ELSEVIER

Operations Research Letters 24 (1999) 29–35

operations research letters

A $\frac{5}{3}$-approximation algorithm for the clustered traveling salesman tour and path problems

Shoshana Anily[a], Julien Bramel[b,*,1], Alain Hertz[c]

[a] *Faculty of Management, Tel-Aviv University, Tel-Aviv 69978, Israel*
[b] *Graduate School of Business, Columbia University, 3022 Broadway, Uris Hall, New York, NY 10027-6902, USA*
[c] *Département de Mathématiques, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland*

Received 1 December 1997; received in revised form 1 September 1998

Abstract

We consider the ordered cluster traveling salesman problem (OCTSP). In this problem, a vehicle starting and ending at a given depot must visit a set of n points. The points are partitioned into K, $K \leqslant n$, prespecified clusters. The vehicle must first visit the points in cluster 1, then the points in cluster 2,..., and finally the points in cluster K so that the distance traveled is minimized. We present a $\frac{5}{3}$-approximation algorithm for this problem which runs in $O(n^3)$ time. We show that our algorithm can also be applied to the path version of the OCTSP: the ordered cluster traveling salesman path problem (OCTSPP). Here the (different) starting and ending points of the vehicle may or may not be prespecified. For this problem, our algorithm is also a $\frac{5}{3}$-approximation algorithm.

Keywords: Analysis of algorithms; Suboptimal algorithms; Transportation; Vehicle routing

1. Introduction

We consider the ordered cluster traveling salesman problem (OCTSP). In this problem, a vehicle starting and ending at a given depot must visit a set of n points partitioned into K disjoint clusters so that points of cluster k are visited prior to points of cluster $k+1$, for $k = 1, 2, \dots, K-1$, and the total distance traveled is minimum.

We assume a complete undirected graph $G = (V, E)$ where $K + 1$ clusters denoted $C_i \subseteq V$, for each $i = 0, 1, 2, \dots, K$, are prespecified. We assume $C_i \cap C_j = \emptyset$ for all $1 \leqslant i, j \leqslant K$, $i \neq j$, and C_0 consists of a single node denoted by $0 \in V$ which we refer to as the *depot*, i.e., $C_0 = \{0\}$. The *ordered cluster TSP* (OCTSP) is the problem of determining the minimum length tour that starts and ends at the depot and visits each node of V in such a way that all nodes of C_i are visited before any of the nodes in C_{i+1}, for $i = 1, 2, \dots, K-1$.

* Corresponding author. E-mail: jdb8@columbia.edu.

[1] This work was supported in part by NSF CAREER Award DMI-97-02596.

PII: S0167-6377(98)00046-7

Since this is a generalization of the TSP (the TSP is the OCTSP with $K = 1$) the problem is $\mathcal{NP}$-hard. In the OCTSP, the clusters as well as the *order* of the clusters are considered *prespecified*. We also consider the path version of this same problem, the ordered cluster traveling salesman path problem (OCTSPP), as well as the version where no depot is specified and we seek only a closed tour through the clusters in the required order.

The OCTSP is a variant of the *free cluster TSP* (FCTSP), where the cluster order is not prespecified. In the FCTSP, the problem is to simultaneously determine the optimal cluster order as well as the routing within and between clusters. These models have well-known applications in automated warehouse routing (see [2,9]) and production planning ([9]). Other applications involve service systems where customers have different service priorities (see [6]). The special case of $K = 2$ is often called the traveling salesman problem with Backhauls (TSPB) (see [5]).

We seek polynomial-time approximation algorithms. An algorithm is an *α-approximation algorithm* if it constructs a solution whose length is guaranteed to be within α of the length of an optimal solution. For the TSP, a well-known $\frac{3}{2}$-approximation algorithm is Christofides' Heuristic [3]. Approximation algorithms for the OCTSP include the 2-approximation $O(n^2)$ algorithm of Gendreau et al. [6]. For the special case of $K = 2$, Gendreau et al. [5] provide a $\frac{3}{2}$-approximation algorithm. For the FCTSP, Arkin et al. [1] provide a 3.5-approximation algorithm, and recently, Guttmann-Beck et al. [7] developed a 2.75-approximation algorithm.

In this paper, we present a $\frac{5}{3}$-approximation algorithm for the OCTSP which runs in $O(n^3)$ time. It is an adaptation of Christofides' Heuristic for the TSP. We also show that the same performance guarantee is achieved when these ideas are applied to the OCTSPP.

2. Preliminaries

Let $V = \bigcup_{i=0}^{K} C_i$ and $|V| = n + 1$. The distance function is given by: dist: $V \times V \to \mathbb{R}^+$ which is assumed to be symmetric and satisfies the triangle inequality: $\forall a, b, c \in V$, $\text{dist}(a, c) \leqslant \text{dist}(a, b) + \text{dist}(b, c)$.

Let OPT denote the cost of the optimal solution to the OCTSP and for any $S \subseteq E$, let $\text{cost}(S) \equiv \sum_{(v,w)\in S} \text{dist}(v, w)$. We define $\overline{d} \equiv \max_{i=1}^{K} \max\{\text{dist}(a, b)\colon a, b \in C_i\}$ as the maximum distance between any two points in the same cluster, and $\hat{d}$ to be the maximum distance between consecutive clusters, i.e., $\hat{d} \equiv \max_{i=0}^{K} \max\{\text{dist}(a, b)\colon a \in C_i,\ b \in C_{i+1}\}$, where $C_{K+1} \equiv C_0$. Let $m > \overline{d}$ and $M > (K + 1)(m + \hat{d})$ be fixed.

We define a set $\text{CON} \subseteq E$ consisting of $K + 1$ edges that represent the shortest connections between consecutive clusters. That is,

$$\text{CON} \equiv \{(0, a_1), (b_1, a_2), (b_2, a_3), \ldots, (b_{K-1}, a_K), (b_K, 0)\},$$

where $a_i, b_i \in C_i$ (for $i = 1, 2, \ldots, K$) are chosen so that CON is of minimum length.

In [6] the authors present a 2-approximation algorithm for the OCTSP which runs in $O(n^2)$ time. The idea is to first build minimum spanning trees in each cluster. Let $\mathscr{F}$ be the forest made up of the union of these K trees. Then the edges of CON are added to this graph. It is obvious that:

Property 2.1.

$$\text{cost}(\mathscr{F}) + \text{cost}(\text{CON}) \leqslant \text{OPT}.$$

By doubling some edges of the spanning trees, a solution can be constructed whose length is at most $2 \cdot \text{OPT}$.

In [5] the authors present a $\frac{3}{2}$-approximation $O(n^3)$ algorithm for the special case of $K = 2$ by combining the minimum spanning trees in each cluster with a specific minimum weight perfect matching over a subset of the nodes. A solution is constructed from this set of edges. The worst-case bound is obtained by using Property 2.1 and by showing that the cost of the matching is at most $\frac{1}{2}\text{OPT}$.

3. The OCTSP heuristic

We now present our heuristic. We assume without loss of generality that $a_i \neq b_i$, for $i=1,2,\dots,K$. If this is not the case we add to C_i ($i=1,2,\dots,K$), without affecting OPT, a dummy node at the same location. In C_0 we duplicate node 0 so that $C_0=\{0,0'\}$, and let $a_0=0$ and $b_0=0'$.

The ordered cluster TSP (OCTSP) heuristic

Step 1: Construct a minimum spanning tree within each cluster C_i ($i=0,1,\dots,K$), and let $\mathscr{F}$ be the union of these $K+1$ trees.

Step 2: For $i=1,2,\dots,K$, augment the graph by duplicating each node $v=a_i$ and $v=b_i$ that is of *even* degree in $\mathscr{F}$. For each duplicate v' of v, add v' to C_i and to V and add the edge (v,v') of zero length to $\mathscr{F}$. Let $\tilde{V}$ be the nodes in the resulting graph, and let $\tilde{C}_i$ be the nodes in cluster i, for $i=0,1,\dots,K$. Let $\mathscr{F}^0$ be the resulting forest. Define $\mathscr{O}$ as the set of odd degree nodes in $\mathscr{F}^0$.

Step 3: Define a symmetric function dist$'$: $\tilde{V}\times\tilde{V}\to\mathbb{R}^+$ as follows:

$$\mathrm{dist}'(v,w) = \begin{cases} \mathrm{dist}(v,w) & \text{if } v,w\in\tilde{C}_i \text{ for } i=1,2,\dots,K, \textit{ or } v=0 \text{ and } w\in\tilde{C}_1 \textit{ or } v=0' \text{ and } w\in\tilde{C}_K, \\ \mathrm{dist}(v,w)+m & \text{if } v\in\tilde{C}_i \text{ and } w\in\tilde{C}_{i+1} \text{ for } i=1,2,\dots,K-1, \\ \mathrm{dist}(v,w)+M & \text{otherwise.} \end{cases}$$

Note dist$'(0,0')=M$. Find a minimum weight perfect matching on $\mathscr{O}$ using the weight function dist$'$. Denote by MATCH* the set of edges of this matching.

Step 4: Combine the edges of $\mathscr{F}^0$ and MATCH*. Construct a feasible tour H from this set of edges.

Note that the graph augmentation performed in Step 2 clearly does not affect the optimal solution value OPT.

By the definitions of m and M and the fact that $|\mathscr{O}\cap\tilde{C}_i|$ is even and at least 2 for each $i=0,1,\dots,K$, MATCH* will never match 0 to $0'$ or two nodes in (different) non-consecutive clusters. Thus, MATCH* contains exactly one edge connecting 0 to $\tilde{C}_1$, one edge connecting $\tilde{C}_i$ to $\tilde{C}_{i+1}$ for $i=1,2,\dots,K-1$, and one edge connecting $\tilde{C}_K$ to $0'$. Then it is clear that combining $\mathscr{F}^0$ and MATCH* in Step 4 results in an Eulerian and connected graph. Therefore, H has exactly one edge connecting 0 to $\tilde{C}_1$, one edge connecting $\tilde{C}_i$ to $\tilde{C}_{i+1}$ for $i=1,2,\dots,K-1$, and one edge connecting $\tilde{C}_K$ to $0'$.

The complexity of the OCTSP Heuristic consists of the time required to construct the minimum spanning trees ($O(n^2)$, see [10]) and the time required to find one perfect matching ($O(n^3)$, see [10]). Therefore, the complexity is $O(n^3)$.

4. Analysis of the worst-case performance

In this section, we prove the $\frac{5}{3}$ bound. We first require the following two lemmata.

Define MATCH^0 to be the set of edges of a minimum weight perfect matching on $\mathscr{O}\setminus\{0,0'\}$ using the edge weights dist$'$. Add the edge $(0,0')$ to MATCH^0. Since m and M are large enough, this matching includes only edges connecting nodes of the same cluster. Recall $\mathrm{cost}(S)\equiv\sum_{(v,w)\in S}\mathrm{dist}(v,w)$. Then:

Lemma 4.1.

$$\mathrm{cost}(\mathrm{MATCH}^*)+\mathrm{cost}(\mathrm{MATCH}^0)\leqslant \mathrm{OPT}.$$

Proof. Consider an optimal tour (on the augmented graph) and focus on the ordering this imposes on the set of nodes $\mathscr{O}$. Node 0 is the starting point and node $0'$ is the ending point. Observe that the number of nodes in $\mathcal{O}$ within each cluster is even. Similarly to [3], decompose the optimal tour on this subset into two different matchings by taking every other arc. Let M_1 and M_2 be these two matchings. Since $|\mathcal{O} \cap \tilde{C}_i|$ is even for $i = 0, 1, \ldots, K$, one of the matchings, say M_1, contains *only* edges connecting nodes within the same cluster. The other matching M_2 contains $K+1$ edges connecting the clusters, in addition to a number of edges connecting nodes within the same cluster. More precisely, M_2 contains one edge connecting each $\tilde{C}_i$ to $\tilde{C}_{i+1}$ for $i = 0, 1, \ldots, K-1$ and one edge connecting $\tilde{C}_K$ to $\tilde{C}_0$.

Clearly $\text{cost}(\text{MATCH}^0) \leqslant \text{cost}(M_1)$ and $\text{cost}(\text{MATCH}^*) \leqslant \text{cost}(M_2)$ giving the desired result. □

Lemma 4.2.

$$\text{cost}(\text{MATCH}^*) \leqslant \tfrac{1}{2}(\text{cost}(\text{MATCH}^0) + \text{cost}(\mathscr{F}^0)) + \text{cost}(\text{CON}).$$

Proof. For each cluster $\tilde{C}_i$, for $i = 1, 2, \ldots, K$, let $Q_i \equiv (\mathcal{O} \cap \tilde{C}_i) \setminus \{a_i, b_i\}$. If Q_i is non-empty then each of its nodes is of odd degree and $|Q_i|$ is even, for $i = 1, 2, \ldots, K$. Find a minimum weight perfect matching on the nodes of $\bigcup_{i=1}^{K} Q_i$ using the weight function dist$'$. This matching clearly matches only nodes within the same cluster. Let M^i be the edges of the matching that are contained in cluster i, for $i = 1, 2, \ldots, K$. Also define $\text{CON}' = \text{CON} \cup \{(b_K, 0')\} \setminus \{(b_K, 0)\}$.

Now observe that the set of edges $(\bigcup_{i=1}^{K} M^i) \cup \text{CON}'$ represents the edges of a feasible matching of the type constructed in Step 3 of the OCTSP heuristic. Therefore, since MATCH^* represents the edges of the cheapest such matching:

$$\begin{aligned} \text{cost}(\text{MATCH}^*) &\leqslant \text{cost}\left(\left(\bigcup_{i=1}^{K} M^i\right) \cup \text{CON}'\right) \\ &= \sum_{i=1}^{K} \text{cost}(M^i) + \text{cost}(\text{CON}'). \end{aligned} \tag{1}$$

For any $U \subseteq \tilde{V}$, let $\text{TSP}(U) \subseteq \tilde{E}$ be the edges of an optimal traveling salesman tour through U. The optimal traveling salesman tour on Q_i defines two matchings on Q_i by taking every other arc. Hence we derive, for each $i = 1, 2, \ldots, K$:

$$\text{cost}(M^i) \leqslant \tfrac{1}{2}\text{cost}(\text{TSP}(Q_i)).$$

This implies

$$\sum_{i=1}^{K} \text{cost}(M^i) \leqslant \frac{1}{2} \sum_{i=1}^{K} \text{cost}(\text{TSP}(Q_i)) \leqslant \frac{1}{2} \sum_{i=1}^{K} \text{cost}(\text{TSP}(\tilde{C}_i)).$$

Consider the set of edges in $(\mathscr{F}^0 \cup \text{MATCH}^0) \cap (\tilde{C}_i \times \tilde{C}_i)$, for $i = 1, 2, \ldots, K$. This is exactly the set of edges that would be used to construct a traveling salesman tour in $\tilde{C}_i$ using Christofides' Heuristic (see [3]). That is, for each cluster, it consists of a minimal spanning tree plus the edges of a minimal weight perfect matching between odd degree nodes. Therefore,

$$\sum_{i=1}^{K} \text{cost}(\text{TSP}(\tilde{C}_i)) \leqslant \text{cost}(\mathscr{F}^0 \cup \text{MATCH}^0),$$

which implies

$$\sum_{i=1}^{K} \text{cost}(M^i) \leqslant \frac{1}{2}(\text{cost}(\mathscr{F}^0) + \text{cost}(\text{MATCH}^0)).$$

This, combined with Eq. (1) and the fact that $\text{cost}(\text{CON}) = \text{cost}(\text{CON}')$, gives the desired result. □

We now prove the main result.

Theorem 4.3.

$$\text{cost}(H) \leqslant \tfrac{5}{3}\text{OPT}.$$

Proof. By construction, the feasible tour H has the following property:

$$\text{cost}(H) \leqslant \text{cost}(\text{MATCH}^*) + \text{cost}(\mathscr{F}^0). \quad (2)$$

We consider two cases.

Case 1: $\text{cost}(\text{MATCH}^*) \leqslant 2\,\text{cost}(\text{MATCH}^0)$. In this case, Lemma 4.1 implies that $\text{cost}(\text{MATCH}^*) \leqslant \frac{2}{3}\text{OPT}$. The result then follows directly from Eq. (2) and Property 2.1.

Case 2: $\text{cost}(\text{MATCH}^*) > 2\,\text{cost}(\text{MATCH}^0)$. In this case, Lemma 4.1 implies that $\text{cost}(\text{MATCH}^0) \leqslant \frac{1}{3}\text{OPT}$. Combined with Lemma 4.2, Eq. (2) implies that

$$\begin{aligned}\text{cost}(H) &\leqslant \tfrac{1}{2}\text{cost}(\text{MATCH}^0) + \tfrac{3}{2}\text{cost}(\mathscr{F}^0) + \text{cost}(\text{CON}) \\ &\leqslant \tfrac{1}{6}\text{OPT} + \tfrac{3}{2}\text{OPT} \quad \text{(by Property 2.1)}\end{aligned}$$

giving the desired result. □

For the special case of $K = 2$, the Eulerian graph constructed by our algorithm (in Step 4) is no longer than the Eulerian graph constructed by the algorithm of [5]. Therefore, it is possible to show that our algorithm is also a $\frac{3}{2}$-approximation algorithm for this case. One can show the bound is tight by applying our algorithm to the example of [4].

For the case $K \geqslant 3$, we show the tightness of the bound by adapting the example from Hoogeveen [8]. Consider the graph in Figure 1:

In Fig. 1, nodes of the same cluster share the same subscript, i.e., cluster i is made up of the six nodes: $\{r_i, s_i, t_i, u_i, v_i, w_i\}$. The depot is replicated on both sides of the graph (but they are at the same physical location). Assume $0 < \varepsilon \ll 1$, and the distance between any two nodes can be found by finding the shortest path in the graph. An optimal solution visits each cluster in the order $r_i \to s_i \to t_i \to u_i \to v_i \to w_i$. The length of this solution is $3K + (6K + 1)\varepsilon$.

Our algorithm first computes minimum spanning trees in each cluster. Say for each cluster these are the edges: $\{(r_i, s_i), (s_i, t_i), (s_i, v_i), (v_i, u_i), (v_i, w_i)\}$. Then $\text{cost}(\mathscr{F}^0) = (3 + 2\varepsilon)K$. Note $\mathscr{O} = V \cup \{0, 0'\}$. An optimal matching of the nodes of $\mathscr{O}$ uses the edges $(0, s_1)$, $(r_1, t_1), (u_1, w_1)$, (v_1, s_2), etc. This has total cost $2K + (3K + 1)\varepsilon$. The solution constructed by the algorithm has each cluster visited in the order: $s_i \to r_i \to t_i \to v_i \to$

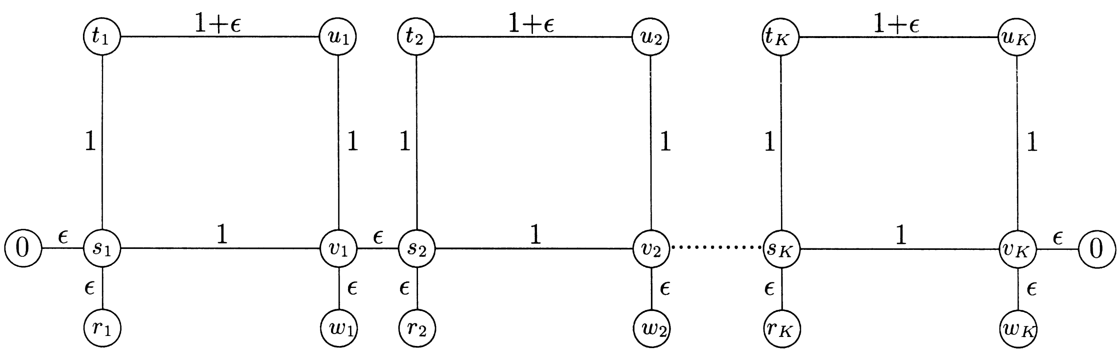

Fig. 1. Example proving the tightness of the $\frac{5}{3}$ bound.

$u_i \to w_i$. The length of this solution is $5K + (5K + 1)\varepsilon$. The worst-case ratio can be made arbitrarily close to $\frac{5}{3}$ by choosing ε small enough.

Remark 1. In the case where a closed tour (with no depot) is saught, our heuristic can be adapted to find a solution whose length is guaranteed to be within $\frac{5}{3}$ of the length of the optimal solution. Select an arbitrary cluster and run the OCTSP heuristic by choosing, one at a time, each node of the cluster as a dummy depot. By choosing the cluster with the least number of nodes, we ensure a running time of $\mathrm{O}(n^3 \min_{i=1}^{K} |C_i|) = \mathrm{O}(n^4)$.

Remark 2. In the case where the number of clusters K is *fixed* (independent of n), our algorithm can be used in a polynomial-time $\frac{5}{3}$-approximation algorithm for the FCTSP. Simply apply our algorithm to each of the $K!/2 = \mathrm{O}(1)$ different permutations of the clusters. The complexity is $\mathrm{O}(n^3)$.

5. The OCTSPP

We now show how these same ideas can be applied to the OCTSPP yielding an identical performance guarantee of $\frac{5}{3}$. In the OCTSPP, the objective is to find a minimum length path visiting all nodes, with the restriction that cluster 1 be visited before cluster 2, cluster 2 before cluster 3, etc. We consider the case where neither the starting nor ending point is prespecified, the other cases can be transformed into this case by adding additional clusters.

First define a dummy depot $0 \in V$ and define $\mathrm{dist}(0, 0') = 0$, $\mathrm{dist}(0, v) = 0$ for $v \in \tilde{V}$ and $\mathrm{dist}(0', v) = 0$ for $v \in \tilde{V}$. (Note that this violates the triangle inequality, but a careful reading of the proof shows that $\frac{5}{3}$-approximation is still guaranteed.) The algorithm then proceeds exactly as in Section 3. If 0 ($0'$) is matched to $a \in \tilde{C}_1$ ($b \in \tilde{C}_K$), then a (b) is the starting (ending) point. It is possible, as above, to show that this path is at most $\frac{5}{3}$ times the length of an optimal path. The example of Fig. 1, with the depot redefined as the first as well as the last cluster, proves the tightness of the bound.

Finally, we remark that this algorithm can also be applied to the traveling salesman path problem (TSPP). In this problem, we seek a path visiting all nodes of a set V. When starting and ending points are not both prespecified, an approach similar to Christofides' heuristic can be applied guaranteeing a $\frac{3}{2}$-approximation (this was also noted in [8]). For the case when both starting and ending points are prespecified, we can apply the path-version of the OCTSP heuristic on $K = 3$, $C_1 = \{a\}$, $C_2 = V \backslash \{a, b\}$, and $C_3 = \{b\}$ yielding a $\frac{5}{3}$-approximation algorithm. The example of [8] shows that these bounds are tight. We note that for the TSPP our heuristic is slightly different from the $\frac{5}{3}$-approximation algorithm of [8]. The main difference in the methods is that [8] constructs a spanning tree on all the nodes while our algorithm excludes the endpoints. Also, the endpoints do not play any special role in the matching of [8], while we force them to be matched to dummy nodes (in C_0). The proofs are quite different, in particular, [8] shows that the minimum spanning tree on all nodes augmented with the optimal traveling salesman path can be decomposed into three perfect matchings on the odd degree nodes of the tree. This is in contrast to our Lemma 4.2 which forms the basis of our proof and is a different result.

References

[1] E.M. Arkin, R. Hassin, L. Klein, Restricted delivery problems on a network, Networks 29 (1994) 205–216.
[2] J.A. Chisman, The clustered traveling salesman problem, Comput. Oper. Res. 2 (1975) 115–119.
[3] N. Christofides, Worst-case analysis of a new heuristic for the traveling salesman problem, Report 388, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA, 1976.
[4] G. Cornuéjols, G.L. Nemhauser, Tight bounds for christofides' traveling salesman heuristic, Math. Programming 14 (1978) 116–121.
[5] M. Gendreau, A. Hertz, G. Laporte, An approximation algorithm for the traveling salesman problem with backhauls, Oper. Res. 45 (1997) 639–641.

[6] M. Gendreau, G. Laporte, J.-Y. Potvin, Heuristic for the clustered traveling salesman problem, Combin. Optim. 1 (1996) 41–56.
[7] N. Guttman-Beck, R. Hassin, S. Khuller, B. Raghavachari, Approximation algorithms with bounded performance guarantees for the clustered traveling salesman problem, Working Paper, Department of Statistics and Operations Research, Tel-Aviv University, 1997.
[8] J.A. Hoogeveen, Analysis of Christofides' heuristic: Some paths are more difficult than cycles, Oper. Res. Lett. 10 (1991) 291–295.
[9] F.C.J. Lokin, Procedures for the traveling salesman problem with additional constraints, European J. Oper. Res. 3 (1978) 135–141.
[10] C.H. Papadimitriou, K. Stieglitz, Combinatorial Optimization: Algorithms and Complexity, Prentice-Hall, Englewood Cliffs, NJ, 1982.