonian path P'' shown in Figure 17 by bypassing

- $v_{4,1}$ between $v_{12,1}$ and $v_{2,1}$, and every other $v_{4,i}$ between $v_{10,i}$ and $v_{2,i}$;
- $v_{8,i}$ between $v_{16,i}$ and $v_{6,i}$, for $1 \leq i \leq k$;
- $v_{12,1}$ between $v_{1,1}$ and $v_{10,1}$, and every other $v_{12,i}$ between $v_{1,i}$ and $v_{9,i}$;
- $v_{14,1}$ between $v_{11,1}$ and $v_{13,1}$, and every other $v_{14,i}$ between $v_{9,i}$ and $v_{13,i}$;
- $v_{18,i}$ between $v_{7,i}$ and $v_{17,i}$, for $1 \leq i \leq k$.

Considering only the edges $\{v_{1,i}, v_{9,i}\}, \{v_{2,i}, v_{10,i}\}, \{v_{6,i}, v_{16,i}\}, \{v_{7,i}, v_{17,i}\}$, $\{v_{9,i}, v_{13,i}\}$, for $2 \leq i \leq k$, we obtain $\text{cost}(P'') \geq 5(k-1)\beta^2$. For every $\beta \geq 1$ and arbitrarily small $\varepsilon > 0$, we have thus shown that there is an implementation I of

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

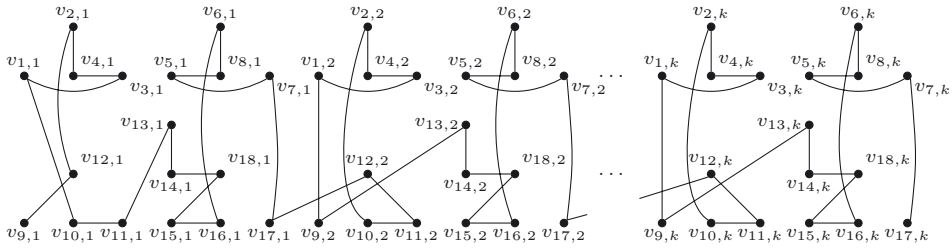


FIGURE 17. The Hamiltonian path P''.

the PMCA-HPP$_2$ such that

$$\frac{\text{cost}(I(G_{18,k}(\beta)))}{\text{Opt}_{\Delta_\beta-\text{HPP}_2}(G_{18,k}(\beta))} \geq \frac{5(k-1)\beta^2}{3k+2\beta^2+21\beta+5-\frac{2\beta^2+\beta}{k}} \geq \frac{5}{3}\beta^2-\varepsilon, \qquad (3.2)$$

for sufficiently large k, *i.e.*, we have shown that the upper bound of $5\beta^2/3$ on the approximation ratio of the PMCA-HPP$_2$ is tight.

4. Conclusion

We proved that the upper bound on the path matching Christofides algorithm (PMCA) as well as the upper bounds on the PMCA-HPP$_l$, for $l \in \{0,1,2\}$, are tight. The implications of these results are twofold. On the one hand, we know now that these algorithms can indeed return a result that is as bad as possible, *i.e.*, the upper bounds cannot be improved. Thus, if we want algorithms that provide better upper bounds, we have to come up with new ones. Indeed, for the special case of the metric HPP$_2$, a recent result due to An *et al.* [1] improves the upper bound to $(1+\sqrt{5})/2 \approx 1.6180$. On the other hand, the structure of the worst-case examples may provide insights into why the algorithms perform badly and may help devise better algorithms.

References

[1] H.-C. An, R. Kleinberg and D.B. Shmoys, Improving Christofides' algorithm for the s-t path TSP. In *Proc. of the 44th symposium on Theory of Computing (STOC 2012)* 875–886.
[2] T. Andreae, On the Traveling Salesman Problem Restricted to Inputs Satisfying a Relaxed Triangle Inequality. *Networks* **38** (2001) 59–67.
[3] M.A. Bender and C. Chekuri, Performance guarantees for the TSP with a parameterized triangle inequality. *Inf. Proc. Lett.* **73** (2000) 17–21.
[4] H.-J. Böckenhauer and J. Hromkovič, Stability of approximation algorithms or parameterization of the approximation ratio. In *Proc. of the 9th International Symposium on Operations Research in Slovenia (SOR 2007)* 23–28.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press

[5] H.-J. Böckenhauer, J. Hromkovič, R. Klasing, S. Seibert and W. Unger, Towards the notion of stability of approximation for hard optimization tasks and the traveling salesman problem. *Theor. Comput. Sci.* **285** (2002) 3–24.
[6] H.-J. Böckenhauer, J. Hromkovič and S. Seibert, Stability of Approximation. In *Handbook of Approximation Algorithms and Metaheuristics*, edited by T.F. Gonzalez. Chapman & Hall, Boca Raton (2007).
[7] N. Christofides, Worst-case analysis of a new heuristic for the travelling salesman problem. Technical Report 388. Carnegie Mellon University, Graduate School of Industrial Administration (1976).
[8] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, *Introduction to Algorithms*. MIT Press, Cambridge (2009).
[9] L. Forlizzi, J. Hromkovič, G. Proietti and S. Seibert, On the Stability of Approximation for Hamiltonian Path Problems. *Alg. Oper. Res.* **1** (2006) 31–45.
[10] E.G. Goodaire and M.M. Parmenter, *Discrete Mathematics with Graph Theory*. Prentice Hall, Upper Saddle River (2005).
[11] J.A. Hoogeveen, Analysis of Christofides' heuristic: Some paths are more difficult than cycles. *Oper. Res. Lett.* **10** (1991) 291–295.
[12] J. Hromkovič, *Algorithmics for Hard Problems. Introduction to Combinatorial Optimization, Randomization, Approximation, and Heuristics*. Springer, Heidelberg (2004).
[13] S. Krug, *Analysis of Approximation Algorithms for the Traveling Salesman Problem in Near-Metric Graphs*. Master's thesis. ETH Zurich, Department of Computer Science (2011).
[14] T. Mömke, Structural Properties of Hard Metric TSP Inputs. In *Proc. 37th Int. Conf. on Current Trends in Theory and Practice of Computer Science (SOFSEM 2011)* 394–405.

Communicated by C. De Figueiredo.
Received December 18, 2012. Accepted July 11, 2013.

https://doi.org/10.1051/ita/2013040 Published online by Cambridge University Press