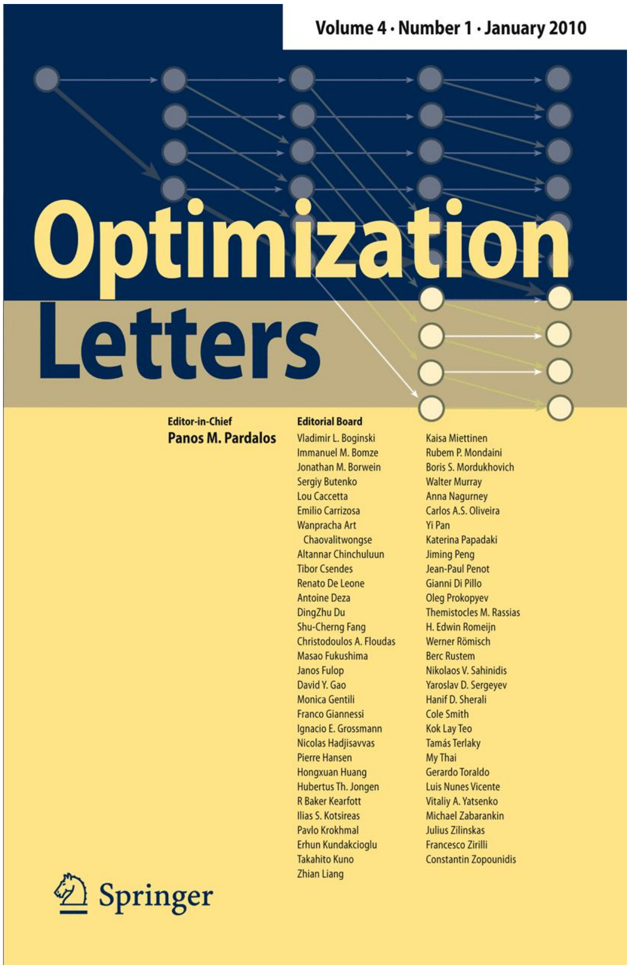
ISSN 1862-4472, Volume 4, Number 1



Volume 4 · Number 1 · January 2010

Optimization Letters

Editor-in-Chief
Panos M. Pardalos

Springer

Optim Lett (2010) 4:7–16
DOI 10.1007/s11590-009-0152-7

ORIGINAL PAPER

The approximability of three-dimensional assignment problems with bottleneck objective

Dries Goossens · Sergey Polyakovskiy · Frits C. R. Spieksma · Gerhard J. Woeginger

Received: 28 May 2009 / Accepted: 5 October 2009 / Published online: 21 October 2009

Abstract We discuss two special cases of the three-dimensional bottleneck assignment problem where a certain underlying cost function satisfies the triangle inequality. We present polynomial time 2-approximation algorithms for the broadest class of these special cases, and we prove that (unless P = NP) this factor 2 is best possible even in the highly restricted setting of the Euclidean plane.

Keywords Bottleneck problem · Multidimensional assignment · Approximation · Computational complexity · Efficient algorithm

1 Introduction

This note deals with the three-dimensional bottleneck assignment problem, which can be described as follows: Given are three n-element sets R, G, and B whose elements are called red, green, and blue points, respectively. A *feasible triple* (or just *triple*, for short) consists of one red point, one green point, and one blue point. Every feasible triple $(i, j, k) \in R \times G \times B$ is associated with a cost c_{ijk}. A *feasible solution* consists

D. Goossens · S. Polyakovskiy · F. C. R. Spieksma (✉)
Operations Research Group, Katholieke Universiteit Leuven, Naamsestraat 69, 3000 Leuven, Belgium
e-mail: frits.spieksma@econ.kuleuven.be

D. Goossens
e-mail: dries.goossens@econ.kuleuven.be

S. Polyakovskiy
e-mail: sergey.polyakovskiy@econ.kuleuven.be

G. J. Woeginger
Department of Mathematics, TU Eindhoven, P.O. Box 513,
5600 MB Eindhoven, The Netherlands
e-mail: gwoegi@win.tue.nl

of n feasible triples that together contain all $3n$ points in $R \cup G \cup B$. The goal in the three-dimensional bottleneck assignment problem is to find a feasible solution that minimizes the maximum cost over all triples in the solution.

We consider two special cases of the three-dimensional bottleneck assignment problem, in which the costs arise from a certain underlying distance function d that assigns to every two points i and j in $R \cup G \cup B$ a corresponding non-negative distance $d(i, j)$. Throughout we will assume that the distances are symmetric with $d(i, j) \equiv d(j, i)$, and that they satisfy the *triangle inequality*:

$$d(i, j) \leq d(i, k) + d(j, k) \quad \text{for all } i, j, k. \tag{1}$$

In the perimeter version B3AP-per of the three-dimensional bottleneck assignment problem, the costs c_{ijk} correspond to the perimeters of the underlying triangles:

$$c_{ijk} = d(i, j) + d(j, k) + d(k, i). \tag{2}$$

In other words, the goal in B3AP-per is to find a partition of the points into triangles such that the largest triangle perimeter is minimized. In the diameter version B3AP-dia of the three-dimensional bottleneck assignment problem, the costs c_{ijk} correspond to the diameters of the underlying triangles:

$$c_{ijk} = \max\{d(i, j), d(i, k), d(j, k)\}. \tag{3}$$

In other words, the goal in B3AP-dia is to find a partition of the points into triangles such that the largest triangle diameter is minimized.

In a more restricted geometric setting, the points in $R \cup G \cup B$ are embedded in the Euclidean plane, and the distances $d(i, j)$ arise as the standard Euclidean distances. We refer to these restricted variants as the *Euclidean special cases* of B3AP-per and B3AP-dia.

1.1 Known results

Two-dimensional bottleneck assignment problems as well as multi-dimensional sum assignment problems have received their share of attention in the literature; see for instance Burkard et al. [1]. In contrast to this, the literature does not seem to contain much work on multi-dimensional bottleneck assignment problems: Enumerative approaches are discussed by Malhotra et al. [9], and by Varthak and Geetha [12]. A solvable special case is described in Klinz and Woeginger [8]. The results of Crama and Spieksma [2] imply that the variations of B3AP-per and B3AP-dia *without* the triangle inequality (1) do not admit any polynomial time approximation algorithm with a constant worst case ratio (unless P = NP).

Gonzalez [6] investigates a closely related clustering problem where the points come without colors. The goal is to find a partition of the points into k clusters (of arbitrary size) such that the maximum cluster diameter is minimized. Gonzalez constructs a polynomial time 2-approximation algorithm for this bottleneck problem, if the points lie in some Euclidean space. Furthermore he shows that (unless P = NP) the 2-dimensional variant does not allow a polynomial time approximation algorithm with worst case ratio strictly better than $\sqrt{3}$, and that the 3-dimensional variant does not allow a worst case ratio strictly better than 2. Feder and Greene [5] improve the 2-dimensional negative result from $\sqrt{3} \approx 1.732$ to $2\cos(10) \approx 1.969$. Hochbaum and Shmoys [7] discuss a variety of bottleneck problems in routing, location, and communication network design. Wang and Du [13] discuss the bottleneck Steiner tree problem in the plane: They give a polynomial time 2-approximation algorithm for the Euclidean distance and the Manhattan distance. They show that (unless P = NP) for the Manhattan distance this result is best possible, and that for the Euclidean distance no approximation guarantee strictly better than $\sqrt{2}$ can be reached in polynomial time.

Finally, let us mention some related variants of the three-dimensional assignment problem under the *sum* objective (where one wants to minimize the sum of the costs of the triples). These cases are relatively well-understood: Crama and Spieksma [2] design polynomial time $\frac{4}{3}$-approximation algorithms for the perimeter problem, if the underlying distances satisfy the triangle inequality. Spieksma and Woeginger [11] establish NP-hardness of the corresponding Euclidean special case. Queyranne and Spieksma [10] exhibit a a 2-approximation algorithm for the diameter problem, again assuming the triangle inequality.

1.2 Contributions of this note

We first derive a (very simple) positive result, and then a (more sophisticated) matching negative result. The positive result concerns the approximability of the general problems B3AP-per and B3AP-dia under the triangle inequality.

Theorem 1 *For both problems* B3AP-per *and* B3AP-dia, *there exists a polynomial time approximation algorithm H that outputs a feasible solution with a cost of no more than twice the cost of an optimal solution.*

The main contribution of this note is that the result stated in Theorem 1 cannot be improved unless P = NP. In fact we will show that even the highly restricted Euclidean special cases do not allow stronger approximation results.

Theorem 2 *For any $\varepsilon > 0$, the Euclidean special cases of problems* B3AP-per *and* B3AP-dia *do not possess a polynomial time $(2-\varepsilon)$-approximation algorithm (unless P = NP).*

Theorems 1 and 2 will be proved in Sects. 2 and 3, respectively.

2 The approximation algorithm

In this section we will prove Theorem 1. Indeed, let us consider some arbitrary instance of B3AP-per. The auxiliary problem of finding a bottleneck assignment between the red points and the green points can be formulated as the following integer program:

$$\begin{array}{lll} \min \max_{i\in R, j\in G} d(i,j)x_{ij} & & \\ \text{s.t.} \quad \sum_{j\in G} x_{ij} = 1 & & \text{for all } i \in R \\ \qquad \sum_{i\in R} x_{ij} = 1 & & \text{for all } j \in G \\ \qquad x_{ij} \in \{0,1\} & & \text{for all } i \in R,\ j \in G. \end{array}$$

In any solution of this integer program, each point from R receives a point from G, and each point from G is assigned to a point from R. Let x^*_{ij} denote an optimal solution, and let L_{RG} denote the corresponding objective value. In an analogous way we set up an integer program for the bottleneck assignment between red points and blue points. Let x^*_{ik} denote an optimal solution to this red-blue integer problem, and let L_{RB} denote the corresponding objective value. Finally we set

$$L = \max\{L_{RG}, L_{RB}\}. \tag{4}$$

Note that the optimal solutions of these two bottleneck assignment problems can be determined in polynomial time $O(n^{2.5}/\sqrt{\log n})$; see for instance Burkard et al. [1]. Based on these optimal solutions x^*_{ij} and x^*_{ik}, we introduce the set

$$X = \{(i,j,k) : x^*_{ij} = 1 \text{ and } x^*_{ik} = 1, \text{ for } i \in R, j \in G, k \in B\}. \tag{5}$$

Note that X constitutes a feasible set of triples for the considered instance of B3AP-per. We denote the corresponding objective value by $H_p(X)$, and we fix a triple (i_p, j_p, k_p) whose cost realizes the objective value $H_p(X)$. Then we have:

$$\begin{aligned} H_p(X) &= d(i_p, j_p) + d(i_p, k_p) + d(j_p, k_p) \\ &\le 2d(i_p, j_p) + 2d(i_p, k_p) \le 2L_{RG} + 2L_{RB} \le 4L. \end{aligned} \tag{6}$$

Here the first inequality follows from the triangle inequality. The second inequality follows from the definition of X, and since the two auxiliary bottleneck assignments match i_p with j_p and i_p with k_p, respectively.

Next consider an optimal solution S with value OPT_p for the instance of B3AP-per. This solution induces feasible solutions for the two auxiliary bottleneck assignment problems. By the definition of L, one of these two induced solutions has bottleneck value at least L. Hence there exists some triple (i_s, j_s, k_s) in S with $d(i_s, j_s) \ge L$ or $d(i_s, k_s) \ge L$; without loss of generality we assume $d(i_s, j_s) \ge L$. Then the triangle inequality yields $d(i_s, k_s) + d(k_s, j_s) \ge L$, and thus the cost of this triple is at least $2L$. This implies

$$\mathrm{OPT}_p \ge 2L. \tag{7}$$

The inequalities (6) and (7) together settle Theorem 1 for problem B3AP-per.

Next let us turn to problem B3AP-dia. We will argue that the same approximation algorithm H can be used. The set X defined in (5) constitutes a feasible solution for

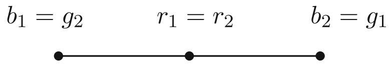


Fig. 1 A worst-case instance for H

the considered instance of B3AP-dia. We denote the corresponding objective value by $H_d(X)$, and we fix a triple (i_d, j_d, k_d) whose cost realizes this objective value $H_d(X)$. The triangle inequality yields

$$H_d(X) = \max\{d(i_d, j_d), d(i_d, k_d), d(j_d, k_d)\} \leq \max\{L, L, 2L\} = 2L. \qquad (8)$$

Consider an optimal solution with value OPT_d for the instance of B3AP-dia. Again we observe that this optimal solution induces feasible solutions for the two auxiliary bottleneck assignment problems. Hence one of the triples will contain an edge of length at least L. This yields

$$\mathrm{OPT}_d \geq L. \qquad (9)$$

The inequalities (8) and (9) settle Theorem 1 for problem B3AP-dia. This completes the proof of Theorem 1.

Finally, we observe that our analysis of algorithm H is tight: The instance in Fig. 1 may lead to the set $X = \{(1, 1, 1), (2, 2, 2)\}$. Then $H_p(X) = 8$ and $\mathrm{OPT}_p = 4$ yield ratio 2 for problem B3AP-per, and $H_d(X) = 4$ and $\mathrm{OPT}_d = 2$ yield ratio 2 for problem B3AP-dia.

3 The lower bound argument

In this section we will prove Theorem 2. Our approach is based on the traditional gap technique: We show that a YES-instance of some well-known NP-complete decision problem corresponds to a solution of our bottleneck problem with cost L, whereas a NO-instance corresponds to an instance of our problem with cost at least $2L$. Then, a polynomial time approximation algorithm with a worst case ratio strictly less than 2 would be able to distinguish the YES-instances of problem X from the NO-instances, and this would imply P = NP.

For the well-known NP-complete decision problem we will choose the planar 3-dimensional matching problem P-3DM, which has been proven to be NP-complete by Dyer and Frieze [4]. An instance of P-3DM consists of three pairwise disjoint sets X, Y, Z with $|X| = |Y| = |Z| = q$, and a set of triples $T \subseteq X \times Y \times Z$ that satisfy the following properties.

(i) Every element of $X \cup Y \cup Z$ occurs in at most three triples of T.
(ii) The following bipartite graph $G = (V_1 \cup V_2, E)$ is planar: For each element $X \cup Y \cup Z$ the graph has a corresponding vertex in V_1, and for each triple in T the graph has a corresponding vertex in V_2. Whenever some element occurs in some triple, the corresponding vertices are connected by an edge in E.

The problem is to decide whether there exists a set $T' \subseteq T$ of q triples such that every element of $X \cup Y \cup Z$ is contained in precisely one triple from T'.

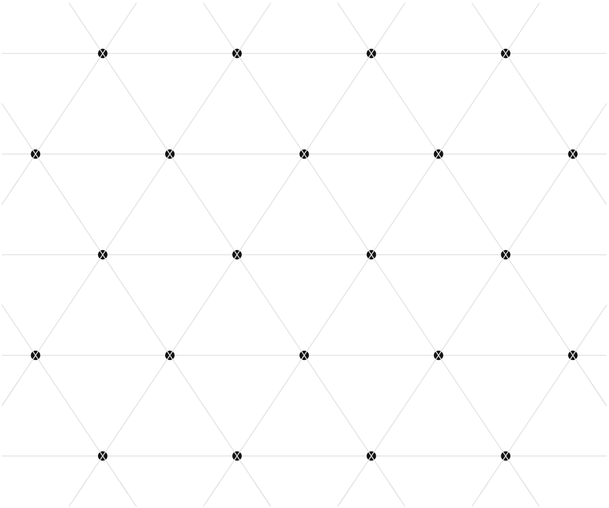


Fig. 2 A triangular grid

Now let us start with an arbitrary instance I of P-3DM, and let us construct a corresponding instance I' of B3AP-per from it. The main idea is to embed the underlying bipartite graph G as specified in (ii) in a triangular grid; see Fig. 2. The literature contains a wide arsenal of polynomial time algorithms for finding all kinds of grid embeddings of planar graphs; see for instance the book [3] by Di Battista et al.

All edges in the triangular grid have length 1, and all points of instance I' will lie on vertices of this grid. There are three types of points in I':

- For each element e in X (Y, respectively Z) there is a corresponding red (green, respectively blue) point in I' lying on a grid vertex. This grid vertex is called an *element vertex* corresponding to e.
- For each triple $t \in T$, there are a corresponding a red point, a corresponding green point, and a corresponding blue point, that together lie on the same vertex of the grid. This grid vertex is called a *triple vertex* corresponding to t.
- For each occurrence of an element $e \in X \cup Y \cup Z$ in some triple $t \in T$ in P-3DM, there is a corresponding set $P(e, t)$ of points in I' that forms a kind of path from the corresponding element vertex to the corresponding triple vertex. This path follows along an odd number $v_1, v_2, \ldots, v_{2k-1}$ (with $k \geq 2$) of grid vertices, such that consecutive vertices are adjacent in the grid. Every vertex v_i with even index i contains a single point; the color of this point coincides with the color of the element point e. Every vertex v_i with odd index i contains two points with two different colors; these are the other two colors that do not show up at the element vertex e.

Note that vertex v_1 in the path $P(e, t)$ is adjacent to the element vertex e, and that vertex v_{2k-1} is adjacent to the triple vertex t. These vertices v_1 and v_{2k-1} are called the *contact vertices* of path $P(e, t)$, whereas the other vertices are the *inner vertices*.

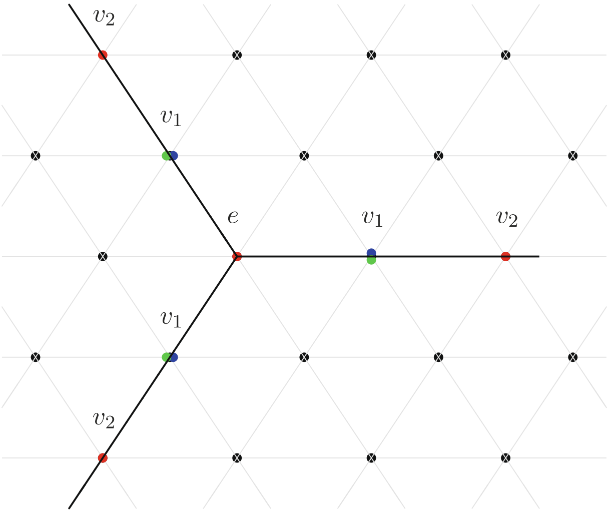


Fig. 3 Three paths leaving an element vertex

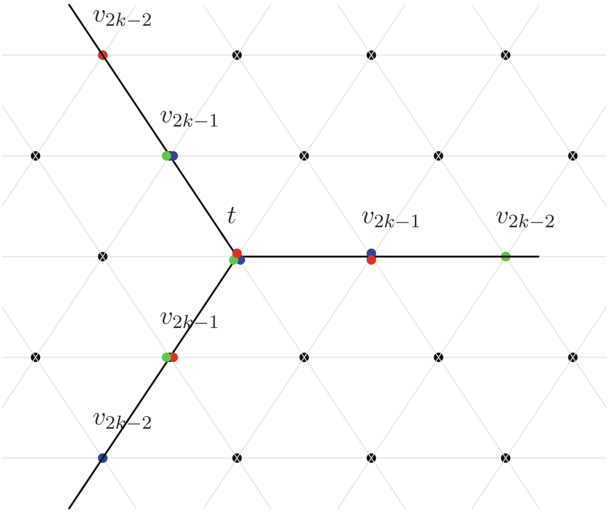


Fig. 4 Three paths leading to a triple vertex

Note that by (i) every element vertex e has at most three leaving paths. These paths are chosen so that they enclose angles of at least 120° at the element vertex e; see Fig. 3 for an illustration. Similarly the three paths leading into a triple vertex t will always enclose angles of 120° at vertex t; see Fig. 4 for an illustration. Furthermore, the paths $P(e, t)$ satisfy the following nice properties:

- On every path $P(e, t)$, the inner vertices have distance larger than 2 to the inner vertices of all other paths. Furthermore, these inner vertices have distance 1 to their immediate neighbors on the path, distance at least $\sqrt{3}$ to the neighbors of their neighbors, and distance at least 2 to all other vertices on the path.

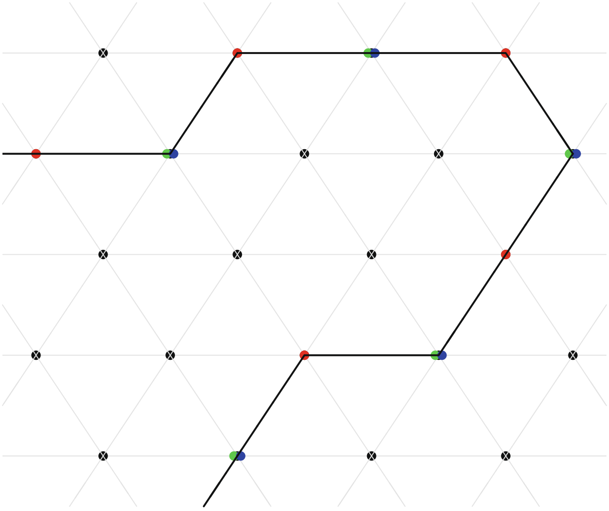


Fig. 5 A connecting path $P(e, t)$

- On every path $P(e, t)$, the contact vertices have distance 1 to their immediate neighbors, and distance at least 2 to all other vertices on the path.
- Whenever a path $P(e, t)$ makes a turn, it is a 120° turn.

It is not difficult to see that the triangular grid allows such systems of connecting paths, if the element and triple vertices are embedded sufficiently far away from each other (see Fig. 5). This completes the description of the instance.

Lemma 3 *If the instance I of problem* P-3DM *is a YES-instance, then the constructed instance I' of problem* B3AP-per *has optimal cost at most* 2.

Proof Since I is a YES-instance, there exists a set $T' \subseteq T$ such that each element of $X \cup Y \cup Z$ occurs in precisely one triple of T'. Consider some element e that occurs in triple $t \in T'$. On the path $P(e, t)$ that connects the element vertex to the triple vertex, we match the point in e with the two points in v_1. Furthermore, for $i = 1, \ldots, k-1$ we match the two points in vertex v_{2i+1} with the single point in vertex v_{2i}. The other paths connect e to some triple not in T'. On these paths we match the two points in vertex v_{2i-1} with the single point in vertex v_{2i}, for $i = 1, \ldots, k-1$. Note that all these triples have perimeter 2.

Finally, in every triple vertex corresponding to some $t \in T$ we match the three points in a triple of cost 0. In every triple vertex corresponding to some $t \notin T$ we match the three points appropriately with the six points in the contact vertices of the incident paths. Again, all these triples have perimeter 2. □

Lemma 4 *If the instance I of problem* P-3DM *is a NO-instance, then the constructed instance I' of problem* B3AP-per *has optimal cost at least* 4.

Proof Suppose for the sake of contradiction that there exists a solution for instance I' with cost strictly less than 4. Consider some path $P(e, t)$ with vertices $v_1, v_2, \ldots,$

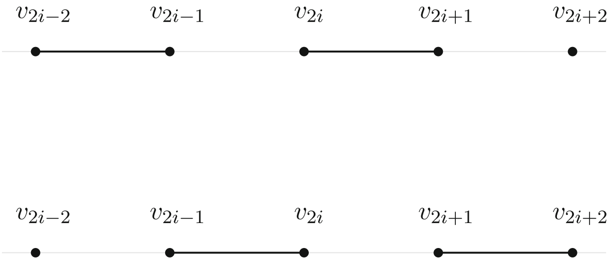


Fig. 6 Two ways of traversing a path

v_{2k-1}. Along this path, any solution with cost strictly less than 4 has exactly two possible ways of forming triples; see Fig. 6. Note that the first solution leaves v_1 unassigned, whereas the second solution leaves v_{2k-1} unassigned; in such a situation we say that the unassigned vertex is a *free* vertex.

Now consider an element vertex e, and its (at most three) adjacent contact vertices (see Fig. 3). If none of these contact vertices is free, then the element point must be matched in a triple of cost at least 4. If two or more of these contact vertices are free, then the points in these vertices must be matched with two or more points with the color of e. Again, this incurs a cost of at least 4. To summarize, exactly one of the adjacent contact vertices must be free, and its corresponding to points are matched with point e.

Next let us look at a triple vertex t. A similar case distinction as above shows that either none or all three of the adjacent contact vertices must be free. In the former case, the three points in the triple vertex are matched to each other. In the latter case, the three points in the triple vertex are matched to the six points in the contact vertices.

All in all, this implies that I is a YES-instance of P-3DM: Indeed, each element vertex is assigned to exactly one path (with a free contact vertex) that leads to a triple vertex. And each triple vertex is either assigned to three paths (in which case the triple is selected into T'), or assigned to no path (in which case the triple is ignored). □

Lemmas 3 and 4 together prove the statement in Theorem 2 on problem B3AP-per. What about the statement on problem B3AP-dia? Interestingly, the above construction goes through again. The entire argument works out essentially as before, except that now a YES-instance corresponds to a solution with cost 1 and a NO-instance corresponds to a solution with cost at least 2. This completes the proof of Theorem 2.

Acknowledgments This research has been supported by the Netherlands Organisation for Scientific Research (NWO) under grant 639.033.403, by BSIK grant 03018 (BRICKS: Basic Research in Informatics for Creating the Knowledge Society), and by OT Grant OT/07/015.

References

1. Burkard, R., Dell'Amico, M., Martello, S.: Assignment problems. SIAM (2009)
2. Crama, Y., Spieksma, F.C.R.: Approximation algorithms for three-dimensional assignment problems with triangle inequalities. Eur. J. Oper. Res. **60**, 273–279 (1992)

3. di Battista, G., Eades, P., Tamassia, R., Tollis, I.G.: Graph Drawing: Algorithms for the Visualization of Graphs. Prentice Hall, Upper Saddle River (1999)
4. Dyer, M.E., Frieze, A.M.: Planar 3DM is NP-complete. J. Algorithms **7**, 174–184 (1986)
5. Feder, T., Greene, D.H.: Optimal algorithms for approximate clustering. In: Proceedings of the 20th Annual ACM Symposium on Theory of Computing, pp. 434–444 (1988)
6. González, T.F.: Clustering to minimize the maximum intercluster distance. Theor. Comput. Sci. **38**, 293–306 (1985)
7. Hochbaum, D.S., Shmoys, D.B.: A unified approach to approximation algorithms for bottleneck problems. J. ACM **33**, 533–550 (1986)
8. Klinz, B., Woeginger, G.J.: A new efficiently solvable special case of the three-dimensional axial bottleneck assignment problem. Lect. Notes Comput. Sci. **1120**, 150–162 (1996)
9. Malhotra, R., Bhatia, H.L., Puri, M.C.: The three-dimensional bottleneck assignment problem and its variants. Optimization **16**, 245–256 (1985)
10. Queyranne, M., Spieksma, F.C.R.: Approximation algorithms for multi-index transportation problems with decomposable costs. Discret. Appl. Math. **76**, 239–253 (1997)
11. Spieksma, F.C.R., Woeginger, G.J.: Geometric three-dimensional assignment problems. Eur. J. Oper. Res. **91**, 611–618 (1996)
12. Vartak, M.N., Geetha, S.: Specially structured precedence constraints in three-dimensional bottleneck assignment problems. J. Oper. Res. Soc. **41**, 339–344 (1990)
13. Wang, L., Du, D.-Z.: Approximations for a bottleneck Steiner tree problem. Algorithmica **32**, 554–561 (2002)