Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 69 (2005)

8/7-Approximation Algorithm for (1,2)-TSP* (Extended Version)

Piotr Berman † Marek Karpinski ‡

Abstract

We design a polynomial time 8/7-*approximation algorithm* for the *Traveling Salesman Problem* in which all distances are either one or two. This improves over the best known approximation factor for that problem. As a direct application we get a 7/6-approximation algorithm for the *Maximum Path Cover Problem*, similarly improving upon the best known approximation factor for that problem. The result depends on a new method of consecutive path cover improvements and on a new analysis of certain related color alternating paths. This method could be of independent interest.

1 Introduction

The metric *Traveling Salesman Problem* (TSP) belongs to the central and one of the oldest NP-hard combinatorial optimization problems. It has been a major open problem for almost three decades to improve upon the best up to now approximation factor 3/2 ([C76]) for that problem. A special case of TSP which played an important role in establishing its NP-hardness is the Traveling Salesman Problem problem with the distances one or two ((1,2)-TSP for short). This special case of the metric TSP can be viewed as a generalization of the *Hamiltonian Cycle Problem* with nonedges represented by edges of length 2. It was also that restriction of TSP which was proven originally to be NP-hard in exact setting by Karp [K72]. The currently best

*Extended Abstract appeared in Proc. 17th ACM-SIAM SODA (2006), pp. 641-648.

†Dept. of Computer Science and Engineering, The Pennsylvania State University. Work partially supported by NSF grant CCR-0208821. Email: berman@cse.psu.edu.

‡Dept. of Computer Science, University of Bonn. Work partially done while visiting CSAIL, MIT. Work supported in part by DFG grant KA 673/7-1, PROCOPE Project, and by IST Grant 1496 (RAND-APX). Email: marek@cs.uni-bonn.de.

ISSN 1433-8092

known explicit inapproximability bound for (1,2)-TSP is 741/740 [EK01]. Some inapproximability issues for special cases of (1,2)-TSP were also studied in [FK99]. In a breakthrough result, Arora [A96] developed a PTAS for the TSP in $\mathbb{R}^2$ under any l_p metric. Trevisan [T97] used in that context a reduction from the restricted (1,2)-TSP to prove approximation hardness of TSP in $\mathbb{R}^{\log n}$ under any l_p metric.

The best up to now approximation factor of 7/6 for (1,2)-TSP was achieved in 1989 by Papadimitriou and Yannakakis [PY93], and very recently it was improved slightly by Bläser and Ram [BR05] to $65/56 \cong 1.1607$.

In this paper we improve the approximation factor for (1,2)-TSP to $8/7 \cong 1.1429$. The underlying method yields also the best up to now approximation factor 7/6 for the *Maximum Path Cover Problem* of constructing for a given graph, a set of node disjoint paths such that the number of edges in all the paths is maximal (isolated nodes are treated as paths with zero edges), as well as for the related *Maximum Traveling Salesman Problem* with distances zero and one, cf. [V92], [BS01]. The problem of *Maximum Path Cover* arises in a number of applications, among others, in parallel programs and distributed systems mappings and the code optimization problems, see [V92] for the references.

We formulate now our main theorem.

Theorem 1. *There exists a polynomial time approximation algorithm for the (1,2)-TSP with approximation ratio* 8/7*.*

The same approximation algorithm with slightly modified analysis (the modifications are given in Section 5) yields polynomial time 7/6-approximation algorithms for the *Maximum Path Cover Problem*, and the *Maximum Traveling Salesman Problem* with distances zero and one, cf. [V92].

Theorem 2. *There exists polynomial time approximation algorithms for the Maximum Path Cover Problem and the Maximum Traveling Salesman Problem with distances zero and one with approximation ratio* 7/6*.*

The formulation of the main algorithm of Theorem 1 is contained in Section 3, and its analysis in Section 4. Proof of Theorem 2 is given in Section 5.

We introduce in this paper a new method of so-called *small step improvements* on the sets of path covers, and the auxiliary notions of justifications, consistency and color alternating paths. This could be of independent interest and we feel that might have also other algorithmic applications. Unlike the previous approximation algorithms for the above problems, we do not use a classical Hardvigsen's algorithm [H84] for computing a minimum cost cover with cycles of length 4 or more. We believe that an extension of our method could possibly result in even better approximation factors for the above problems.

2 The Equivalent Statement

We can represent an instance of (1,2)-TSP as a graph G in which nodes are points of the metric and edges are pairs of points in distance 1. Suppose that G has n nodes and we can find a *path cover* with k paths (these paths have to be simple and node-disjoint). Then these paths have $n-k$ edges and we can connect them into a tour, with steps from a path end to a path beginning having cost 2; thus the cost of this tour is $n+k$. Thus our problem is to minimize k. Moreover, if an optimum solution has cost $n+k^*$, and our goal is to approximate it within factor $\frac{8}{7}$, it suffices to find a path cover with no more than $\frac{1}{7}n+\frac{8}{7}k^*$ paths.

3 Small Step Improvement Algorithm

We will investigate the following approach. We maintain a tentative solution that is represented as edge set A (A stands for *algorithm*'s solution), A is a 2-matching (*i.e.* no more than two edges of A are incident to any given node) that defines, say, k_A paths and cycles with m_A nodes in the cycles, and s_A A-*singletons*, connected components of size 1. We can alter this solution using an edge set C (C stands for *change*) into a new solution $A \oplus C$ (here $\oplus$ is the symmetric difference). We say that C improves A if

① $A \oplus C$ is a 2-matching;

② either $k_{A\oplus C} < k_A$ or

③ $k_{A\oplus C} = k_A$ and $m_{A\oplus C} > m_A$, or

④ $k_{A\oplus C} = k_A$, $m_{A\oplus C} = m_A$ and $s_{A\oplus C} < s_A$.

Suppose that for a certain constant K the following holds true:

(✷) either $k_A \le \frac{1}{7}n+\frac{8}{7}k^*$, or there exists a C that improves A and $|C| \le K$.

Then we can use the following algorithm K-**IMPROV**

```
start with $A = \varnothing$;
while you can find $C$ of size at most $K$ that improves $A$
        replace $A$ with $A \oplus C$.
```

Clearly, we cannot perform n improvements of kind ② because we would get zero as the number of paths and cycles. We also cannot perform n improvements of kind ③ without an improvement of kind ② because we would get more than n nodes in the cycles. Similarly we cannot perform $n+1$ consecutive improvements of kind ④ because we would get less than 0 A-singletons. Hence we cannot perform n^3 improvements. Each search for an improvement takes a polynomial time (where the polynomial depends on K) and when it fails, we terminate and A is a satisfactory solution.

Obviously, K-**IMPROV** runs in time $O(n^{K+4})$. In the remaining part of the paper we will prove (✷) for $K = 15$. This crude estimate of the running time can be improved essentially, Section 4.11 contains a much tighter analysis, in particular, for the case when the graph formed by the distance one edges is *sparse* (degree of that graph is bounded by a constant). We believe that a much tighter analysis of our algorithm is still possible.

The above algorithm and the method of its analysis was initially motivated by some constructions and the analysis technique used in [BHK02].

4 Alternating paths

4.1 Definitions.

To analyze the algorithm, we introduce the notions of the *auxiliary graph* $\mathcal{G}$*, paths, cycles,* possible *initial* edges and nodes, *consistency* of initial nodes, $\mathcal{AP}s$ and *justification points.*

In the analysis of Small Improvement Algorithm we fix an optimum solution, a 2-Matching B such that $k_B = k^*$ (B stands for the *best*). Using B define graph $\mathcal{G}$ which has all the same nodes as G. Let D be the set of edges that have both ends in the same cycle of A. $\mathcal{G}$ has edge set $A \cup B - D$ which we divide into three *colors*, white color $A - B - D$, black color $B - A - D$ and *gray* color $A \cap B - D$.

An *alternating path*, $\mathcal{AP}$ for short, is a path that starts and ends with a black edge and in which black and white edges alternate. The set of alternating path is used only to the analysis of the program; $\mathcal{AP}s$ are always assumed to be edge disjoint. We start by using an arbitrary decomposition of the white and black edges into $\mathcal{AP}s$, but later, when needed, we show how to alter this decomposition. At that point we may admit gray edges instead of white ones.

Paths and cycles in a solution A will be called A*-objects.* For the A-objects we define *initial nodes.* A node is *initial* if we allow it to be the first or last node of an $\mathcal{AP}$ and it is *owned* by an A-object. For an A-path, the initial nodes are the endpoints. For an A-cycle C we designate a pair of initial nodes such that C has a Hamiltonian path with these nodes as the endpoints, and this path can be extended with two black edges to another two nodes. We will show that such node pairs can be found in A-cycles with fewer than 8 nodes.

In this proof we will use the notion of *justification points*. A *justification point* is a unit of weight (a *divisible* pebble) which will be associated with the nodes. The initial distribution of those points among the nodes is defined by the weights (numbers) which are assigned to the nodes. The total number of the justification points is the sum of all these weights. If B consists of k^* paths, the optimum cost is $n + k^*$ and A is good enough if it has cost at most $\frac{8}{7}(n+k^*)$, *i.e.* $n+k \leq \frac{8}{7}(n+k^*)$, equivalently, $7k \leq n+8k^*$. We create $n+8k^*$ *justification points*, and to prove that solution A is good enough each A-object has to collect 7 points. A node that is incident to $2-a$ edges of B has $1+4a$ points (*i.e.* its weight is $1+4a$); the sum of these a's equals $2k^*$. A path starts with the justification points of its endpoints and a cycle starts with the justifications of all its nodes. The remaining points will be collected by $\mathcal{A}Ps$; an $\mathcal{A}P$ gives the collected points to the A-objects that contain its initial nodes. An $\mathcal{A}P$ collects points as follows: for every node on this path that is not already assigned to an A-object (endpoint of an A-path or a node of an A-cycle) it collects 0.5 points. The other 0.5 points can be collected by the other $\mathcal{A}P$ to which this node belongs, or redistributed according to one of our rules. After we "break" some $\mathcal{A}Ps$ into two, each of the new $\mathcal{A}Ps$ contains only one initial node and thus deliver the collected points to only one A-object.

Typically, an A-path has two endpoints and each of them is an initial node of an $\mathcal{A}P$ that should give it $2\frac{1}{2}$ points. Similarly, a typical cycle has $4+a$ nodes, it has two initial nodes of $\mathcal{A}Ps$ that should give it $(3-a)/2$ points.

There can be several deviations from the typical case. An A-object can have fewer than two initial nodes; in such a case it collects more justifications from the nodes it contains. If an A-object is an A-singleton, than each initial edge should give it 3 points. If a cycle has more than 6 nodes, it will own no initial nodes. A node incident to only 1 edge of B has 4 additional points and we omit easy special cases provided by such nodes.

4.2 Very Small Improvements.

In some situations we have small improvements that insert only one edge. We will discuss this cases, and in further analysis we may assume that they do not occur.

A black edge e that connects two initial nodes is one such case. If e connects initial nodes from two different A-objects, inserting e merges these two objects into one; when we merge an A-cycle we have to remove one of its edges. If e connects initial nodes of a single A-object, this must be an A-path and inserting e converts that path into a cycle.

An edge e that connects an A-singleton with another A-objects is another such case, except when A connect an A singleton with a midpoint of an A-path with exactly 3 nodes. Otherwise we have an improvement of kind ④.

Now suppose that we do not have a very small improvement and we have an $\mathcal{AP}$, say $\mathcal{R}$, that starts at u, and $\{u\}$ is an A-singleton. Then for an A-path (v, w, x) and some y, path $\mathcal{R}$ starts with (u, w, v, y). When we consider $\mathcal{R}$ as a possible part of an improvement, we have an option of using an ”abbreviated” version that starts with (v, y); on one hand we will ”forget” that $\mathcal{R}$ starts at an A-singleton and thus needs to collect an extra $\frac{1}{2}$ point; one the other, we will forget that $\mathcal{R}$ collected $\frac{1}{2}$ point at node w.

4.3 Examples with $\mathcal{AP}s$.

An alternating path by itself can define an improvement. The $\mathcal{AP}$ at the left of Fig. 1 consists of 4 black edges and 3 white ones. If we apply its set of edges as a change, we will get 4 paths (shown underneath) where before we had 5. What happened is that the $\mathcal{AP}$ changed the number of solution edges that are incident to path ends from 1 to 2, and it did not change this number for intermediate nodes. Therefore we decreased the number of path ends by 2, hence the number of paths by 2.

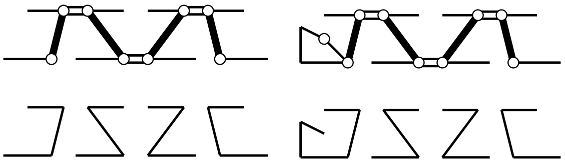


Figure 1

If the ends of the $\mathcal{AP}$ belong to cycles, the situation is similar; in the second example in Fig. 1 the number of path ends does not change but we have one less cycle, and therefore fewer objects. An interesting special case (see Fig. 2, right) occurs when $\mathcal{AP}$ starts and ends at the same cycle. Then we obtain an improvement if the first and the last nodes of this $\mathcal{AP}$ are the initial nodes of the cycle; because these nodes are *consistent*, we can include the traversal of this cycle in the improved solution.

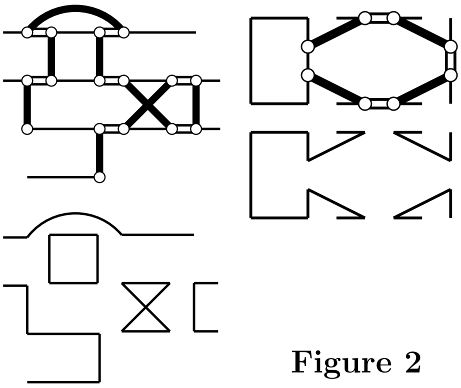


Figure 2

Finally, an $\mathcal{A}P$ may fail to provide an improvement if it creates cycles. Even if the number of path ends decreases by two the number of paths and cycles may increase if we create two new cycles in the process. In the example at the left of Fig. 2 we started with 4 paths and we changed them into 3 paths and 2 cycles.

4.4 Initial Edges of Cycles.

Let C is a cycle of A with at most 7 nodes with $|C|$ *justification* points (*i.e.* with all nodes adjacent to two edges of B).

Let $\hat{C}$ be the set of nodes of C and $\hat{K} \subset \hat{C}$ be the set of nodes incident to black edges. In this subsection we show that certain two nodes of $\hat{K}$ are *consistent* in the sense that they are endpoints of a Hamiltonian path of $\hat{C}$.

If $|\hat{K}| = 2$, then $\hat{K}$ is a *consistent pair* because the set of edges of B that are contained in C forms a single path. Hence we assume that $|\hat{K}| \geq 3$.

Suppose that two nodes of $\hat{K}$, u and v, are adjacent on the cycle C, then u and v are consistent because we can form a path by removing edge $\{u, v\}$ from C. Therefore we can assume that nodes of $\hat{K}$ are not adjacent on C, hence $|\hat{K}| \leq |\hat{C}|/2$; this means that $|\hat{K}| = 3$ and $|\hat{C}| = 6$.

Because $\hat{K}$ has 3 nodes, B covers $\hat{C}$ with two paths, one of these paths has 1 node and no edges, the other has 5 nodes and 4 edges, hence 2 edges of B are contained in $\hat{C} - \hat{K}$. W.l.o.g. C is a cycle $(u_0, \ldots, u_5)$, $\hat{K} = \{u_0, u_2, u_4\}$ and $\{u_1, u_3\} \in B$; thus we can traverse $\hat{C}$ with $(u_0, u_5, u_4, u_3, u_1, u_2)$.

4.5 $\mathcal{A}Ps$ with Deficit—General Method.

According to our rules, an $\mathcal{A}P$, say $\mathcal{R}$, collects $\frac{1}{2}$ point for every of its nodes except endpoints of paths and cycle nodes; the reason we do not *a priori* collect more is that each of these nodes may belong to two different $\mathcal{A}Ps$.

We will use several methods to give $\mathcal{R}$ more points. One is to distribute the points of gray edges. In the second method, we break $\mathcal{A}Ps$ that traverse through a cycle created by $\mathcal{R}$; if one of the broken $\mathcal{A}Ps$, say $\mathcal{P}$, is short, we can merge this cycle with the A-object that owns the initial point of $\mathcal{P}$, if $\mathcal{P}$ is long, it does not need to collect points from its edge in the cycle, and we can transfer this point to $\mathcal{R}$.

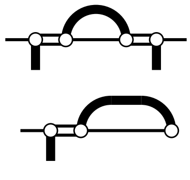
4.6 Avoiding Bad Cases.

4.6.1 S-arcs— Avoiding Them or Finding Extra Points for Them.

Arcs, the black edges contained in paths are potentially troublesome because they make it possible for a short $\mathcal{A}P$ to create more cycles as they allow to obtain a cycle from a single A-path fragment and one black edge (the arc). In this case the in the $\mathcal{A}P$ this arc is preceded and followed by a white edge directed away from it — or by a path endpoint. We will call it S-arc, for Short cycle making arc. The number of nodes on the path fragments that connects the endpoints of an arc will be called the *length* of this arc.

We will avoid the creation of S-arcs by making decisions about the decomposition of black and white edges into a set of $\mathcal{A}Ps$. When we will not be able to make such a decision, we will be able to endow such S-arcs with gray edges that will provide them with extra points.

To apply these techniques we consider a *chain* of arcs, say $\mathcal{Q}$, which is a path formed by arcs that are contained in an A-path, say $\mathcal{P}$. For the sake of uniformity, we extend $\mathcal{P}$ in both direction with a single *phony white edge*. We will make decisions how to connect the elements of a chain with the adjacent white edges when we are forming $\mathcal{A}Ps$. Connecting an arc to a phony white edge implicitly designates it to be an initial edge of its $\mathcal{A}P$.



S-arcs

Figure 3

The decisions about the decomposition within a chain are dependent: a node u incident to two arcs of $\mathcal{Q}$, say a_0 and a_1, is also incident to two white edges of $\mathcal{P}$, say e_0 and e_1. The *decision at* u may connect a_0 with e_0 and a_1 with e_1, or, alternatively, it may connect a_0 with e_1 and a_1 with e_0.

We first pick an arc a_0 of $\mathcal{Q}$ as the "least priority"; next, we assure that no other arc in $\mathcal{Q}$ is an S-arc. We start from any end of $\mathcal{Q}$ and proceed toward a. Initially we we make an arbitrary decision at that endpoint of $\mathcal{Q}$. Inductively, we consider the endpoint of arc $a \neq a_0$ for which we made decision at its other end; if the latter decision connected a with a white edge directed away from it, at the other and we connect a with a white edge directed inward, otherwise we make an arbitrary choice.

Good case ❶. An arc a in chain $\mathcal{Q}$ is *directly hit* from an endpoint of an A-path, *i.e.* a node inside arc a is connected by a black edge with this endpoint. We give a the least priority and we have no adverse consequences. If we want to create an improvement using an $\mathcal{A}P$ that contains S-arc a, then we increase the number of objects by creating a cycle, but then we decrease it back by inserting the edge of the direct hit, and removing an adjacent edge from the cycle.

Good case ❷. An endpoint of $\mathcal{Q}$, adjacent to its first arc a, is adjacent to a white edge b directed inward a; we can leave a as the ”least priority” and the final decision connecting a with b will assure that it is not an S-arc.

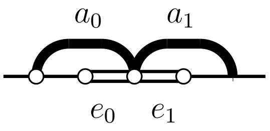


Figure 4

Good case ❸. A node u is adjacent to two arcs of $\mathcal{Q}$ and two white edges, positioned as shown in Fig. 4. We can give a_0 the ”least priority”, and the decision at u will connect a_0 with e_0 and a_1 with e_1 which assures that neither becomes an S-arc.

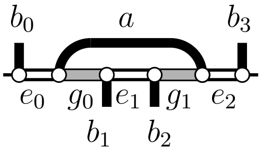


Figure 5

Good case ❹. $\mathcal{Q}$ consists of one arc only, a, of length 4, and a becames an S-arc, *i.e.* a is adjacent to two gray edges that are inside it; we must have the configuration as show in Fig. 5. We have two $\mathcal{A}Ps$, one with the sequence of edges b_0, e_0, a, e_2, b_3 and another with sequence b_1, e_1, b_2. We can alter the connections to have sequences b_0, e_0, a, g_1, b_2 and b_1, g_0, a, e_2, b_3, and each of these two $\mathcal{A}Ps$ can get $2\frac{1}{2}$ points.

Good case ❺. $\mathcal{Q}$ contains two consecutive arcs that differ in length by exactly 2, so we have a situation from Fig. 6. We can give b_0 the least priority because it is in a similar situation to a direct hit. In particular, in an $\mathcal{A}P$ we can replace the edge sequence e_0, b_0, e_1 with a ”detour” e_0, b_1, e_2.

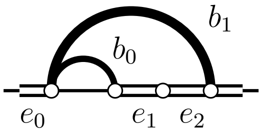


Figure 6

Consequently, if $\mathcal{Q}$ consists of only one arc a and we make it an S-arc then a has length at least 5 and is adjacent to two gray edges inside it.

Good case ❻. $\mathcal{Q}$ contains two consecutive arcs, say b_0 and b_1, where b_1 is adjacent to an end of path $\mathcal{P}$ and b_0 is not — imagine that in Fig. 6 edge e_2 is the first edge of $\mathcal{P}$. Then we can give the lowest priority to b_0, because b_1 delivers a direct hit to b_0 in case the latter becomes an S-arc.

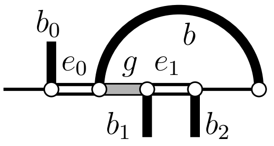


Figure 7

Good case ❼. $\mathcal{Q}$ contains an arc b adjacent to an endpoint of $\mathcal{P}$, as shown in Fig. 7. If we do not have case ❷, edge g is gray and edge e_0 is white. Suppose that e_1, the other edge of $\mathcal{P}$ that is adjacent to g, is white. We will use a similar method to case ❹. We have two $\mathcal{A}Ps$, one starts with edge sequence b, e_0, b_0, the other has a fragment b_1, e_1, b_2. We replace them by connecting b with b_1 (rather than b_0), edge sequence b, g, b_1, as well as connecting b_0 with b_2, edge sequence b_0, e_0, b, e_1, b_2. Both of these new $\mathcal{A}Ps$ use edge b, but the former removes g from $\mathcal{P}$ and collects 1 point, and the latter removes e_0 and e_1 and collects 2 points.

The chains which do not fall into either of the seven good cases are *troublesome.* They may form a *superchain* connected with gray edges. If such a chain contains an endpoint of $\mathcal{P}$, we say that it is *terminal.* Note that a troublesome terminal chain cannot be connected into a superchain with others, because it provides neighboring chains with direct hits.

A terminal troublesome chain contains only arcs adjacent to the endpoint, say u, because we cannot apply the methods of cases ❸ and ❻, and because ❷ and ❼ are not applicable, and the other end(s) of its arc(s) are adjacent to a pair of gray edges. Thus if this chain consists of one arc, the $\mathcal{A}P$ starting at this edge collects $2\frac{1}{2}$ points from 3 gray edges, and this arc has length at least 5; and if this chains consists of two arcs, it can collect points from 4 gray edges and the longer arc has length at least 7, the $\mathcal{A}P$ starting at the longer arc collects 4 points.

A nonterminal troublesome chain may have one arc only, in this case it is an arc of length 5 or more and is adjacent to two gray edges, both directed inward. We can pick any of these gray edges to provide an extra point to this arc.

A nonterminal troublesome chain with more than one arc has is adjacent to two gray edges at its endpoints. Let us decide that we will collect an extra point from one of them. We will give the least priority to an arc of length at least 6 that contains this gray edge in its interior. Say that the arc adjacent to this gray edge is the first. If the first arc has length 6 or more, we can choose it (to give the least priority). If the first arc has length below 6, we follow the chain until we have the first length increase of more than 1, or from 5 to 6. Observe that after length increase of 1 we cannot finish the chain, and after length increase of 2 we can apply the method of case ❺, so after the first larger increase we must have length at least $3 + 3 = 6$.

As we follow the chain, we visit nodes, starting from the two nodes of the initial gray edge, and we cannot return to a visited node, as black and gray edges cannot form a short cycle. If we increase the arc length by 1, we increase the span of the visited nodes without introducing a new hole, and if we decrease the arc length, we fill a hole. If we start with an arc of length 3, we do not have a hole so we cannot decrease arc length before the first large increase. If we start with an arc of length 4, we have a single hole, so we can decrease the arc length once, but this cannot be the end of the chain, and we cannot start with a decrease (this would make an arc of length 2, which is not an arc), so if we avoid length 6 we have length sequence 4,5,4, and then only increases. If we start with length 5, the first decrease cannot be 1, that closes the black/gray cycle, it cannot be 2, case ❺, and it cannot be 3 — no arcs of length 2, so we start with an increase.

4.6.2 Avoiding Terminal Short Cycles.

Let $\mathcal{R}$ be an $\mathcal{A}P$ starting at an endpoint of an A-path, say $\mathcal{P}$. We wish to avoid a situation when the first 4 edges of $\mathcal{R}$— two black and two white — define a change that creates a cycle, say $\mathcal{C}$. If $\mathcal{C}$ consists of one fragment of an A-path, it contains an S-arc and receives extra point(s), and this we accept. If $\mathcal{C}$ consists of two fragments, they need to be flanked on 4 sides; one flank can be provided by the beginning node of $\mathcal{R}$, but three flanks would have to be provided by white edges, so one white edge is used twice, hence $\mathcal{R}$ starts as shown in Fig. 8 with edges b_0, e_0, b_1, e_3. In this case we treat the triple of edges b_0, e_0, b_1 as if it was a single arc, and we apply the methods of the previous section.

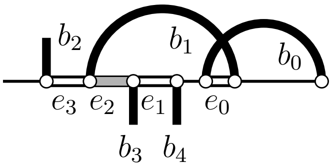


Figure 8

If e_2 is a white edge, we change $\mathcal{R}$ to follow e_2 rather than e_3; if that conflicts S-arc avoidance at that node, the other black edge at this node is an arc that is hit either by b_0 or by b_0, e_0, b_1. If e_2 is a gray edge, we give its point to $\mathcal{R}$ and we consider e_1 (if e_2 connects b_1 with b_0, we define e_1 as the edge on $\mathcal{P}$ adjacent to e_0). The reason why e_2 and e_1 can be considered in this fashion is that we can "reconfigure" $\mathcal{P}$ in such a way that e_2 (or e_1) becomes its initial edge; we replace the pair of $\mathcal{A}Ps$— $\mathcal{R}$ and the $\mathcal{A}P$ that contains e_1 with: an $\mathcal{A}P$ that uses b_2 and e_3, thus merging some object with one part of $\mathcal{P}$ and creating a cycle from the initial part, and follows with (quite arbitrarily chosen) black edge that is incident to that cycle; the second $\mathcal{A}P$ makes reconfiguration that makes e_2 the new initial edge and follows with the black edge incident to the new endnode.

If both e_2 and e_1 are gray, $\mathcal{R}$ gets 2 extra points.

4.7 $\mathcal{A}Ps$ with Deficit—Cycle to Cycle.

An $\mathcal{A}P$, say $\mathcal{R}$, that connects two cycles should collect $1\frac{1}{2} + 1\frac{1}{2} = 3$ points. Consider the cases when $\mathcal{R}$ does not form an improvement.

In that case $\mathcal{R}$ creates a cycle, so it creates c path fragments and uses b S-arcs where $c + b \geq 2$. In turn, a path fragments must have $2a$ *flanks* where they are separated from their paths, and these flanks have to be created by white edges of $\mathcal{R}$; at least two white edges create one flank, and no white edge creates more than 2, so we need at least $c + 1$ white edges, hence $\mathcal{R}$ collects at least $c + 1 + b \geq 3$ points.

4.8 $\mathcal{A}Ps$ with Deficit—Large Cycle to Path Endpoint.

In this section we consider an $\mathcal{A}P$, say $\mathcal{R}$, that connects an path $\mathcal{P}$ and a cycle $\mathcal{C}_0$ of length 6 and which does not define an improvement. It should collect at least $2\frac{1}{2}+\frac{1}{2}=3$ points.

If $\mathcal{R}$ creates two cycles, we can show that $\mathcal{R}$ collects at least 4 points. Two cycles require b S-arcs and c separated path fragments where $b+c\geq 4$. To separate c path fragments we need to create $2c$ fragment endpoints; only one fragment end can be created by the endpoint of $\mathcal{P}$, and the rest, $2c-1$ of them, requires $a\geq c$ white edges, hence $\mathcal{R}$ collects $a+b\geq 4$ points.

Now we can assume that $\mathcal{R}$ creates exactly one cycle, say $\mathcal{C}$. The reasoning from the cycle-to-cycle case does not apply only if $\mathcal{C}$ is created from fragments of A-paths and one of the endpoints of these fragments is the initial point of $\mathcal{R}$, say u, that is an endpoint of $\mathcal{P}$. If $\mathcal{R}$ does not collect 3 points, it has at most 2 white edges and $\mathcal{C}$ is a *terminal cycle* as we discussed in case ❼ of subsection 4.6.1 and in subsection 4.6.2. As we showed there, we can choose a decomposition of white and black edges into $\mathcal{A}Ps$ so that either a terminal cycle is avoided, or it receives additional 2 points from a gray edge (both points are needed if $\mathcal{C}$ is created with an S-arc).

4.9 $\mathcal{A}Ps$ with Deficit—Small Cycle to Path Endpoint.

In this section we consider an $\mathcal{A}P$, say $\mathcal{R}$, that connects an path $\mathcal{P}$ and a cycle $\mathcal{C}_0$ of length 4 or 5; $\mathcal{R}$ should collect $2\frac{1}{2}+1\frac{1}{2}=4$ points (or $3\frac{1}{2}$ points if $\mathcal{C}_0$ has length 5). If $\mathcal{R}$ creates two cycles, we have a situation already discussed in the previous subsection.

Suppose that $\mathcal{R}$ creates only one cycle $\mathcal{C}$; we would have an improvement if $\mathcal{C}$ contains more than 5 nodes.

Suppose that $\mathcal{C}$ is created with a S-arc and that it has exactly 5 nodes; imagine that (u_2,u_5) in Fig. 9 is a black arc; note that at both ends of $\mathcal{C}$ we have gray edges. If the left of these edges is adjacent to another possibly troublesome chain of arcs (besides the arc that defines $\mathcal{C}$), the first arc of this chain is directly hit by edge (u_7,u_6) and thus that it does not need an extra point. If the right of these edges is adjacent to another possibly troublesome chain of arcs, then the first arc of this chain is hit by (u_7,u_6,u_5,u_2). Therefore $\mathcal{R}$ can collect points from both of these gray edges, and thus it has 4 points.

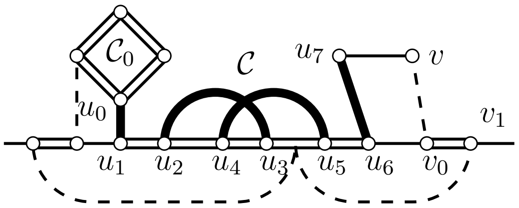


Figure 9

If $\mathcal{R}$ contains three white edges, we must have a configuration of Fig. 9

or Fig. 10 with $\mathcal{R} = (u_0, \ldots, u_7)$.

We will break every $\mathcal{A}P$, say $\mathcal{Q}$, containing an edge of $\mathcal{C}$; we claim that there will be enough points for all these $\mathcal{A}Ps$ plus one point for $\mathcal{R}$.

Suppose that we have broken $\mathcal{Q}$ into $\mathcal{Q}_0$ and $\mathcal{Q}_1$ by removing an edge e that belongs to $\mathcal{C}$. We consider $\mathcal{Q}_i$ as a possible improvement for $A \oplus \mathcal{R}$.

If $\mathcal{Q}_i$ starts at an A-cycle and it is not an improvement, $\mathcal{Q}_i$ must create a cycle from a fragment(s) of an A-path, and that requires at least two white edges; in that case $\mathcal{Q}_i$ collects at least $\frac{1}{2}$ point more than needed for that A-cycle. There is one exception: $\mathcal{Q}_i$ may start at $\mathcal{C}_0$ that in solution $A \oplus \mathcal{R}$ is an ending portion of a path and we may have an arc that was not an arc in solution A; then $\mathcal{Q}_i$ may have exactly one white edge and it still creates a cycle. However, this case describes an improvement, as the new cycle must have more nodes than $\mathcal{C}_0$.

If $\mathcal{Q}_i$ starts at an A path and it is not an improvement, $\mathcal{Q}_i$ must create a cycle from a fragment(s) of an A path, and this requires collecting at least 3 points; if $\mathcal{Q}_i$ creates a cycle with 2 points, it has at most 2 white edges, so this is a terminal cycle that gets another 2 points. Again, $\mathcal{Q}_i$ has a surplus of $\frac{1}{2}$ point. An exception occurs if $\mathcal{Q}_i$ has at most two white edges and it uses an arc that was not an arc in solution A, *i.e.* when $\mathcal{Q}_i$ creates a cycle from a single fragment of a path that contains edge (u_6, u_7).

If there exist two such exceptions, such $\mathcal{Q}_i$ and $\mathcal{Q}_j$ form an improvement in conjunction with $\mathcal{R}$. The reason is that both expectional $\mathcal{A}P$ fragment create a cycle from a single path fragment that contains a particular edge; it follows that one of these fragments must have an endpoint inside the other, suppose that the cycle created by $\mathcal{Q}_j$ has an endpoint inside the cycle created by $\mathcal{Q}_i$. Now, applying $\mathcal{R}$ as a change creates replaces cycle $\mathcal{C}_0$ with $\mathcal{C}$ and creates a "composite" path with edge (u_6, u_7); applying $\mathcal{Q}_i$ fuses $\mathcal{C}$ with a path but creates a cycle that contains (u_6, u_7) edge; now the initial portion of $\mathcal{Q}_j$, with at most one white edge, reaches the latter cycle, so it must be an improvement.

Note that an exception $\mathcal{Q}_i$ must have at least one white edge.

Now we can perform the balance of points. If we break $\mathcal{Q}$ and one of the paths is an exception, one branch of $\mathcal{Q}$ has a surplus of $\frac{1}{2}$, the second branch has a deficit of $1\frac{1}{2}$, and we have also the white edge that was removed from $\mathcal{Q}$; as a result we have neither a deficit nor a surplus. If none of the paths is an exception, we have a surplus of 2 points. Finally, if we had a gray edge, we have 1 point to collect. Note that the interior of $\mathcal{C}$ has at least two edges, so at least one of them brings the surplus.

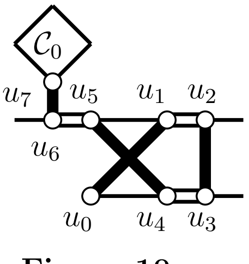


Figure 10

The difference between the case of Fig. 9 and the case of Fig. 10 is the following: one of the nodes of $\mathcal{C}$ is an endpoint of an A-path, so we cannot collect $\frac{1}{2}$ point from that node — this A-path already collected both halfs; as a result we can be $\frac{1}{2}$ point short. This requires that everything is tight: $\mathcal{C}_0$ has 4 nodes only, the interior of $\mathcal{C}$ has only two edges, one is gray, so we break only one $\mathcal{A}P$, say $\mathcal{Q}$, and one of the created branches, say $\mathcal{Q}_0$, starts at a path endpoint, creates a cycle and has one white edge only.

This means that $\mathcal{Q}_0$ starts at an endpoint of a new path joined with edge (u_2, u_3), uses a new arc — that surely is not an arc in solution A — and that its 4-th node is u_0, u_1, u_4 or u_5. If it is u_0, $\mathcal{Q}_0$ forms an $\mathcal{A}P$ between two endpoints of A-path, with one white edge only — surely an improvement. If it is u_5, we extend $\mathcal{Q}_0$ with (u_5, u_6, u_7) and we get an $\mathcal{A}P$ from an A-path to an A-cycle with two white edges and no arc — again an improvement. If it is u_4, we extend $\mathcal{Q}_0$ with $(u_4, u_3, u_2, u_1, u_0)$ and we get an $\mathcal{A}P$ between two A-path endpoints with no S-arc, and 3 white edges that are located on two A-paths, in the next subsection we will see that it has to be an improvement. The case of u_1 is similar: we extend $\mathcal{Q}_0$ with $(u_1, u_5, u_4, u_3, u_2, u_1, u_0)$.

4.10 $\mathcal{A}Ps$ with Deficit—Path Endpoint to Path Endpoint.

An $\mathcal{A}P$ that connects two path endpoints should collect 5 points or form a small improvement. Otherwise, we have an $\mathcal{A}P$, say $\mathcal{R}$, that creates at least 2 cycles and collects at most 4 points.

If $\mathcal{R}$ creates c cycles, they must contain $2c - a$ fragments of A-paths and a S-arcs. To separate the fragments of A-paths from their paths we need to create $2(2c - a)$ flanks, and at most 2 of them can be the path endpoints, so at least $2(2c - a - 1)$ of them is created by white edges, which requires at least $2c - a - 1$ white edges.

Suppose that $\mathcal{R}$ creates 3 cycles. Then it collects at least $2 \times 3 - 1 = 5$ points from white edges and S-arcs.

Suppose that $\mathcal{R}$ creates 2 cycles and it collects only 3 points. Then none of the cycles is a terminal one, and we have a configuration from Fig. 11, or a similar one, with 2 white edges and one S-arc. We can collect the extra point using the $\mathcal{A}P$ breaking method from the previous subsection.

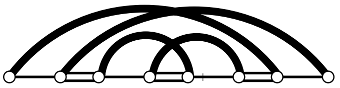


Figure 11

Note that $\mathcal{R}$ converted an A-path, say $\mathcal{P}$, into a pair of cycles. Suppose that there exists an $\mathcal{A}P$ that starts at another A-path and extends to $\mathcal{P}$, and that it collects less than 3 points before reaching $\mathcal{P}$. Then this $\mathcal{A}P$ fragment does not create a new cycle but it merges one of the cycles that replaced

$\mathcal{P}$ with another A-path; as a result we have the same number of objects as before the change, but we have more nodes in cycles, an improvement. Now we can assume that such an $\mathcal{A}P$ collects at least 3 points, so it has $\frac{1}{2}$ point surplus before reaching $\mathcal{P}$.

Suppose that there exists an $\mathcal{A}P$ that starts at an A-cycle and extends to $\mathcal{P}$ and it collects less than 2 points before reaching $\mathcal{P}$, this $\mathcal{A}P$ decreases the number of objects in solution $A \oplus \mathcal{R}$; if there are such $\mathcal{A}Ps$ reaching each of the two new cycles that cover $\mathcal{P}$, we have an improvement. Now we can assume that one of the new cycles, say $\mathcal{C}$, is not reached by any such $\mathcal{A}P$.

We can break all $\mathcal{A}Ps$ that contain an edge of $\mathcal{C}$ and collect points from its gray edges. In the worst case, $\mathcal{C}$ is the "outer" cycle, and it has only 2 edges on $\mathcal{P}$, one of them gray; we collect surplus of two partial $\mathcal{A}Ps$, 1 point, and the points from the edges themselves that were not allocated already to $\mathcal{P}$, 2-nd point.

The configuration of Fig. 11 was based on an assumption that every white edge provides two endpoints for the path fragments of the cycles created by $\mathcal{R}$. In every other case $\mathcal{R}$ collect at least 4 points, and we can use the techniques of the last two subsections to collect another one.

Now suppose that $\mathcal{R}$ creates 2 cycles and collects only 4 points. We can repeat the above arguments and collect points from one of the cycles created by $\mathcal{R}$, say $\mathcal{C}$, that is not reached by a partial $\mathcal{A}P$ with a deficit — that extends from an A-cycle. However, one $\mathcal{R}$ has a "spare" black edge that merges two fragments of A-paths into a new path, and as a result new arcs and new terminal cycles may exists. Thus we experience the same problems as in the previous subsection. As a result, we need a more detailed analysis for the case when $\mathcal{C}$ contains only one white edge, only one gray edge, and it contains an endpoint of an A-path.

In a case analysis we can omit the case when $\mathcal{R}$ includes an S-arc, because it is treated identically to the case when $\mathcal{R}$ creates a cycle like the inner cycle in Fig. 11. Therefore we will assume that $\mathcal{R}$ contains exactly 4 white edges, creates 4 path fragments combined into 2 cycles, hence it creates 8 fragment flanks, and *either*

there are 3 white edges that create two flanks each, 1 white edge creating 1 flank, and 1 flank is created by a common endpoint of $\mathcal{R}$ and an A-path, hence or 4 path fragments are contiguous as in Fig. 11, *or*

there are 2 white edges that create two flanks each, 2 white edges creating 1 flank each and 2 flanks are created by endpoints of $\mathcal{R}$.

In the first case, it is as if we added to Fig. 11 a part of Fig. 9 that is to the left of u_5. Suppose that edge incident to the beginning of $\mathcal{R}$ is white, then we are breaking $\mathcal{Q}$ that contains that edge. Let $\mathcal{Q}_0$ be the branch starting at the beginning of $\mathcal{R}$, we can show that it should have a surplus of $1\frac{1}{2}$ point.

Observe that $\mathcal{Q}_0$ can be used as a complete $\mathcal{A}P$, and suppose that it does not create an improvement. If $\mathcal{Q}_0$ connects to another endpoint of an A-path, it can collect only 3 points only if it is exactly as in Fig. 11, so it converts an A-path into two cycles. However, edge (u_7, u_6) delivers a direct hit to one of these cycles, so combined with $\mathcal{Q}_0$ it does not change the number of objects but it creates one new cycle. If $\mathcal{Q}_0$ connects to an A-cycle and has at most 2 white edges, than it creates a terminal cycle and gets at least 3 points. Thus we got the extra point that we need.

If that edge is gray, we break $\mathcal{Q}$ that contains the rightmost edge of Fig. 11, in the position of (u_3, u_5) of Fig. 9. Suppose that the exception $\mathcal{Q}_0$ comes to u_5, we can extend it to u_3 and than to the beginning of $\mathcal{R}$, and this is a complete $\mathcal{A}P$ with 2 white edges, an improvement. Suppose that $\mathcal{Q}_0$ comes to u_3; we view it as a change to solution $A \oplus \mathcal{R}$; from the outer cycle we remove (u_3, u_5), we add back edge (u_5, u_6) and we remove (u_6, u_7). As a result, we merge $\mathcal{C}$ with the cycle created by $\mathcal{Q}_0$, and the net change is one more cycle.

We skip the case analysis for the second case because it involves the same ideas.

4.11 Running Time Estimate.

We need to make improvements of kind ④ as long as an isolated node is connected with an edge to anything except the center node of a path with 3 nodes. With a single scan of all edges we can reach a situation when no such improvement is possible.

The number of improvements of kind ③ and ② that can be perform in the execution of our algorithm is $O(n^2)$, thus the running time is determined by: the size of the search space, and the cost of determining that a small change is an improvement.

An improvement that we need is one that does not include augmenting paths that deliver a sufficient number of justification points to the respective A-objects. We need most points when we connect a pair of A-paths, namely 5, so an insufficiently long path has 4 white edges — and 5 black ones. Note that when we pick a black edge, we have only two possibilities to pick a white one, as it must belong to the A-path where the last end of a selected black belongs.

Such a path does not form an improvement when it forms at least 2 cycles. If a cycle is formed with an S-arc, we have an extra point, so we have only 3 white edges; there is an exception, "good case ❶" in subsection 4.6.1. Upon closer inspection, this exception is used only when a S-arc connects two nodes, and one of these two nodes is connected with the end of its own

A-path. The latter connection should be the part of the improvement, but here we have only two possibilities to consider in our searching space (so it is a matter of a constant factor).

One can see that the number of conceivable paths with 4 white edges is $O(n^5)$ (in graphs of bounded degree, $O(n)$). If such a path forms two cycles and we cannot find more points, we have shown in subsection 4.10 that we can obtain an improvement by connecting either (a) one of the cycles with an end of an A-path, where the connection was an alternating path with at most two white edges – and at most 3 black ones, or (b) each of these two cycles with an A-cycle, via an alternating path with at most one white edge.

Strictly speaking, an alternating path with one white edge that connects two cycles has to be extended by edges inside each cycle to form an improvement that satisfies condition ①. Because those edges can be selected arbitrarily, we can represent the improvemet by a set that does not include them; this way the largest improvement set has $9 + 3 + 3 = 15$ edges. If we connect a path to a cycle, this connection has 5 edges (again, not counting the necessary but implicit edge in the cycle) and this leads to an improvement with $9 + 5 = 14$ edges.

However, the search space that we need has to include only improvements needed in our proof (either one of them exists or we have enough points) and these sets are far from arbitrary. Hence the search space is much smaller than $\Omega(m^{15})$.

First, let us count the number of ways to choose our alternating path with 4 white edges. This case is possible only when (i) $\mathcal{P}$ connects two ends of A-paths, and (ii) each of these paths contains a white edge of $\mathcal{P}$. Thus we freely choose the first node on that path and the second (two unrestricted choices of nodes of the first black edge), then we choose the third in two ways (the first white edge, restricted choice of a node), after three more unrestricted choices of nodes we have selected all the white edges, and we finish by selecting an end of the "visited" A-path — a restricted choice. Thus we have a search space with $O(n^5)$ paths. If this is a bounded degree graph (when we consider the edges of cost 1), that searching space has $O(n)$ paths.

In the further analysis let us assume that the shorter of the two cycles created by $\mathcal{P}$ has a nodes. The first implication is that at least two unrestricted choices are actually restricted to $O(a)$ possibilities, so we can decrease the estimate of the search space size by the factor of $(n/a)^2$ (but we have no such "reward" for the bounded degree case).

Now, consider the improvement of type (a). The effect of this improvement is that the number of A-objects remains unchanged while one of the cycles created by $\mathcal{P}$ is new, and no cycles are destroyed. The reasoning is that we can break $\mathcal{A}Ps$ going through both cycles, and it may happen that these

$\mathcal{A}Ps$ are short for one of the two cycles, thus providing no extra points, but they come from A-cycles, so the use of one of these $\mathcal{A}Ps$ does not increase the number of cycles. However, if $a > 6$, this is an improvement even so: we replace a cycle with at most 6 nodes with one that has more. Therefore we have to consider an improvement of this kind only if $a \leq 6$.

In the bounded degree case, this makes all choices restricted, and thus the searching space remains $O(n)$. In the general case, we make three unrestricted choices in case (a): two to select the first black edge (and white), the third to select the second pair of black and white edge, and the final choice is restricted to at most 12 possibilities. So we have the searching space of size $O(n^5 \times n^3/n^2 = n^6)$. In case (b) we make 4 unrestricted choices, so we have a searching space of size $O(n^7)$.

If $a > 6$, then our improvement increases the number of nodes in cycles by at least $a - 6$, so if we count n^2 improvement, this improvement counts as $a - 6$ of them. Moreover, the improvement includes only two $\mathcal{A}Ps$, rather then three. In the general case, we have a searching space of size $O(n^6a^2)$ and the improvement counts as a improvements, so the worst case is for $a\tilde{n}$, which is equivalent to the case (b) above. In the bounded degree case we have a searching space of size $O(na)$, so we pay $O(n)$ per "standard" improvement.

Summarizing, we can estimate the running time as $O(n^9)$ in the general case and $O(n^3)$ in the bounded degree case. We believe that a much tighter running time analysis is still possible.

5 Proof of Theorem 2

We start with a modified analysis for the *Maximum Path Cover.* We use the same technique of *justification points* as in Section 4. However, we need to define those points slightly differently.

Given n points, suppose that an optimum path cover $\mathcal{C}^*$ covers them with p paths where isolated nodes are counted as paths. Then these paths contain $n-p$ edges. To obtain an 7/6-approximation it suffices to find a cover $\mathcal{C}$ with q paths such that

$$\frac{6}{7}(n-p) \leq n-q \ , \qquad \text{i.e.} \qquad q \leq \frac{1}{7}n + \frac{6}{7}p \ .$$

According to the latter sufficient condition, for each path of $\mathcal{C}$ we need to find 7 points, provided we get 1 point for each node and 6 points for each path from $\mathcal{C}^*$.

The 6 points of a path from $\mathcal{C}^*$. can be distributed between its two ends (which may be located at a single node if this is a degenerate path with no edges), so we place 1 point on every node and 3 points on every endpoint of a path from $\mathcal{C}^*$.

In turn, in a phase of Small Improvement algorithm we try to collect 7 points for every path of the current solution $\mathcal{C}$, and failing that, we have to find a small improvement. The only difference with our previous algorithm and analysis is that there we were giving 4 points to every endpoint of a path in $\mathcal{C}^*$ rather than 3. However, 3 extra points are still sufficient.

Suppose that we have an isolated point in $\mathcal{C}^*$. Then this point does not belong to any $\mathcal{A}Ps$, and it delivers 1+6 points to a path in $\mathcal{C}$ where it belongs.

Suppose that we have a path endpoint of $\mathcal{C}^*$. The analysis is similar as before, except that one $\mathcal{A}P$ can terminate at this point — we give the extra 3 points to that $\mathcal{A}P$. Note that "one-ended" $\mathcal{A}P$ needs to collect at most 2.5 points (besides the point from its "end"), so these 3 points always suffice.

A similar modification of a definition of *justification points* and an analysis of a phase of Small Improvement algorithm gives a 7/6-approximation of the *Maximum Traveling Salesman Problem* with distances zero and one.

Given an instance of that problem, we disregard first the edges with profit 0, so a valid solution is either a collection of node disjoint simple paths or a Hamiltonian Cycle. Thus, in the absence of a Hamiltonian Cycle the problem is *identical* to Maximum Path Cover. In the analysis of the case with a Hamiltonian Cycle, we do not have to consider path endpoints of the optimum solution, so we merely give 1 point to each node and the goal is to find 7 points for each object of the solution.

□

Acknowledgments

We thank Lars Engebretsen and Johan Håstad for stimulating discussions related to the topic of this paper.

References

[A96] S. Arora, *Polynomial Time Approximation Schemes for Euclidean TSP and other Geometric Problems*, Proc. 37th IEEE FOCS (1996), pp. 2-11.

[BHK02] P. Berman, S. Hannenhalli and M. Karpinski, *1.375-Approximation Algorithm for Sorting by Reversals*, Proc. 10th ESA (2002), LNCS 2461, Springer, 2002, pp. 200-210.

[BR05] M. Bläser and L. Shankar Ram, *An Improved Approximation for TSP with Distances One and Two*, Proc. 15th FCT (2005), LNCS 3623, Springer, 2005, pp. 504-515.

[BS01] M. Bläser and B. Siebert, *Computing Cycle Covers without Short Cycles*, Proc. 9th ESA (2001), LNCS 2161, Springer, 2001, pp. 368-379.

[C76] N. Christofides, *Worst-Case Analysis of a New Heuristic for the Traveling Salesman Problem*, Technical Report, GSIA, Carnegie-Mellon University, 1976.

[EK01] L. Engebretsen and M. Karpinski, *Approximation Hardness of TSP with Bounded Metrics*, Proc. 28th ICALP (2001), LNCS 2076, Springer, 2001, pp. 201-212; the full paper appeared in J. Comput. System Sci. 72 (2006), pp. 509-546.

[FK99] W. Fernandez de la Vega and M. Karpinski, *On the Approximation Hardness of Dense TSP and other Path Problems*, Information Processing Letters 70 (1999), pp. 53-55.

[H84] D. B. Hardvigsen, *Extensions of Matching Theory*, Ph. D. Thesis, Carnegie Mellon University, 1984.

[K72] R. M. Karp, *Reducibility among Combinatorial Problems*, in R. E. Miller and J. W. Thatcher (Eds.), Plenum, New York, 1972.

[PY93] C. H. Papadimitriou and M. Yannakakis, *The Traveling Salesman Problem With Distances One and Two*, Math. Oper. Res. 18 (1993), pp. 1-11.

[T97] L. Trevisan, *When Hamming Meets Euclid: The Approximability of Geometric TSP and MST*, Proc. 29th ACM STOC (1997), pp. 21-29.

[V92] S. Vishwanathan, *An Approximation Algorithm for the Asymmetric Travelling Salesman Problem with Distances One and Two*, Information Processing Letters 44 (1992), pp. 297-302.