An Analysis of Approximations for Finding a Maximum Weight Hamiltonian Circuit

Author(s): M. L. Fisher, G. L. Nemhauser and L. A. Wolsey
Source: *Operations Research*, Vol. 27, No. 4 (Jul. - Aug., 1979), pp. 799-809
Published by: INFORMS
Stable URL: http://www.jstor.org/stable/170295
Accessed: 08/05/2014 11:24

An Analysis of Approximations for Finding a Maximum Weight Hamiltonian Circuit

M. L. FISHER
University of Pennsylvania, Philadelphia, Pennsylvania
G. L. NEMHAUSER
Cornell University, Ithaca, New York
L. A. WOLSEY
University of Louvain, Louvain-La-Neuve, Belgium

(Received May 1977; accepted April 1978)

We give bounds on heuristics and relaxations for the problem of determining a maximum weight hamiltonian circuit in a complete, undirected graph with non-negative edge weights. Three well-known heuristics are shown to produce a tour whose weight is at least half of the weight of an optimal tour. Another heuristic, based on perfect two-matchings, is shown to produce a tour whose weight is at least two-thirds of the weight of an optimal tour. Assignment and perfect two-matching relaxations are shown to produce upper bounds that are, respectively, at most 2 and 3/2 times the optimal value. By defining a more general measure of performance, we extend the results to arbitrary edge weights and minimization problems. We also present analogous results for directed graphs.

A *HAMILTONIAN CIRCUIT* (*tour*) of an undirected graph is a subset of edges that form a circuit containing all of the vertices of the graph. Given a complete, undirected graph $G = (V, E)$, where $V = \{1, \ldots, n\}$, $E = \{e_{ij}: 1 \leqq i < j \leqq n\}$, e_{ij} is the edge joining vertices i and j, and c_{ij} is the weight of e_{ij}, we consider heuristics and relaxations for the problem of finding a maximum weight tour. An instance of this problem is specified by n and the weights c_{ij}. In Section 1 we will assume $c_{ij} \geqq 0$ for all i and j, which makes the problem equivalent to a traveling salesman problem with $c_{ij} \leqq 0$.

Let $x = \{x_{ij}\}$, where $x_{ij} \in \{0, 1\}$, $1 \leqq i < j \leqq n$, and let $E(x) = \{e_{ij}: x_{ij} = 1\}$. The edge induced subgraph $G(x) = (V, E(x))$ of G is a tour if and only if x satisfies

$$\textstyle\sum_{j>i} x_{ij} + \sum_{j<i} x_{ji} = 2, \qquad i = 1, \ldots, n \tag{1}$$

$$\textstyle\sum_{i \in S} \sum_{j \in S, j>i} x_{ij} \leqq |S| - 1, \qquad \text{for all } S \subset V \text{ with } |S| \geqq 3 \tag{2}$$

$$x_{ij} \in \{0, 1\}, \qquad 1 \leqq i < j \leqq n. \tag{3}$$

Operations Research
Vol. 27, No. 4, July-August 1979
0030-364X/79/2704-0799 $01.25

Therefore, the problem of finding a maximum weight tour is

$$Z = \max \textstyle\sum_{i=1}^{n-1} \sum_{j>i} c_{ij}x_{ij}, \qquad x \in X = \{x: x \text{ satisfies (1)–(3)}\}. \qquad (4)$$

This formulation was first given in [3].

Upper bounds on Z can be obtained by solving

$$Z^M = \max \textstyle\sum_{i=1}^{n-1} \sum_{j>i} c_{ij}x_{ij}, \quad x \in X^M = \{x: x \text{ satisfies (1) and (3)}\} \quad (5)$$

and

$$Z^A = \max \textstyle\sum_{i=1}^{n-1} \sum_{j>i} c_{ij}x_{ij}, \qquad x \in X^A = \{x: x \text{ satisfies (1)}, x_{ij} \geqq 0\}. \qquad (6)$$

From the definitions of the constraints in (4)–(6), we have $Z^A \geqq Z^M \geqq Z$. The quantities Z^A and Z^M can be found by fast (polynomially bounded) algorithms. Z^M is the value of a maximum weight *perfect two-matching* in which each edge is included at most once. It can be obtained by an algorithm of Edmonds [4]. It is well-known that problem (6) can be formulated as an *assignment problem.*

We will consider four heuristics for problem (4). The *greedy heuristic* first chooses an edge of maximum weight and then continues to choose edges of maximum weight, subject to the requirement that the collection be contained in a tour.

The *best-neighbor heuristic* [6] specifies an initial vertex and chooses an edge of maximum weight adjacent to it. It then continues to select an edge of maximum weight adjacent to the vertex just reached, subject to the requirement that the collection be contained in a tour.

The *two-interchange heuristic* [8] begins with an arbitrary tour and tries to improve it by deleting two edges and replacing them by the unique two edges that yield another tour. The procedure continues until no such replacement yields a tour of larger weight.

The *matching heuristic* begins by solving (5). If the maximum weight, perfect two-matching is not a tour, then we delete the smallest weight edge from each cycle of the two-matching and replace the deleted edges by any subset of edges that yield a tour.

Each of these heuristics can, for a given problem instance, produce tours of different weight, depending on how unspecified selections are resolved. For example, in the greedy and best-neighbor heuristics an edge of maximum weight that can be selected may not be unique; also, there can be arbitrary choices in the matching and two-interchange heuristics. Therefore, for a problem instance each heuristic defines a non-empty collection of tours. Since we are concerned with worst-case analysis, we assume that a minimum weight tour is selected from the collection of tours that can be produced. With this assumption we have a tour of specified weight associated with a given heuristic and problem instance. We denote this weight by Z_G, Z_N, Z_I, and Z_M for the greedy, best-neighbor, two-interchange, and matching heuristics, respectively. These values depend, of course, on the problem instance, but we have suppressed this dependence for notational simplicity.

In Section 1 our objective is to compute bounds that are valid for all problem instances and have the form

$$Z_H \geqq \alpha_H{}^R Z^R, \tag{7}$$

where H refers to one of the heuristics, G, N, I, or M; R refers to one of the relaxations M or A; and $\alpha_H{}^R$ is a real number between zero and one, independent of the problem instance. Assuming $Z^R > 0$, (7) says that the heuristic H is guaranteed to achieve at least 100 $\alpha_H{}^R$% of Z^R and thus, of course, at least 100 $\alpha_H{}^R$% of the optimal value Z. Specifically, we obtain the results $Z_G \geqq \frac{1}{2}Z^A$, $Z_N \geqq \frac{1}{2}Z^A$, $Z_I \geqq \frac{1}{2}Z^A$ and $Z_M \geqq \frac{2}{3}Z^M$. These bounds are sharp and imply $Z_G \geqq \frac{1}{2}Z$, $Z_N \geqq \frac{1}{2}Z$, $Z_I \geqq \frac{1}{2}Z$, $Z_M \geqq \frac{2}{3}Z$, $Z^A \leqq 2Z$ and $Z^M \leqq \frac{3}{2}Z$. The result $Z_G \geqq \frac{1}{2}Z$ is also implicit in the works of Jenkyns [5] and Korte [7].

For any optimizing algorithm the problem of finding a minimum weight hamiltonian circuit (the traveling salesman problem) is mathematically equivalent to the maximization problem. Unfortunately, this equivalence does not extend to the analysis of approximations. In particular, our bounds are not valid for the traveling salesman problem, which would have negative edge weights when converted to a maximization problem. Moreover, it is shown in [10] that the problem of finding a tour guaranteed to have a weight within a fixed percentage of the minimum weight is itself NP-hard. However, in Section 2 we define a measure of the performance of an approximation that is more general than the percentage of optimality measure. Using this measure, we extend the results of Section 1 to problems in which edges can have negative weights and to minimization problems.

In Section 3 all the approximations are generalized for the case of directed graphs. Let Z, Z_H, and Z^R now denote optimal, heuristic, and relaxation values for the directed problem. We give a series of examples for which Z_N/Z and Z_I/Z approach 0 as n approaches infinity and show that $Z^A = Z^M \leqq 2Z_M$ and $Z_G/Z^A \geqq \frac{1}{3}$. This last inequality implies $Z_G/Z \geqq \frac{1}{3}$, which also follows from results in [5] and [7].

1. THE BOUNDS FOR NON-NEGATIVE EDGE WEIGHTS

THEOREM 1. (a) $Z_G \geqq \frac{1}{2}Z^A$, $Z_N \geqq \frac{1}{2}Z^A$, $Z_I \geqq \frac{1}{2}Z^A$.

(b) *Each of the bounds in (a) is sharp in the sense that (i) there exists a graph for which* $Z = \frac{1}{2}Z^A$ *and (ii) there exists a family of* $(3k - 1)$-*vertex graphs such that* $Z_G = Z_N = Z_I = [k/(2k - 1)]Z$, $k = 3, 4, \ldots$.

Proof. (a) Let $u = (u_1, \ldots, u_n)$ be a vector of dual variables for the constraints (1). By Lagrangian duality

$$Z^A \leqq \max_{x_{ij} \geqq 0} \sum_{i=1}^{n-1} \sum_{j>i} (c_{ij} - u_i - u_j)x_{ij} + 2 \sum_{i=1}^{n} u_i.$$

For $H \in \{G, N, I\}$ we will give $u(H)$ such that $c_{ij} - u_i(H) - u_j(H) \leqq 0$ and $\sum_{i=1}^{n} u_i(H) = Z_H$, which implies $Z^A \leqq 2 \sum_{i=1}^{n} u_i(H) = 2Z_H$. We assume, without loss of generality, that the edges selected by heuristic H are e_{12}, $e_{23}, \ldots, e_{n-1,n}, e_{1n}$.

In the best-neighbor heuristic suppose that the edges have been selected in the order $e_{12}, e_{23}, \ldots, e_{n-1,n}, e_{1n}$. Then[1] set $u_i(N) = c_{i,i+1}$, $i = 1, \ldots, n$ so that $\sum_{i=1}^{n} u_i(N) = Z_N$ and $u(N) \geqq 0$. Consider indices i and j satisfying $1 \leqq i < j \leqq n$. Since, by hypothesis, the partial tour reached vertex i before vertex j, we have $c_{ij} - u_i(N) \leqq 0$. But then $u_j(N) \geqq 0$ implies $c_{ij} - u_i(N) - u_j(N) \leqq 0$.

Set $u_i(G) = ½(c_{i-1,i} + c_{i,i+1})$, $i = 1, \ldots, n$ so that $\sum_{i=1}^{n} u_i(G) = Z_G$ and $u(G) \geqq 0$. Now if $j = i + 1$, then $u_i(G) + u_j(G) = c_{ij} + ½(c_{i-1,i} + c_{j,j+1}) \geqq c_{ij}$. If $j \neq i + 1$, then since e_{ij} was not selected by the greedy heuristic, c_{ij}

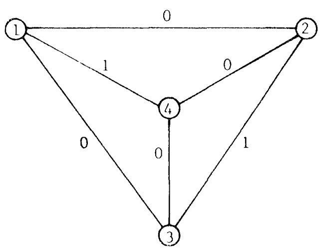

Figure 1. Example for Theorem 1 (b) (i).

is equal to or less than the second largest number in the set $\{c_{i-1,i}, c_{i,i+1}, c_{j-1,j}, c_{j,j+1}\}$. This implies that $c_{ij} \leqq u_i(G) + u_j(G)$.

Set $u(I) = u(G)$ so that $\sum_{i=1}^{n} u_i(I) = Z_I$ and $u(I) \geqq 0$. If $j = i + 1$, then $u_i(I) + u_j(I) \geqq c_{ij}$ as above for the greedy heuristic. If $j \neq i + 1$, by the interchange property, $c_{ij} + c_{i+1,j+1} \leqq c_{i,i+1} + c_{j,j+1}$ and $c_{ij} + c_{i-1,j-1} \leqq c_{i-1,i} + c_{j-1,j}$. Hence, $c_{ij} \leqq ½(c_{i-1,i} + c_{i,i+1} + c_{j-1,j} + c_{j,j+1}) = u_i(I) + u_j(I)$.

(b) (i) In the graph of Figure 1, an optimal solution to problem (6) is given by $x_{14} = x_{23} = 2$, $x_{ij} = 0$ otherwise, which yields $Z^A = 4$. Clearly, $Z = 2$.

(b) (ii) For $k \geqq 2$, we define G_k to be the complete graph on $3k - 1$ vertices with edge weights of 1 for all edges in $E_1 \cup E_2$ where $E_1 = \{e_{i,i+1}: i = 1, \ldots, k\}$ and $E_2 = \{e_{1,k+1}\} \cup \{e_{k-j,k+2j+2}, e_{k-j,k+2j+3}: j = 0, \ldots, k-2\}$. All other edges have weight zero. The graphs G_2 and G_3 are shown in Figure 2. Solid edges have weight 1; dashed and omitted edges have weight zero.

[1] Throughout, index arithmetic is performed modulo n. Thus, if $i = n$, then $i + 1 = 1$. In this case we sometimes abuse notation by denoting the edge e_{1k} by e_{k1}.

An optimal tour for G_k is given by the vertex sequence $(1, k + 1, \{k + 2 + 2j, k - j, k + 3 + 2j, j = 0, \ldots, k - 2\}, 1)$. The edges of this tour of positive weight are precisely those contained in E_2, and hence the tour has weight $|E_2| = 2(k - 1) + 1 = 2k - 1$. To prove its optimality, observe that $Z \leqq 2k - 1$ because every edge in the set $E_1 \cup E_2 - \{e_{1,k+1}\}$ is incident to a vertex in $\{2, \ldots, k\}$. Hence by constraint (1) a feasible tour can contain at most $2k - 2$ of these edges.

The greedy and best-neighbor heuristics can select the edges $T = \{e_{i,i+1}: i = 1, \ldots, n\}$ in natural order. The edges of this tour with positive weight are those contained in E_1. Thus, $Z_G = Z_N = |E_1| = k$.

Finally, we show that for $k \geqq 3$ the tour with edge set T cannot be improved by 2-interchanges so that $Z_I = k$. Consider an arbitrary 2-interchange and denote the deleted edges by $e_{i,i+1}$ and $e_{j,j+1}$ with $i + 1 < j$, so that the unique replacement edges are e_{ij} and $e_{i+1,j+1}$. For this

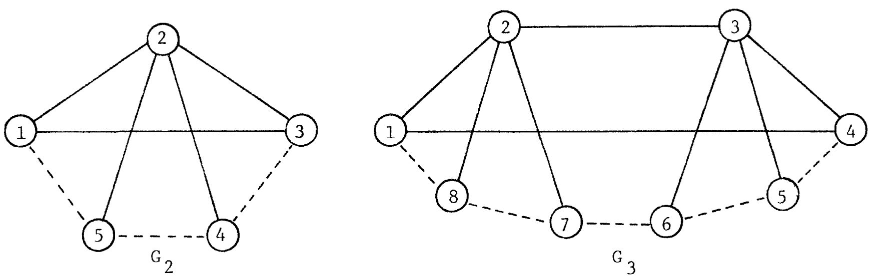

Figure 2. Examples for Theorem 1 (b) (ii).

interchange to yield a tour of larger weight than the weight of T, either (1) both deleted edges are in $T - E_1$ and at least one replacement edge is in E_2, or (2) one deleted edge is in $T - E_1$, the other is in E_1, and both replacement edges are in E_2. Case (1) is not possible since for any $e_{rs} \in E_2$, $E_1 \cup \{e_{rs}\}$ is not contained in any tour. Case (2) is not possible since, for $k \geqq 3$, $e_{ij} \in E_2$ implies $e_{i+1,j+1} \notin E_2$.

Note that the family described in part (b) of Theorem 1 implies that we cannot improve the bounds given in part (a) by using any combination of the greedy, best-neighbor, and interchange heuristics.

THEOREM 2. (a) $Z_M \geqq \frac{2}{3} Z^M$.

(b) *This bound is tight. In fact, there exist graphs for which* $Z_M = Z = \frac{2}{3} Z^M$ *and graphs for which* $Z_M = \frac{2}{3} Z = \frac{2}{3} Z^M$.

Proof. (a) Consider a perfect two-matching that is not a tour. Let $P \geqq 2$ be the number of cycles, let C_p be the edge set of cycle p, and let $k_p = |C_p|$, $p = 1, \ldots, P$.

Then

$$Z_M \geqq \sum_{p=1}^{P} ((k_p - 1)/k_p) \sum_{(i,j)\in C_p} c_{ij} \geqq (2/3) \sum_{p=1}^{P} \sum_{(i,j)\in C_p} c_{ij} = 2/3 Z^M,$$

where the first inequality follows from the deletion of the smallest weight edge in each cycle and the second from $k_p \geqq 3$.

(b) Consider the graph of Figure 3, in which the solid edges have weight equal to one, omitted edges have weight equal to zero, and the two dashed edges have weight of zero or one as explained below. A maximum weight, perfect two-matching is given by the solid edges so that $Z^M = 6$. If both dashed edges have weight equal to zero, then $Z = 4$ and $Z = 2/3 Z^M$. On the other hand, if both dashed edges have weight equal to one, then $Z = 6$. But there are several tours of weight 4 that contain two solid edges from each triangle. Thus, with arbitrary tie-breaking we can have $Z_M = 4$ and $Z_M = 2/3 Z$.

The proof of Theorem 2 shows, more generally, that if $Z^{M(k)}$ is the value of maximum weight, perfect two-matching whose smallest cycle contains no fewer than k edges and $Z_{M(k)}$ is the weight of a tour constructed from

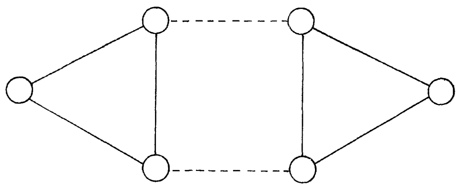

Figure 3. Example for Theorem 2 (b).

this two-matching, then $Z_{M(k)} \geqq [(k - 1)/k]Z^{M(k)}$. However, no fast algorithms are known for finding such perfect two-matchings for $k > 3$. Note that by allowing two parallel edges between each pair of vertices and regarding two parallel edges as a cycle of length two, we obtain $Z^{M(2)} = Z^A$. Thus, this type of heuristic can be used in conjunction with the assignment problem to obtain $Z_{M(2)} \geqq 1/2 Z^A$.

2. GENERALIZATION FOR ARBITRARY EDGE WEIGHTS

Here we extend the results of Section 1 by generalizing the heuristics and obtaining bounds on their performance for graphs with arbitrary edge weights. These bounds may, however, depend on the problem data. (See [2] for a related approach to a different problem.)

Define $Y = \{y: y_i + y_j \leqq c_{ij}$ for all $i, j\}$ and $W = \{w: w_i + w_j \geqq c_{ij}$ for all $i, j\}$, and let y^* and w^* be optimal solutions of $Z_* = \max\{2\sum_{i=1}^{n} y_i: y \in Y\}$ and $Z^* = \min\{2\sum_{i=1}^{n} w_i: w \in W\}$, respectively.

Let H be some heuristic for the problem of the previous section for which $Z_H \geqq \lambda Z$ when all edge weights are non-negative. For a problem with arbitrary edge weights and any $y \in Y$, define the heuristic $H(y)$ as follows. Apply heuristic H with edge weights $c_{ij} - y_i - y_j \geqq 0$ and take the resulting tour as the solution. Let $Z_{H(y)}$ be the weight of this tour relative to the original edge weights. Note that the tours produced by all of the heuristics of Section 1 can depend on y.

PROPOSITION 1. *If we apply heuristic $H(y)$ to the problem of finding a maximum weight tour with arbitrary edge weights, then $Z_{H(y)} \geqq \lambda Z + (1 - \lambda)\,[2\sum_{i=1}^{n} y_i]$ for all $y \in Y$.*

Proof. Let Z' and Z_H' be the values of the optimal and heuristic solutions relative to edge weights $c_{ij} - y_i - y_j$, so that $Z_H' \geqq \lambda Z'$. Now $Z' = Z - 2\sum_{i=1}^{n} y_i$, $Z_H' = Z_{H(y)} - 2\sum_{i=1}^{n} y_i$, and the result follows by substitution.

Note that $y = y^*$ gives the largest value of $2\sum_{i=1}^{n} y_i$ and thus yields the largest lower bound of $Z_{H(y^*)} \geqq \lambda Z + (1 - \lambda)Z_*$. Some limited computational experience suggests that the value y^* also tends to give improved values for the heuristic solutions. We also note that when the edge weights are non-negative, $0 \in Y$; hence $H(0)$ gives the results of the last section. However, even with non-negative edge weights a better bound may be obtained by choosing $y \neq 0$.

Now consider the problem of finding a minimum weight tour. For all $w \in W$, define the heuristic $H(w)$ as follows. Apply heuristic H to the maximization problem with edge weights $w_i + w_j - c_{ij} \geqq 0$ and take the resulting tour as the solution. Let $Z_{H(w)}$ be the weight of this tour relative to the original weights. Let Z_s be the value of a minimum weight tour. Analogous to Proposition 1 we have

PROPOSITION 2. *If we apply heuristic $H(w)$ to the problem of finding a minimum weight tour with arbitrary edge weights, then $Z_{H(w)} \leqq \lambda Z_s + (1 - \lambda)(2\sum_{i=1}^{n} w_i)$ for all $w \in W$.*

Choosing $w = w^*$, we obtain the least upper bound of $Z_{H(w^*)} \leqq \lambda Z_s + (1 - \lambda)Z^*$. Here, however, if the edge weights are non-negative, as in the traveling salesman problem, $0 \notin W$ and we have not obtained a bound that is independent of the data. In fact, for the traveling salesman problem, it is shown in [10] that finding a tour whose weight is no more than kZ_s is an NP-hard problem for all $k \geqq 1$. On the other hand, if the edge weights satisfy the triangle inequality, Christofides [1] gives a heuristic that finds a tour whose weight does not exceed $3/2$ times the minimum weight. Other results for traveling salesman problem heuristics are given in [9].

3. GENERALIZATION FOR DIRECTED GRAPHS

In this section we suppose that $G = (V, E)$ is a complete *directed* graph with vertices $V = \{1, \ldots, n\}$ edges $E = \{e_{ij}: 1 \leqq i \leqq n, 1 \leqq j \leqq n, i \neq j\}$, and weight $c_{ij} \geqq 0$ on edge e_{ij}. We will restate all the relaxations and heuristics for the directed problem and analyze their worst-case performance. We use notation from the introduction with the understanding that all symbols are now defined for the directed problem.

Let $x = \{x_{ij}\}$, where $x_{ij} \in \{0, 1\}$, $1 \leqq i \leqq n$, $1 \leqq j \leqq n$, $i \neq j$, and let $E(x) = \{e_{ij}: x_{ij} = 1\}$. The edge-induced subgraph $G(x) = (V, E(x))$ of G is a directed tour if and only if x satisfies the following conditions, which are analogous to (1)–(3) for the undirected problem.

$$\textstyle\sum_{j \neq i} x_{ij} = 1, \qquad i = 1, \ldots, n \tag{1a$'$}$$

$$\textstyle\sum_{i \neq j} x_{ij} = 1, \qquad j = 1, \ldots, n \tag{1b$'$}$$

$$\textstyle\sum_{i \in S} \sum_{j \in S, i \neq j} x_{ij} \leqq |S| - 1, \qquad \text{for all } S \subset V \text{ with } |S| \geqq 2 \tag{2$'$}$$

$$x_{ij} \in \{0, 1\},\ 1 \leqq i \leqq n,\ 1 \leqq j \leqq n,\ i \neq j. \tag{3$'$}$$

Analogous to (4)–(6), the problem of finding a maximum weight tour is

$$Z = \max \textstyle\sum_{i=1}^{n} \sum_{j \neq i} c_{ij} x_{ij}, \qquad x \in X = \{x: x \text{ satisfies } (1')\text{–}(3')\}, \tag{4$'$}$$

and upper bounds on Z can be obtained by solving

$$Z^M = \max \textstyle\sum_{i=1}^{n} \sum_{j \neq i} c_{ij} x_{ij}, \qquad x \in X^M = \{x: x \text{ satisfies } (1'), (3')\} \tag{5$'$}$$

and

$$Z^A = \max \textstyle\sum_{i=1}^{n} \sum_{j \neq i} c_{ij} x_{ij}, \qquad x \in X^A = \{x: x \text{ satisfies } (1'),\ x_{ij} \geqq 0\}. \tag{6$'$}$$

The four heuristics given for the undirected problem can also be restated. The greedy and best-neighbor heruistics are applicable as originally stated, and the matching heuristic requires only the replacement of (5) by (5$'$). Taken literally, the two-interchange heuristic is not applicable to the directed problem because it is not possible to replace two edges of a directed tour without changing other edges in the tour. However, by permitting a broader class of interchanges one can define a natural two-interchange heuristic for the directed problem. Without loss of generality, assume that the current tour consists of the edges $\{e_{12}, e_{23}, \ldots e_{n-1,n}, e_{n1}\}$. When two arbitrary edges $e_{i-1,i}$ and $e_{j-1,j}$, $i < j$, are deleted, three possible replacements are considered, as indicated in Figure 4.

(i) Edges a_1 and a_2 are added to the tour and the current paths from j to $i - 1$ and i to $j - 1$ are reversed. (Each edge $(k, k + 1)$ on the path is replaced by $(k + 1, k)$.)

(ii) Edges b_1 and b_2 are added and the path from j to $i - 1$ is reversed.

(iii) Edges c_1 and c_2 are added and the path from i to $j - 1$ is reversed.

As with the undirected problem, we let Z_G, Z_N, Z_I, and Z_M denote the minimum weight that can be produced by greedy, best-neighbor, interchange, and matching heuristics for a given problem instance. Several observations about worst-case performance are straightforward. The upper bounds Z^A and Z^M are equal. Because the solution to (5′) may have cycles with two edges, $Z_M \geqq \frac{1}{2}Z^M$. There exist simple examples analogous to those given in the proof of Theorem 2(b) for which $Z_M = Z = \frac{1}{2}Z^M$ and $Z_M = \frac{1}{2}Z = \frac{1}{2}Z^M$. The class of examples with edge weights given by $c_{i,i+1} = 1$, $i = 1, \ldots, n$, $c_{1n} = 1$ and all other $c_{ij} = 0$ has $Z_N = 1$ and $Z = n$.

These simple observations settle the worst-case performance of all approximations under consideration except the greedy and interchange heuristics, which are addressed in Theorems 3 and 4.

Theorem 3. (a) $Z_G \geqq \frac{1}{3}Z^A$.

(*b*) *There exists a 3-vertex graph for which* $Z_G = 1$ *and* $Z = Z^A = 3$.

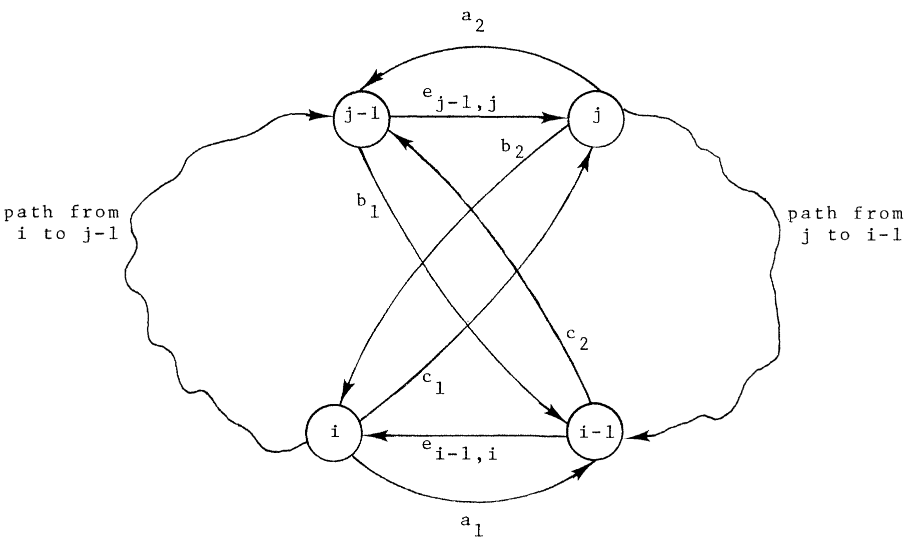

Figure 4. Two-Interchange for directed graphs.

Proof. (a) Let $u = (u_1, \ldots, u_n)$ and $v = (v_1, \ldots, v_n)$ be vectors of multipliers for the constraints (1a′) and (1b′). By Lagrangian duality

$$Z^A \leqq \max_{x_{ij} \geqq 0} \textstyle\sum_{i=1}^n \sum_{j \neq i} (c_{ij} - u_i - v_j)x_{ij} + \sum_{i=1}^n (u_i + v_i).$$

We will give u and v such that $\sum_{i=1}^n (u_i + v_i) = 3Z_G$ and $c_{ij} - u_i - v_j \leqq 0$, which implies $Z^A \leqq 3Z_G$. Assume without loss of generality that the greedy heuristic selects edges $e_{12}, e_{23}, \ldots, e_{n-1,n}, e_{n1}$ and set $u_i = c_{i,i+1} + \frac{1}{2}c_{i-1,i}$ and $v_j = c_{j-1,j} + \frac{1}{2}c_{j,j+1}$. Thus, $\sum_{i=1}^n (u_i + v_i) = 3Z_G$.

Obviously, $c_{ij} - u_i - v_j \leqq 0$ if $j = i + 1$. If $j = i - 1$, then $c_{ij} - u_i - v_j = c_{i,i-1} - c_{i-2,i-1} - c_{i-1,i} - c_{i,i+1}$. The first edge chosen by the greedy heuristic from the set $\{e_{i-2,i-1}, e_{i-1,i}, e_{i,i+1}\}$ has weight no less than $c_{i,i-1}$; hence $c_{ij} - u_i - v_j \leqq 0$. Finally, suppose $|j - 1| \geqq 2$. Then there are four edges of the tour produced by the greedy heuristic incident to i and j, as shown in

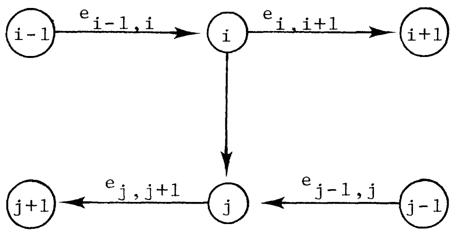

Figure 5. Illustration of Theorem 3 (a).

Figure 5. We distinguish three possibilities depending on the order in which these edges are chosen by the greedy heuristic.

(i) $e_{i-1,i}$ and $e_{j,j+1}$ are chosen as the first two edges in either order. The greedy rule then implies $c_{i-1,i} \geqq c_{ij}$ and $c_{j,j+1} \geqq c_{ij}$.

(ii) $e_{i-1,i}$ and $e_{j,j+1}$ are not the first two edges chosen and $e_{i,i+1}$ is chosen before $e_{j-1,j}$. Then $c_{i,i+1} \geqq c_{ij}$.

(iii) $e_{i-1,i}$ and $e_{j,j+1}$ are not the first two edges chosen and $e_{j-1,j}$ is chosen before $e_{i,i+1}$. Then $c_{j-1,j} \geqq c_{ij}$.

In each of these cases $c_{ij} - u_i - v_j \leqq 0$ follows directly from the definition of u_i and v_j.

(b) In the graph shown in Figure 6 (omitted edges have weight 0), the greedy heuristic can choose the tour with vertex sequence 1, 3, 2, 1 while 1, 2, 3, 1 is optimal.

THEOREM 4. *There exists a family of graphs with $n \geqq 5$ and odd for which $Z_I = 4$ and $Z = n$.*

Proof. All edges have weight 0 or 1. The edges with weight 1 are (1, 2), (2, 3), $((n + 1)/2, (n + 3)/2)$, $((n + 3)/2, (n + 5)/2)$, and $(i, i - 2)$, $i = 1, \ldots, n$. The graph for $n = 7$ is displayed in Figure 7. (Edges e_{ij}, $j \neq i + 1$ with $c_{ij} = 0$ are not shown.)

The tour $1, n - 1, n - 3, \ldots, 4, 2, n, n - 2, \ldots, 3, 1$ has weight n and is clearly optimal. We will show that the tour $1, \ldots, n$ with weight 4 cannot be improved by 2-interchanges. Consider the replacement of general edges $e_{i,i+1}$ and $e_{j,j+1}$, with i and j defined so that the path $e_{i,i+1}$, $e_{i+1,i+2}, \ldots, e_{j-1,j}$ has fewer edges than the path $e_{j,j+1}, e_{j+1,j+2}, \ldots, e_{i-1,i}$. This interchange can improve the tour only if at least one of the entering

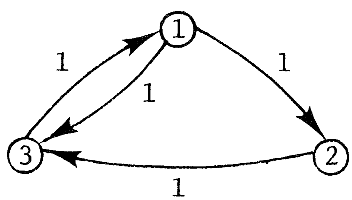

Figure 6. Example for Theorem 3 (b).

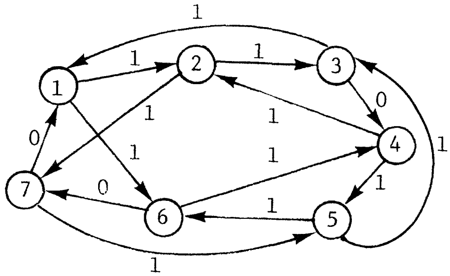

Figure 7. Example for Theorem 4 with $n = 7$.

edges has weight 1. This happens only if $j = i + 2 (\mathrm{mod}\ n)$, $e_{i+2,i}$ and $e_{i+3,i+1}$ enter the tour and the path from $i + 3(\mathrm{mod}\ n)$ to i is reversed. But this interchange is not improving because either (a) reversing the path from $i + 3(\mathrm{mod}\ n)$ to i reduces the tour weight by 2 or (b) $n = 5$ and $c_{i,i+1} = c_{i+2,i+3} = 1$.

ACKNOWLEDGMENTS

This work was supported in part by NSF Grant ENG76-20274 to the University of Pennsylvania and NSF Grant ENG75-00568 to Cornell University.

REFERENCES

1. N. Christofides, "Worst-Case Analysis of a New Heuristic for the Traveling-Salesman Problem," Carnegie Mellon Report 388, February 1976.
2. G. Cornuejols, M. L. Fisher, and G. L. Nemhauser, "Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms," *Management Sci.* **23,** 789–810 (1977).
3. G. B. Dantzig, D. R. Fulkerson, and S. M. Johnson, "Solution of a Large Scale Traveling Salesman Problem," *Opns. Res.* **2,** 393–410 (1954).
4. J. Edmonds, "Maximum Matching and a Polyhedron with 0-1 Vertices," *J. Res. Natl. Bur. Stat.* **69B,** 125–130 (1965).
5. T. A. Jenkyns, "The Efficiency of the Greedy Algorithm," *Proc. 7th S.E. Conf. on Combinatorics, Graph Theory and Computing,* pp. 341–350, Utilitas Mathematica, Winnipeg, 1976.
6. L. L. Karg and G. L. Thompson. "A Heuristic Approach to Traveling Salesman Problems," *Management Sci.* **10,** 225–248 (1964).
7. B. Korte and D. Hausmann, "An Analysis of the Greedy Heuristic for Independence Systems," Report No. 7645, Institute for Econometrics and Operations Research, University of Bonn, May 1976.
8. S. Lin, "Computer Solution of the Traveling Salesman Problem," *Bell System Tech. J.* **44,** 2245–2269 (1965).
9. D. J. Rosenkrantz, R. E. Stearns, and P. M. Lewis, "An Analysis of Several Heuristics for the Traveling Salesman Problem," *Siam J. Comput.* **6,** 563–581 (1977).
10. S. Sahni and T. Gonzalez, "*P*-Complete Approximation Problems," *J. Assoc. Comput. Mach.* **23,** 555–565 (1976).