SIAM J. CONTROL
Vol. 7, No. 1, February 1969

ON THE TOURS OF A TRAVELING SALESMAN*

KATTA G. MURTY†

Abstract. Adjacency properties of tours on their convex hull are discussed. A rule is given by which it can be tested whether any two tours are adjacent vertices on this convex hull or not. Based on this rule an algorithm is described for generating all the adjacent tours of a given tour.

1. Introduction. The traveling salesman problem is the problem of finding a minimal cost tour covering a set of n cities given the costs of traveling between every possible pair of cities. Here a tour is a path covering all the cities, each city being covered once and only once in the path. A precise mathematical definition of a tour is given later on.

Let us denote the cities by $1, 2, \cdots, n$. We put

$$x_{ij} = \begin{cases} 1 & \text{if in the tour the salesman goes from } i \text{ to } j, \\ 0 & \text{otherwise.} \end{cases}$$

Then the matrix $X = (x_{ij})$, which is a cyclic permutation matrix, represents the tour.

If in a tour the salesman goes from i_1 to i_2, then (i_1, i_2) is called an *arc* or *cell* in that tour.

We use the letters i, j to denote cities.

Let c_{ij} = cost of traveling from i to j, $i \neq j$; $c_{ii} = \alpha$, an arbitrarily chosen very large positive number. Then $C = (c_{ij})$ is the cost matrix for the problem and this is given. Starting from any city, the salesman can choose to go to any of the remaining $n-1$ cities initially. From that city he can go to any of the remaining $n-2$ cities and so on. Thus, the total number of distinct tours is $(n-1)!$. The set of all possible tours is denoted by T and their convex hull by K_T. We shall use the letters t or s to denote tours.

Given any tour t_1 we describe an algorithm in this paper for generating the adjacent tours of t_1 on the convex polyhedron K_T.

2. Notation. The convex polyhedron K_A is the set of all feasible solutions

$$\begin{aligned} X &= (x_{ij}), && \text{an } n \times n \text{ matrix, where the } x_{ij} \text{ satisfy} \\ \sum_i x_{ij} &= 1, && j = 1, \cdots, n, \\ \sum_j x_{ij} &= 1, && i = 1, \cdots, n, \\ x_{ij} &\geqq 0. \end{aligned} \tag{1}$$

* Received by the editors November 7, 1967, and in revised form May 6, 1968.

† Department of Industrial Engineering, University of Michigan, Ann Arbor, Michigan 48104.
his research was supported by the National Science Foundation under Grant GP-4593 with the niversity of California at Berkeley.

An extreme point of K_A is called an *assignment*. Every assignment is a permutation matrix, i.e., it is an $n \times n$ matrix with a single nonzero entry equal to 1 in each row and column. We use the letters a, b to denote assignments.

Occasionally it is convenient to denote an assignment by its unit cells, i.e., the cells in the matrix X representing the assignment which have unit entries in them. All the other cells have zero entries, of course. Thus,

$$a = \{(1, j_1), \cdots, (n, j_n)\} \tag{2}$$

is an assignment, where $j_1, \cdots, j_n$ is a permutation of the numbers $1, 2, \cdots, n$.

We also write

$$(r, j_r) \in a$$

which means that in the matrix X representing the assignment a, the entry in the cell (r, j_r) is 1. The same fact is also expressed by saying that (r, j_r) is a cell in the assignment a, or that the assignment a has an allocation in the cell (r, j_r).

For any assignment a we shall denote specifically by $\{a\}$ the set of cells of a, i.e., if a is the assignment given by (2), then

$$\{a\} = \{(1, j_1), \cdots, (n, j_n)\}.$$

A *tour* is an assignment whose cells can be written down as a complete path covering all the cities and then returning to the starting point, without any subtours. In other words a tour t is an assignment whose cells can be written down as

$$t = \{1, j_1), (j_1, j_2), \cdots, (j_{n-1}, 1)\},$$

where $j_1, j_2, \cdots, j_{n-1}$ is a permutation of the numbers $2, 3, \cdots, n$. To be specific, we can say that t is a tour covering the cities $\{1, 2, \cdots, n\}$. Thus, $T \subset A$ and $K_T \subset K_A$.

By a *self-loop at a city* we mean a cell of the form (i, i). It corresponds to an allocation along the principal diagonal of the matrix X representing an assignment. Any cell of the form (i, i) is also called a *diagonal cell*. Any cell of the form (i, j) where $i \neq j$ is called a *nondiagonal cell*.

Pick any subset S of the cities $\{1, 2, \cdots, n\}$ such that $S \subset \{1, 2, \cdots, n\}$ and $S \neq \{1, 2, \cdots, n\}$. Then any tour covering the cities in S only is known as a *subtour*.

A *nontour* is an assignment which is not a tour *and which has no self-loops*. In other words it is an assignment without any allocation along the principal diagonal, whose unit cells constitute at least two subtours.

D.A. is an abbreviation for the diagonal assignment which is the assignment represented by the unit matrix.

Two assignments a_1 and a_2 are called *adjacent assignments* if the line segment joining them is an edge of the convex polyhedron K_A, i.e., if and only if every point of the form $\lambda a_1 + (1 - \lambda) a_2$ for all $0 \leqq \lambda \leqq 1$ has a unique representation as a convex combination of assignments.

Two tours t_1 and t_2 are called *adjacent tours* if the line segment joining them forms an edge of the convex polyhedron K_T, i.e., if and only if every point of the form $\lambda t_1 + (1 - \lambda)t_2$ for all $0 \leqq \lambda \leqq 1$ has a unique representation as a convex combination of tours. Since $K_T \subset K_A$, two tours which are not adjacent as assignments may be adjacent as tours.

Suppose the tour $t = \{(i_1, i_2), (i_2, i_3), \cdots, i_n, i_1)\}$. Then the tour $\bar{t} = \{(i_2, i_1), (i_3, i_2), \cdots, (i_1, i_n)\}$ is called the *reflection* of the tour t.

The θ-loop of a nonbasic cell. Consider a basis for (1) representing an assignment a. Such a basis consists of $2n - 1$ basic cells, the n cells of a which are at value 1 and $n - 1$ other independent cells which are at value 0 in the basis.

Let us try to obtain a new basis by bringing the nonbasic cell (i_1, j_1) into the basis. To do this, we put an entry of $+\theta$ in the nonbasic cell (i_1, j_1). Since the sum of all the entries in each row and column should equal 1, we should put a $-\theta$ entry somewhere else in column j_1 and row i_1. Make all these subsequent entries among the basic cells only. Taking up from column j_1, put alternate entries of $-\theta$ and $+\theta$ among columns and rows until the $+\theta$ entry in each row and column is canceled by a $-\theta$ entry. The set of all the basic cells along the $-\theta$ and $+\theta$ path is called the *θ-loop* of the nonbasic cell (i_1, j_1) in this basis. The maximum value which θ can take without the resulting solution violating the nonnegativity constraint of the x_{ij}'s is known as the *value* with which the nonbasic cell (i_1, j_1) enters the basis.

ZBC is an abbreviation for any zero-valued basic cell in any basis for (1). In any basis for (1), if a nonbasic cell (i_1, j_1) enters the basis with a value of zero, then it can be brought into the basis as a ZBC replacing any of the old ZBC's in its θ-loop. If it enters the basis with a unit value, then it can be brought into the basis by replacing one of the unit-valued cells in its θ-loop. But in this process some of the other unit-valued basic cells might become ZBC's.

3. Mathematical theory. We shall first of all look at a characterization of the set of all tours T as a subset of the set of all assignments A. This leads to the corollary that the traveling salesman problem is a special case of the general problem of finding the minimal cost adjacent vertex of a given vertex in a linear programming problem. This can be solved easily when the linear programming problem is nondegenerate. But if the given vertex is a degenerate vertex, the problem of finding its minimal cost adjacent vertex becomes very hard, which explains the difficulty in solving the traveling salesman problem.

THEOREM 1. *Considering* K_A, *the set of all feasible solutions to* (1), *we have*:
(i) *all tours are adjacent assignments to D.A.*;
(ii) *every nontour is not an adjacent assignment of D.A.*;
(iii) *the class of all adjacent assignments of D.A. consists of*
(a) *all the tours*,
(b) *all the subtours in a smaller number of cities with self-loops at the remaining cities.*

This theorem has been proved by Heller in [1].

(i) can be proved by taking a basis for (1) representing the D.A., with $(1, 2), (2, 3), \cdots, (n - 1, n)$ as ZBC's. In this basis for (1) if the nonbasic cell $(n, 1)$ is brought into the basis, the tour $\{(1, 2), (2, 3), \cdots, (n, 1)\}$ is obtained. Thus the tour $\{(1, 2), (2, 3), \cdots, (n, 1)\}$ is obtained by performing a single pivot in a basis for (1) representing the D.A., and hence it is an adjacent assignment of the D.A. A similar argument holds for every other tour.

(iii) is proved by a similar argument.

(ii) follows because any $2n - 1$ of the cells among those of the D.A. and any nontour are not linearly independent and hence cannot constitute a basis for (1). Thus any nontour cannot be obtained by a single pivot step in any basis representing the D.A., which proves (ii).

COROLLARY 1. *The traveling salesman problem is a special case of the following problem: given a feasible vertex V (i.e., an extreme point) in a linear programming problem, find the minimal cost adjacent vertex of V.*

Proof. Consider the assignment problem with C as the cost matrix, i.e., the problem of minimizing $Z = \sum_{i,j} c_{ij}x_{ij}$ subject to the constraints (1).

The cost of any self-loop is α, which is a very large positive number. Hence, (iii) of Theorem 1 implies that the minimal cost tour is the minimal cost adjacent assignment of D.A.

COROLLARY 2. *Consider any assignment a which has no self-loops:*

$$a = \{(i_1, j_1), \cdots, (i_n, j_n)\}, \qquad i_r \neq j_r, \quad r = 1, \cdots, n.$$

If the cells of a together with any $n - 1$ of the diagonal cells as ZBC's form a basis for the system of constraints (1), *then a must be a tour and conversely.*

Proof. This follows easily because if a contains at least two subtours, then any $2n - 1$ of the cells $\{(1, 1), \cdots, (n, n), (i_1, j_1), \cdots, (i_n, j_n)\}$ cannot constitute a basis for (1) as in (ii) of Theorem 1 and conversely.

4. Properties of nonadjacent tours. The following theorem provides a test for determining whether two given tours are adjacent tours or not.

THEOREM 2. *Two tours t_1 and t_2 are not adjacent tours if and only if it is possible to form another tour t_3, distinct from t_1 and t_2, by taking some cells out of t_1 and the others out of t_2, but no cells outside those of t_1 and t_2. Such a tour t_3 contains all the common cells of t_1 and t_2. In other words, t_1 and t_2 are not adjacent tours if and only if there exists a tour t_3, $t_3 \neq t_1$, $t_3 \neq t_2$ such that*

$$\{t_3\} \subset \{t_1\} \cup \{t_2\} \quad \text{and} \quad \{t_1\} \cap \{t_2\} \subset \{t_3\}.$$

Proof. If t_1 and t_2 are not adjacent tours, then by definition there exists $0 < \alpha < 1$ such that

$$\alpha t_1 + (1 - \alpha)t_2 = \sum_{i=1}^{r} \beta_i s_i, \tag{3}$$

where $\beta_i > 0$, $\sum_{i=1}^{r} \beta_i = 1$, each of the s_i for $i = 1, \cdots, r$ is a tour and at least one of them, say s_1, is distinct from t_1 and t_2.

In (3) none of the s_i for $i = 1, \cdots, r$ can contain any cell outside those of t_1 and t_2 since $\beta_i > 0$ for all $i = 1$ to r.

It also implies that each of the s_i must contain all the common cells of t_1 and t_2, since $\beta_i > 0$ for $i = 1, \cdots, r$.

Hence, the tour s_1 which is distinct from t_1 and t_2 satisfies all the requirements in the proposition for the tour t_3.

On the other hand, if there exists a tour like t_3 above, then t_4 such that

$$\{t_4\} = [\{t_1\} \cap \{t_2\}] \cup [\{t_1\} \cup \{t_2\} \setminus \{t_3\}],$$

where $\setminus$ indicates set theoretic difference, represents another tour by Lemma 1, which follows. And, $\frac{1}{2}t_1 + \frac{1}{2}t_2 = \frac{1}{2}t_3 + \frac{1}{2}t_4$. Hence, t_1 and t_2 are not adjacent tours.

DEFINITION. Consider any tour t, where

$$t = \{(i_1, i_2), (i_2, i_3), \cdots, (i_n, i_1)\}.$$

Then, a subset of t like

$$\{(i_1, i_2), \cdots, (i_{r-1}, i_r)\}$$

is called a *segment of t from i_1 to i_r*. It consists of all the cells of t along a path from i_1 to i_r in t. The arc (i_1, i_2) itself may be considered as a segment of t from i_1 to i_2.

LEMMA 1. *Suppose t_1 and t_2 are two distinct tours and t_3 is another tour such that*

$$t_3 \neq t_1, \quad t_3 \neq t_2,$$

$$\{t_3\} \supset \{t_1\} \cap \{t_2\},$$

$$\{t_3\} \subset \{t_1\} \cup \{t_2\}.$$

Then, the cells

$$\{t_4\} = [\{t_1\} \cap \{t_2\}] \cup [\{t_1\} \cup \{t_2\} \setminus \{t_3\}],$$

where $\setminus$ indicates set theoretic difference, represent another tour.

Proof. Since both $\{t_3\}$ and $\{t_4\}$ contain all the common cells of t_1 and t_2, it is sufficient to prove the lemma for the case when t_1 and t_2 have no common cells.

In $\{t_1\} \cup \{t_2\}$ there are two cells in each row and column. Of these t_3 contains one in each row and column, since t_3 is a tour. Thus, t_4, which consists of the remaining cells, contains one cell from each row and column. Hence, t_4 is an assignment.

It remains to show that in t_4 there is a path from any city to any other.

Since t_3 is a tour, it must consist of some segments of t_1 and some of t_2. Actually, it consists of alternating segments from t_1 to t_2 respectively, i.e., it may consist of a segment from i_{r_1} to i_{r_2} of t_1, then a segment from i_{r_2} to i_{r_3} of t_2, then again a segment from i_{r_3} to i_{r_4} of t_1, etc.

Thus, t_4, which consists of the remaining segments of t_1 and t_2 (after striking off those in common with t_3) contains a path from each city to each other. Hence, t_4 is a tour.

LEMMA 2. *t and $\bar{t}$, the reflection of t, are always adjacent tours for $n \geqq 3$.*

Proof. Consider

$$t = \{(1, 2)(2, 3)(3, 4)(4, 5)(5, 6)(6, 1)\},$$

$$\bar{t} = \{(2, 1)(3, 2)(4, 3)(5, 4)(6, 5)(1, 6)\}.$$

If these are not adjacent tours, then by Theorem 2 it is possible to form a tour s distinct from t and $\bar{t}$ from the cells $\{t\} \cup \{\bar{t}\}$.

Suppose $(1, 2) \in s$. Then, since s contains only one cell from each row and column, $(1, 6) \notin s$ and $(3, 2) \notin s$. So, $(5, 6) \in s$, $(3, 4) \in s$. Hence, $(5, 4) \notin s$. Now, since s cannot contain any subtours, $(1, 2) \in s$ implies that $(2, 1) \notin s$. Similarly, $(6, 5) \notin s$, $(4, 3) \notin s$. Hence, $(2, 3) \in s$, $(6, 1) \in s$, $(4, 5) \in s$. Hence, $s = t$. Hence, it is not possible to form a tour distinct from t and $\bar{t}$ with the cells of $\{t\} \cup \{\bar{t}\}$. Therefore, by Theorem 2, t and $\bar{t}$ are adjacent tours.

In general, by renumbering the cities, we can assume that

$$t = t^* = \{(1, 2), (2, 3), \cdots, (n-1, n), (n, 1)\}.$$

By a construction similar to the above, we verify that the only tours that can be formed using only the cells $\{t^*\} \cup \{\bar{t}^*\}$ are t^* and $\bar{t}^*$. Hence, by Theorem 2, t^* and $\bar{t}^*$ are adjacent tours. Only when $n = 2$, $t = \bar{t} = \{(1, 2), (2, 1)\}$.

LEMMA 3. *Suppose $n \geqq 4$ and $r \leqq n - 3$. Let*

$$t = \{(1, i_1), (i_1, i_2), (i_2, i_3), \cdots, (i_{n-2}, i_{n-1}), (i_{n-1}, 1)\}$$

be any tour. Pick any r of the cells of t. Then there exists an adjacent tour t_1 of t containing exactly those r cells in common with t.

Proof. The tour t may be represented by the sequence

$$1 i_1 i_2 \cdots i_{n-2} i_{n-1}$$

indicating the order in which the cities are visited in the tour t.

The sequence which represents $\bar{t}$, the reflection of t, is obtained by reversing the order in which the cities occur in the sequence representing t. Thus $\bar{t}$ is represented by the sequence

$$i_{n-1} i_{n-2} \cdots i_2 i_1 1.$$

Case 1. Suppose the r cells which were picked constitute a segment of t from 1 to i_r, say. We wish to find an adjacent tour of t which contains this entire segment. For this we shall treat all these cities from 1 to i_r along the segment as a single block of cities. This is indicated by enclosing the segment from 1 to i_r within brackets, in the sequence representing t, which then becomes

$$[1 i_1 i_2 \cdots i_r] i_{r+1} \cdots i_{n-2} i_{n-1}.$$

We treat this entire block as if it were one location. Any arc entering this block enters at 1 and any arc leaving the block leaves from i_r. In t, the $n - r$ cells which are not on the segment from 1 to i_r form a tour in the cities $i_{r+1}, \cdots, i_{n-1}$ and the block, the reflection of which has all the properties desired of t_1. To generate it we write down the reverse sequence obtained by reversing the order of the cities $i_{r+1}, \cdots, i_{n-1}$ and the block in the sequence for t. In reversing the order of the cities, we treat the block as if it were another super-city, and we reverse its position in the sequence, but keep the order of the cities within it unchanged. This gives rise to the sequence

$$i_{n-1} i_{n-2} \cdots i_{r+1} [1 i_1 \cdots i_r].$$

The tour represented by this sequence

$$t_1 = \{(i_{n-1}, i_{n-2}), \cdots, (i_{r+2}, i_{r+1}), (i_{r+1}, 1), (1, i_1), \cdots, (i_{r-1}, i_r), (i_r, i_{n-1})\}$$

is an adjacent tour of t which has all the cells of the segment from 1 to i_r in common with t.

Case 2. Suppose the r cells which were picked constitute k nonoverlapping segments of t, say from 1 to i_{l_1}, from i_{l_2} to i_{l_3}, etc.

As before, write down the sequence representing the tour t and in that sequence represent each of the k segments above as a block:

$$[1 i_1 \cdots i_{l_1}] i_{l_1+1} \cdots [i_{l_2} \cdots i_{l_3}] \cdots.$$

Any city which is not in any block is known as an *out of block city*.

Now reverse the order of the out of block cities and the blocks in the above sequence, without changing the order of the cities inside each block. This gives a new sequence and let t_1 be the tour represented by it. Then t_1 is an adjacent tour of t and its common cells with t are exactly the r cells which were picked (contained within the blocks).

As an illustration, if

$$t = \{(1,3), (3,2), (6,5), (9,8); (2,7), (4,9), (5,1), (7,10), (8,6), (10,4)\},$$

the tour t_1 obtained by the above procedure, containing the first four cells in t, is

$$t_1 = \{(1,3), (3,2), (6,5), (9,8); (2,6), (5,9), (8,4), (4,10), (10,7), (7,1)\}.$$

LEMMA 4. *When $n \geqq 6$, it is always possible to find a pair of nonadjacent tours.*

Proof. If $n = 6$, let

$$t_1 = \{(1,2), (2,3), (3,4), (4,5), (5,6), (6,1)\},$$
$$t_2 = \{(1,3), (3,2), (2,4), (4,6), (6,5), (5,1)\},$$
$$t_3 = \{(1,2), (2,3), (3,4), (4,6), (6,5), (5,1)\},$$

and if $n > 6$, let

$$t_1 = \{(1,7), (7,8), \cdots, (n-1, n), (n,2), (2,3), (3,4), (4,5), (5,6), (6,1)\},$$
$$t_2 = \{(1,n), (n, n-1), (n-1, n-2), \cdots, (8,7), (7,3), (3,2), (2,4), (4,6), (6,5), (5,1)\},$$
$$t_3 = \{(1,7), (7,8), \cdots, (n-1, n), (n,2), (2,3), (3,4), (4,6), (6,5), (5,1)\}.$$

Then, $t_3 \neq t_1$, $t_3 \neq t_2$ and $\{t_3\} \subset \{t_1\} \cup \{t_2\}$. Hence, by Theorem 2, t_1 and t_2 are not adjacent tours.

LEMMA 5. *When $n \geqq 6$, the number of adjacent tours of any given tour is*

$$\geqq 2^n - \left[1 + n + \binom{n}{2}\right].$$

Proof. When $n \geqq 6$ and $r \leqq n - 3$, by Lemma 3 we know that there exists at least one adjacent tour of t containing exactly any selected r cells of t in common with it. Hence, if U_n is the number of adjacent tours of a given tour, then

$$U_n \geqq \sum_{n=0}^{n-3} \binom{n}{r} = 2^n - \left[1 + n + \binom{n}{2}\right].$$

This indicates that the number of adjacent tours of a given tour goes up at least in the order of 2^n. This completes the proof.

The important steps in the simplex algorithm for minimizing a linear function on a convex polyhedral set described by a set of linear inequalities are the following:

(i) An easy method has been developed by which adjacent vertices of any given vertex may be obtained.

In the simplex method this is done by bringing a nonbasic variable into the basis (one pivot step).

(ii) If the present vertex does not minimize the linear function on the solution set, then a simple criterion has been developed, by which one can obtain an adjacent vertex at which the linear function takes a value less than or equal to that at the present vertex.

In the simplex method this is done by bringing into the basis a nonbasic variable whose relative cost coefficient is negative.

Even though it is not easy to describe the convex polyhedral set K_T by a set of linear inequalities, it is possible to develop a simple method by which adjacent tours of a given tour may be obtained. This corresponds to Step (i) of the simplex method discussed above.

The method for obtaining adjacent tours of a given tour uses pivot steps on the assignment matrix, which is characterized by the set of linear constraints (1). This is discussed below.

4.1. An algorithm for generating an adjacent tour of a given tour. Any basis for the system of constrains (1) with the $n - 1$ ZBC's along the principal diagonal represents a tour by Corollary 2. Such a basis is known as a *diagonal basis* (DB) of that tour. Using the test developed in Theorem 2 and Lemma 2, an algorithm which starts with a DB of a given tour and leads to a DB of an adjacent tour is described below.

Consider a given tour t. Then, the cells of t are known as the *original basic cells* (OBC's).

Step 1. Start with any DB for t. Bring any nonbasic cell which is not a diagonal cell into the basis replacing an OBC (or a diagonal cell if this is not possible) in its row or column.

The new cells that are brought into the basis are called the *new basic cells* (NBC's),

At any stage an OBC in the row or column of an NBC is known as an *excess cell*. A row (or column) is known as a *deficit row* (*column*) if it has

(i) only one basic cell in it and if this is either a diagonal cell or an excess cell;

(ii) only two basic cells in it and if one of them is a diagonal cell and the other an excess cell.

TABLE 1

Step	Current basis	Excess cells	Deficit rows and columns	NBC or diagonal cell brought in	OBC or diagonal cell removed
	(1, 2) (2, 3) (3, 4) (4, 5) (5, 6) (6, 7) (7, 8) (8, 9) (9, 10) (10, 1); (1, 1) (2, 2) (3, 3) (4, 4) (5, 5) (6, 6) (7, 7) (8, 8) (9, 9)			(2, 7)	(2, 3)
1	(2, 7) (7, 8) (8, 9) (9, 10) (10, 1) (1, 2) (3, 3) (4, 4) (5, 5) (6, 6); (1, 1) (2, 2) (3, 4) (4, 5) (5, 6) (7, 7) (8, 8) (9, 9) (6, 7)	(6, 7)	row 3 col. 3	(6, 5)	(5, 5)
2	(2, 7) (6, 5) (5, 6) (7, 8) (8, 9) (9, 10) (10, 1) (1, 2) (3, 3) (4, 4); (1, 1) (2, 2) (3, 4) (6, 6) (7, 7) (8, 8) (9, 9) (6, 7) (4, 5)	(6, 7 (4, 5)	row 4 col. 3	(4, 9)	(4, 5)
3	(2, 7) (6, 5) (5, 6) (7, 8) (9, 10) (10, 1) (1, 2) (3, 3) (4, 4) (8, 9); (4, 9) (1, 1) (2, 2) (3, 4) (6, 6) (7, 7) (8, 8) (9, 9) (6, 7)	(8, 9) (6, 7)	row 8 col. 3	(8, 6)	(6, 7)
4	(2, 7) (6, 5) (7, 8) (9, 10) (10, 1) (1, 2) (3, 3) (4, 4) (5, 6) (8, 9); (4, 9) (8, 6) (1, 1) (2, 2) (3, 4) (6, 6) (7, 7) (8, 8) (9, 9)	(5, 5) (8, 9)	row 5 col. 3	(5, 1)	(1, 1)
5	(2, 7) (6, 5) (1, 2) (7, 8) (9, 10) (3, 3) (4, 4) (5, 6) (8, 9) (10, 1); (4, 9) (5, 1) (8, 6) (2, 2) (3, 4) (6, 6) (7, 7) (8, 8) (9, 9)	(5, 6) (8, 9) (10, 1)	row 10 col. 3	(10, 4)	(10, 1)
6	(2, 7) (10, 4) (4, 9) (8, 6) (6, 5) (5, 1) (1, 2) (7, 8) (9, 10) (3, 3); (2, 2) (4, 4) (6, 6) (7, 7) (8, 8) (9, 9) (3, 4) (5, 6) (8, 9)	(3, 4) (5, 6) (8, 9)	row 3 col. 3	(3, 2)	(3, 4)
7	(2, 7) (10, 4) (4, 9) (8, 6) (6, 5) (5, 1) (3, 3) (7, 8) (9, 10) (1, 2); (3, 2) (2, 2) (4, 4) (6, 6) (7, 7) (8, 8) (9, 9) (5, 6) (8, 9)	(1, 2) (5, 6) (8, 9)	row 1 col. 3	(1, 3)	(1, 2)
8	(8, 6) (6, 5) (5, 1) (1, 3) (3, 2) (2, 7) (10, 4) (4, 9) (7, 8) (9, 10); (2, 2) (3, 3) (4, 4) (6, 6) (7, 7) (8, 8) (9, 9) (5, 6) (8, 9)	(5, 6) (8, 9)		(5, 5)	(5, 6)
9	(8, 6) (6, 5) (5, 1) (1, 3) (3, 2) (2, 7) (10, 4) (4, 9) (7, 8) (9, 10); (2, 2) (3, 3) (4, 4) (5, 5) (6, 6) (7, 7) (8, 8) (9, 9) (8, 9)	(8, 9)		(1, 1)	(7, 8)
10	(10, 4) (4, 9) (1, 1) (2, 2) (3, 3) (5, 5) (6, 6) (7, 7) (8, 8) (9, 10); (8, 6) (6, 5) (5, 1) (1, 3) (3, 2) (2, 7) (4, 4) (9, 9) (8, 9)	(8, 9)	row 7	(7, 10)	(8, 9)
11	(10, 4) (4, 9) (1, 1) (2, 2) (3, 3) (5, 5) (6, 6) (7, 7) (8, 8) (9, 10); (8, 6) (6, 5) (5, 1) (1, 3) (3, 2) (2, 7) (7, 10) (4, 4) (9, 9)	(9, 10)	row 9	(9, 8)	(9, 10)
12	(1, 3) (3, 2) (2, 7) (7, 10) (10, 4) (4, 9) (9, 8) (8, 6) (6, 5) (5, 1); (1, 1) (2, 2) (3, 3) (4, 4) (5, 5) (6, 6) (7, 7) (8, 8) (9, 9)				

Subsequent steps. Bring into the basis a nonbasic cell which is not a diagonal cell and which is in a deficit row or column and not in a row or column of any NBC, replacing if possible an OBC in its row or column or otherwise a diagonal basic cell in the same row or column.

The process terminates when a DB is reached.

If at any stage a DB is not reached, but there is no deficit row or column, then the number of diagonal basic cells must be $< n - 1$. Bring a nonbasic diagonal cell back into the basis replacing an excess cell if possible, or otherwise an OBC in its θ-loop. When $n - 1$ diagonal basic cells are again in the basis, either a DB is obtained or some deficit rows and columns are created.

The steps are repeated until a DB is reached. The new DB represents the DB of an adjacent tour of t by Theorem 2.

Also let t be any tour and t_1 an adjacent tour of t. Start with a DB for t and bring successively the cells of $\{t_1\} \setminus \{t\}$ (where $\setminus$ indicates set theoretic difference) as NBC's in the above algorithm. By Theorem 2 there does not exist any other tour t_2 distinct from t and t_1 whose cells form a subset of $\{t\} \cup \{t_1\}$. Hence the above algorithm will terminate only when all the cells of t_1 are brought into the basis.

Thus by an appropriate choice of NBC's at the various steps, all the adjacent tours of a given tour can be obtained by the above algorithm.

4.2. A numerical example. Let $t = \{(1, 2), (2, 3), \cdots, (9, 10), (10, 1)\}$. Starting with a DB for t, we obtain an adjacent tour of t. The bases for (1) during the various steps of the algorithm are given in Table 1.

In the table, the basic cells at each stage of the algorithm are arranged in two groups; the cells listed before the symbol ";" are unit-valued basic cells and those that follow the ";" are ZBC's.

Since Step 12 gave a DB, the tour

$$t_1 = \{(1, 3), (3, 2), (2, 7), (7, 10), (10, 4), (4, 9), (9, 8), (8, 6), (6, 5), (5, 1)\}$$

is an adjacent tour of t.

REFERENCES

[1] I. HELLER, *On the traveling salesman's problem*, Proc. 2nd Symposium in Linear Programming, National Bureau of Standards, Washington D.C., 1955, pp. 643–665.

[2] ———, *Neighbour relations on the convex of cyclic permutations*, Pacific J. Math. 6 (1956), pp. 467–477.

[3] M. M. FLOOD, *The traveling salesman problem*, Operations Res., 4 (1956), pp. 61–75.

[4] H. W. KUHN, *On certain convex polyhedra*, Abstract, Bull. Amer. Math. Soc., 61 (1955), pp. 557–558.

[5] R. E. GOMORY, *The traveling salesman problem*, Proc. IBM Scientific Computing Symposium on Combinatorial Problems, White Plains, New York, 1964, pp. 93–121.