1987

Computational aspects and statistical applications of the transportation problem of linear programming

Miriam Bridget C. Tirol

Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/rtd

Part of the Statistics and Probability Commons

Recommended Citation


https://lib.dr.iastate.edu/rtd/8596

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
INFORMATION TO USERS

While the most advanced technology has been used to photograph and reproduce this manuscript, the quality of the reproduction is heavily dependent upon the quality of the material submitted. For example:

- Manuscript pages may have indistinct print. In such cases, the best available copy has been filmed.

- Manuscripts may not always be complete. In such cases, a note will indicate that it is not possible to obtain missing pages.

- Copyrighted material may have been removed from the manuscript. In such cases, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, and charts) are photographed by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each oversize page is also filmed as one exposure and is available, for an additional charge, as a standard 35mm slide or as a 17”x 23” black and white photographic print.

Most photographs reproduce acceptably on positive microfilm or microfiche but lack the clarity on xerographic copies made from the microfilm. For an additional charge, 35mm slides of 6”x 9” black and white photographic prints are available for any photographs or illustrations that cannot be reproduced satisfactorily by xerography.
Computational aspects and statistical applications of the transportation problem of linear programming

Tirol, Miriam Bridget C., Ph.D.
Iowa State University, 1987
PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark √.

1. Glossy photographs or pages
2. Colored illustrations, paper or print
3. Photographs with dark background
4. Illustrations are poor copy
5. Pages with black marks, not original copy
6. Print shows through as there is text on both sides of page
7. Indistinct, broken or small print on several pages √
8. Print exceeds margin requirements
9. Tightly bound copy with print lost in spine
10. Computer printout pages with indistinct print
11. Page(s) lacking when material received, and not available from school or author.
12. Page(s) seem to be missing in numbering only as text follows.
13. Two pages numbered. Text follows.
14. Curling and wrinkled pages
15. Dissertation contains pages with print at a slant, filmed as received
16. Other

University Microfilms International
Computational aspects and statistical applications of the
transportation problem of linear programming

by

Miriam Bridget C. Tirol

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of the
Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Co-majors: Statistics
Industrial Engineering

Approved:

Signature was redacted for privacy.
In Charge of Major Work

Signature was redacted for privacy.
For the Major Departments

Signature was redacted for privacy.
For the Graduate College

Iowa State University
Ames, Iowa
1987
# TABLE OF CONTENTS

1. INTRODUCTION  
   1.1 A Review of Some Basic Concepts in Linear Programming  
   1.2 Definition of the Transportation Problem  
   1.3 Properties of the Transportation Problem  
   1.4 Computational Procedure for Solving the Transportation Problem  
   1.5 Overview  

2. A COMPUTATIONAL STUDY OF METHODS FOR FINDING AN INITIAL BASIC FEASIBLE SOLUTION FOR TRANSPORTATION PROBLEMS  
   2.1 Description of Methods for Finding an Initial Basic Feasible Solution  
   2.2 Experimental Setup  
   2.3 Computational Results and Their Interpretation  

3. THE CAPACITATED TRANSPORTATION PROBLEM  
   3.1 Definition of the Capacitated Transportation Problem  
   3.2 Finding an Initial Feasible Solution  
   3.3 Obtaining an Optimum Feasible Solution  

4. STATISTICAL APPLICATIONS OF THE TRANSPORTATION PROBLEM  
   4.1 Least Absolute Value Estimation for the Two-way Classification Model  
   4.2 The Controlled Rounding Problem  

5. SUMMARY  

6. REFERENCES  

7. ACKNOWLEDGEMENTS
8. APPENDIX A: SIMPLEX PROCEDURE FOR PROBLEMS WITH UPPER BOUND CONSTRAINTS 106

9. APPENDIX B: A FORTRAN PROGRAM FOR FINDING LEAST ABSOLUTE VALUE ESTIMATES FOR THE TWO-WAY CLASSIFICATION MODEL 109
1. INTRODUCTION

A special type of linear programming problem that arises quite frequently in practical application is the transportation problem. The transportation problem arises when we must determine the shipping schedule that minimizes the total cost of shipment given that:

1) there are known fixed quantities of a commodity available for shipment at each of m origins;

2) given quantities of a commodity are required to be shipped to each of n destinations;

3) the total shipments from all origins equal the total requirements of all destinations;

and

4) the minimum cost of shipping a unit of commodity from any origin to any destination is known, and from which the total cost is obtained by taking the sum of individual costs.

The standard form of the transportation problem as we know it today was originally stated by Hitchcock (1941) and later discussed in detail by Koopsman (1947). An earlier approach was described by Kantorovich (1939). The linear programming formulation and the associated method of solution developed by specializing the general simplex method to the special structure of the transportation problem was first described by Dantzig (1951). Some of the other methods for solving the transportation problem that have appeared in literature include the stepping-stone method of Charnes and Cooper (1954), the dual method of Ford and
Fulkerson (1963), a primal Hungarian method by Balinski and Gomory (1964), and methods of reduced matrices by Dwyer (1966).

The transportation problem has a wide range of applications. It has been used to solve problems in production planning [Bowman (1956); Henderson (1958); Sadleir (1970)], allocation problems [Ferguson and Dantzig (1956); Lederman et al. (1966)], and contract award problems [Waggener and Suzuki (1967); Beged-Dov (1970); Gass (1970)]. It has also been used in geographic studies [Stevens (1961); Barr and Smillie (1972); Hay (1977)]. A more recent application involved using the transportation algorithm as an aid to chromosome classification [Tso and Graham (1983)].

An overview of the various aspects of the transportation problem that are dealt with in this thesis is presented in Section 1.5.

1.1 A Review of Some Basic Concepts in Linear Programming

Before we proceed with defining the transportation problem and its properties, we shall first state some of the basic concepts in linear programming that will be used in our discussion of the transportation problem. The material for this section is contained in linear programming textbooks by Hadley (1962), Cooper and Steinberg (1974), Gass (1975), Sposito (1975), and Hillier and Lieberman (1986).

We state a general linear programming problem as follows:

We need to find values for a set of $n$ variables $x_j$ satisfying $m$ linear inequalities or equalities (the constraints) of the form

$$a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n \{ \leq \geq \} b_i, \ i = 1, \ldots, m,$$

(1.1)
where one and only one of the signs ≤, =, ≥ holds for each constraint, but the sign can vary from one constraint to another. Furthermore, the variables are required to be nonnegative,

\[ x_j \geq 0, \ j=1,...,n, \quad (1.2) \]

(the nonnegativity restriction), and are to maximize or minimize a linear function (the objective function)

\[ z = c_1x_1 + c_2x_2 + \ldots + c_nx_n. \quad (1.3) \]

All the \( a_{ij}, b_1, \) and \( c_j \) are assumed to be known constants, and \( m < n \).

We now state a number of standard definitions and theorems that describe some characteristics of a solution to the general linear programming problem. The proofs of the theorems that are given below can be found in the references listed at the beginning of this section.

**Definition 1.1**

A feasible solution to the linear programming problem is a vector \( X = \{x_1, \ldots, x_n\} \) that satisfies conditions (1.1) and (1.2).

**Definition 1.2**

A basic solution to (1.1) is a solution obtained by setting \( n - m \) variables equal to zero and solving for the remaining \( m \) variables, provided that the vectors corresponding to these \( m \) variables are linearly independent. These \( m \) variables are called basic variables.
Definition 1.3

A basic feasible solution is a basic solution that also satisfies (1.2); that is, all basic variables are nonnegative.

Definition 1.4

A nondegenerate basic feasible solution is a basic feasible solution with exactly \( m \) positive \( x_j \); that is, all basic variables are strictly positive.

Definition 1.5

Any feasible solution that maximizes or minimizes the objective function \( z \) (1.3) is called an optimal feasible solution.

Using the following definitions, we will be able to describe the set of all feasible solutions, commonly called the feasible region.

Definition 1.6

A set \( S \) is convex if for any two points or vectors \( \bar{x} \) and \( \bar{z} \) in \( S \), then
\[
\hat{z} = \lambda \bar{x} + (1-\lambda)\bar{z}
\]
is in \( S \) for any \( \lambda \in [0,1] \).

Definition 1.7

A point \( \bar{z} \) is an extreme point of a convex set if and only if there do not exist other points \( y, z, y \neq z \), in the set such that
\[
\bar{z} = \lambda y + (1-\lambda)z, \quad 0 < \lambda < 1.
\]

Theorem 1.1

The set of all feasible solutions to the linear programming problem is a convex set.
Theorem 1.2

The objective function (1.3) assumes its maximum (or minimum) at an extreme point of the convex set generated by the set of feasible vectors of the linear programming problem. If it assumes its maximum (or minimum) at more than one extreme point, then it takes on the same value for every convex combination of those particular points.

The best-known and most widely used procedure for solving linear programming problems is called the simplex method. The simplex method is an algebraic iterative procedure that will solve exactly any linear programming problem in a finite number of steps, or give an indication that there is an unbounded solution or the problem is infeasible. The simplex method can be given a very simple geometrical interpretation in terms of the concepts that have already been introduced. We have stated above that if there is an optimal solution, one of the extreme points will identify this solution. There is only a finite number of extreme points. The simplex method is a procedure for moving step by step from a given extreme point to an optimal extreme point. At each step, the simplex method moves along an edge of the feasible region from one extreme point to a neighboring extreme point. Of all the neighboring extreme points, the one chosen is that which gives the greatest increase (or decrease) in the objective function value. At each extreme point, the simplex method determines whether that extreme point is optimal, and if not, what the next extreme point will be. If at any stage the
simplex method comes to an extreme point which has an edge leading to infinity, and if the objective function can be increased (or decreased) by moving along that edge, the simplex method informs us that there is an unbounded solution.

The theory and computational details on the simplex method are found in the references given at the beginning of the section.

Finally, we describe the concept of duality. Let us express the linear programming problem in matrix form. Let a linear programming problem be defined as,

\[
\begin{align*}
\text{minimize} & \quad c'x \\
\text{subject to} & \quad Ax \geq b \\
& \quad x \geq 0
\end{align*}
\]

where \(A\) is an \(m \times n\) matrix of \(a_{ij}\), \(b' = [b_1, \ldots, b_m]\), \(c' = [c_1, \ldots, c_n]\), \(x' = [x_1, \ldots, x_n]\), and \(a_{ij}, b_i, \text{ and } c_j\) are given constants \((i=1, \ldots, m; j=1, \ldots, n)\).

Then the dual problem of the primal problem (1.4) is

\[
\begin{align*}
\text{maximize} & \quad b'y \\
\text{subject to} & \quad A'y \leq c \\
& \quad y \geq 0
\end{align*}
\]

where \(y' = [y_1, \ldots, y_m]\), and \(y_i, i=1, \ldots, m,\) are the dual variables.

If instead of \(Ax \geq b\) the primal problem (1.4) has equality constraints, \(Ax = b\), its dual problem is also (1.5) without the nonnegativity restriction on \(y\).
We now state some fundamental properties of dual problems.

**Theorem 1.3: Weak Duality Theorem**

If \( \bar{x} \) and \( \bar{y} \) are feasible solutions of problems (1.4) and (1.5), respectively, then \( c'\bar{x} \geq b'\bar{y} \).

A corollary to Theorem 1.3 states:

If \( \bar{x} \) and \( \bar{y} \) are feasible solutions to problems (1.4) and (1.5), respectively, and \( c'\bar{x} = b'\bar{y} \), then \( \bar{x} \) is an optimal solution of problem (1.4) and \( \bar{y} \) is an optimal solution of problem (1.5).

### 1.2 Definition of the Transportation Problem

The transportation problem is a linear programming problem which has the following special structure:

\[
\begin{align*}
\text{minimize} \quad & z = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\
\text{subject to} \quad & \sum_{j=1}^{n} x_{ij} = a_i, \quad i=1, \ldots, m \\
& \sum_{i=1}^{m} x_{ij} = b_j, \quad j=1, \ldots, n \\
& x_{ij} \geq 0, \text{ for all } i \text{ and } j.
\end{align*}
\]  

The above linear programming problem may be considered as one in which various amounts of a commodity are to be shipped from each of m origins to each of n destinations. The amount available for shipment from origin \( i \) is \( a_i, i=1, \ldots, m \); the amount required by destination \( j \) is
$b_j$, $j=1,...,n$. The cost of shipping one unit of commodity from origin $i$ to destination $j$ is denoted as $c_{ij}$. The unknown quantity to be shipped from origin $i$ to destination $j$ is denoted by $x_{ij}$. It is also assumed that the total available supply equals the total demand, that is

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j,$$

in order for the transportation problem defined above to have a feasible solution.

The special structure of the transportation problem also enables us to express it in tableau form, with $m$ rows and $n$ columns, as shown by:

$$
\begin{array}{cccc}
  & c_{11} & c_{12} & \cdots & c_{1n} \\
 x_{11} & x_{12} & \cdots & \cdots & x_{1n} \\
 x_{12} & c_{22} & \cdots & \cdots & x_{2n} \\
 \vdots & \vdots & \ddots & \vdots & \vdots \\
 x_{m1} & c_{m2} & \cdots & c_{mn} \\
 b_1 & b_2 & \cdots & b_n \\
\end{array}
$$

The dual of the transportation problem is defined as follows:

$$\text{maximize } \sum_{i=1}^{m} a_i u_i + \sum_{j=1}^{n} b_j v_j$$

subject to $u_i + v_j \leq c_{ij}$, $i=1,...,m$; $j=1,...,n$. 


1.3 Properties of the Transportation Problem

In this section we will state some fundamental theorems describing the properties of the transportation problem. The material in this section is described in Strum (1972), Cooper and Steinberg (1974), and Gass (1975).

**Theorem 1.4**

The transportation problem always has a feasible solution.

**Proof:**

Since $\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j = t$, we have the feasible solution $x_{ij} = a_i b_j / t$ for all $i$ and $j$. Each $x_{ij} \geq 0$ and (1.7) is satisfied, since

$$\sum_{j=1}^{n} x_{ij} = \sum_{j=1}^{n} \left( a_i b_j / t \right) = a_i, \quad i=1,\ldots,m$$

and (1.8) is satisfied, since

$$\sum_{i=1}^{m} x_{ij} = \sum_{i=1}^{m} \left( a_i b_j / t \right) = b_j, \quad j=1,\ldots,n$$

**Theorem 1.5**

A basis for the transportation problem consists of at most $m+n-1$ variables.

**Proof:**

From (1.7), if we sum on $i$, and from (1.8) if we sum on $j$, we have
Summing the first \( n-1 \) constraints given by (1.8) yields

\[
\sum_{j=1}^{n-1} \sum_{i=1}^{m} x_{ij} = \sum_{j=1}^{n-1} \sum_{i=1}^{n} a_i = \sum_{j=1}^{n} b_j. \tag{1.10}
\]

Subtracting (1.10) from (1.9), we obtain

\[
\sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij} - \sum_{i=1}^{m} \sum_{j=1}^{n-1} x_{ij} = \sum_{i=1}^{n} b_i - \sum_{j=1}^{n} b_j - \sum_{i=1}^{m} \sum_{j=1}^{n} x_{ij} = 0
\]

which is the remaining \( n \)th constraint of (1.9). Therefore, the last constraint is dependent on the first \( m+n-1 \) constraints. This implies that a basis for the transportation problem consists of at most \( m+n-1 \) variables.

**Theorem 1.6**

A basic feasible solution of the transportation problem has a row or column containing only one occupied cell in the transportation tableau.
Proof:

We suppose $m < n$; if this is not the case we may simply interchange the roles of $m$ and $n$. Assume that every column in the transportation tableau contains at least two occupied cells. Then, the number $N$ of occupied cells satisfies the inequalities

$$N \geq 2n > m + n > m + n - 1.$$ 

But, this is contrary of hypothesis, and hence, there must be some column which contains only one occupied cell. Theorem 1.6 will be used to prove the following theorem.

Theorem 1.7

Assuming that each $a_i$ and $b_j$ are nonnegative integers, then every basic feasible solution has integral values.

Proof:

By Theorem 1.6, there is a column containing only one occupied cell. Let this be column $j$, and suppose $x_{ij} \neq 0$. Then, $x_{ij} = b_j$, clearly an integer. Now, consider the remaining table after we delete column $j$ and replace $a_i$ by $a_i^{(1)} = a_i - b_j$. This new table has $m$ rows and $n-1$ columns; it has at most $m+n-2$ occupied cells, and all rim entries are positive integers. Hence, Theorem 1.6 can be applied to this reduced table; and so on. After $k$ reductions, we are left with a table containing a total of $m+n-k$ rows and columns. The capacities $a_i^{(k)}$ and requirements $b_j^{(k)}$ of this reduced table are of the form
\[ a_{i}^{(k)} = a_{i} + \text{(sum of } a_{i}'s \text{ corresponding to the deleted rows)} \]
\[ - \text{(sum of } b_{j}'s \text{ corresponding to the deleted columns)} \]
\[ b_{j}^{(k)} = b_{j} + \text{(sum of } b_{j}'s \text{ corresponding to the deleted columns)} \]
\[ - \text{(sum of } a_{i}'s \text{ corresponding to the deleted rows).} \]

As long as there exist reduced tables, Theorem 1.6 can be applied to locate some row (or column) that contains no more than one basic \( x_{ij} \), and the value of this basic variable is an \( a_{i}^{(k)} \) or \( b_{j}^{(k)} \). This shows that the amount contained in an occupied cell is the difference between a partial sum of row supply and a partial sum of column demand.

**Theorem 1.8**

The transportation problem is never unbounded.

**Proof:**

Each variable \( x_{ij} \) appears in exactly two constraints, both times with a coefficient of +1. Therefore, it is easy to see that \( x_{ij} \) is bounded by

\[ 0 \leq x_{ij} \leq \min\{a_{i}, b_{j}\} . \]

1.4 Computational Procedure for Solving the Transportation Problem

We have already seen that a basic feasible solution will contain no more than \( m+n-1 \) positive variables, with the remaining variables being zero. However, not every feasible solution containing this number of positive variables is basic. For such a solution to be basic, the \( m+n-1 \)
columns of the coefficients corresponding to these variables must be linearly independent. Thus, we need to be able to determine whether a given set of \( m+n-1 \) columns of coefficients corresponding to a set of \( m+n-1 \) variables is either linearly dependent or linearly independent.

The methods of solution that will be described in this section is applied on the transportation tableau, so that the focus of our discussion will now be on the transportation tableau. We introduce the following terminology:

1) The "box" in row \( i \) and column \( j \) of the tableau is called the \((i,j)\) cell;

2) A loop is a sequence of cells such that:
   a) each adjacent pair of cells lies in either the same row or the same column;
   b) no group of three or more consecutive cells in the sequence lies in the same row or in the same column; and
   c) the first and last cells in the sequence are in either the same row or the same column.

It should be noted that given any loop, each row of the tableau contains either no cells or an even number of cells, and similarly, every column of the tableau either contains no cells or an even number of cells. Also, in moving from one cell to an adjacent cell in a loop, we alternate between moving to a cell in the same row and between moving to a cell in the same column.
These facts suggest the following theorem.

**Theorem 1.9**

Every loop contains an even number of cells.

**Proof:**

Suppose we have a loop which contains \( N \) cells. Let these cells be numbered \( 1, 2, \ldots, N \) where the numbering corresponds to the ordering of the construction of the loop. Cells 1 and 2 must lie in the same row or column, by definition of a loop. Assume that they are in the same row. Then, the step from cell 2 to cell 3 is a movement in the same column, the step from cell 3 to cell 4 a movement in the same row, etc. Alternating in this manner, the step from any cell \( t \) to \( t+1 \) must be a movement in the same row if \( t \) is odd, and a movement in the same column if \( t \) is even. Hence, the step from cell \( N \) to cell 1 must be a movement in the same column, since it was assumed that the step from cell 1 to cell 2 was a movement in the same row. Thus, \( N \) must necessarily be even.

Theorem 1.9 leads us to the following theorem.

**Theorem 1.10**

Let \( \bar{a}_{ij} \) denote a column corresponding to the coefficients of variable \( x_{ij} \) in the transportation problem, and let \( T \) denote a subset of columns \( \bar{a}_{ij} \). Then, the columns of \( T \) are linearly dependent if and only if the corresponding cells, or a subset of them, can be arranged in a sequence which forms a loop.
Proof:

See pages 210-212 of Cooper and Steinberg (1974).

Now, we are ready to state a general rule for finding an initial solution to the transportation problem. We will also show that such a rule for obtaining an initial solution will give a basic solution.

As a candidate for the first basic variable, choose any variable, \( x_{pq} \), and make it as large as possible, so that \( x_{pq} = \min\{a_p, b_q\} \). As a result of this allocation, three possible situations may arise:

1) If \( a_p \) is less than \( b_q \), then all other variables in the \( p \)th row are given the value zero and designated as nonbasic. The \( p \)th row is then deleted, the value of \( b_q \) is reduced to \( b_q - a_p \), and the process of evaluating a variable is done in the same manner in the reduced array of \( m-1 \) rows and \( n \) columns.

2) If \( a_p \) is greater than \( b_q \), then all other variables in the \( q \)th column are given the value zero and designated as nonbasic. The \( q \)th column is deleted, the value of \( a_p \) is reduced to \( a_p - b_q \), and the process of evaluating a variable is done in the same manner in the reduced array of \( m \) rows and \( n-1 \) columns.

3) If \( a_p \) equals \( b_q \), then delete either the row or column, but not both. However, if several columns, but only one row remain in the reduced array, then drop the \( q \)th column, and conversely, if several rows, and one column remain, drop the \( p \)th row.

This rule will necessarily select \( m+n-1 \) variables for the basic set. Let us now show that the variables chosen by the rule for finding an
initial solution constitute a basic set. This is done by demonstrating that a loop cannot be formed by the cells corresponding to the variables in the initial solution.

Suppose that at step $k$ a positive value is assigned to $x_{ij}$, and cell $(i,j)$, together with some or all of the $k-1$ cells corresponding to previously determined positive values of the variables, form a loop. Let the loop be described by the ordered set

$$(i,j), (i,r), (s,r), \ldots, (v,u), (v,j), (i,j).$$

Since we are allowed to assign a positive value to $x_{ij}$, then it must be true that when $x_{ir}$ was assigned a positive value the column constraint $r$ was satisfied rather than the row constraint $i$. This means that a positive value must have been assigned to $x_{sr}$ before $x_{ir}$, and at that step, the row constraint $s$ has been satisfied. Proceeding in this way, the column constraint $j$ was satisfied by assigning a positive value to $x_{vj}$. However, we are now assigning a positive value to $x_{ij}$, which contradicts the fact that the constraint $j$ has been satisfied. Therefore, no loop can be formed, and a basic solution is obtained.

After identifying an initial feasible basic solution, we would like to know if this solution is optimal, and if not, how can further improvements be made. In particular, we would like to introduce a nonbasic variable in the solution to replace one of the present basic variables in such a way that feasibility is maintained and the value of the objective function is improved.
The condition used for selecting an entering nonbasic variable is derived as follows:

\[ z - \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} = 0. \]  

(1.11)

Multiplying \( \sum_{j=1}^{n} x_{ij} = a_i \) by \( u_i \), for each \( i \), and summing across \( i \), we add this expression to (1.11). Likewise, multiplying \( \sum_{i=1}^{m} x_{ij} = b_j \) by \( v_j \), for each \( j \), and summing across \( j \), we also add this expression to (1.11). So that the expression added to (1.11) is

\[ \sum_{i=1}^{m} \sum_{j=1}^{n} u_i a_i + \sum_{j=1}^{n} v_j b_j. \]

Hence, (1.11) becomes

\[ z + \sum_{i=1}^{m} \sum_{j=1}^{n} (u_i + v_j - c_{ij}) x_{ij} = \sum_{i=1}^{m} u_i a_i + \sum_{j=1}^{n} v_j b_j. \]  

(1.12)

The coefficients of the present basic values are zero, for the proper choice of \( u_i \) and \( v_j \). Moreover, by definition of row 0 of the simplex tableau

\[ z + \sum_{i=1}^{m} \sum_{j=1}^{n} (z_{ij} - c_{ij}) x_{ij} = V \]  

(1.13)

where \( z_{ij} - c_{ij} = 0 \) for each basic \( x_{ij} \) and \( V \) is the present value of the objective function.
Comparing (1.12) and (1.13), we see that

\[ z_{ij} - c_{ij} = u_i + v_j - c_{ij} \]

and

\[ V = \sum_{i=1}^{m} a_{iu_i} + \sum_{j=1}^{n} b_{jv_j} . \]

Therefore, by knowing the present set of basic variables, an appropriate set of values for \( u_i \) and \( v_j \) can be computed from the equation

\[ u_i + v_j - c_{ij} = 0 \text{ for } x_{ij} \text{ basic} . \]

For these values of \( u_i \) and \( v_j \), the corresponding value for \( z_{ij} - c_{ij} \), which is

\[ u_i + v_j - c_{ij} , \tag{1.14} \]

can be computed for the nonbasic \( x_{ij} \)'s. Hence, if (1.14) is positive for any nonbasic \( x_{ij} \), then the present set of basic variables is not optimal and a further reduction of the present value of \( V \) is possible.

We now list the steps to an algorithm to solve the transportation problem.

1) Find an initial basic feasible solution.

2) Calculate \( c_{ij} - u_i - v_j \) for each nonbasic variable.
   a) If \( c_{ij} - u_i - v_j \geq 0 \) for all nonbasic variables, then the present set of basic variables is optimal.
   b) Choose the nonbasic variable \( x_{pq} \) as the new entering basic variable such that
\[ c_{pq} - u_p - v_q = \min_{i,j} \{ c_{ij} - u_i - v_j | c_{ij} - u_i - v_j < 0 \}. \]

4) Determine a loop which connects this nonbasic cell with a subset of the present basic cells. Label the cells in the loop alternating between "+" and "-", starting with a "+" for the incoming nonbasic cell.

5) Choose the level of the new entering variable, denoted by \( \theta \), which will preserve feasibility; that is choose \( \theta \) according to the following condition

\[ \theta = \min_{k} \{ x_{bk} | x_{bk} \text{ corresponds to cell labeled "-"} \}. \]

6) Compute a new set of basic variables by adding \( \theta \) to the cells with a "+" label and subtracting \( \theta \) from cells with a "-" label. One of the basic variables that now has a zero value will be nonbasic.

7) Return to step 2.

Note that the \( u_i, v_j \) variables described above are the variables for the dual of the transportation problem. Also, in the optimal tableau, the value of the \( u_i, v_j \) variables correspond to the optimal solution of the dual problem.

1.5 Overview

The preceding sections of this chapter gave an introduction on some of the basic concepts relating to the transportation problem. These included a brief review of some linear programming concepts, a statement of the properties of the transportation problem, and a description of an algorithm for solving the transportation problem.
In Chapter 2, we focus on the different computational procedures for finding an initial basic feasible solution. Studies comparing initial start procedures have been done by Srinivasan and Thompson (1973), Glover et al. (1974), and Ross et al. (1975). Since many different parameters had to be included in these studies, they limited the size of their studies and only considered a small number of samples which were generated for each group of problems tested. A more thorough comparison of various start procedures is considered in Chapter 2. A start method which is a variation of the Modified Minimum Row rule, the start procedure recommended in previous studies, is introduced with the expectation that it might give a better initial basis than the Modified Minimum Row rule. The different start procedures are compared based on the CPU time it took to find an initial basis and the closeness of the objective function value of the initial basis to the optimum. Upon comparing several procedures based on these two criteria, it is shown that a different computational procedure, Large Amount-Low Cost method, yields a better result than those considered in previous studies, especially for very large rectangular (number of origins < number of destinations) transportation problems. For square (number of origins = number of destinations) and smaller transportation problems, results of previous studies are verified with the Modified Minimum Row rule being the better start procedure.

The capacitated transportation problem is discussed in Chapter 3. It is illustrated that in finding an initial feasible basis for the capacitated case a general rule, such as the computational procedures presented in
Chapter 2, do not necessarily yield a feasible solution, so that procedures for adjusting an initial allocation to make it feasible are needed. Three such iterative adjustment procedures found in the literature are described. For some special forms of the capacitated transportation problem, like the application on least absolute value estimation presented in Chapter 4, it is possible to construct a rule for obtaining an initial feasible basic allocation that do not require any iterative adjustment procedure. In Chapter 4, one such rule is derived. In addition to finding an initial basis, an efficient algorithm for obtaining an optimal solution for the capacitated transportation problem is also described. This algorithm is a modification of an algorithm for solving uncapacitated transportation problems with the application of some concepts used in the simplex procedure for simple upper bounds.

In Chapter 4, two applications of the transportation problem to some statistical problem are presented, namely, obtaining least absolute value estimates for the two-way classification model and solving the problem of controlled rounding. A simple upper bounds procedure to obtain optimal solutions to these problems by solving an equivalent capacitated transportation problem is developed and demonstrated through numerical examples. In solving for least absolute value estimates for the two-way classification model, a new procedure for finding an initial feasible basis is developed. This procedure for finding an initial basis and the simplex procedure for simple upper bounds was implemented in a FORTRAN code and is given in Appendix B.

Finally, in Chapter 5 we present a summary of all the topics discussed in this thesis.
2. A COMPUTATIONAL STUDY OF METHODS FOR FINDING AN INITIAL BASIC FEASIBLE SOLUTION FOR TRANSPORTATION PROBLEMS

The first step of the simplex method for the transportation problem is to determine an initial basic feasible solution. The simplest procedure for finding an initial basic feasible solution was proposed by Dantzig (1951) and was termed the Northwest Corner rule by Charnes and Cooper (1954).

Using the notations defined in Chapter 1, the Northwest Corner rule proceeds in the following manner:

Starting at the northwest corner cell of the transportation tableau, allocate as much as possible to variable \( x_{11} \). By doing so will result in either exhausting the supply at origin 1 or satisfying the demand at destination 1. If the former occurs, continue by proceeding to row 2; if the latter occurs, continue in row 1 by allocating as much as possible to \( x_{12} \), which will either exhaust the supply at origin 1 or completely satisfy the demand of destination 2. This process is continued until all demands are met and all supplies are exhausted.

It is possible that at some step, other than the last, a supply is exhausted and a demand is satisfied simultaneously. In this case, a zero is arbitrarily assigned to the next variable in either the same row or the same column, and then proceed as described above.

Since the Northwest Corner rule does not take into account costs when allocating to the variables, it may yield an initial solution that gives an objective function value that is far from the optimum. This
may then result in an excessive number of iterations to get to the optimum solution.

There are other methods for finding an initial feasible solution that have appeared in literature. These methods are more efficient than the Northwest Corner rule because they take into account costs when selecting variables to become basic. The methods that will be examined in this chapter are the Row Minimum rule (Dennis, 1958; Hadley, 1962; Cooper and Steinberg, 1974). Column Minimum rule (Hadley, 1962; Cooper and Steinberg, 1974). Modified Row Minimum rule and Modified Column Minimum rule (Srinivasan and Thompson, 1973), Large Amount Low Cost method (Lee, 1968), and Vogel's method (Reinfeld and Vogel, 1958). These methods will be described in detail in Section 2.1.

Studies which have been devoted to finding the most efficient technique for solving transportation problems have been conducted by Srinivasan and Thompson (1973), Glover, Karney, Klingman, and Napier (1974), and Ross, Klingman, and Napier (1975). All of these studies included comparisons of several primal start methods.

Srinivasan and Thompson looked at 21 sample problems of size 175 x 175 that were 100 percent dense in comparing seven primal start procedures.

In the study by Glover et al., they compared the primal start methods using both dense and nondense square transportation problems of sizes ranging from 10 x 10 to 200 x 200. For the 100 percent dense problems, comparisons were based on median time to find an initial basis from a sample of ten problems for each problem size. The nondense problems involved five sample problems for each problem size.
The study by Ross et al. examined the effect of rectangularity of the transportation problem on solution times. The problems they looked at included 37 different combinations of the number of origins and destinations (see Table 2.1) with densities of 15, 20, 25, 30, and 35 percent for each problem size. A sample of five problems for each problem dimension and density was generated and the data corresponding to the problem in the sample associated with the median solution time for finding an initial basis was selected to represent that sample.

Table 2.1. Problem dimensions that were used in the study by Ross et al.

|                | 10 x 500 | 20 x 250 | 25 x 50 | 50 x 100 | 75 x 100 | 100 x 100 | 10 x 750 | 20 x 375 | 25 x 75 | 50 x 150 | 75 x 150 | 100 x 125 | 20 x 500 | 25 x 100 | 50 x 200 | 75 x 200 | 100 x 150 | 25 x 125 | 50 x 250 | 75 x 225 | 100 x 200 | 25 x 150 | 50 x 300 | 75 x 300 | 100 x 300 | 25 x 200 | 50 x 400 | 75 x 375 | 100 x 400 | 25 x 300 | 75 x 400 | 100 x 500 | 25 x 400 | 75 x 450 | 100 x 600 | 25 x 500 | 25 x 600 |
|----------------|----------|----------|---------|----------|----------|----------|----------|----------|---------|----------|----------|----------|----------|----------|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|

Since all of these studies were interested in finding an overall efficient method for solving transportation problems, many different parameters had to be included in their study. Thus, to limit the size of their study, only a small number of samples were generated for each group of problems.
The focus of this chapter is to do a more thorough comparison of initial feasible solution methods. A method will be preferred from the other methods if it gives an initial basic feasible solution that yields an objective function value that is closest to the optimum and does not use up a lot of computing time to obtain it. Since this study will deal only on the initial step of finding a feasible basis, more sample problems can be looked at and thus a more conclusive result can be reached.

2.1 Description of Methods for Finding an Initial Basic Feasible Solution

2.1.1 Row minimum rule

Begin in row 1, and find the smallest $c_{ij}$. Allocate as much as possible to the corresponding $x_{ij}$. If this allocation satisfies a destination demand continue in row 1, finding the smallest remaining $c_{ij}$, and allocating as much as possible for this corresponding variable. Continue in this manner until the supply at origin 1 has been exhausted. Then, proceed to row 2 and repeat the process, each time allocating as much as possible to the variable whose cost is the smallest, and moving to the next row whenever a supply is exhausted. If a supply is exhausted simultaneously with a demand being satisfied, arbitrarily assign zero to the variable in that row whose cost is the smallest of the remaining unallocated variables in the row.

2.1.2 Column minimum rule

This method is essentially the transpose of the Row Minimum rule. Instead of proceeding in a row by row fashion as in the row minimum rule,
one moves from column to column.

2.1.3 Modified row minimum rule

Begin in row 1, and find the smallest $c_{ij}$. Allocate as much as possible to the corresponding $x_{ij}$. Then, proceed to row 2, allocating as much as possible to the $x_{ij}$ with the smallest $c_{ij}$. Continue down through the rows, allocating only to one variable at a time. If the last row is reached and not all the row supplies have been exhausted, proceed back up beginning with the first encountered row with some unallocated supply remaining. Continue allocating in this manner until all the row supplies have been allocated.

This method is similar to the Row Minimum rule. It differs in that only one cell is selected each time a row is examined, whereas the Row Minimum rule continues to select cells in a row until the total supply of the row has been exhausted.

2.1.4 Modified column minimum rule

This method is just the transpose of the Modified Row Minimum rule. Instead of proceeding row by row as in the Modified Row Minimum rule, one proceeds by moving across the columns.

2.1.5 Large amount low cost method

Find the largest rim entry $a_i$, row supply, or $b_j$, column demand. In the corresponding row or column select the $x_{ij}$ having the lowest cost,
and allocate as much as possible to this variable. Continue allocating to the largest remaining rim entries, until all row supplies and column demands are satisfied.

2.1.6 Vogel's approximation method

For each row and each column, find the two smallest costs and take their difference. In the row or column that corresponds to the maximum difference, find the variable \( x_{ij} \) that has the lowest cost, \( c_{ij} \), and allocate as much as possible to this variable. If this allocation exhausts a supply, delete the corresponding row from further consideration. If this allocation satisfies a demand, delete the corresponding column from further consideration. If both occur simultaneously, delete only the row from further consideration. Repeat this process until all but one row or all but one column have been deleted. The remaining allocation in this one row or column will then be deterministic.

Vogel's method can be viewed as a penalty approach method. The difference between the two smallest costs in a row or column is the cost penalty that would be incurred if the allocation was not made to this row or column. One would then avoid incurring the largest cost penalty by allocating to the variable with the lowest cost in the row or column that has the largest penalty.

2.1.7 Minimum in both the row and column rule

This method is a variation of the Modified Row Minimum rule. The allocation is done in the same manner as the Modified Row Minimum rule
with the additional condition that an allocation is made to a variable \( x_{ij} \) with minimum cost \( c_{ij} \) in the row only if it is also minimum in its column. This method is included in the study because it may result in giving a better initial feasible solution than the Modified Row Minimum rule.

2.1.8 Other methods

There are other start procedures that were not included in our study, two of which are the Matrix Minimum rule (Hadley, 1962; Dantzig, 1963; Gaver and Thompson, 1973; Cooper and Steinberg, 1974), and Russell's Approximation method (Russell, 1969).

The Matrix Minimum rule involves searching for the smallest \( c_{ij} \) in the entire cost matrix from among rows \( i \) and columns \( j \) for which the supplies have not been exhausted or the demands have not been satisfied, respectively.

Russell's method proceeds in the following manner. For each row supply \( i \) remaining under consideration, determine its \( \bar{u}_i \), which is the largest unit cost, \( c_{ij} \), still remaining in that row. For each column demand \( j \) remaining under consideration, determine its \( \bar{v}_j \), which is the largest unit cost, \( c_{ij} \), still remaining in that column. For each variable \( x_{ij} \) not previously allocated in these rows and columns, calculate \( \Delta_{ij} = c_{ij} - \bar{u}_{ij} - \bar{v}_{ij} \). Allocate as much as possible to the variable having the largest negative value of \( \Delta_{ij} \).

These two methods involve searching through the entire transportation tableau to find the next variable to allocate. This would
involve considerable amount of time especially for large problems, and thus, are not very desirable.

2.2 Experimental Setup

The computer code that was used to generate the transportation problems and the methods for finding an initial basic feasible solution was written in VS FORTRAN and was tested on the National Advanced Systems AS/9160 computer at the Iowa State University Computation Center.

The problems used in the study consisted of 100 percent dense uncapacitated transportation problems of seven different dimensions. The exact dimensions of the problems used are presented in Table 2.2.

<table>
<thead>
<tr>
<th>Dimensions of transportation problems used in the study</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 x 20</td>
</tr>
<tr>
<td>100 x 150</td>
</tr>
</tbody>
</table>

For each problem dimension, a sample of 200 problems was generated. In every problem, total supply and total demand were set equal to 1000n, n being the number of destinations. A uniform probability distribution was used to randomly generate supply values within the range 100(n/m) ± 100(n/m) and demand values within the range 1000 ± 100(n/m),
m and n being the number of sources and the number of destinations, respectively. Each cost coefficient was an integer in the range 1 to 20.

The time it took to find an initial basic feasible solution and the objective function value that corresponded to this initial basis were recorded for each problem. From the time data, statistics such as the minimum, maximum, median, and mean time to find an initial basis were calculated. The objective function values that were obtained from the initial basis using the different methods for each problem were compared to determine which method would result in an objective function value that is closest to the optimum. A start procedure is then recommended to be the best among the methods tested after these two criteria have been examined.

2.3 Computational Results and Their Interpretation

Tables 2.3 to 2.9 show some summary statistics on the time it took to find an initial basis using different start procedures. The method that took the shortest time was the Minimum Column rule, while Vogel's method took the longest time. Vogel's method took seven times longer than the Minimum Column rule for the smallest problem tested, and as much as 28 times longer for the 100 x 600 problem.

Comparing the methods in terms of the objective function value that corresponded to their initial basis, Vogel's method dominated all the other methods tested for problem dimensions 10 x 20, 50 x 50, 100 x 100, 100 x 150, and 100 x 300 (see Tables 2.10 to 2.14). However, the
**Table 2.3.** Time to find an initial basis for transportation problems of dimension 10 x 20

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Mean time (msec)</th>
<th>Std. dev.</th>
<th>Min. time (msec)</th>
<th>Max. time (msec)</th>
<th>Median time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>0.465</td>
<td>0.038</td>
<td>0.402</td>
<td>0.867</td>
<td>0.464</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0.281</td>
<td>0.015</td>
<td>0.247</td>
<td>0.410</td>
<td>0.280</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>0.506</td>
<td>0.022</td>
<td>0.452</td>
<td>0.642</td>
<td>0.504</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0.343</td>
<td>0.020</td>
<td>0.301</td>
<td>0.472</td>
<td>0.345</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>0.980</td>
<td>0.055</td>
<td>0.695</td>
<td>1.220</td>
<td>0.980</td>
</tr>
<tr>
<td>Vogel's</td>
<td>2.315</td>
<td>0.287</td>
<td>1.940</td>
<td>3.931</td>
<td>2.256</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>1.171</td>
<td>0.096</td>
<td>0.960</td>
<td>1.500</td>
<td>1.165</td>
</tr>
</tbody>
</table>

**Table 2.4.** Time to find an initial basis for transportation problems of dimension 50 x 50

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Mean time (msec)</th>
<th>Std. dev.</th>
<th>Min. time (msec)</th>
<th>Max. time (msec)</th>
<th>Median time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>3.278</td>
<td>0.120</td>
<td>3.058</td>
<td>4.259</td>
<td>3.264</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>3.208</td>
<td>0.095</td>
<td>3.023</td>
<td>3.662</td>
<td>3.200</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>3.450</td>
<td>0.097</td>
<td>3.271</td>
<td>4.291</td>
<td>3.441</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>3.484</td>
<td>0.112</td>
<td>3.131</td>
<td>4.287</td>
<td>3.467</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>9.784</td>
<td>0.345</td>
<td>8.698</td>
<td>11.353</td>
<td>9.710</td>
</tr>
<tr>
<td>Vogel's</td>
<td>25.689</td>
<td>0.733</td>
<td>23.838</td>
<td>27.683</td>
<td>25.648</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>13.434</td>
<td>0.634</td>
<td>11.990</td>
<td>16.180</td>
<td>13.385</td>
</tr>
</tbody>
</table>
Table 2.5. Time to find an initial basis for transportation problem of dimension 100 x 100

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Mean time (msec)</th>
<th>Std. dev.</th>
<th>Min. time (msec)</th>
<th>Max. time (msec)</th>
<th>Median time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>12.118</td>
<td>0.137</td>
<td>11.603</td>
<td>12.462</td>
<td>12.122</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>12.059</td>
<td>0.154</td>
<td>11.588</td>
<td>12.576</td>
<td>12.072</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>12.781</td>
<td>0.181</td>
<td>12.364</td>
<td>13.880</td>
<td>12.774</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>13.249</td>
<td>0.141</td>
<td>12.628</td>
<td>13.602</td>
<td>13.242</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>36.691</td>
<td>0.256</td>
<td>34.836</td>
<td>37.623</td>
<td>36.715</td>
</tr>
<tr>
<td>Vogel's</td>
<td>117.875</td>
<td>3.078</td>
<td>110.094</td>
<td>125.285</td>
<td>118.046</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>58.002</td>
<td>3.371</td>
<td>51.538</td>
<td>68.381</td>
<td>57.322</td>
</tr>
</tbody>
</table>

Table 2.6. Time to find an initial basis for transportation problem of dimension 100 x 150

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Mean time (msec)</th>
<th>Std. dev.</th>
<th>Min. time (msec)</th>
<th>Max. time (msec)</th>
<th>Median time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>24.652</td>
<td>0.230</td>
<td>23.913</td>
<td>25.174</td>
<td>24.645</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>15.215</td>
<td>0.166</td>
<td>14.825</td>
<td>15.551</td>
<td>15.204</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>23.314</td>
<td>0.270</td>
<td>22.666</td>
<td>24.389</td>
<td>23.284</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>16.062</td>
<td>0.169</td>
<td>15.665</td>
<td>16.679</td>
<td>16.057</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>56.973</td>
<td>0.280</td>
<td>55.998</td>
<td>57.962</td>
<td>56.977</td>
</tr>
<tr>
<td>Vogel's</td>
<td>189.106</td>
<td>3.365</td>
<td>181.077</td>
<td>196.882</td>
<td>189.409</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>104.414</td>
<td>5.521</td>
<td>92.940</td>
<td>119.685</td>
<td>104.060</td>
</tr>
</tbody>
</table>
Table 2.7. Time to find an initial basis for transportation problems of dimension 100 x 300

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Mean time (msec)</th>
<th>Std. dev.</th>
<th>Min. time (msec)</th>
<th>Max. time (msec)</th>
<th>Median time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>70.796</td>
<td>0.654</td>
<td>69.379</td>
<td>75.210</td>
<td>70.707</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>27.384</td>
<td>0.227</td>
<td>26.835</td>
<td>28.444</td>
<td>27.363</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>73.417</td>
<td>0.813</td>
<td>71.368</td>
<td>75.949</td>
<td>73.266</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>26.389</td>
<td>0.287</td>
<td>25.835</td>
<td>27.216</td>
<td>26.313</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>159.497</td>
<td>1.866</td>
<td>155.741</td>
<td>170.233</td>
<td>159.072</td>
</tr>
<tr>
<td>Vogel's</td>
<td>477.181</td>
<td>9.152</td>
<td>452.557</td>
<td>503.267</td>
<td>477.819</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>285.583</td>
<td>16.689</td>
<td>251.725</td>
<td>363.687</td>
<td>283.404</td>
</tr>
</tbody>
</table>

Table 2.8. Time to find an initial basis for transportation problems of dimension 100 x 600

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Mean time (msec)</th>
<th>Std. dev.</th>
<th>Min. time (msec)</th>
<th>Max. time (msec)</th>
<th>Median time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>246.551</td>
<td>3.377</td>
<td>239.068</td>
<td>257.519</td>
<td>246.129</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>44.671</td>
<td>0.891</td>
<td>43.417</td>
<td>49.696</td>
<td>44.476</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>260.746</td>
<td>5.272</td>
<td>253.227</td>
<td>281.728</td>
<td>259.418</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>46.950</td>
<td>0.800</td>
<td>45.335</td>
<td>50.378</td>
<td>46.856</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>553.581</td>
<td>18.832</td>
<td>520.942</td>
<td>602.953</td>
<td>556.411</td>
</tr>
<tr>
<td>Vogel's</td>
<td>1274.188</td>
<td>23.053</td>
<td>1221.925</td>
<td>1221.925</td>
<td>1274.210</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>994.188</td>
<td>51.253</td>
<td>867.348</td>
<td>867.348</td>
<td>993.076</td>
</tr>
</tbody>
</table>
Table 2.9. Time to find an initial basis for transportation problems of dimension 200 x 400

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Mean time (msec)</th>
<th>Std. dev.</th>
<th>Min. time (msec)</th>
<th>Max. time (msec)</th>
<th>Median time (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>142.551</td>
<td>0.703</td>
<td>140.569</td>
<td>146.199</td>
<td>142.550</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>73.126</td>
<td>0.339</td>
<td>72.131</td>
<td>74.111</td>
<td>73.129</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>156.311</td>
<td>2.691</td>
<td>151.334</td>
<td>168.122</td>
<td>155.647</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>81.076</td>
<td>1.017</td>
<td>79.032</td>
<td>85.825</td>
<td>80.891</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>366.683</td>
<td>7.745</td>
<td>350.707</td>
<td>392.581</td>
<td>366.286</td>
</tr>
<tr>
<td>Vogel's</td>
<td>1725.815</td>
<td>35.477</td>
<td>1651.412</td>
<td>1100.261</td>
<td>1726.190</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>948.659</td>
<td>48.588</td>
<td>852.225</td>
<td>1822.663</td>
<td>947.168</td>
</tr>
</tbody>
</table>

Largest Amount Low Cost method appeared to be giving better initial basis for the more rectangular problems as shown in Tables 2.15 and 2.16.

Although Vogel's method resulted in determining a better initial basis than the other methods, it also takes the longest CPU time to find an initial basis. A comparison of the other methods, excluding Vogel's method, was done to find an alternative method to Vogel's method that would still be better than the other methods in terms of the objective function value but would not take as long to execute. This resulted with the Largest Amount Low Cost method dominating the other methods for all of the problem dimensions except for the 50 x 50 problem where the Minimum in Both the Row and Column rule did as well as the Largest
Table 2.10. Comparison of methods for finding an initial basis for transportation problems of dimension 10 x 20 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent^a out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Std. Mean dev. Maximum Median</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>2.50</td>
<td>30.52 16.51 84.89 28.93</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0.50</td>
<td>26.80 17.30 81.97 24.07</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>6.50</td>
<td>16.79 11.70 55.20 16.04</td>
</tr>
<tr>
<td>Modified Min. Column</td>
<td>1.50</td>
<td>23.87 15.29 77.96 20.63</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>9.50</td>
<td>15.26 11.64 67.43 14.90</td>
</tr>
<tr>
<td>Vogel's</td>
<td>71.50</td>
<td>1.78  3.71  25.11  0</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>8.00</td>
<td>17.28 12.45 55.95 15.75</td>
</tr>
</tbody>
</table>

^aNumber of times out of 200 problems a method gave the smallest objective function value.

Table 2.11. Comparison of methods for finding an initial basis for transportation problems of dimension 50 x 50 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent^a out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Std. Mean dev. Maximum Median</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>0.50</td>
<td>35.39 15.95 79.80 34.47</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>1.00</td>
<td>35.12 15.93 72.91 35.65</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>0.50</td>
<td>28.74 13.60 82.86 27.91</td>
</tr>
<tr>
<td>Modified Min. Column</td>
<td>1.00</td>
<td>30.19 14.12 71.39 28.50</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>2.50</td>
<td>27.45 13.09 61.88 27.36</td>
</tr>
<tr>
<td>Vogel's</td>
<td>92.50</td>
<td>0.32  1.55  12.30  0</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>2.00</td>
<td>26.49 13.23 66.67 26.08</td>
</tr>
</tbody>
</table>

^aNumber of times out of 200 problems a method gave the smallest objective function value.
Table 2.12. Comparison of methods for finding an initial basis for transportation problems of dimension 100 x 100 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Percent out of 200</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>0.50</td>
</tr>
<tr>
<td>Vogel's</td>
<td>99.50</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>0</td>
</tr>
</tbody>
</table>

*Number of times out of 200 problems a method gave the smallest objective function value.

Table 2.13. Comparison of methods for finding an initial basis for transportation problems of dimension 100 x 150 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Percent out of 200</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>19.00</td>
</tr>
<tr>
<td>Vogel's</td>
<td>81.00</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>0</td>
</tr>
</tbody>
</table>

*Number of times out of 200 problems a method gave the smallest objective function value.
Table 2.14. Comparison of methods for finding an initial basis for transportation problems of dimension 100 x 300 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Percent a out of 200</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>1.00</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>30.50</td>
</tr>
<tr>
<td>Vogel's</td>
<td>68.50</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>0</td>
</tr>
</tbody>
</table>

*aNumber of times out of 200 problems a method gave the smallest objective function value.

Table 2.15. Comparison of methods for finding an initial basis for transportation problems of dimension 100 x 600 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Percent a out of 200</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>71.00</td>
</tr>
<tr>
<td>Vogel's</td>
<td>29.00</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>0</td>
</tr>
</tbody>
</table>

*aNumber of times out of 200 problems a method gave the smallest objective function value.
Table 2.16. Comparison of methods for finding an initial basis for transportation problems of dimension 200 x 400 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Row</td>
<td>0</td>
<td>14.43 3.76 7.28 36.53 14.07</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0</td>
<td>12.71 3.62 4.67 26.57 12.55</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>0</td>
<td>5.81 3.11 0.05 20.14 5.25</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0</td>
<td>10.02 2.88 4.63 25.16 9.61</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>51.00</td>
<td>0.72 1.02 0 5.36 0</td>
</tr>
<tr>
<td>Vogel's</td>
<td>49.00</td>
<td>1.02 1.80 0 11.04 0.07</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>0</td>
<td>10.21 2.90 3.04 22.45 3.04</td>
</tr>
</tbody>
</table>

aNumber of times out of 200 problems a method gave the smallest objective function value.

Amount Low Cost method (see Tables 2.17 to 2.23). Comparing the Largest Amount Low Cost method to the method that took the shortest time to find an initial basis, it took 3.5 times longer than the shortest time for the 10 x 20 problem, and 12.5 times longer for the 100 x 600 problem. If it were compared to Vogel's method, it is 2 times faster for the 10 x 20 problem, and about 4 times faster for the 200 x 400 problem.

The studies that were done by Srinivasan and Thompson (1973), Glover et al. (1974), and Ross et al. (1975) recommended the Modified Minimum Row rule as the most efficient method for finding an initial basis. In our study, the Modified Row Minimum rule was second to the Largest Amount Low Cost method in terms of giving the smallest objective function value.
Table 2.17. Comparison of methods for finding an initial basis (excluding Vogel's method) for transportation problems of dimension 10 x 20 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. dev.</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>6.50</td>
<td>22.06</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>4.00</td>
<td>18.43</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>24.00</td>
<td>9.19</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>3.50</td>
<td>15.76</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>35.50</td>
<td>7.73</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>21.50</td>
<td>9.56</td>
</tr>
</tbody>
</table>

*Percent a Number of times out of 200 problems a method gave the smallest objective function value.

Table 2.18. Comparison of methods for finding an initial basis (excluding Vogel's method) for transportation problems of dimension 50 x 50 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. dev.</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>6.00</td>
<td>14.96</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>7.00</td>
<td>14.70</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>18.00</td>
<td>9.32</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>14.00</td>
<td>10.53</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>27.00</td>
<td>8.25</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>28.00</td>
<td>7.41</td>
</tr>
</tbody>
</table>

*Percent a Number of times out of 200 problems a method gave the smallest objective function value.
Table 2.19. Comparison of methods for finding an initial basis (excluding Vogel's method) for transportation problems of dimension 100 x 100 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent difference from the smallest objective function value a</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Percent out of 200</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>3.50</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>5.00</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>15.50</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>15.00</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>42.50</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>18.50</td>
</tr>
</tbody>
</table>

^aNumber of times out of 200 problems a method gave the smallest objective function value.

Table 2.20. Comparison of methods for finding an initial basis (excluding Vogel's method) for transportation problems of dimension 100 x 150 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent difference from the smallest objective function value a</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Percent out of 200</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>0</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>3.00</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>97.00</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>0</td>
</tr>
</tbody>
</table>

^aNumber of times out of 200 problems a method gave the smallest objective function value.
Table 2.21. Comparison of methods for finding an initial basis (excluding Vogel's method) for transportation problems of dimension 100 x 300 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. dev.</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>24.48</td>
<td>6.13</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>20.65</td>
<td>5.43</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>7.05</td>
<td>4.08</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>16.72</td>
<td>4.58</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>0.01</td>
<td>0.08</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>16.57</td>
<td>4.23</td>
</tr>
</tbody>
</table>

^aNumber of times out of 200 problems a method gave the smallest objective function value.

Table 2.22. Comparison of methods for finding an initial basis (excluding Vogel's method) for transportation problems of dimension 100 x 600 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percent out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>Std. dev.</td>
</tr>
<tr>
<td>Minimum Row</td>
<td>30.29</td>
<td>8.17</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>25.65</td>
<td>7.78</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>12.49</td>
<td>9.61</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>23.07</td>
<td>7.65</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>21.56</td>
<td>6.86</td>
</tr>
</tbody>
</table>

^aNumber of times out of 200 problems a method gave the smallest objective function value.
Table 2.23. Comparison of methods for finding an initial basis (excluding Vogel's method) for transportation problems of dimension 200 x 400 in terms of their objective function value

<table>
<thead>
<tr>
<th>Method for finding initial basis</th>
<th>Percenta out of 200</th>
<th>Percent difference from the smallest objective function value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean</td>
</tr>
<tr>
<td>Minimum Column</td>
<td>0</td>
<td>11.91</td>
</tr>
<tr>
<td>Modified Minimum Row</td>
<td>0.50</td>
<td>5.07</td>
</tr>
<tr>
<td>Modified Minimum Column</td>
<td>0</td>
<td>9.26</td>
</tr>
<tr>
<td>Large Amount Low Cost</td>
<td>99.50</td>
<td>0</td>
</tr>
<tr>
<td>Minimum in Row and Column</td>
<td>0</td>
<td>9.44</td>
</tr>
</tbody>
</table>

aNumber of times out of 200 problems a method gave the smallest objective function value.

(when Vogel's method was excluded from the comparison) for the larger problems. The Minimum in Both the Row and Column rule performed as well as the Modified Minimum Row rule in this criterion for the 10 x 20, 50 x 50, and 100 x 100 problems. However, the Minimum in Both the Row and Column rule would not even be considered as an efficient method to use since it takes more time than the Largest Amount Low Cost method. The Modified Minimum Row rule is about three times faster than the largest Amount Low Cost method for square problems. For rectangular problems, the Modified Minimum Row rule is only two times faster than the Largest Amount Low Cost method. The more elongated the problem the less difference in time is observed. However, the opposite is observed in the difference between the objective function values of the two methods.
This suggests that for square and not very rectangular problems, the Modified Minimum Row rule would do as well as the Largest Amount Low Cost method. For very large rectangular problems, the most efficient method to use is the Largest Amount Low Cost method.

Summarizing the results, if time is not of any concern then the method that would give the best initial feasible basis for square and not very rectangular problems is Vogel's method. However, if computation time is also considered important, the Modified Minimum Row rule would give the second best initial feasible basis for the same class of problems next to Vogel's method without using up as much CPU time. The Large Amount Low Cost method is considered to be the best method to use for very large rectangular problems.
3. THE CAPACITATED TRANSPORTATION PROBLEM

One variation of the transportation problem involves imposing capacity restrictions on the quantities to be transported between sources and destinations. This variation of the transportation problem is known as the capacitated transportation problem [Hadley (1962); Dantzig (1963); Spivey and Thrall (1970); Gass (1975)].

In this chapter, we will focus on the capacitated transportation problem, which includes methods for finding an initial basis, and algorithms for finding an optimum solution. The latter will involve a modification of an algorithm for solving uncapacitated transportation problems with the use of some concepts used in the simplex procedure for problems with simple upper bounds.

3.1 Definition of the Capacitated Transportation Problem

A capacitated transportation problem with m origins and n destinations involves finding \( x_{ij} \), the quantity to be shipped from origin \( i \) to destination \( j \), that will

\[
\text{minimize} \quad \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}
\]

subject to

\[
\sum_{j=1}^{n} x_{ij} = a_i, \quad i = 1,2,...,m \\
\sum_{i=1}^{m} x_{ij} = b_j, \quad j = 1,2,...,n \\
0 \leq x_{ij} \leq t_{ij}, \text{ for all } i \text{ and } j.
\]
The amount available for shipment from origin \( i \) is \( a_i \); the amount required by destination \( j \) is \( b_j \). The cost of shipping each unit from origin \( i \) to destination \( j \) is \( c_{ij} \). The upper bound restriction set on the quantity that can be shipped from origin \( i \) to destination \( j \) is \( r_{ij} \).

As in the uncapacitated case, we assume that \( \sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \). However, it is no longer true that this condition guarantees a feasible, and hence an optimal solution. Because of the upper bounds, there may not be any feasible solution.

3.2 Finding an Initial Feasible Solution

While simple rules have been devised for finding an initial feasible solution for the uncapacitated transportation problem, it does not appear possible, in general, to construct such a rule for the capacitated case. However, for some special forms of the capacitated transportation problem it is possible to construct a simple procedure for finding an initial feasible solution. This will be presented in Chapter 4, where specific applications of the transportation problem are discussed.

In this section, we will describe some procedures that have appeared in literature which handles the problem of finding an initial feasible solution for the capacitated case. These procedures will be described by solving the following numerical example.

Consider the following transportation tableau:
where the supply and demand requirements are specified to the right and below the above 3 x 4 transportation matrix, respectively. We shall assume that $0 \leq x_{ij} \leq 22$ for all $i$ and $j$.

We begin by allocating as much as possible to a variable $x_{ij}$ which will satisfy either a row supply or column demand making the variable basic. If this is not possible, because of the capacity restriction, the variable is given the value of its upper bound and remain nonbasic. Any of the start procedures that were presented in Section 2.1 can be used. For this example, using the Modified Minimum Row rule, we have the following allocation:
where the $x_{ij}'$s enclosed in squares are basic variables and the starred $x_{ij}'$s are nonbasic variables at their upper bound.

After using the Modified Minimum Row rule on our example, the allocation in (3.2) is not a feasible solution. Six units are lacking in row 3 and 3 units each are lacking in columns 3 and 4. Therefore, adjustments have to be made to (3.2) that will result in an initial feasible solution.

Procedures for adjusting this initial allocation to make it feasible are presented in Dantzig (1963), Krekô (1968), and Vajda (1975). Each of these procedures will now be described in detail.

3.2.1 Procedure presented by Dantzig (1963)

To (3.1), cells which will contain the shortages in the allocation are added to the table. These cells will be placed in an additional row and column which will be referred to as row $i = 0$ and column $j = 0$, respectively. The additional cells are placed in columns 3 and 4 of row $i = 0$, and row 3 of column $j = 0$. The original $c_{ij}'$s are replaced by $d_{ij} = 0$, with the shortage cells having $d_{ij} = 1$. The modified transportation table is shown in (3.3). We now have a problem of minimizing the sum of the artificial variables, in particular, $x_{03} + x_{04} + x_{30}$, subject to the same constraints as the original problem.
We then proceed to calculate the values of the dual variables, $u_i$ and $v_j$ for all $i$ and $j$, starting with $u_0$ and $v_0$ which are assigned a value of zero. Note that by doing this $u_3$, $v_3$, and $v_4$ must equal $d^0 = d^0 = d^g = 1$, respectively. A cell is then selected for which the still unallocated units are to be placed. The criteria for this is the same as that of finding an optimum allocation for a capacitated transportation problem which is: A solution is optimum if $u_i + v_j < c_{ij}$ for the nonbasic variables with value zero and $u_i + v_j \geq c_{ij}$ for the nonbasic variables with value at their upper bound. This condition for optimality will be discussed in detail in Section 3.3.2. Thus, we will choose to allocate to a nonbasic variable, $x_{ij}$, if it corresponds to 
\[
\max[-(c_{ij} - u_i - v_j) > 0 \text{ for } x_{ij} = 0; \ c_{ij} - u_i - v_j > 0 \text{ for } x_{ij} = t_{ij}].
\]
Referring to (3.4), we allocate 3 units to $x_{31}$.

\[
\begin{array}{cccc}
\hline
 & & & \\
20 & 5 & & \\
 & 0 & 0 & 0 \\
15 & & & 10 \\
1 & & & \\
\hline
\end{array}
\]

\[
\begin{array}{cccc}
\hline
 & 15 & 20 & 30 & 35 \\
\hline
1 & & & & \\
22 & 22 & & & \\
\hline
\end{array}
\]
This results in (3.5) with 3 units still lacking in row 3 and column 3.

We continue with a second iteration in which 3 units are allocated to $x_{22}$. This is shown by (3.6).
This second iteration furnishes us with a feasible solution. This is given by (3.7) with the original cost values, \( c_{ij} \), restored.
3.2.2 Procedure presented by Krekő (1968)

For calculations, we use problem (3.1) and add a row and a column with the costs for all the cells in this row and column being a large number, $M$, except for the intersection cell of the row and column which has zero cost. The extended table looks like this:

\[
\begin{array}{ccccc}
2 & 1 & 3 & 5 & M \\
4 & 2 & 10 & 6 & M \\
5 & 3 & 4 & 8 & M \\
M & M & M & M & 0 \\
\end{array}
\]

The extension is necessary because it is not known in advance whether the problem is solvable. The additional row and column is used to provide space for the quantities that could not otherwise be allocated. We allocate the units using the Modified Row Minimum rule assigning elements to the last row and last column only after we are unable to assign any further quantities to the original $3 \times 4$ matrix. We place the units that we are not able to assign in the additional fourth row and fifth column, assigning 6 units to $x_{35}$, 3 units to $x_{43}$, and 3 units
to $x_{44}$ and fixing the totals of the extra row and column at 6. The result is the transportation table in (3.8).

\[
\begin{array}{cccccc}
2 & 1 & 3 & 5 & M \\
4 & 2 & 10 & 6 & M \\
5 & 3 & 4 & 8 & M \\
M & M & M & M & 0 \\
\end{array}
\]

We now try to eliminate the fictitious shipments that have been assigned to the high-cost elements, M. This can be done only if it is possible to assign the 6 units to $x_{45}$ in the end, otherwise the original problem does not have a feasible solution.

We begin by determining the value of the dual variables, $u_i$ and $v_j$ for all $i$ and $j$. Then, by using the optimality criteria for the capacitated transportation problem, as stated in Section 3.2.2, we select a nonbasic variable to enter the solution and find the loop associated with the entering variable.

We augment (3.8) with the values of the dual variables.
We select \( x_{32} \) to be the entering variable. Constructing the loop associated with variable \( x_{32} \), we have:

We could move around 3 units in the loop, making \( x_{32} \) basic. The new table with the values of the dual variables is:
Cell (1,1) having \( c_{11} - u_1 - v_1 = 6 - 2M \) indicates that \( x_{11} \) should become basic. The loop corresponding to \( x_{11} \), as shown by (3.9), shows that 3 units can be moved around it, making \( x_{11} \) basic with an assigned value of 3.
The variable $x_{45}$ now has a value of 6, which means we have reached a feasible solution and can now delete the fourth row and fifth column. The feasible solution is given by (3.10).

\[
\begin{array}{|c|c|c|c|}
\hline
2 & 14 & 8 & 5 \\
\hline
4 & 2 & 10 & 6 \\
\hline
12 & 3 & 22^* & 4 \\
\hline
6 & 22^* & 8 & 50 \\
\hline
\end{array}
\]

(3.10)

3.2.3 Procedure presented by Vajda (1975)

We begin with the allocation in (3.2) in which 6 units are still missing. We deal with this by a succession of + and - in a circuit, starting and finishing at a starred entry, so that the starred entry receives a minus sign. This latter sign has actually no effect on the allocation of the starred entry. In our example, this gives us (3.11).

\[
\begin{array}{|c|c|c|c|}
\hline
2 & 1 & 3 & 5 \\
\hline
20 & - & 5 & + \\
\hline
4 & 2 & 10 & 6 \\
\hline
15 & 3 & 22^* & 4 \\
\hline
5 & 22^* & 8 & 50 \\
\hline
\end{array}
\]

(3.11)
We then obtain (3.12) which is still missing 3 units. So, we go through a second iteration of putting a succession of + and − in a circuit.

This results in (3.13) which identifies a feasible solution.

3.3 Obtaining an Optimum Feasible Solution

We have described in Section 3.2 procedures for obtaining an initial feasible solution for the capacitated transportation problem. From an initial feasible solution, we then proceed to find an optimum feasible
solution. To do this, we will apply some of the concepts from the simplex procedure for simple upper bounds to modify the transportation algorithm that was described in Chapter 1.

Before we proceed to apply the concepts of the simplex procedure for simple upper bounds to solve the capacitated transportation problem, we shall first look at solving the capacitated transportation problem using the transportation algorithm that we already know without any modifications and apply the modifications instead on the transportation matrix.

3.3.1 Applying the capacity restrictions as additional rows and columns in the transportation matrix

We will again use example (3.1), and instead of assuming an upper bound restriction of 22 on all the variables we will put upper bound restrictions on only two variables, \( 0 \leq x_{12} \leq 10 \) and \( 0 \leq x_{34} \leq 10 \).

The discussion that follows is based from Vajda (1981).

In the transportation problem, the marginal row total or column total puts implicit restrictions on the values of the variables in that row or column. With this in mind, the upper bound restrictions on \( x_{12} \) and \( x_{34} \) are put in the transportation matrix as additional rows and columns, one row and one column for each variable restriction. The capacity upper bounds are written as totals of additional rows as well as columns. Thus, our example showing the variables of the transportation problem will have the following form:
The original marginal totals are still valid, and the restrictions are applied through the additional rows and columns. The \( y_{ij} \) variables are included, because we must make sure that the variables which appear twice, e.g., \( x_{12} \), will have the same value in each case. This is ensured by the implied equation \( x_{12} + y_{12} = 10 \) appearing twice, with \( y_{12} \) written down only once.

The costs associated with each of the variables are now determined. The cost values associated with the \( x_{ij} \)'s which are merely restricted by the marginal totals stay the same. The restricted variables appear twice but their costs should only be counted once, so we attach the original cost to one of the positions and zero cost on the other. The \( y_{ij} \)'s have zero cost, while the cells where an entry is not needed is assigned a high cost, \( M \). Therefore, our example problem with its associated cost values is as follows:

<table>
<thead>
<tr>
<th></th>
<th>( x_{11} )</th>
<th>( x_{13} )</th>
<th>( x_{14} )</th>
<th>( x_{12} )</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_{21} )</td>
<td>( x_{22} )</td>
<td>( x_{23} )</td>
<td>( x_{24} )</td>
<td></td>
<td>25</td>
</tr>
<tr>
<td>( x_{31} )</td>
<td>( x_{32} )</td>
<td>( x_{33} )</td>
<td></td>
<td></td>
<td>50</td>
</tr>
<tr>
<td>( y_{12} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>( y_{34} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
</tr>
</tbody>
</table>

The costs associated with each of the variables are now determined. The cost values associated with the \( x_{ij} \)'s which are merely restricted by the marginal totals stay the same. The restricted variables appear twice but their costs should only be counted once, so we attach the original cost to one of the positions and zero cost on the other. The \( y_{ij} \)'s have zero cost, while the cells where an entry is not needed is assigned a high cost, \( M \). Therefore, our example problem with its associated cost values is as follows:
The transportation matrix in (3.14) can now be solved by applying the transportation algorithm as described in Chapter 1. No modification to the algorithm is needed since the capacity restrictions on the variables have already been applied through the modification of the transportation matrix.

This method for solving transportation problems with upper bound restrictions on the \( x_{ij} \)'s is not an efficient method to use if most of the \( x_{ij} \)'s have upper bound restrictions, since this will result in a large transportation matrix. If all the variables in a 3x4 transportation problem, like our example, have upper bound restrictions on them, this method will require us to solve a 15x16 transportation problem.
3.3.2 Using the simplex procedure for simple upper bounds to find an optimum solution for the capacitated transportation problem

The application of some of the concepts used in the simplex procedure for solving problems with simple upper bounds (SUB) to the algorithm for solving capacitated transportation problems have been discussed by Dantzig (1963), Krekó (1968), and Van de Panne (1976).

We shall illustrate this modification using example (3.1). We start with an initial feasible solution. We can use any one of the feasible solutions obtained in Section 3.2. Let us use the feasible allocation in (3.10), which is:

\[
\begin{array}{cccccc}
3 & 14 & 8 & 5 & 25 \\
4 & 2 & 10 & 6 & 25 \\
12 & 5 & 3 & 4 & 8 & 50 \\
15 & 20 & 30 & 35 & 25 & 25 \\
\end{array}
\]

As in the simplex procedure for SUB, the starred variables are nonbasic variables at their upper bound.

To determine the \( z_{ij} - c_{ij} \) coefficients for these basic variables, we need to modify this step, as in the SUB procedure (see Appendix A for a description of the simplex procedure for solving problems with SUB), by only considering the nonstarred feasible solution. This yields a set of \((u_i, v_j)\) as shown in (3.15).
Then \( c_{ij} - u_i - v_j \) for the nonbasic variables are

\[
\begin{array}{cccccc}
\hline
\textbf{v}_j & 2 & 1 & 3 & 4 \\
\textbf{u}_i & \\ 0 & 2 & 1 & 3 & 5 \\
2 & 4 & 2 & 10 & 6 \\
2 & 5 & 3 & 4 & 8 \\
\hline
\end{array}
\]

(3.15)

The second modification of the transportation algorithms must now consider the following:

(a) let \( \Delta_1 = \min \{c_{ij} - u_i - v_j \mid c_{ij} - u_i - v_j < 0 \text{ and the (i,j) cell is not starred} \} \).

(b) let \( \Delta_2 = \max \{c_{ij} - u_i - v_j \mid c_{ij} - u_i - v_j > 0 \text{ and the (i,j) cell is starred} \} \).

(c) let \( \Delta = \max \{-\Delta_1, \Delta_2 \} \).
Referring to (3.16), we have \( \Delta = 2 = \max\{-(-1), 2\} \). This implies that \( x_{34} \) is to enter the solution as a basic variable. Note that since \( x_{34} \) is at its upper bound, it can only decrease in value. Constructing an appropriate loop, we have the following:

\[
\begin{array}{ccccc}
2 & 1 & 3 & 5 \\
3 + \theta & 14 - \theta & 8 & \\
4 & 2 & 10 & 6 \\
12 - \theta & \\
5 & 3 & 4 & 8 \\
6 + \theta & * & * - \theta & 50 \\
15 & 20 & 30 & 35 & \\
\end{array}
\]

Now, \( \theta = \min\{22, 22-13, 12, 22-3, 14, 22-6\} = 9 \) implying that \( x_{34} \) should become basic and decrease in value by 9 units. Our next feasible solution will now be:

\[
\begin{array}{ccccc}
2 & 1 & 3 & 5 \\
12 & 5 & 8 & \\
4 & 2 & 10 & 6 \\
3 & \\
5 & 3 & 4 & 8 \\
15 & * & 13 & 50 \\
15 & 20 & 30 & 35 & \\
\end{array}
\]

Checking if this tableau is optimal, we must determine the values of \((u_i, v_j)\) for each basic variable. We therefore have:
The \( c_{ij} - u_i - v_j \) entries for the nonbasic variables are:

\[
\begin{array}{cccccc}
& 2 & 1 & 3 & 6 \\
\hline
u_1 & 2 & 1 & 3 & 5 \\
0 & 4 & 2 & 10 & * \\
2 & 3 & 5 & * & 8 \\
2 & 15 & 4 & 13 & . \\
\end{array}
\]

We have \( \Delta = -(-1) \), which corresponds to \( x_{14} \) and \( x_{22} \). Suppose \( x_{22} \) is chosen to enter the next solution as a basic variable. Constructing an appropriate loop, we have:
Thus, $\theta = \min\{22, 5, 22-12, 3\} = 3$. This implies that our next feasible solution is:

```
<table>
<thead>
<tr>
<th></th>
<th>2</th>
<th>1</th>
<th>3</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 + $\theta$</td>
<td>5 - $\theta$</td>
<td>8</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>10</td>
<td>6</td>
<td>25</td>
</tr>
<tr>
<td>3 - $\theta$</td>
<td>$\theta$</td>
<td></td>
<td></td>
<td>25</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>4</td>
<td>8</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td></td>
<td>13</td>
<td></td>
</tr>
</tbody>
</table>
```

Checking again for optimality, the values of $(u_i, v_j)$ for the basic variables are determined and then $c_{ij} - u_i - v_j$ is computed for each of the nonbasic variables. The result is the tableau given in (3.17).
This tableau is still not optimal since $c_{14} - u_1 - v_4 = -1$. So, we now let $x_{14}$ become basic. Determining a loop, we see that $x_{14}$ can only be increased by 2 since $\theta = \min\{22, 13, 22-15, 2\} = 2$.

We now have the following feasible solution:
This tableau is seen to be optimal as shown below:

<table>
<thead>
<tr>
<th></th>
<th>v_1</th>
<th>v_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>u_1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>17</td>
<td>11</td>
</tr>
</tbody>
</table>

yielding an optimal solution with \( x_{11} = 15, x_{13} = 8, x_{14} = 2, x_{22} = 3, \)
\( x_{24} = x_{33} = 22, x_{32} = 17, \) and \( x_{34} = 11. \)
4. STATISTICAL APPLICATIONS OF THE TRANSPORTATION PROBLEM

The relationship between linear programming and statistics has been recognized for the last three decades. Charnes, Cooper, and Ferguson (1955) chose, as an alternative to the least squares approach to linear regression, to minimize the sum of the absolute deviations [also known as least absolute value (LAV) estimation]. They were the first to point out the equivalence between LAV estimation and a linear programming problem. Other statistical problems that can be viewed as linear programming problems include determining the solutions of Tchebycheff inequalities, the generalized Neyman-Pearson problem, and a special class of stochastic programming often called chance-constrained programming [Sposito (1975)]. The general area of mathematical programming even covers a wider area of application in statistics including sampling, design and analysis of experiments, and cluster analysis [Arthanari and Dodge (1981)].

In this chapter we will deal specifically with the application of the transportation problem to statistics. Two applications will be discussed in the succeeding sections, namely, obtaining least absolute value estimates for the two-way classification model and solving the problem of controlled rounding.

4.1 Least Absolute Value Estimation for the Two-way Classification Model

The usual technique for obtaining estimators for the classification models are based on the method of least squares being applied to the
resulting regression problems. Inference theory in least squares relies on certain normality assumptions being satisfied. However, these assumptions may not always be valid and a more robust technique may be needed. One such technique is least absolute value (LAV) estimation. Among the desirable properties of LAV estimators are their resistance to outliers in the data and their robustness to heavy-tailed error distributions [Rice and White (1964); Barrodale (1968); Gentle (1977); Gentle, Kennedy, and Sposito (1977)].

The equivalence between the problem of obtaining least absolute value estimates for the two-way classification model and the capacitated transportation problem was demonstrated by Armstrong, Elam, and Hultz (1977). This relationship will be described here and LAV estimators for the two-way classification model will be obtained using the method described in Section 3.3.2.

### 4.1.1 Two-way classification model

Consider the classification model

\[ y_{ij} = \alpha_i + \beta_j + \epsilon_{ij}, \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n, \]

where \( y_{ij} \) is an observation at the \( i \)th level of the first factor and the \( j \)th level of the second factor. There are \( m \) levels of the first factor and \( n \) levels of the second factor. \( \alpha_i \) represents the effect of the \( i \)th level of the first factor and \( \beta_j \) represents the effect of the \( j \)th level of the second factor.
Suppose we wish to find estimates of $\alpha_i$ and $\beta_j$, for all $i$ and $j$, under the criterion of minimizing the sum of absolute deviations (also known as LAV estimates). In this situation we can obtain such estimates by solving the problem:

$$\minimize \sum_{i=1}^{m} \sum_{j=1}^{n} |y_{ij} - (\alpha_i + \beta_j)|. \quad (4.1)$$

By using the techniques for the general LAV regression problems first developed by Charnes, Cooper, and Ferguson (1955), (4.1) can be expressed as the following linear programming problem:

$$\minimize \sum_{i=1}^{m} \sum_{j=1}^{n} (e^+_{ij} + e^-_{ij}) \quad (4.2)$$

subject to $\alpha_i + \beta_j + e^+_{ij} - e^-_{ij} = y_{ij}$

$$e^+_{ij} > 0, \quad e^-_{ij} > 0, \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n,$$

where $e^+_{ij}$ and $e^-_{ij}$ are the positive and negative deviations of the regression equation from $y_{ij}$, respectively.

The dual of (4.2) is

$$\maximize \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} y_{ij}$$

subject to $\sum_{i=1}^{n} d_{ij} = 0, \quad i = 1, 2, \ldots, m$
Letting $f_{ij} = d_{ij} + 1$, for all $i$ and $j$, we have the following capacitated transportation problem:

\[
\begin{align*}
\text{maximize} & \quad \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}y_{ij} - \sum_{i=1}^{m} \sum_{j=1}^{n} y_{ij} \\
\text{subject to} & \quad \sum_{j=1}^{n} f_{ij} = n, \quad i = 1, 2, \ldots, m \\
& \quad \sum_{i=1}^{m} f_{ij} = m, \quad j = 1, 2, \ldots, n \\
& \quad 0 \leq f_{ij} \leq 2, \text{ for all } i \text{ and } j.
\end{align*}
\]

This problem is solvable since a feasible solution is $f_{ij} = 1$, for all $i$ and $j$, and thus a feasible solution also exists for the primal problem (4.2).

We can solve this problem using the algorithm described in Section 3.3.2 by first changing the objective function to

\[
\begin{align*}
-\min & \quad \sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}y_{ij} - \sum_{i=1}^{m} \sum_{j=1}^{n} y_{ij} \\
\end{align*}
\]

Expressing this problem in tabular form, we have
with capacity restrictions $0 \leq f_{ij} \leq 2$.

### 4.1.2 Obtaining an initial feasible solution

Before we can apply the algorithm in Section 3.3.2, we need to determine an initial feasible solution. It was shown in Chapter 3 that the rules devised for finding an initial feasible solution for the uncapacitated transportation problem do not always result in a feasible solution when applied to the capacitated case so that some iterative procedure will have to be applied to make the solution feasible. However, for the specific case of solving the dual of the two-way classification model, a simple rule can be constructed that will result in an initial feasible solution without making use of any iterative adjustment procedure. In applying this allocation procedure it must
be kept in mind that the basis that will make up an initial feasible solution should consist of \( m+n-1 \) basic variables.

The allocation rule proceeds as follows:

1. Take the first two rows and apply the Modified Minimum Row rule in the same manner as allocating to a \( 2 \times n \) transportation matrix with supply of \( n \) units per row and demand of 2 units per column. The variable that is the last to be allocated in each row is considered basic. This is done to assure that there will be \( m+n-1 \) basic variables. The other variables for which allocations were made are nonbasic and at their upper bound.

2. (a) If \( m \), the number of rows, is even continue as in step 1 until only two rows remain to be allocated.

   (b) If \( m \) is odd continue as in step 1 until only one row is left to be allocated.

3. (a) If only two rows remain to be allocated, allocate the same way as in step 1 except consider all the variables that have been allocated to as basic. Also for this case, if the number of columns, \( n \), is even the number of basic variables at this point is only \( m+n-2 \). A nonbasic variable at its upper bound is selected to become basic, making sure that no closed path results among the basic variables by the addition of this variable to the basis. If this is not possible, a nonbasic variable with zero value will have to be made basic.

   (b) If only one row remains to be allocated, all the variables in the last row are allocated one unit and are all basic.

The following examples illustrate how the procedure just described are applied to capacitated transportation problems of the type given in (4.3). The examples given are of different combinations of odd and even,
m and n values. The allocations enclosed in squares represent the basic variables and the starred allocations represent the nonbasic variables at the upper bound.

**Example 1: m=6 n=3**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2*</td>
<td>3</td>
<td>2*</td>
<td>1</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>2*</td>
<td>1</td>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>2*</td>
<td>6</td>
<td>5</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>5</td>
<td></td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

**Example 2: m=6 n=4**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>6</td>
<td>3</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>2*</td>
<td>6</td>
<td>4</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>6</td>
<td>5</td>
<td>1</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

6 6 6 6
The variable in cell (1,1) of Example 2 was arbitrarily selected among the nonbasic variables at their upper bounds to be basic so that the basis will consist of \(6 + 4 - 1 = 9\) variables.

**Example 3:** \(m=5\) \(n=3\)

- \(\begin{array}{|c|c|c|c|}
  \hline
  2^* & 1 & 5 & 6 \\
  \hline
  3 & 2 & 1 & 6 \\
  \hline
  2^* & 3 & 2 & 1 \\
  \hline
  4 & 2 & 3 & 1 \\
  \hline
  1 & 1 & 1 & 1 \\
  \hline
\end{array}
\)

**Example 4:** \(m=5\) \(n=4\)

- \(\begin{array}{|c|c|c|c|c|}
  \hline
  2^* & 1 & 5 & 6 & 3 \\
  \hline
  3 & 2 & 1 & 6 & 4 \\
  \hline
  2^* & 2 & 1 & 2 & 4 \\
  \hline
  1 & 1 & 1 & 1 & 6 \\
  \hline
\end{array}
\)

### 4.1.3 Obtaining an optimum feasible solution

We shall now apply the simplex procedure for simple upper bounds for solving capacitated transportation problems, which was described in
Section 3.3.2, to the problem of finding least absolute value estimators for a two-way classification model. This will be illustrated by means of an example.

Consider the following two-way table:

<table>
<thead>
<tr>
<th></th>
<th>j=1</th>
<th>j=2</th>
<th>j=3</th>
</tr>
</thead>
<tbody>
<tr>
<td>i=1</td>
<td>30</td>
<td>29</td>
<td>46</td>
</tr>
<tr>
<td>i=2</td>
<td>35</td>
<td>34</td>
<td>33</td>
</tr>
<tr>
<td>i=3</td>
<td>31</td>
<td>30</td>
<td>32</td>
</tr>
<tr>
<td>i=4</td>
<td>16</td>
<td>11</td>
<td>21</td>
</tr>
</tbody>
</table>

This two-way table expressed as a transportation matrix with the appropriate supply and demand requirements, and cost coefficients as denoted in (4.3) is given as follows:

\[
\begin{array}{ccc}
    f_{11} & f_{12} & f_{13} \\
    f_{21} & f_{22} & f_{23} \\
    f_{31} & f_{32} & f_{33} \\
    f_{41} & f_{42} & f_{43} \\
\end{array}
\]

We now find an initial feasible solution using the procedure given in Section 4.1.2. This results in the following initial solution:
The values of \((u^i, v^j)\) are determined for each basic variable and 
\(c^i_j - u^i - v^j\) is then computed for the nonbasic variables. This is shown in the following table:

\[
\begin{array}{ccc}
0 & -30 & -29 & -46 \\
4 & -35 & -34 & -33 \\
4 & -31 & -30 & -32 \\
0 & -16 & -11 & -21 \\
\end{array}
\]

We have \(\Delta = 8 = \max\{4, -(-8)\}\). This implies that the solution can be improved by allocating some quantity \(\theta\) in cell \((4,3)\). In particular,
where $\theta = \min\{2,1,1,2\} = 1$.

Letting $\theta = 1$ gives us our next feasible solution

Checking for optimality, we have that the $c_{ij} - u_i - v_j$'s are
We still do not have an optimal solution since $c_{31} - u_3 - v_1 = -4$. So, we need to allocate $\theta = \min\{2,1,1,2\} = 1$ to cell $(3,1)$ as shown by the following table:

$$
\begin{array}{cccc}
-30 & -29 & 2^* & -46 \\
2^* & -35 & -34 & -33 \\
-31 & -30 & -32 & \\
0 & -16 & -11 & -21 \\
\end{array}
$$

Our next feasible solution is

$$
\begin{array}{cccc}
-30 & -29 & 2^* & -46 \\
2^* & -35 & -34 & -33 \\
-31 & -30 & -32 & \\
0 & -16 & -11 & -21 \\
\end{array}
$$

This solution is optimal as shown by the table below:

$$
\begin{array}{cccc}
-30 & -29 & -35 \\
0 & -30 & -29 & -46 \\
-5 & -35 & -34 & -33 \\
0^* & -31 & -30 & -32 \\
-1 & -16 & -11 & -21 \\
14 & 4 & & \\
\end{array}
$$
After obtaining an optimal solution for the dual of the two-way classification model, how do we determine the LAV estimates for the $\alpha_i$'s and $\beta_j$'s. Recall that the transportation problem we solved is equivalent to the following:

\[
\begin{align*}
    -\min & \quad -\left[ \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} y_{ij} \right] \\
    \text{subject to} & \quad \sum_{j=1}^{n} d_{ij} = 0, \quad i = 1, 2, \ldots, m \\
    & \quad \sum_{i=1}^{m} d_{ij} = 0, \quad j = 1, 2, \ldots, n \\
    & \quad -d_{ij} \geq -1 \\
    & \quad d_{ij} \geq -1, \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n
\end{align*}
\]  

where $d_{ij} = f_{ij} - 1$, for all $i$ and $j$.

We know that the $u_i$'s and $v_j$'s in the optimal tableau is the optimal solution of the dual problem of (4.4) [Kreko (1968); Gass (1975); McLewin (1982)]

\[
\begin{align*}
    -\max & \quad -\sum_{i=1}^{m} \sum_{j=1}^{n} (e_{ij}^+ + e_{ij}^-) \\
    \text{subject to} & \quad u_i + v_j - e_{ij}^+ + e_{ij}^- = -y_{ij} \\
    & \quad e_{ij}^+ > 0, \quad e_{ij}^- > 0, \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n.
\end{align*}
\]  

This is also equivalent to:
\[
\min \sum_{i=1}^{m} \sum_{j=1}^{n} (e_{ij}^+ + e_{ij}^-)
\]

subject to \[-u_i - v_j + e_{ij}^+ - e_{ij}^- = y_{ij},\]

\[e_{ij}^+ \geq 0, \quad e_{ij}^- \geq 0, \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n.\]

Letting \(-u_i = \alpha_i\) and \(-v_j = \beta_j\), for all \(i\) and \(j\), we have our original linear programming problem for LAV estimation of the \(\alpha_i's\) and \(\beta_j's\).

Therefore, the LAV estimates for the \(\alpha_i's\) and \(\beta_j's\) correspond to the negative value of the \(u_i's\) and \(v_j's\), respectively, in the optimal transportation tableau.

For our example, we then have \(\alpha_1 = 0, \alpha_2 = 5, \alpha_3 = 1, \alpha_4 = -14, \beta_1 = 30, \beta_2 = 29, \) and \(\beta_3 = 35.\)

The LAV estimator for \(\alpha_i\) and \(\beta_j\) is not unique. The value of the \(u_i's\) and \(v_j's\) to which the \(\alpha_i's\) and \(\beta_j's\) correspond to, respectively, are determined by solving a system of \(m+n-1\) equations in \(m+n\) unknowns, e.g., solving the set of \(m+n-1\) equations \(c_{ij} - u_i - v_j = 0\) corresponding to the \(m+n-1\) basic variables for \(u_i, i = 1, 2, \ldots, m\) and \(v_j, j = 1, 2, \ldots, n.\)

This makes it necessary to add an equation setting one of the \(u_i's\) or \(v_j's\) to zero and then solving for the rest. So the values for the \(u_i's\) and \(v_j's\) will vary depending on which of the \(u_i's\) or \(v_j's\) is set equal to zero.

In our example, if instead of having \(u_1 = 0,\) we have \(v_2 = 0,\) then our estimates for the \(\alpha_i's\) and \(\beta_j's\) will be \(\alpha_1 = 29, \alpha_2 = 34, \alpha_3 = 30, \)
\( \alpha_4 = 15, \beta_1 = 1, \beta_2 = 0, \) and \( \beta_3 = 6 \) as determined from the following tableau:

\[
\begin{array}{ccc}
  v_j & -1 & 0 & -6 \\
 u_i & & & \\
  -29 & -30 & -29 & -46 \\
   -35 & -34 & -33 & 2^* \\
   -31 & -30 & -32 & 1 \\
   -16 & -11 & -21 & 2 \\
\end{array}
\]

There are other methods that can be used to solve for the LAV parameter estimates of the two-way classification model. Armstrong, Elam, and Hultz developed an algorithm which is a specialization of the Barrodale and Roberts' (1973) algorithm to the two-way classification problem. The method they developed is a special purpose network algorithm which solves the primal problem (4.2) directly. It is a combination of the techniques used in the general LAV algorithm and the approaches to solving network problems.

4.1.4 Computational results

A FORTRAN code was written to implement the procedure for obtaining an initial feasible solution described in Section 4.1.2 and the algorithm for finding an optimum solution for the capacitated transportation problem described in Section 3.3.2 specifically applied to solving the problem of
LAV estimation of the two-way classification model (see Appendix B for the listing of the FORTRAN program). This computer code was used to solve two-way classification problems of nine different dimensions with a sample of twenty problems for each dimension. For each problem solved, the CPU time to find an initial feasible solution, the average CPU time per iteration, the total CPU time to solve the problem, and the number of iterations needed to reach the optimum solution were recorded. The results are summarized in Table 4.1.

Table 4.1. Mean CPU time and mean number of iterations obtained from solving 20 sample problems for each problem dimension

<table>
<thead>
<tr>
<th>Problem dimension</th>
<th>Time to find an initial solution^a (msec)</th>
<th>No. of iter. to find opt. solution^a</th>
<th>Avg. time for each iteration^a (msec)</th>
<th>Total solution time^a (msec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 x 10</td>
<td>0.369 (0.019)</td>
<td>17 (3.4)</td>
<td>0.303 (0.015)</td>
<td>5.421 (0.978)</td>
</tr>
<tr>
<td>5 x 20</td>
<td>0.755 (0.024)</td>
<td>34 (4.0)</td>
<td>0.533 (0.029)</td>
<td>18.774 (2.664)</td>
</tr>
<tr>
<td>5 x 30</td>
<td>1.129 (0.034)</td>
<td>49 (5.5)</td>
<td>0.755 (0.024)</td>
<td>37.997 (4.734)</td>
</tr>
<tr>
<td>10 x 20</td>
<td>1.783 (0.026)</td>
<td>84 (8.9)</td>
<td>1.206 (0.051)</td>
<td>103.500 (12.514)</td>
</tr>
<tr>
<td>10 x 40</td>
<td>4.206 (0.114)</td>
<td>172 (13.3)</td>
<td>2.417 (0.086)</td>
<td>420.515 (38.174)</td>
</tr>
<tr>
<td>10 x 60</td>
<td>6.402 (0.116)</td>
<td>256 (20.8)</td>
<td>3.581 (0.093)</td>
<td>923.016 (77.035)</td>
</tr>
<tr>
<td>15 x 30</td>
<td>3.773 (0.061)</td>
<td>193 (12.4)</td>
<td>2.869 (0.096)</td>
<td>558.923 (42.556)</td>
</tr>
<tr>
<td>15 x 45</td>
<td>6.656 (0.353)</td>
<td>304 (20.2)</td>
<td>4.432 (0.134)</td>
<td>1352.470 (101.199)</td>
</tr>
<tr>
<td>15 x 60</td>
<td>8.873 (0.122)</td>
<td>397 (21.5)</td>
<td>5.881 (0.194)</td>
<td>2341.690 (151.085)</td>
</tr>
</tbody>
</table>

^aValues in parentheses are the corresponding standard deviations.
4.2 The Controlled Rounding Problem

The controlled rounding problem is the problem of optimally rounding real-valued entries in a two-way tabular array to adjacent integer values in a manner that preserves the additive structure of the array. The various applications of controlled rounding were described by Causey et al. (1985). Controlled rounding may be used to uniformize tabular data values for analysis. It may also be applied to statistical problems for which a complete solution requires that real numbered values be replaced by integers with minimum overall distortion to the original tabular data array. Another application of controlled rounding is to control statistical disclosure in tabular presentation of frequency counts. This is done because small counts can be inferred from released tables, that is, with frequency counts of small magnitude an individual respondent can be associated with a small and maybe identifiable subset of the respondent population, and thus a solution to prevent disclosure of confidential respondent data is to round all entries in the published tables to an appropriately chosen base with minimum overall distortion of the data. Other applications of controlled rounding include the problem of iterative proportional fitting or raking in two-way tables of counts as considered by Ireland and Kullback (1968), and the problem of controlled selection in the area of survey design as considered by Ernst (1981).

In this section, we will define and describe the solution of the controlled rounding problem, formulated as a capacitated transportation problem, developed by Cox and Ernst (1982).
4.2.1 Definition of the controlled rounding problem

Let \( A \) be a two dimensional array of real numbers with \( m+1 \) rows and \( n+1 \) columns. \( A \) is denoted in the form

\[
\begin{array}{c|c}
(a_{ij})_{m \times n} & (a_{i.})_{m \times 1} \\
(a_{.j})_{1 \times n} & (a_{..})_{1 \times 1}
\end{array}
\]

where \( (a_{ij})_{m \times n} \) make up the internal entries, \( (a_{i.})_{m \times 1} = \sum_{j=1}^{n} a_{ij} \) are the row total entries, \( (a_{.j})_{1 \times n} = \sum_{i=1}^{m} a_{ij} \) are the column total entries, and \( (a_{..})_{1 \times 1} = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij} \) is the grand total entry.

A solution to the controlled rounding of \( A \) to integer multiples of a positive integer base \( B \) is defined to be a function \( R^* \) that satisfies the following conditions:

1. For each entry \( a \) of \( A \), including totals, \( R^*(a) = B\lfloor a/B \rfloor \) or \( B(\lfloor a/B \rfloor + 1) \), where \( \lfloor \rfloor \) is the greatest integer function.

2. The array of rounded values is also tabular; that is,

\[
\begin{array}{c|c}
{R^*(a_{ij})}_{m \times n} & {R^*(a_{i.})}_{m \times 1} \\
{R^*(a_{.j})}_{1 \times n} & {R^*(a_{..})}_{1 \times 1}
\end{array}
\]

3. Given \( p, 1 \leq p < \infty \), \( R^* \) minimizes

\[
L_p(R, A) = \left( \sum_{i=1}^{m} \sum_{j=1}^{n} |R(a_{ij}) - a_{ij}|^p \right)^{1/p}
\]

or for \( p = \infty \),

\[
L_{\infty}(R, A) = \max\{|R(a_{ij}) - a_{ij}| : 1 \leq i \leq m, 1 \leq j \leq n\}.
\]
The solution to the controlled rounding problem that satisfies the above conditions is an optimal controlled rounding of $A$.

All controlled rounding problems can be simplified to the case where $0 \leq a_{ij} \leq 1$ and $B = 1$. This is done by dividing each of the $A$ values by $B$, a positive integer base to which values are to be rounded to, and then replacing each internal $a_{ij}$ entry by $a_{ij} - \lfloor a_{ij} \rfloor$ and adjusting all total entries accordingly.

4.2.2 The controlled rounding problem formulated as a capacitated transportation problem

The formulation of the controlled rounding problem as a capacitated transportation problem will be based on the simplifying assumptions that for a given $p$, $1 \leq p \leq \infty$, $0 \leq a_{ij} \leq 1$ and $B = 1$. This will involve defining a correspondence between controlled roundings $R$ of $A$ and a set of variables $X$ of a transportation problem, formulating a transportation problem type system of linear constraints consistent with $A$, and constructing an objective function which is a linear function in the $X$ variables whose set of minimizing solutions correspond with the set of controlled roundings of $A$ that minimizes the $L_p$ function given in condition 3 of Section 4.2.1.

Observe that the controlled roundings $R$ of $A$ correspond one-to-one with the $\{0, 1\}$ solutions of the $X$ variables to the system of linear equations defined by the tabular array

$$
\begin{array}{ccc}
(x_{ij})_{mxn} & ([a_{i.}] + x_{i.})_{mx1} \\
([a_{.j}] + x_{.j})_{lxn} & ([a_{..}] + x_{..})_{lx1}
\end{array}
$$

(4.5)
where \( x \) denotes an arbitrary \( X \) variable. It should also be noted that \( x_i, x_j, \) and \( x \) denote \( \{0, 1\} \) variables, not sums of the corresponding \( x_{ij} \)'s.

The tabular array (4.5) do not correspond to a transportation problem type system of linear constraints. Some modification to (4.5) is needed to bring it to the desired tabular array form. This is done as follows:

Add \((1 - x_i.)\) to each of the \( m \) rows, and adjust the corresponding row, column, and grand totals. Next, add \((1 - x_j.)\) to each of the \( n \) columns, and adjust the corresponding row, column, and grand totals. Lastly, add \( x \) to the cell in the \( m+1 \) row and \( n+1 \) column, and again adjust the row, column, and grand totals. The result is the following tabular array:

\[
\begin{array}{c|c|c}
(x_{ij})_{mxn} & (1 - x_i.)_{mxl} & ([a_i.] + 1)_{mxl} \\
(1 - x_j.)_{lxn} & (x.)_{lxl} & \{ \sum_{j=1}^{n} (1 - x_j.) + x \}._{lxl} \\
([a_j.] + 1)_{lxn} & \{ \sum_{i=1}^{m} (1 - x_i.) \} + x. & \{ \sum_{j=1}^{n} (1 - x_j.) + x \}._{lxl} \\
& & \\
\end{array}
\]

(4.6)

The tabular array (4.6) can still be further simplified.

From (4.5), \( x \) can be expressed as

\[
\sum_{i=1}^{m} ([a_i.] + x_i.) - [a.i.] = x.. \quad (4.7)
\]
or as
\[
\sum_{j=1}^{n} (a_{i,j} + x_{j}) - [a_{..}] = x_{..}.
\]  
(4.8)

Using (4.7), the row total of row \(m+1\) in (4.6) can be written as
\[
\sum_{i=1}^{m} (1-x_{i,j}) + x_{..} = \sum_{i=1}^{m} (1-x_{i,j}) + \sum_{i=1}^{m} (a_{i,j} + x_{j}) - [a_{..}]
\]
\[
= \sum_{i=1}^{m} [a_{i,j} + 1] - [a_{..}],
\]
(4.9)

and using (4.8), the column total of column \(n+1\) in (4.6) can be written as
\[
\sum_{j=1}^{n} (1-x_{i,j}) + x_{..} = \sum_{j=1}^{n} (1-x_{i,j}) + \sum_{j=1}^{n} (a_{i,j} + x_{j}) - [a_{..}]
\]
\[
= \sum_{j=1}^{n} [a_{i,j} + 1] - [a_{..}],
\]
(4.10)

By (4.9) and (4.10), the grand total in (4.6) can then be expressed as
\[
\sum_{i=1}^{m} \sum_{j=1}^{n} [a_{i,j} + 1] + \sum_{j=1}^{n} [a_{i,j} + 1] - [a_{..}].
\]

Therefore, (4.5) can be re-expressed by the following equivalent tabular array
\[
\begin{array}{ccc}
(x_{i,j})_{mxn} & (1-x_{i,j})_{mx1} & ([a_{i,j}] + 1)_{mx1} \\
(1-x_{i,j})_{1xn} & (x_{..})_{1xl} & (\sum [a_{i,j} + 1] - [a_{..}])_{1xl} \\
([a_{i,j} + 1])_{1xn} & (\sum [a_{i,j} + 1])_{1xl} & ([a_{i,j} + 1] + \sum [a_{i,j} + 1])_{1xl} - [a_{..}]_{1xl} \\
\end{array}
\]
(4.11)
The tabular array (4.11) corresponds to a system of linear constraints in the X variables of a transportation problem with capacity restriction 
0 \leq x_{ij} \leq 1, for all i and j. The entries corresponding to the row and column totals of (4.11) are always positive integers, which assures that the basic feasible solution to (4.11) are integer valued, namely, 0 or 1. It was shown by Cox and Ernst (1982) that a feasible solution satisfying (4.11) always exists.

To construct the objective function, condition 3 of Section 4.2.1 is used. For 1 \leq p \leq \infty, let the objective function be defined as

\[ z_p = z_p(R,A) = \{l_p(R,A)\}^p = \sum_{i=1}^{m} \sum_{j=1}^{n} |R(a_{ij}) - a_{ij}|^p. \quad (4.12) \]

The x_{ij}'s which correspond to the R(a_{ij})'s define two sets, a set D = \{(i,j): x_{ij} = 0 in R\} and a set U = \{(i,j): x_{ij} = 1 in R\}. Then, (4.12) can be written as

\[ z_p = \sum_{(i,j) \in D} (a_{ij})^p + \sum_{(i,j) \in U} (1 - a_{ij})^p \]

\[ = \sum_{i=1}^{m} \sum_{j=1}^{n} (a_{ij})^p (1 - x_{ij}) + \sum_{i=1}^{m} \sum_{j=1}^{n} (1 - a_{ij})^p x_{ij} \]

\[ = \sum_{i=1}^{m} \sum_{j=1}^{n} \{(1 - a_{ij})^p - (a_{ij})^p\} x_{ij} + \sum_{i=1}^{m} \sum_{j=1}^{n} (a_{ij})^p, \quad (4.13) \]

which is linear in the X variables.

Thus, the controlled rounding problem expressed as a capacitated transportation problem involves finding the x_{ij}'s that minimizes the
objective function (4.13) subject to the system of linear constraints in (4.11) with capacity restrictions $0 \leq x_{ij} \leq 1$, for all $i$ and $j$.

4.2.3 Obtaining a starting solution for the controlled rounding problem

The method for obtaining a starting solution for the controlled rounding problem that will be presented in this section was developed by Cox and Ernst (1982).

First, assume that the sequence $a_1, \ldots, a_m$ is nonincreasing. Let

$$R(a_{11}) = \lfloor a_{11} \rfloor,$$

$$R(a_{1i}) = \lfloor a_{1i} \rfloor,$$

and

$$R(a_{ii}) = \left[ \sum_{k=1}^{i-1} a_{ik} \right] - \left[ \sum_{k=1}^{i-1} a_{ki} \right], \quad i=2, \ldots, m.$$

To determine initial roundings for the internal entries of $A$, we proceed by recursing on $i$, starting with $i=1$ and ending with $i=m$. For a given $i$, assign a rounding $R(a_{ij}) = 1$ to $R(a_{ii})$ elements in row $i$ which correspond to the columns that have the largest value for

$$[a_{ij}] - \sum_{k=1}^{i-1} R(a_{kj}),$$

and a rounding $R(a_{ij}) = 0$ for all the other elements in row $i$. Finally, let

$$R(a_{i.}) = \sum_{j=1}^{n} R(a_{ij}), \quad j=1, \ldots, n.$$
From the initial roundings, an initial feasible basis for the transportation problem can be obtained. The initial value of the $x_{ij}$'s correspond to the value of the $R(a_{ij})$'s. The initial values of the $x_i$'s, $x_j$'s, and $x_{..}$ are $x_i = R(a_{i.}) - [a_{i.}]$, $i=1,...,m$; $x_j = R(a_{.j}) - [a_{.j}]$, $j=1,...,n$; and $x_{..} = 0$.

The methods for finding an initial feasible basis presented in Chapters 2 and 3 can also be used to obtain a starting solution for the controlled rounding problem. For a given $p$, say for $p=1$, the controlled rounding problem expressed in the form of a transportation problem tableau is

$$
\begin{array}{cccc}
1-2a_{11} & 1-2a_{12} & \cdots & 1-2a_{1n} \\
x_{11} & x_{12} & \cdots & x_{1n} & y_{1.} \\
1-2a_{21} & 1-2a_{22} & \cdots & 1-2a_{2n} & y_{2.} \\
& & & & \vdots \\
1-2a_{m1} & 1-2a_{m2} & \cdots & 1-2a_{mn} & y_{m.} \\
x_{m1} & x_{m2} & \cdots & x_{mn} & \eta_j \\
y_{1.} & y_{2.} & \cdots & y_{n.} & x_{..} & 0
\end{array}
$$

where $y_{.j} = 1 - x_{.j}$, for $j = 1, ..., n$; $y_{i.} = 1 - x_{i.}$, for $i = 1, ..., m$; and $0 \leq x_{ij} \leq 1$, $0 \leq x_{i.} \leq 1$, $0 \leq x_{.j} \leq 1$, $0 \leq x_{..} \leq 1$, for all $i$ and $j$.

$$
[a_{1.}+1] \\
[a_{2.}+1] \\
\vdots \\
[a_{m.}+1] \\
\eta_j = \sum_{i=1}^{n} (a_{ij}+1) - a_{..}
$$

(4.14)
Applying any of the methods for finding an initial basis for the capacitated transportation problem (4.14) is equivalent to obtaining an initial solution for the controlled rounding problem under the criterion of minimizing an $L_p$ function with $p = 1$.

Let us apply this latter method for finding an initial solution for a controlled rounding problem to an example.

Suppose we want to find a controlled rounding to the base 25 for the following two-way frequency counts table:

\[
\begin{array}{ccc}
30 & 1 & 28 \\
48 & 23 & 20 \\
18 & 11 & 14 \\
96 & 35 & 62 \\
\end{array}
\]

\[91 \]  \[91 \]  \[91 \]

(4.15)

The two-way table (4.15) can be simplified to a two-way table whose internal entries have values in the range of 0 to 1 and the resulting controlled rounding is to the base 1, as described in Section 4.2.1. The resulting equivalent controlled rounding problem is

\[
\begin{array}{ccc}
.20 & .04 & .12 \\
.92 & .92 & .80 \\
.72 & .44 & .56 \\
1.84 & 1.40 & 1.48 \\
\end{array}
\]

\[0.36 \]  \[2.64 \]  \[1.72 \]  \[4.72 \]

(4.16)

Expressing (4.16) in the form of a transportation problem tableau, as in (4.14), we have
Using the Modified Minimum Row method, an initial feasible basis is obtained for (4.17). This is shown by the following tableau:

\[
\begin{array}{cccccc}
 x_{11} & x_{12} & x_{13} & y_1 & 0 & 1 \\
 x_{21} & -0.84 & -0.84 & x_{23} & -0.60 & y_2 & 0 & 3 \\
 x_{31} & -0.44 & x_{32} & 0.12 & x_{33} & -0.12 & y_3 & 0 & 2 \\
 y_1 & 0 & y_2 & 0 & y_3 & 0 & x_\cdot & 0 & 2 \\
\end{array}
\]

\[(4.17)\]

where the variables enclosed in squares are basic, and the starred variables are nonbasic at its upper bound. From (4.18), an initial controlled rounding of (4.16) is

\[
\begin{array}{cccccc}
 0.60 & 0.92 & 0.76 & 0 & 1 \\
 1\cdot & -0.84 & -0.84 & -0.60 & 0 & 3 \\
 1\cdot & -0.44 & 0.12 & -0.12 & 0 & 2 \\
 0 & 1\cdot & 0 & 0 & 0 & 2 \\
\end{array}
\]

\[(4.18)\]
4.2.4 Obtaining an optimal solution for the controlled rounding problem

To obtain an optimal solution for the controlled rounding problem, its equivalent capacitated transportation problem is solved using the simplex procedure for simple upper bounds which was described in Section 3.3.2.

Starting with the initial feasible basis (4.18) of the controlled rounding example in the previous section, we proceed to find an optimal controlled rounding.

The values of \((u^*, v^*)\) are determined for each basic variable and \(c_{ij} - u^* - v_j\) is computed for each of the nonbasic variables. This is shown by the following tableau:

<table>
<thead>
<tr>
<th>( u^* )</th>
<th>(-.92)</th>
<th>(-.84)</th>
<th>(-.60)</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>( v_j )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>.60</td>
<td>.92</td>
<td>.76</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1.52</td>
<td>1.76</td>
<td>1.36</td>
<td>0</td>
</tr>
<tr>
<td>(.08)</td>
<td>-.84</td>
<td>-.84</td>
<td>-.60</td>
<td>0</td>
</tr>
<tr>
<td>(.48)</td>
<td>-.44</td>
<td>.12</td>
<td>-.12</td>
<td>0</td>
</tr>
<tr>
<td>(.60)</td>
<td>.32</td>
<td>.24</td>
<td>0</td>
<td>-.60</td>
</tr>
</tbody>
</table>
From the above tableau, we can see that the objective function value can be improved by decreasing variable $x_{21}$ or $y_2$, or increasing variable $y_3$ or $x_3$. Allocating some quantity $\theta$ to $x_3$ results in the largest improvement to the objective function value. From the following tableau,

<table>
<thead>
<tr>
<th></th>
<th>.60</th>
<th>.92</th>
<th>.76</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1*</td>
<td>-84</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

\( \theta \) is found to be equal to 1. Thus, our next feasible solution is

<table>
<thead>
<tr>
<th></th>
<th>.60</th>
<th>.92</th>
<th>.76</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1*</td>
<td>-84</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

We continue with the iterations until the optimality conditions are satisfied, that is, \( c_{ij} - u_i - v_j \geq 0 \) for all nonbasic variables.
equal to zero; and \( c_{ij} - u_i - v_j \leq 0 \) for all nonbasic variables equal to 1. After two more iterations, the following optimal solution is obtained:

\[
\begin{array}{cccc}
0.60 & 0.92 & 0.76 & 1 \\
-0.84 & 1^* & -0.84 & 1^* -0.60 & 0 \\
-0.44 & 0.12 & 1 & -0.12 & 0 \\
0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 2 & 1 \\
2 & 1 & 2 & 5 & .
\end{array}
\] (4.19)

So that an optimal controlled rounding of (4.16) is .

\[
\begin{array}{cccc}
0 & 0 & 0 & 0 \\
1 & 1 & 1 & 3 \\
1 & 0 & 1 & 2 \\
2 & 1 & 2 & 5 \\
\end{array}
\]

An optimal controlled rounding to the base \( B=25 \) of the entries in the original problem (4.15) is determined from the optimal transportation tableau (4.19) as follows:

\[
R^*(a_{ij}) = B([a_{ij}/B] + x_{ij}), \text{ for all internal entries;}
\]

\[
R^*(a_{i.}) = B([a_{i.}/B] + x_{i.}), \text{ for all row totals;}
\]

\[
R^*(a_{.j}) = B([a_{.j}/B] + x_{.j}), \text{ for all column totals;}
\]

and

\[
R^*(a_{..}) = B([a_{..}/B] + x_{..}), \text{ for the grand total.}
\]
Therefore, an optimal controlled rounding to the base 25 of (4.15) is

<table>
<thead>
<tr>
<th></th>
<th>25</th>
<th>0</th>
<th>25</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>25</td>
<td>25</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>0</td>
<td>25</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>----</td>
<td>---</td>
<td>----</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>25</td>
<td>75</td>
<td>200</td>
<td></td>
</tr>
</tbody>
</table>
In this thesis we have presented several aspects of the transportation problem. Various start procedures for generating an initial feasible basis were examined and compared to determine which of these methods would give a "better" initial feasible basis. The capacitated case of the transportation problem was also one of the topics discussed. Of interest too was the application of the methods for determining an optimal solution to the transportation problem to solve some statistical problems.

In Chapter 1, we defined some terms and stated some properties in linear programming that were used in the discussion of the transportation problem. A definition of the transportation problem was also given, and some theorems were presented to describe some of the properties of the transportation problem. An algorithm for finding an optimal solution to the transportation was also described.

Chapter 2 focused on the different computational procedures for finding an initial basic feasible solution. The start procedures that were compared included the Minimum Row rule, the Minimum Column rule, the Modified Minimum Row rule, the Modified Minimum Column rule, the Large Amount Low Cost method, Vogel's method, and the Minimum in Both Row and Column rule. The latter procedure was a variation of the Modified Minimum Row rule, the start procedure that was favored in previous studies, and was introduced in this study with the expectation that it would give a better initial basis than the Modified Minimum Row rule. The transportation problems that were used to compare the different start procedures
consisted of 100 percent dense uncapacitated transportation problems of seven different dimensions. A sample of 200 problems was generated for each problem dimension. The start procedures were compared based on the CPU time it took to find an initial basis and the closeness of the objective function value associated with an initial basis to the optimum. Vogel's method gave the smallest objective function value for square and not very rectangular problems, but it used the most CPU time. The Minimum Row, Minimum Column, and Modified Minimum Column rules did not do as well as the other methods in terms of objective function value, but the column methods used the least CPU time. The Minimum in Both Row and Column rule when compared with the Large Amount Low Cost method and the Modified Minimum Row rule did not do as well in terms of objective function value and used more CPU time than these two start procedures. The Large Amount Low Cost method gave the best objective function results for large rectangular problems compared to the other start procedures and used less CPU time than Vogel's method. The Modified Minimum Row rule was most appropriate for square and not very rectangular problems. For these type of problems, it was second to Vogel's method in terms of objective function value and used less CPU time than Vogel's method. It was concluded that the best method for obtaining an initial feasible basis for large rectangular transportation problems is the Large Amount Low Cost method, and for square and not very rectangular transportation problems is the Modified Minimum Row rule.
The capacitated transportation problem was discussed in Chapter 3. It was shown that in determining an initial feasible basis for the capacitated case a general rule, such as the computational procedures that were presented in Chapter 2, do not necessarily yield a feasible solution. Some iterative adjustment procedure would be needed to make such a solution feasible. In this chapter, we described three such iterative procedures that were found in the literature. Methods for obtaining an optimal basic feasible solution were also discussed. One method for solving the capacitated transportation problem involved expressing the capacity restrictions as additional rows and columns in the transportation tableau, and then solving the expanded tableau by using the method for the uncapacitated case. This method would not be efficient to use especially if most of the variables had capacity restrictions which would result in a very large transportation tableau. A more efficient algorithm for obtaining an optimal solution for the capacitated transportation problem was developed. This algorithm is a modification of an algorithm for solving the uncapacitated transportation problem with the application of some of the concepts used in the simplex procedure for problems with simple upper bounds.

In Chapter 4, two statistical problems which could be expressed in the form of a transportation problem were presented. The first problem discussed was obtaining least absolute value parameter estimates for the two-way classification model. When expressed as a linear programming problem, its associated dual problem has the form of a capacitated
transportation problem. For this specific type of a capacitated transportation problem it was shown that a certain rule for determining an initial basic feasible solution that would not require any iterative adjustment procedure is more efficient. This procedure for finding an initial basis and the simplex procedure for simple upper bounds was implemented in a FORTRAN code (see Appendix B). The second statistical problem discussed was the controlled rounding problem. This problem could also be expressed as a capacitated transportation problem. It was shown that a solution to this problem could be obtained by the application of the methods described in Chapter 3. The application of the simple upper bounds procedure to obtain optimal solutions for these two problems was demonstrated through numerical examples.
6. REFERENCES


7. ACKNOWLEDGEMENTS

I would like to express my sincere thanks to Dr. Vince Sposito for the guidance, encouragement, and patience that he has extended to me during the completion of this dissertation.

I am grateful to the Department of Statistics at Iowa State University for giving me the opportunity and the resources to be able to pursue and complete my doctorate degree.

I feel privileged to have been associated with the staff and graduate students in the Numerical Analysis Section. The four years that I have spent working with them will long be remembered.

Thanks to Darlene Wicks for her eager typing of this dissertation.

I thank my parents Socrates and Laura, my sisters Vickylu and Agnes, my brothers Plato and Aristotle, and my nieces and nephews Lemuel, Lynn, Lala, and Lance, though they are thousands of miles away, have always shared their love and support in all my endeavors.

Thank you to Dean and Wendy Zimmerman, for giving me a family away from home.

To Dale Zimmerman, my very best friend, thank you for your unfailing love and constant encouragement.

Finally, I thank God for making everything possible.
8. APPENDIX A: SIMPLEX PROCEDURE FOR PROBLEMS WITH UPPER BOUND CONSTRAINTS

The material being presented here can be found in Van de Panne (1971). Suppose we have the following linear programming problem:

\[
\text{maximize } z = \sum_{j=1}^{n} c_j x_j \\
\text{subject to } \sum_{j=1}^{n} a_{ij} x_j = b_i, \quad i=1,\ldots,m \\
x_j \leq t_j, \quad j=1,\ldots,n.
\]

where \( t_j \) is referred to as the upper bound of \( x_j \).

The rules for the simplex procedure for finding an optimal solution for linear programming problems with upper bound constraints, such as the one given above, is as follows.

The values of basic variables are indicated by \( b_i \). The indices of the columns with variables having an upper bound are said to belong to a set \( U \). If the nonbasic variable in column \( j \) has an upper bound, this bound is indicated by \( t_j \); similarly, an upper bound of the basic variable in row \( i \) is indicated by \( t^i \). Columns of nonbasic variables put at their upper bounds are said to belong to a set \( S \); the other columns are said to belong to a set \( N \).

I. Selection of the New Basic Variable

Select the column associated with the minimum \( (z_j - c_j) \);

\[
\min \{ (z_j - c_j) | z_j - c_j < 0; \, j \in N \}, \quad [-(z_j - c_j) | z_j - c_j > 0; \, j \in S] .
\]

Let \( k \) denote the column of this new basic variable. If \( z_j - c_j \geq 0 \) for
all $j \in N$ and $z_j - c_j \leq 0$ for all $j \in S$, then the optimal solution has been obtained.

II. Selection of the Leaving Variable $k$

If $k \in N$, select the row associated with

$$\min \left\{ \begin{array}{l}
(i) \frac{b_i}{a_{ik}} | a_{ik} > 0,
(ii) \frac{b_i - t^i}{a_{ik}} | a_{ik} < 0, i \in U,
(iii) t_k | k \in U;
\end{array} \right.$$  

if $k \in S$, select the row associated with

$$\min \left\{ \begin{array}{l}
(iv) \frac{b_i}{-a_{ik}} | a_{ik} < 0,
(v) \frac{t^i - b_i}{a_{ik}} | a_{ik} > 0, i \in U,
(vi) t_k | k \in U.
\end{array} \right.$$  

Let the row associated with the minimum in other cases than (iii) and (vi) be the $r$th row.

III. Transformation of the Tableau

Case (i): Transform the tableau with $a_{rk}$ as a pivot.

Case (ii): Subtract $t^r$ from $b_r$, transform with $a_{rk}$ as a pivot and include the column of the leaving basic variable in $S$.

Case (iii): Subtract $a_{ik} t^k$ from $b_i$ for all rows, include $k$ in $S$. 
Case (iv): Transform with $a_{rk}$ as a pivot and add $t^k$ to the new basic variable. Include the new basic variable in $N$.

Case (v): Subtract $t_r$ from $b_r$; transform with $a_{rk}$ as a pivot and add $t^k$ to the value of the new basic variable. Include the new basic variable in $N$ and the column of the leaving basic variable in $S$.

Case (vi): Add $a_{ik} t^k$ to $b_i$ for all rows. Include $k$ in $N$. 

9. APPENDIX B: A FORTRAN PROGRAM FOR FINDING LEAST ABSOLUTE VALUE
ESTIMATES FOR THE TWO-WAY CLASSIFICATION MODEL

THIS PROGRAM SOLVES A SPECIAL CASE OF THE TRANSPORTATION PROBLEM.
THIS SPECIAL CASE IS THE DUAL PROBLEM OF THE TWO-WAY CLASSIFICATION
MODEL, UNDER THE CRITERION OF MINIMIZING THE SUM OF ABSOLUTE
DEVIATIONS.

REAL C(60,60),COST(60),U(60),V(60),MAXCST

INTEGER CINDEX(60),INDEX(60,60),ALLOC(60,60),BASIC(60,60),
+ NROWB(60),NCOILB(60),LOOPR(120),
+ LOOPC(120),THETA,DIFF,OUTR,OUTC,OUTN

LOGICAL EVENR,EVENC,COLAL(60),RLABEL(60),CLABEL(60)

COMMON C,ALLOC

READ IN THE DATA
M -- THE NUMBER OF ROWS
N -- THE NUMBER OF COLUMNS
C(I,J) -- THE COST FOR THE Ith ROW AND Jth COLUMN

READ(1,901) M,N
901 FORMAT(2I3)
1000 DO 1 I=1,M
      READ(1,902,END=1001) (C(I,J),J=1,N)
902 FORMAT(60F8.3)
1 CONTINUE

DETERMINE IF THE NUMBER OF ROWS AND THE NUMBER OF COLUMNS ARE
ODD OR EVEN.

EVENR=.FALSE.
EVENC=.FALSE.
IF(MOD(M,2).EQ.O) EVENR=.TRUE.
IF(MOD(N,2).EQ.O) EVENC=.TRUE.
MN=M/2
NN=N/2
IF(EVENR) MN=MN - 1
IF(EVENC) NN=NN - 1

SORT THE COSTS IN EACH ROW IN ASCENDING ORDER

MS=M
IF (.NOT. EVENR) MS = M - 1
DO 4 I = 1, MS
   DO 2 J = 1, N
      CINDEX(J) = J
      COST(J) = C(I, J)
   2 CONTINUE
   CALL SORT (CINDEX, COST, N)
   DO 3 J = 1, N
      INDEX(I, J) = CINDEX(J)
   3 CONTINUE
   4 CONTINUE
C
C******************************************************************************
C  INITIALIZE VARIABLES TO ZERO
C  ALLOC(I, J) -- ALLOCATION FOR CELL (I, J)
C  BASIC(I, J) -- IDENTIFIES WHETHER CELL (I, J) IS BASIC
C      IF BASIC(I, J) = 0  NONBASIC
C           = 1  AT UPPERBOUND NONBASIC
C           = 2  BASIC
C  NROWB(I) -- NUMBER OF BASIC VARIABLES IN ROW I
C  NCOLB(J) -- NUMBER OF BASIC VARIABLES IN COLUMN J
C******************************************************************************
DO 6 I = 1, M
   DO 5 J = 1, N
      ALLOC(I, J) = 0
      BASIC(I, J) = 0
   5 CONTINUE
   6 CONTINUE
DO 25 I = 1, M
   NROWB(I) = 0
25 CONTINUE
DO 26 J = 1, N
   NCOLB(J) = 0
26 CONTINUE
C
C******************************************************************************
C  FIND AN INITIAL BASIC FEASIBLE SOLUTION.
C  THIS IS DONE BY ALLOCATING TO A PAIR OF ROWS AT A TIME USING THE
C  MODIFIED MINIMUM ROW METHOD. THE MAXIMUM THAT CAN BE ALLOCATED TO
C  A CELL IS 2. THE ROW SUPPLY IS EQUAL TO THE NUMBER OF COLUMNS AND
C  THE COLUMN DEMAND IS EQUAL TO THE NUMBER OF ROWS. IF THERE ARE AN
C  ODD NUMBER OF ROWS, EACH CELL IN THE LAST ROW WILL BE ALLOCATED
C  1 UNIT.
C******************************************************************************
ITER = 0
DO 14 I = 1, MN
   DO 7 J = 1, N
      COLAL(J) = .FALSE.
   7 CONTINUE
14 CONTINUE
CONTINUE
J1=0
J2=0
DO 10 J=1,NN
  J1=J1 + 1
  IF(.NOT.COLAL(INDEX(I*2-1,J1))) THEN
    ALLOC(I*2-1,INDEX(I*2-1,J1))=2
    BASIC(I*2-1,INDEX(I*2-1,J1))=1
    COLAL(INDEX(I*2-1,J1))=.TRUE.
  ELSE
    GO TO 8
  ENDIF
  J2=J2 + 1
  IF(.NOT.COLAL(INDEX(I*2,J2))) THEN
    ALLOC(I*2,INDEX(I*2,J2))=2
    BASIC(I*2,INDEX(I*2,J2))=1
    COLAL(INDEX(I*2,J2))=.TRUE.
  ELSE
    GO TO 9
  ENDIF
CONTINUE
IF(EVEN) THEN
  J1=J1 + 1
  IF(.NOT.COLAL(INDEX(I*2-1,J1))) THEN
    ALLOC(I*2-1,INDEX(I*2-1,J1))=2
    BASIC(I*2-1,INDEX(I*2-1,J1))=2
    COLAL(INDEX(I*2-1,J1))=.TRUE.
    NROWB(I*2-1)=NROWB(I*2-1) + 1
    NCOLB(INDEX(I*2-1,J1))=NCOLB(INDEX(I*2-1,J1)) + 1
  ELSE
    GO TO 11
  ENDIF
  J2=J2 + 1
  IF(.NOT.COLAL(INDEX(I*2,J2))) THEN
    ALLOC(I*2,INDEX(I*2,J2))=2
    BASIC(I*2,INDEX(I*2,J2))=2
    NROWB(I*2)=NROWB(I*2) + 1
    NCOLB(INDEX(I*2,J2))=NCOLB(INDEX(I*2,J2)) + 1
  ELSE
    GO TO 12
  ENDIF
ELSE
  J1=J1 + 1
  IF(.NOT.COLAL(INDEX(I*2-1,J1))) THEN
    ALLOC(I*2-1,INDEX(I*2-1,J1))=1
    BASIC(I*2-1,INDEX(I*2-1,J1))=2
    NROWB(I*2-1)=NROWB(I*2-1) + 1
    ALLOC(I*2,INDEX(I*2-1,J1))=1
    BASIC(I*2,INDEX(I*2-1,J1))=2
NR0WB(I*2) = NR0WB(I*2) + 1
NCOLB(INDEX(I*2-1, J1)) = NCOLB(INDEX(I*2-1, J1)) + 2
ELSE
   GO TO 13
ENDIF
ENDIF

CONTINUE
IF(EVENR) THEN
   MN = MN + 1
   DO 15 J = 1, N
      COLAL(J) = .FALSE.
   CONTINUE
ENDIF
IF(EVENC) NN = NN + 1
J1 = 0
J2 = 0
DO 18 J = 1, NN
   J1 = J1 + 1
   IF(.NOT. COLAL(INDEX(MN*2-1, J1))) THEN
      ALLOC(MN*2-1, INDEX(MN*2-1, J1)) = 2
      BASIC(MN*2-1, INDEX(MN*2-1, J1)) = 2
      COLAL(INDEX(MN*2-1, J1)) = .TRUE.
      NROWB(MN*2-1) = NROWB(MN*2-1) + 1
      NCOLB(INDEX(MN*2-1, J1)) = NCOLB(INDEX(MN*2-1, J1)) + 1
   ELSE
      GO TO 16
   ENDIF
   J2 = J2 + 1
   IF(.NOT. COLAL(INDEX(MN*2, J2))) THEN
      ALLOC(MN*2, INDEX(MN*2, J2)) = 2
      BASIC(MN*2, INDEX(MN*2, J2)) = 2
      COLAL(INDEX(MN*2, J2)) = .TRUE.
      NROWB(MN*2) = NROWB(MN*2) + 1
      NCOLB(INDEX(MN*2, J2)) = NCOLB(INDEX(MN*2, J2)) + 1
   ELSE
      GO TO 17
   ENDIF
CONTINUE
IF(EVENC) THEN
   M2 = M - 2
   DO 184 I = 1, M2
      DO 181 J = 1, N
         IF(BASIC(I, J).EQ.2) THEN
            MM = M - 1
         ELSE
            GO TO 182
         ENDIF
      CONTINUE
   CONTINUE
   DO 183 J = 1, N
      IF((BASIC(MM, J).EQ.2).AND.(BASIC(I, J).EQ.1)) THEN
         MM = M
      ELSE
         GO TO 182
      ENDIF
CONTINUE
   CONTINUE
   GO TO 182
ENDIF
BASIC(I,J)=2
NROWB(I)=NROWB(I) + 1
NCOLB(J)=NCOLB(J) + 1
GO TO 28
ENDIF

183 CONTINUE
184 CONTINUE

DO 185 J=1,N
IF((BASIC(MM,J).EQ.2).AND.(BASIC(M2,J).EQ.O)) THEN
BASIC(M2,J)=2
NROWB(M2)=NROWB(M2) + 1
NCOLB(J)=NCOLB(J) + 1
GO TO 28
ENDIF

185 CONTINUE
ELSE
J1=J1 + 1
IF(.NOT.COLAL(INDEX(MN*2-1,J1))) THEN
ALLOC(MN*2-1,INDEX(MN*2-1,J1))=1
BASIC(MN*2-1,INDEX(MN*2-1,J1))=2
NROWB(MN*2-1)=NROWB(MN*2-1) + 1
ALLOC(MN*2,INDEX(MN*2-1,J1))=1
BASIC(MN*2,INDEX(MN*2-1,J1))=2
NROWB(MN*2)=NROWB(MN*2) + 1
NCOLB(INDEX(MN*2-1,J1))=NCOLB(INDEX(MN*2-1,J1)) + 2
ELSE
GO TO 19
ENDIF
ENDIF
ELSE
DO 21 J=1,N
ALLOC(M,J)=1
BASIC(M,J)=2
NCOLB(J)=NCOLB(J) + 1
21 CONTINUE
NROWB(M)=N
ENDIF
C

C COMPUTE FOR U(I) AND V(J), ASSUMING U(1)=0
C
28 DO 29 I=1,M
RLABEL(I)=.FALSE.
29 CONTINUE
DO 30 J=1,N
CLABEL(J)=.FALSE.
30 CONTINUE
U(1)=O.
RLABEL(1)=.TRUE.
NCL=0
NRL=1
DO 31 J=1,N
   IF(BASIC(1,J).EQ.2) THEN
      V(J)=C(1,J) - U(1)
      CLABEL(J)=.TRUE.
      NCL=NCL + 1
   ENDIF
31 CONTINUE
32 IF(NRL.LT.M) THEN
   DO 34 I=2,M
      IF(.NOT.RLABEL(I)) THEN
         DO 33 J=1,N
            IF((BASIC(I,J).EQ.2).AND.(CLABEL(J))) THEN
               U(I)=C(I,J) - V(J)
               RLABEL(I)=.TRUE.
               NRL=NRL + 1
               GO TO 34
            ENDIF
         ENDIF
      ENDIF
33 CONTINUE
34 CONTINUE
   ENDIF
   IF(NCL.LT.N) THEN
      DO 37 J=1,N
         IF(.NOT.CLABEL(J)) THEN
            DO 36 I=1,M
               IF((BASIC(I,J).EQ.2).AND.(RLABEL(I))) THEN
                  V(J)=C(I,J) - U(I)
                  CLABEL(J)=.TRUE.
                  NCL=NCL + 1
                  GO TO 37
               ENDIF
            ENDIF
         ENDIF
      ENDIF
36 CONTINUE
37 CONTINUE
   ENDIF
38 IF((NCL.LT.N).OR.(NRL.LT.M)) GO TO 32
C******************************************************************************
C COMPUTE PER UNIT COST CHANGE, C(I,J) - U(I) - V(J), AND DETERMINE
C WHICH VARIABLE TO ENTER AS A NEW BASIS
C******************************************************************************
MAXCST=0
INR=0
INC=0
DO 41 I=1,M
   DO 40 J=1,N
      IF(BASIC(I,J).NE.2) THEN
UC = C(I,J) - U(I) - V(J) 
IF(((BASIC(I,J).EQ.0).AND.(UC.LT.0)).OR. 
+ ((BASIC(I,J).EQ.1).AND.(UC.GT.0.))) THEN 
  IF(ABS(UC).GE.ABS(MAXCST)) THEN 
    INR = I 
    INC = J 
    MAXCST = UC 
  ENDIF 
ENDIF 
ENDIF 
40 CONTINUE 
41 CONTINUE 
C CHECK FOR OPTIMALITY 
C ************************************************************ 
C NOT OPTIMAL SOLUTION, CONTINUE ITERATION. 
C ************************************************************ 
C IF((INR.NE.0).AND.(INC.NE.0)) THEN 
  ITER = ITER + 1 
C FIND THE PATH DETERMINED BY THE ENTERING VARIABLE 
C ************************************************************ 
LOOPR(1) = INR 
LOOPC(1) = INC 
NLOOP = 1 
JJ = 1 
43 DO 44 J = JJ, N 
  IF((J.NE.LOOPC(NLOOP)).AND.(BASIC(LOOPR(NLOOP),J).EQ.2).AND. 
+ ((J.EQ.LOOPC(1)).OR.(NCOLB(J).GT.1))) THEN 
    NLOOP = NLOOP + 1 
    LOOPR(NLOOP) = LOOPR(NLOOP - 1) 
    LOOPC(NLOOP) = J 
  ENDIF 
  IF(J.EQ.LOOPC(1)) GO TO 50 
  IF(J.EQ.LOOPC(1)) GO TO 50 
  JJ = J + 1 
44 CONTINUE 
II = LOOPR(NLOOP) + 1 
NLOOP = NLOOP - 1 
IF(II.GT.M) GO TO 48
DO 47 I=II,M
   IF((I.NE.LOOPR(NLOOP)).AND.(BASIC(I,LOOPC(NLOOP)).EQ.2).AND.  
   (NROWB(I).GT.1)) THEN
      NLOOP=NLOOP + 1
      LOOPR(NLOOP)=I
      LOOPC(NLOOP)=LOOPC(NLOOP-1)
      JJ=1
      GO TO 43
   ENDIF
47 CONTINUE
48 JJ=LOOPC(NLOOP) + 1
   NLOOP=NLOOP - 1
   IF(JJ.GT.N) GO TO 45
   GO TO 43

C
C FIND THE LEAVING VARIABLE
C *********************************************************************************

50 THETA=2
   IF(BASIC(LOOPR(I),LOOPC(I)).EQ.0) THEN
      DO 51 I=2,NLOOP
         IF(MOD(I,2).EQ.0) THEN
            DIFF=ALLOC(LOOPR(I),LOOPC(I))
         ELSE
            DIFF=2 + ALLOC(LOOPR(I),LOOPC(I))
         ENDIF
         IF((DIFF.LT.THETA).OR.  
            ((DIFF.EQ.THETA).AND.(MOD(I,2).EQ.0))) THEN
            THETA=DIFF
            OUTR=LOOPR(I)
            OUTC=LOOPC(I)
            OUTN=I
         ENDIF
51 CONTINUE
   DO 52 I=1,NLOOP
      IF(MOD(I,2).EQ.0) THEN
         ALLOC(LOOPR(I),LOOPC(I))=ALLOC(LOOPR(I),LOOPC(I)) - THETA
      ELSE
         ALLOC(LOOPR(I),LOOPC(I))=ALLOC(LOOPR(I),LOOPC(I)) + THETA
      ENDIF
52 CONTINUE
   IF(MOD(OUTN,2).EQ.0) THEN
      BASIC(OUTR,OUTC)=0
   ELSE
      BASIC(OUTR,OUTC)=1
   ENDIF
ELSE
   DO 54 I=2,NLOOP
      IF(MOD(I,2).EQ.0) THEN
DIFF = 2 - ALLOC(LOOPR(I), LOOPC(I))
ELSE
  DIFF = ALLOC(LOOPR(I), LOOPC(I))
ENDIF

IF((DIFF .LT. THETA) .OR. ((DIFF .EQ. THETA) .AND. (MOD(I, 2) .NE. 0))) THEN
  THETA = DIFF
  OUTF = LOOPR(I)
  OUTC = LOOPC(I)
  OUTN = I
ENDIF

CONTINUE

DO 55 I = 1, NLOOP
  IF(MOD(I, 2) .EQ. 0) THEN
    ALLOC(LOOPR(I), LOOPC(I)) = ALLOC(LOOPR(I), LOOPC(I)) + THETA
  ELSE
    ALLOC(LOOPR(I), LOOPC(I)) = ALLOC(LOOPR(I), LOOPC(I)) - THETA
  ENDIF
55 CONTINUE

IF(MOD(OUTN, 2) .EQ. 0) THEN
  BASIC(OUTR, OUTC) = 1
ELSE
  BASIC(OUTR, OUTC) = 0
ENDIF

BASIC(LOOPR(1), LOOPC(1)) = 2
NROWB(LOOPR(1)) = NROWB(LOOPR(1)) + 1
NCOLB(LOOPC(1)) = NCOLB(LOOPC(1)) + 1
NROWB(OUTR) = NROWB(OUTR) - 1
NCOLB(OUTC) = NCOLB(OUTC) - 1

C
C NEW BASIC SOLUTION HAS BEEN FOUND.
C GO TO 28 --- COMPUTE NEW U(I) AND V(J)
C
GO TO 28

C
C SOLUTION IS OPTIMAL
C
ELSE
  WRITE(6, 903) ITER
  903 FORMAT(1H1,'OPTIMUM SOLUTION WAS REACHED AFTER', I3, + ' ITERATIONS.')
  WRITE(6, 904) (I, -1*U(I), I = 1, M)
  904 FORMAT(/12(5(2X,'ALPHA(', I2, ')=', F12.4))/)
  WRITE(6, 905) (J, -1*V(J), J = 1, N)
  905 FORMAT(/12(5(2X,'BETA(', I2, ')=', F12.4))/)
WRITE(6,999)
999 FORMAT(1H1)
ENDIF
GO TO 1000
1001 STOP
END

C
C SUBROUTINE USED FOR SORTING ROW COSTS
C
SUBROUTINE SORT(INDEX,COST,N)
C
REAL COST(N)
INTEGER INDEX(N)
C
I=1
101 IF(I-N) 102,102,103
102 I=I+1
GO TO 101
103 M=I-1
104 M=M/2
IF(M) 111,111,105
105 K=N-M
DO 110 J=1,K
   I=J+M
   IF(I) 110,110,107
   L=I+M
   IF(COST(INDEX(L)) .GE. COST(INDEX(I))) GO TO 110
   IX=INDEX(I)
   INDEX(I)=INDEX(L)
   INDEX(L)=IX
   GO TO 106
110 CONTINUE
GO TO 104
111 RETURN
END