Mathl Comput. Modelling, Vol. 12, No. 8, pp. 1035–1044, 1989 0895-7177/89 \$3.00 + 0.00
Printed in Great Britain. All rights reserved Copyright © 1989 Maxwell Pergamon Macmillan plc

A REFINED SIMPLEX ALGORITHM FOR THE CLASSICAL TRANSPORTATION PROBLEM WITH APPLICATION TO PARAMETRIC ANALYSIS

H. Arsham and A. B. Kahn
Department of Information and Quantitative Sciences, University of Baltimore, Baltimore, MD 21201, U.S.A.

(*Received October 1988; accepted for publication December 1988*)

Communicated by E. Y. Rodin

Abstract—The well-known sensitivity analysis (SA) of linear programming (LP) is rarely performed on transportation problems (TP) because it is not directly applicable to TP algorithms such as Stepping Stone or its modified versions. These algorithms do not provide the final LP tableau which is prerequisite to all post-optimality analysis. This prerequisite is automatically fulfilled by simplex or dual simplex. However, they are very tedious because they increase computational complexity through the introduction of additional variables. A further complication arises in that TP always has dependencies require that SA receive the more general treatment of parametric analysis (PA). This paper presents an efficient PA for TP.

INTRODUCTION

Post-optimality, i.e. sensitivity analysis (SA) and in general parametric analysis (PA), is an important part of linear programming (LP). Much work has been done since Saaty [1] introduced the concepts of parametric programming and critical values in 1959. A good review, including stochastic perturbation, is given by Deif [2].

Adaptation of SA for transportation problems (TP) has been slow. Formulation of TP as LP becomes tedious because the customary treatment of equality constraints requires artificial variables which increase computational complexity. As a result it is traditional to use Stepping Stone (SS) or one of its modifications to solve TP.

However SS is not general purpose. Although Shih [3] was able to handle capacitation he required extensive and tedious modifications to SS. Furthermore SS completely fails with the so called "degenerate case" which is a frequent and important real-life situation. Arsham and Kahn [4] have resolved some of these difficulties by adapting the refined simplex method [5] to TP. Since the TP formulation as a simplex tableau results in a "unimodular" matrix of coefficient, considerable simplification of the algorithm was possible.

The current paper adapts the available PA of LP to TP. We start with a nominal solution in the form of a simplex tableau derived with a new algorithm. We present two techniques: one for perturbation of the costs, and a second for perturbation of the supply/demand on the R.H.S. of the tableau.

FORMULATION OF TP AS AN LP

The transportation problem is a special case of the general class known as LP problems. The elements of the problem are:

—A set of m origins: $O_1, O_2, \ldots, O_m$. For example, warehouses or plants.
—A set of n destinations: $D_1, D_2, \ldots, D_n$. For example, markets.
—A set of supplies of identical units: a_1 units at O_1, a_2 units at $O_2, \ldots, a_m$ units at O_m.
—A set of requirements: e_1 units required at D_1, e_2 units required at $D_2, \ldots, e_n$ units required at D_n.
—A set of unit shipment cost: C_{ij} is the cost of shipping 1 unit from O_i to D_j. There are $m \cdot n$ such costs.

The goal is to find a shipment plan that satisfies all the requirements with minimum total shipment cost. There are $m \cdot n$ decision variables, x_{ij}, representing the number of units to ship from O_i to D_j. The total cost of the shipment schedule, as a function of the decision variables,

$$E = \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij}$$

A "balanced" transportation problem is one in which the total supply exactly equals the total requirement. There is no loss in generality to assume that this is true because all unbalanced problems can be easily converted to balanced problems by addition of a "slack" or "dummy" row or column having just enough supply or requirement to balance the problem. In symbols, a balanced problem is one where

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} e_j.$$

The usual presentation of a balanced TP cost matrix is given in Table 1.

Table 1. TP cost matrix

	D_1	D_2	$D_{..}$	D_n	Avl
O_1	C_{11} X_{11}	C_{12} X_{12}	$C_{1.}$ $X_{1.}$	C_{1n} X_{1n}	a_1
O_2	C_{21} X_{21}	C_{22} X_{22}	$C_{2.}$ $X_{2.}$	C_{2n} X_{2n}	a_2
$O_.$	$C_{.1}$ $X_{.1}$	$C_{.2}$ $X_{.2}$	$C_{.}$ $X_{.}$	$C_{.n}$ $X_{.n}$	$a_{..}$
O_m	C_{m1} X_{m1}	C_{m2} X_{m2}	$C_{m.}$ $X_{m.}$	C_{mn} X_{mn}	a_n
Reqrmnt	e_1	e_2	$e_{..}$	e_n	

All of the constraints involve equality. The supply constraints are

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad \text{for } i = 1, 2, \ldots, m,$$

and the requirement constraints are

$$\sum_{i=1}^{m} x_{ij} = e_j \qquad \text{for } j = 1, 2, \ldots, n.$$

The TP in matrix form is

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} \cdot X_{ij}$$

$$\text{s.t. } AX = b,$$

where

$$b = (a_1, a_2, \ldots a_m, e_1, e_2, \ldots, e_n).$$

Clearly, in any balanced problem, one of these $m + n$ constraints is redundant. Rather than arbitrarily eliminate a constraint, the new algorithm, presented in the next section eliminates the constraint which will most reduce the number of iterations.

THE ALGORITHM

Notation

TP	Transportation problem
LP	Linear programming
SS	Stepping stone
GJP	Gauss–Jordan pivoting
BV	Basic variable
BVS	Basic variable set
FE	Feasibility
PR	Pivot row (row to be assigned to the variable coming in)
PC	Pivot column (column associated with the variable to come in)
PE	Pivot element
OR	Open row. A row not yet assigned a BV. Labelled (?)
(?)	Label for a row which is not yet assigned a BV (open row)
R.H.S.	Right-hand side
C/R	Column ratio R.H.S./PC

The algorithm consists of initialization *preliminaries* followed by two phases. The first, uses a *BV iteration*, to develop a BVS which may or may not be feasible. The second, uses an *FE iteration*, to develop a feasible and optimum solution. Both use the GJP transformation. However, they differ in the method used to select the PE. The BV iteration uses simplex criteria modified to select only open rows not already assigned to a BV. This strategy pushes towards an optimal vertex, and sometimes past it into non-FE. The FE iteration, if needed, pulls the solution back to FE using the dual simplex criteria for selecting the PE.

These major phases include two independent sets of optional operations which improve efficiency. Steps 0.1 and 0.2 eliminate the constraint which will most reduce the number of iterations. The second group comprises three operations: 1.2c, 2.2a and 2.2d which together progressively reduce the size of the tableau. Omission of either set would cause extra computation, but not prevent the attainment of optimality.

0. Preliminaries

Step 0.0. *TP cost matrix formulation.*

Step 0.1. *Row–column reduction* (*or column–row reduction*). From each row subtract the smallest cost. Accumulate the effect of each row reduction into the base cost.
Similarly, subtract the smallest cost in each column. Accumulate the effect of each column reduction in the base cost.

Step 0.2. *Eliminate redundant constraint.*
Identify row, or column with the most zeros for reduced costs.
Break any ties arbitrarily.
Eliminate the corresponding constraint.

Step 0.3. *Set up the simplex tableau.*
Use a row for each constraint and a column for each variable.
Do not add artificial variables.

Step 0.4. *Identify BVs*
For each column which is a unit vector label the row containing the one with the name of the variable for the column.
Label the remaining rows with question mark (?).

Step 0.5. *Delete BV columns.*

1. BV phase

Step 1.0. *BVS iteration termination test.*
IF (?) label exists, (there are ORs)
THEN continue the BV iteration
OTHERWISE BVS is complete, start FE phase (Step 2.0).

Step 1.1. *BV selection of PE.*
PC: Select the smallest C_{ij} and any ties as candidate column(s).
PR: Select ORs as candidate rows.
PE: Select the candidate row and column with the smallest non-negative C/R. Arbitrarily break ties.
If no non-negative C/R chose the C/R with the smallest absolute value.
If the PE is zero, select the next best C_{ij}.

Step 1.2. *BVS augmentation.*
(a) Perform GJP
(b) Replace the (?) row label with the variable name.
(c) Remove PC from the tableau.

Continue BVS iteration (loop back to Step 1.0).

2. FE phase

Step 2.0. *FE iteration termination test.*
IF R.H.S. is non-negative
THEN tableau is optimal. Interpret the result.
OTHERWISE continue FE iteration (Step 2.1).

Step 2.1. *FE selection of PE*
PR: row with the most negative R.H.S.
PC: column with a negative element in the PR.
Tie breaker: column with the smallest C_{ij}.
Further tie breaker arbitrary.

Step 2.2. *FE transportation*
(a) Save PC outside the tableau.
(b) Perform usual GJP.
(c) Exchange PC and PR labels.
(d) Replace the new PC with old PC saved in (a).

Continue FE iteration (loop back to Step 2.0).

End of algorithm

The first phase of the algorithm can be characterized as seeking a BVS while pushing toward an optimal vertex. The second phase, if necessary, pulls back to FE. In both phases GJP is used. However the selection criteria is unique to each phase. The first phase uses the customary simplex criteria, modified so as to restrict selection to OR. If necessary FE is relaxed. The second phase uses the criteria normally associated with dual simplex which ensures termination of the algorithm. For the detailed proofs see Ref. [4].

A NUMERICAL EXAMPLE OF THE NEW TP ALGORITHM

This section demonstrates the algorithm by walking through the problem described by Davis *et al.* [6]. This 4 × 2 problem was selected because it is small enough to conserve space yet rich enough to demonstrate the algorithm. The cost matrix is shown in Table 2.

Table 2. Cost matrix of a TP problem from Ref. [6]

	Dest 1	Dest 2	Dest 3	Dest 4	Avl.
	21	15	18	9	550
	10	14	16	23	650
Reqrmnt	200	250	400	350	

0. Preliminaries

Step 0.0. *TP cost matrix formulation.*
The matrix is given in Table 2, it is a balanced TP.

Step 0.1. *Row–column reduction (or column–row reduction).*
Row reduction reduces the first row by 9, the second row by 10.
Column reduction reduces the second column by 4 and third by 6.
The reduced matrix is shown in Table 3.

Table 3. Row–column reduced matrix

	Dest 1	Dest 2	Dest 3	Dest 4	Avl.
	12	2	3	0	550
	0	0	0	13	650
Reqrmnt	200	250	400	350	

Step 0.2. *Eliminate redundant constraint.*
The second row has the most zeros.

Step 0.3. *Formulate the simplex tableau.*
Table 4 shows the tentative initial tableau with the second row as the redundant constraint.

Table 4. Tentative simplex tableau

	Var	X_{11}	X_{12}	X_{13}	X_{14}	X_{21}	X_{22}	X_{23}	X_{24}	R.H.S.
Orgn 1	?	1	1	1	1					550
Orgn 2	?					1	1	1	1	650
Dest 1	?	1				1				200
Dest 2	?		1				1			250
Dest 3	?			1				1		400
Dest 4	?				1				1	350
	Cost	12	2	3	0	0	0	0	13	

Step 0.4. *Identify BVs.*
The initial tableau in Table 5 shows three variables in the BVS and two open rows.

Table 5. Initial tableau

	Var	X_{11}	X_{12}	X_{13}	X_{14}	X_{21}	X_{22}	X_{23}	X_{24}	R.H.S.
Orgn 1	?	1	1	1	1					550
Dest 1	X_{21}	1				1				200
Dest 2	X_{22}		1				1			250
Dest 3	X_{23}			1				1		400
Dest 4	?				1				1	350
	Cost	12	2	3	0	0	0	0	13	

Step 0.5. *Delete BV columns.*
See Table 6.

Table 6. Reduced initial tableau

	Var	X_{11}	X_{12}	X_{13}	X_{14}	X_{24}	R.H.S.
Orgn 1	?	1	1	1	(1)		550
Dest 1	X_{21}	1					200
Dest 2	X_{22}		1				250
Dest 3	X_{23}			1			400
Dest 4	?				[1]	1	350
	Cost	12	2	3	0	13	

1. BV phase

Step 1.0. *BVS iteration termination test.*
There are (?), i.e. ORs.
THEREFORE continue with BVS iteration in Step 1.1.

Step 1.1. *BV selection of PE*
PC: Smallest C_{ij} is zero.
PR: The first and fifth rows are ORs.
PE: Non-zero candidate PEs enclosed by [ ] and ( ) in Table 7.
Term in [ ] has smallest non-negative C/R.

Step 1.2. *BVS augmentation*
(a) Perform GJP
(b) Replace the (?) row label with the variable X_{14}.
(c) Remove PC from the tableau.
The results are shown in Table 7.

Table 7. Reduced second tableau

	Var	X_{11}	X_{12}	X_{13}	X_{24}	R.H.S.
Orgn 1	?	1	[1]	1	-1	200
Dest 1	X_{21}	1				200
Dest 2	X_{22}		1			250
Dest 3	X_{23}			1		400
Dest 4	X_{14}				1	350
	Cost	12	2	3	13	

End? of BVS iteration (loop back to Step 1.0).

Step 1.0. *BVS iteration termination test.*
(?) Label exists (there are OR).
THEREFORE continue the BV iteration in Step 1.1.

Step 1.1. *BV selection of PE*
PC: Smallest C_{ij} is 2 belonging to X_{12}.
PR: Only OR is the first.
(Resulting PE is non-zero.)
PE: Enclosed by [ ] in Table 7.

Step 1.2. *BVS Augmentation*
(a) Perform GJP
(b) Replace the (?) row label with the variable name.
(c) Remove PC from the tableau.
The results are shown in Table 8.

Table 8. Basic tableau (non-feasible?)

	Var	X_{11}	X_{13}	X_{24}	R.H.S.
Orgn 1	X_{12}	1	1	-1	200
Dest 1	X_{21}	1			200
Dest 2	X_{22}	-1	-1	1	50
Dest 3	X_{23}		1		400
Dest 4	X_{14}			1	350
	Cost	10	1	15	

Continue BVS iteration (loop back to Step 1.0).

Step 1.0. *BVS iteration termination test.*
NO (?) label exists (there NO ORs).
BVS is complete, start FE iteration, Step 2.0.

2. FE phase

Step 2.0. *FE iteration termination test.*
R.H.S. is non-negative.
THEREFORE terminate FE iteration.
TABLEAU IS NOW OPTIMUM.

The results are:
$X_{11} = 0$, $X_{12} = 200$, $X_{13} = 0$, $X_{14} = 350$,
$X_{21} = 200$, $X_{22} = 50$, $X_{23} = 400$, $X_{24} = 0$.
The optimum cost is $15,250.

PARAMETRIC ANALYSIS OF TP

SA begins as soon as we get the final tableau. All information we need to perform PA are contained in the initial and final tableaux:

Initial basic tableau			Final tableau			
Basis		R.H.S.	BV	Basis		R.H.S.
B	N	b	—	I	$B^{-1} \cdot N$	$B^{-1}b$

As shown above the initial basic tableau may be partitioned into B, the basic mixing coefficients, N, the non-basic mixing coefficients and b, the initial R.H.S. Where the basis is related to the basic variables in the final tableau. In the final tableau the basic partition becomes I, the identity matrix, and the remaining partitions are multiplied by B^{-1} the inverse of B.

(a) Perturbation of the cost vector

The cost vector, **C**, is composed of elements, C_{ij} signifying the cost of shipping one unit from the ith source to the jth sink. Similarly, we define a perturbation vector **P** consisting of elements v_{ij} specifying a perturbation in each cost element. Introducing a scalar parameter, r, we extend the original problem to a TP with the following objective function:

$$\min \sum \sum (c_{ij} + r \cdot v_{ij})x_{ij}, \qquad \text{over all } i\text{s and } j\text{s}.$$

The matrix form of the extended problem is:

$$\min(\mathbf{C} + r \cdot \mathbf{P})X \qquad \text{s.t. } AX = b,\ X \geqslant 0.$$

Note that **P** establishes the direction of the perturbation and r the magnitude. At $r = 0$ this reduces to the original problem.

We would like to find out how far we can move in the direction of **P** while still maintaining optimality of the current solution. This is equivalent to determining how large r may increase while maintaining FE.

Decomposing the final tableau into $[I, Q]$, where Q contains the columns of non-basic variables, and updating the tableau, we have

$$\text{new } C_{ij} = \text{old } C_{ij} + r(\mathbf{P}_n - \mathbf{P}_b \cdot Q).$$

where $\mathbf{P}_n$ and $\mathbf{P}_b$ are the sub-vectors of **P** corresponding to non-basic and basic variables, respectively. Note that C_{ij} are given in the last row of the final tableau.

Define

$$S = \{ij\colon \mathbf{P}_n - \mathbf{P}_b \cdot Q < 0\}.$$

If $S = 0$, then the current solution is optimal for all values of $r \geqslant 0$. Otherwise, calculate r' as follows:

$$r' = \min[-\text{old } C_{ij}/(\mathbf{P}_n - \mathbf{P}_b \cdot Q)_{ij} < 0].$$

For $r \in [0, r']$ the current solution is optimal.

(b) Perturbation of the R.H.S.

This section treats changes in supply quantities and/or the demand quantities. They correspond to changes in the R.H.S. Assume the original problem is

$$\min \sum \sum (C_{ij} X_{ij})$$

$$\text{s.t.} \quad AX = b, \quad X \geqslant 0.$$

We treat changes depending on a scalar parameter r along a vector $\mathbf{f}$ by introducing the perturbation $r\mathbf{f}$:

$$\text{s.t.} \quad AX = b + r\mathbf{f},$$

where $\mathbf{f} = (f_1, f_2, \ldots, f_m, \ldots, f_{n+m})$, is a vector with m perturbations in supply followed by n perturbations in demand. Balance of supply and demand in TP requires

$$\sum_{i=1}^{m} f_i = \sum_{i=1+m}^{n+m} f_i.$$

Finally, each element of the perturbed supply and demand must be non-negative.

Our goal is to find the admissible upper and lower endpoints for parameter r under these conditions.

In the original problem if B is the optimal basis found in the initial tableau and the initial R.H.S. $= b$, then the final R.H.S. $= B^{-1} \cdot b$, where B^{-1} is the inverse of B. In the perturbed problem we replace b with $b + r\mathbf{f}$ to obtain:

$$\text{final perturbed R.H.S.} = B^{-1} \cdot (b + r\mathbf{f}).$$

As long as $B^{-1} \cdot (b + r\mathbf{f})$ is non-negative, the current basis remains optimal.

Define

$$b' = B^{-1} \cdot b \quad \text{and} \quad b'' = B^{-1} \cdot \mathbf{f}.$$

The value of r at which the optimal basis changes can be determined as follows: let

$$S = \{i \colon b'_i + rb''_i < 0\}.$$

If $S = 0$, then the current basis is optimal for all values of $r \geqslant 0$. Otherwise let

$$r' = \min\{-b_i/b''_i\}, \quad \text{over all } i,\ i \in S.$$

For $r \in [0, r']$ the optimal solution is

$$X = B^{-1} \cdot (b + r\mathbf{f}).$$

A NUMERICAL EXAMPLE

This section demonstrates the algorithm by walking through the problem of Davis *et al.* [6] solved earlier.

Table 9. Cost matrix of a TP from Ref. [6]

	Dest 1	Dest 2	Dest 3	Dest 4	Avl.
	21	15	18	9	550
	10	14	16	23	650
Reqrmnt	200	250	400	350	

Table 10. Initial tableau

	Var	X_{11}	X_{12}	X_{13}	X_{14}	X_{21}	X_{22}	X_{23}	X_{24}	R.H.S.
Orgn 1	?	1	1	1	1					550
Dest 1	X_{21}	1				1				200
Dest 2	X_{22}		1				1			250
Dest 3	X_{23}			1				1		400
Dest 4	?				1				1	350
	Cost	12	2	3	0	0	0	0	13	

Table 11. Final tableau

Var	X_{11}	X_{13}	X_{24}	R.H.S.
X_{12}	1	1	-1	200
X_{21}	1			200
X_{22}	-1	-1	1	50
X_{23}		1		400
X_{14}			1	350
Cost	10	1	15	

(a) Perturbation of the cost vector

The final tableau rearranged in order of x_{ij}, $i = 1, 2, \ldots, n$; is:

BV	x_{11}	x_{13}	x_{24}	R.H.S.
x_{12}	1	1	-1	200
x_{14}			1	350
x_{21}	1		0	200
x_{22}	-1	-1	1	50
x_{23}		1		400
c_{ij}	10	1	15	

Let us perturb the cost vector along $\mathbf{P} = (-2, 1, -3, -1, -1, 1, 1, 1)$,

$$\mathbf{C}' = \mathbf{C} + r\mathbf{P} = (21 - 2r, 15 + r, 18 - 3r, 9 - r, 10 - r, 14 + r, 16 + r, 23 + r).$$

Now by undating the final tableau new $C_{ij} =$ old $C_{ij} - r(\mathbf{P}_\mathrm{n} - \mathbf{P}_\mathrm{b} \cdot B^{-1} \cdot N)$, where

$$\mathbf{P}_\mathrm{n} - \mathbf{P}_\mathrm{b} \cdot B^{-1} \cdot N = (-2, -3, 1) - (1, -1, -1, 1, 1) \begin{bmatrix} 1 & 1 & -1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ -1 & -1 & 1 \\ 0 & 1 & 0 \end{bmatrix} = (-1, -4, 2).$$

Define

$$S = \{ij: \mathbf{P}_\mathrm{n} - \mathbf{P}_\mathrm{b} \cdot B^{-1} \cdot N < 0\} = \{11, 13\}.$$

Therefore

$$r' = \min\{-10/-1, -1/-4\} = 1/4,$$

which implies that the current solution is optimal for $r \in [0, 1/4]$.

(b) Perturbation of the R.H.S.

We modify this example by considering an increase in demand in Dest. D1. Some increase might be caused by the customer switching from Dest. D3 to D1. The final tableau is shown below:

BV	x_{11}	x_{13}	x_{24}	R.H.S.
x_{12}	1	1	-1	200
x_{21}	1	0	0	200
x_{22}	-1	-1	1	50
x_{23}	0	1	0	400
x_{14}	0	0	1	350
	10	1	15	

Here the R.H.S. is being perturbed along $(0, 0, 1, 0, -1, 0)$ with $r \geqslant 0$. The coefficient vector of the parameter r in this example is $\mathbf{f} = (0, 1, 0, -1, 0)$. From the initial tableau,

$$B = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad B^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

To find the range over which the above basis is optimal, first compute

$$b'' = B^{-1} \cdot \mathbf{f} = (0, 1, 0, -1, 0)^T, \qquad S = \{4\}.$$

Therefore

$$r' = \min\{-b_i'/b_i''\} = 400,$$

which implies that the basis remains optimal over the integer interval [0, 400].

CONCLUDING REMARKS

We have introduced an algorithm which has the following special features:

—No artificial variables (as in simplex) or extra slack/surplus variables (as in dual simplex) are needed.
—Appears faster than other LP methods, see Ref. [5].
—Simplicity: the only operation is Gauss–Jordan pivoting.
—An adaptation of all LP post-optimality analyses for TP is made available.

Acknowledgement—This work received support from the University of Baltimore Educational Foundation.

REFERENCES

1. T. Saaty, Coefficient perturbation of a constrained extremum. *Ops Res.* **7**, 294–302. (1959).
2. A. Deif, *Sensitivity Analysis in Linear Systems.* Springer, New York (1986).
3. W. Shih, Modified stepping-stone method as a teaching aid for capacitated transportation problems. *Decis. Sci.* **18**, 662–676 (1987).
4. H. Arsham and A. Kahn, The open basic variable set algorithm for the transportation problem. Manuscript, Univ. of Baltimore, Md (1988).
5. H. Arsham, A tabular simplex type algorithm as a teaching aid for general LP models. *Mathl Comput. Modelling* **12**, 1051–1056 (1989).
6. K. R. Davis, P. G. McKeown and T. R. Rakes, *Management Science—an Introduction*, pp. 311–314. Kent Publishing, Boston, Mass. (1986).