MANAGEMENT SCIENCE
Vol. 14, No. 3, November, 1967
Printed in U.S.A.

A PRIMAL METHOD FOR MINIMAL COST FLOWS WITH APPLICATIONS TO THE ASSIGNMENT AND TRANSPORTATION PROBLEMS*†

MORTON KLEIN‡

Columbia University

A simple procedure is given for solving minimal cost flow problems in which feasible flows are maintained throughout. It specializes to give primal algorithms for the assignment and transportation problems. Convex cost problems can also be handled.

1. Introduction

Suppose we have a network with vertices $V = \{1, \cdots, n\}$, directed edges $E = \{(i, j) \varepsilon V \times V\}$, a non-negative integral valued function k giving the maximum allowable flow k_{ij} over every edge, and a non-negative cost function a giving the cost a_{ij} associated with a unit of flow over any edge. A flow X of value v is an integral valued function defined on E satisfying

$$0 \leqq x_{ij} \leqq k_{ij}, \qquad (i, j) \varepsilon E. \tag{1.1}$$

$$\begin{aligned} \textstyle\sum_{j=1}^{n} (x_{ij} - x_{ji}) &= v, && i = 1, \\ &= 0, && i = 2, \cdots, n-1, \\ &= -v, && i = n. \end{aligned} \tag{1.2}$$

Vertices 1 and n are, respectively, called the flow *source* and *sink*.

The minimal cost flow problem is to find, among all flows X of value v, one which minimizes

$$Q(X) = \textstyle\sum_{(i,j)\varepsilon E} x_{ij} a_{ij}. \tag{1.3}$$

The main computational procedures developed to date for solving this problem are the primal-dual type algorithms of Ford and Fulkerson [7], Busacker and Gowen (described in [3]), and Jewell [12]. These are dual methods in which feasible flows (those satisfying (1.1) and (1.2)) become available when the computations terminate. A similar method for convex cost flow problems has been given by Hu [9]. Fulkerson's "Out-of-Kilter" algorithm (described in [7]) is essentially a primal method in that it can be started with a feasible flow or one becomes available at an early stage. Another primal approach for problems with convex costs has also been suggested recently by Menon [14].

The assignment and transportation problems can be thought of as special minimal cost flow problems in the sense that their networks have a particular

* Received September 1966, revised April 1967, and accepted May 1967.

† This research is supported by the Army, Navy, Air Force, and N.A.S. under a contract administered by the Office of Naval Research; Contract Nonr 266(55). Reproduction in whole or in part is permitted for any purpose of the United States Government.

‡ The author is indebted to T. C. Hu and M. Florian for helpful suggestions.

bipartite form. Kuhn's "Hungarian Method" [13], a primal-dual type algorithm, and two variants: ([15] and one given in [7]) provide the most popular methods for solving these problems. Other methods are described by Flood [5] and Hoffman and Markowitz [8]. Primal methods are also available: Dantzig's adaptation of the simplex method (described in [4]), the methods given by Beale [2], Flood [6], and, most recently, by Balinski and Gomory [1].

The purpose of this paper is to suggest that one more primal method can be added to the above arsenal for both minimal cost flow and assignment-transportation problems. With slight modification it can also be used for problems involving convex costs. It shares, with other primal methods, the property that it can be started with a "good" solution and a better one is always available in case early termination of computations is required.

2. A Minimal Cost Flow Algorithm

In this section we give a procedure for solving the minimal cost flow problem. We assume a familiarity with the maximal flow and shortest route problems together with the Ford-Fulkerson [7] methods for solving them.

The method suggested here is to first find a flow satisfying (1.1) and (1.2) by, say, the Ford and Fulkerson maximum flow routine [7] (pp. 17–18). Given such a flow X, we then construct an associated network $G(X)$ which has the same vertices as the original network and directed edges, as follows:

$$\text{(2.1)} \qquad \begin{aligned} E(X)\colon (i, j), &\quad \text{if} \quad x_{ij} < k_{ij} \quad \text{and} \quad x_{ji} = 0, \\ (j, i), &\quad \text{if} \quad x_{ij} > 0, \end{aligned}$$

with revised capacities:

$$\text{(2.2)} \qquad \begin{aligned} k'\colon k'_{ij} &= k_{ij} - x_{ij}, \quad \text{if} \quad x_{ij} < k_{ij} \quad \text{and} \quad x_{ji} = 0, \\ k'_{ji} &= x_{ij}, \qquad\quad \text{if} \quad x_{ij} > 0, \end{aligned}$$

and with revised edge costs:

$$\text{(2.3)} \qquad \begin{aligned} a'\colon a'_{ij} &= a_{ij}, \quad \text{if} \quad x_{ij} < k_{ij} \quad \text{and} \quad x_{ji} = 0, \\ a'_{ji} &= -a_{ij}, \quad \text{if} \quad x_{ij} > 0. \end{aligned}$$

These revised edge costs are simply those associated with increasing or cancelling the flow by one unit on these edges: The revised capacities indicate the extent to which this can be done.

Now, we use a result proved in Busacker and Saaty ([3], pp. 256–257).

Theorem. X is a minimal cost flow *if and only if* there is no directed cycle C in $G(X)$ such that the sum of the costs around C's edges are negative. (A directed cycle is a sequence of distinct directed edges of the form $\{(i_0, i_1)\ (i_1, i_2) \cdots (i_p, i_q)\ (i_q, i_0)\}$ involving distinct vertices.)

An immediate consequence of this theorem is that a test of the optimality of the flow X is at hand if $G(X)$ can be checked for the existence of a negative cost directed cycle. Further, if such a cycle is found an improved flow is obtained by sending a positive unit flow around this cycle. Such a flow alteration obviously leads to a lower total cost and also leaves the flow value v unchanged.

Fortunately, there are known methods for locating negative cost directed cycles. The Fulkerson "Out-of-Kilter" procedure [7] (pp. 162–169) is one. Various shortest route algorithms may also be used; of these, we shall use a "Matrix Multiplication" procedure due to Murchland [16] (as described by Hu [10]) which is designed to find the shortest routes between every pair of vertices in a network. Although this method (in common with other shortest route algorithms) requires the assumption of non-negative directed cycle costs; it also may be used to locate such a cycle.

We need only put the above observations together to obtain an

Algorithm for Minimal Cost Flows

1. Use the Ford-Fulkerson maximal flow routine (or any other) to find a flow X of value v.

2. Form the associated network $G(X)$ according to (2.1), (2.2), and (2.3).

3. Test for the existence of a negative directed cycle C in $G(X)$ using the matrix multiplication method described in [10] as follows:

a) Form an $n \times n$ matrix $D(X)$ in which each row and column is associated with a vertex of $G(X)$, with entries $d_{ij} = a'_{ij}$ as given by (2.3), and with all other entries equal to infinity.

b) *Pivot Steps*: For each value $k = 1, 2, \cdots, n$, in turn, perform the *replacement operation*,

$$d_{ij} = \min \; (d_{ij}, \; d_{ik} + d_{kj}) \tag{2.4}$$

for all $i, j \neq k$.

c) Repeat b) until some entry d_{rr} becomes negative (indicating the existence of a negative cost cycle involving vertex r) or until the replacement operation has been completed for $k = n$ with all $d_{ii} \geqq 0$. If the latter occurs, it indicates that $G(X)$ does not contain a negative cycle and that the flow being tested is optimal.

d) If a negative cost directed cycle C is found to involve vertex r, we shall call r the *initial point*. We now suggest that it is computationally convenient to use the Ford-Fulkerson shortest route index reduction method to trace the edges of C by using it to find the shortest route from the initial point to every other vertex in $G(X)$. (Step 4, below) (It is possible to keep track of the edges which give rise to the negative cycle in the matrix multiplication method as indicated by Hu in [11]. However, "keeping track" is cumbersome and requires the storage of much unneeded data. The index reduction routine appears to be more economical for this purpose.)

4. The negative cycle tracing routine is as follows:

a) Let r be the initial point. Assign it the label $(-, \pi(r) = 0)$. Assign all other vertices labels of the form $(-, \pi(i) = \infty)$.

b) Search for an edge (j, k) such that

$$\pi(j) + d_{jk} \leqq \pi(k)$$

and then either change the label on vertex k to $(j, \pi(j) + d_{jk})$, if the inequality is strict (case a) or, change the label to $(\{k, j\}, \pi(k))$ if equality holds (case b). Continue until the initial point receives a label of the form $(i^*, \pi(r) < 0)$.

c) The negative cycle C can now be found by tracing backwards from r to $i^*, \cdots$, to r. If vertices are encountered with more than one label (case b in (b)), then sub-cycles of cost zero will be encountered; these are discarded. For example, if tracing yields a sequence of the form (r, i_1) (i_1, i_2) (i_2, i_1) where vertex i_1 had the double label, say, $(\{i_2, i_3\}, d)$, the sub-cycle (i_1, i_2) (i_2, i_1) has associated cost zero. The tracing process is then continued from vertex i_3. Since there may be more than one negative cycle, one can, in tracing, encounter a negative cycle which does not involve the initial point. In this case, this cycle can be used as a basis for the improved flow.

5. Given C, an improved flow X' is given by

$$\begin{aligned} x'_{ij} &= x_{ij}, && \text{if} \quad (i, i) \not\varepsilon C, \\ &= x_{ij} - \delta, && \text{if} \quad (j, j) \varepsilon C \quad \text{and} \quad a'_{ji} \leqq 0, \\ &= x_{ij} + \delta, && \text{if} \quad (i, j) \varepsilon \text{C} \quad \text{and} \quad a'_{ij} \geqq 0, \end{aligned} \tag{2.5}$$

where $\delta = \min_{(i,j)\varepsilon C} \{k'_{ij}\}$, *i.e.*, δ is the largest amount of flow which can be sent around C.

Return to Step 2.

Example. Suppose a minimal cost flow of value 4 is to be imposed on the following network from vertex 1 to vertex 5. Costs and capacities are indicated according to the legend {cost, capacity} on each edge.

We start with an arbitrary flow X of value 4, indicated in Figure 2 (omitting Step 1 for a problem of this size).

Step 2. $G(X)$ is then as shown in Fig. 3, where negative cost edges are indicated by dashed lines.

Step 3. The matrix multiplication routine starts with $D(X)$ as given below.

	1	2	3	4	5
1	∞	4	1	∞	∞
2	∞	∞	∞	∞	1
3	−1	2	∞	3	∞
4	∞	∞	−3	∞	∞
5	∞	∞	∞	−2	∞

It ends (at pivot step $k = 4$) when $d_{55} = -2$,

	1	2	3	4	5
1	0	3	1	−4	4
2	∞	∞	∞	∞	1
3	−1	−2	0	3	3
4	−4	−1	−2	0	0
5	−6	−3	−4	−2	−2

Hence, vertex 5 is the initial point for the cycle tracing routine.

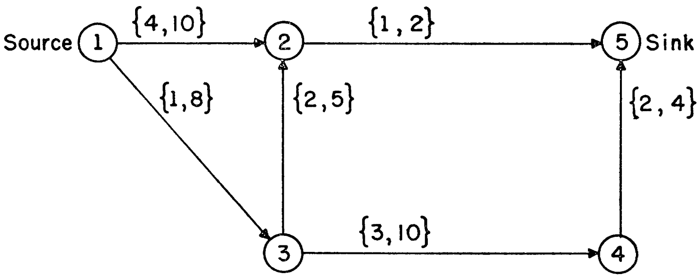

FIGURE 1

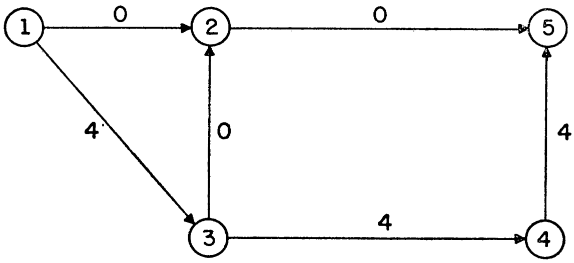

FIGURE 2

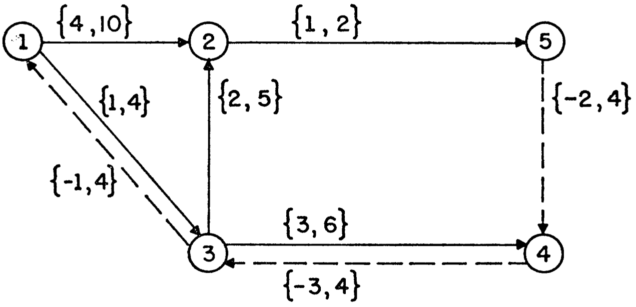

FIGURE 3

Step 4. After application of the cycle tracing routine $G(X)$ has vertex labels as indicated in Figure 4. C is found by tracing back from vertex 5 to 2, from 2 to 3, from 3 to 4, and from 4 to 5. That is, $C = \{(5, 4)\ (4, 3)\ (3, 2)\ (2, 5)\}$.

Step 5.

$$\begin{aligned}\delta &= \min\{k'_{54}, k'_{43}, k'_{32}, k'_{25}\}\\ &= \min\{4, \quad 4, \quad 5, \quad 2\} = 2\end{aligned}$$

and the new flow X' is

$$\begin{aligned}x'_{45} &= 4 - 2 = 2\\ x'_{34} &= 4 - 2 = 2\end{aligned}$$

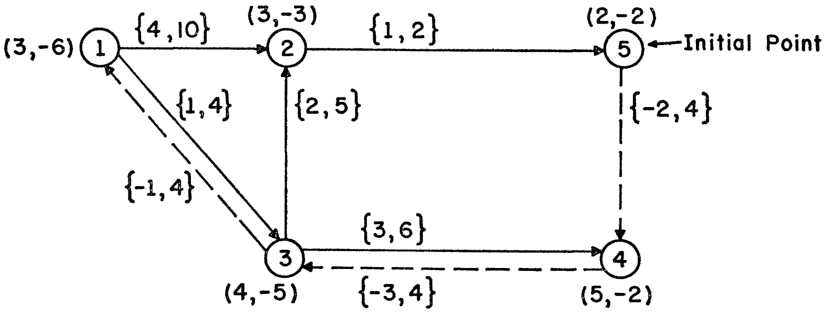

FIGURE 4

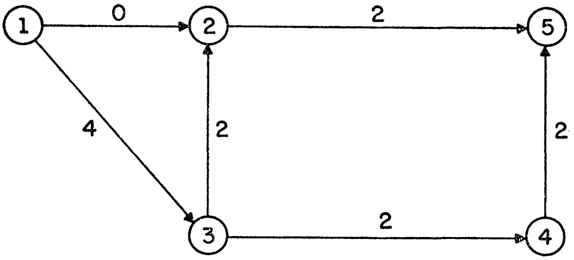

FIGURE 5

$$
\begin{aligned}
x'_{32} &= 0 + 2 = 2 \\
x'_{25} &= 0 + 2 = 2 \\
x'_{13} &= X_{13} = 4 \\
x'_{12} &= X_{12} = 0.
\end{aligned}
$$

All other flow values are unchanged. The new flow is shown in Figure 5. Now $G(X')$ is as shown in Figure 6.

The matrix multiplication routine yields

	1	2	3	4	5
1	0	3	1	4	6
2	−3	0	−2	1	3
3	−1	2	0	3	5
4	−4	1	−3	0	2
5	−6	3	−5	−2	0

Since none of the elements on the main diagonal is negative, the new flow X' is optimal.

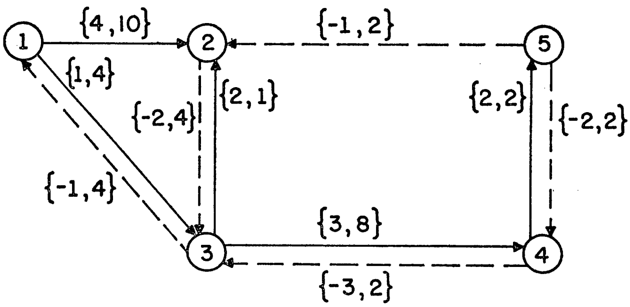

FIGURE 6

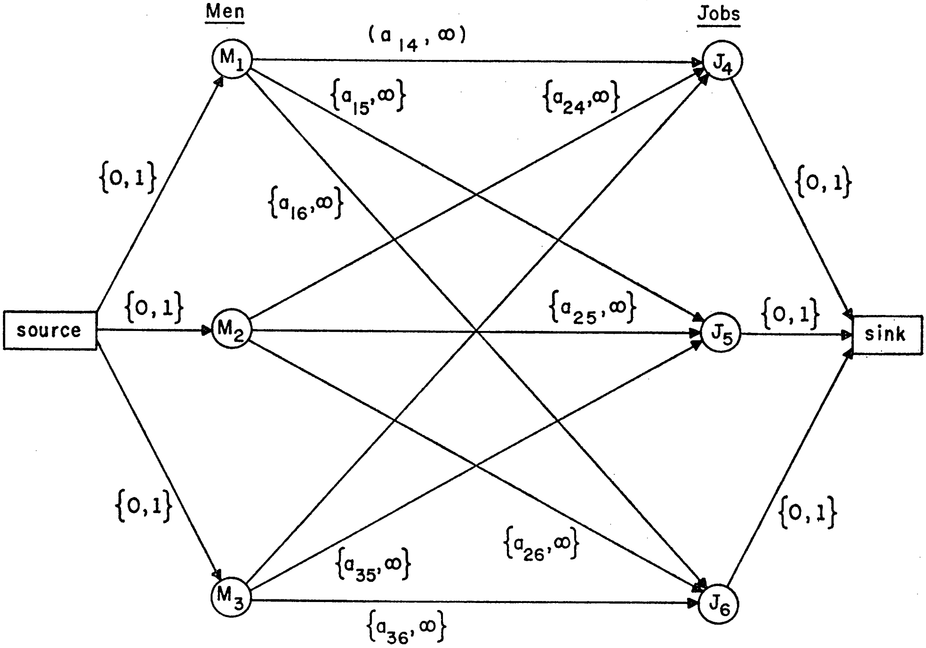

FIG. 7. Network representation of an assignment problem

3. Application to the Assignment Problem

In this section we show how the method described above specializes for solving the assignment problem.

The assignment problem is to fill n jobs by as many men at least total cost. If a_{ij} represents the cost of using man i in job j, then a mathematical statement of the problem is to find a permutation matrix $X = (x_{ij})$ of order n, which minimizes the total cost $Q(X) = \sum x_{ij}a_{ij}$, where $x_{ij} = 1$ implies that man i ($i = 1, \cdots, n$) is assigned to job j ($j = n + 1, \cdots, 2n$).

The equivalent minimum cost flow problem (following Ford and Fulkerson) is illustrated for the case $v = n = 3$ in Figure 7.

Algorithm for Assignment Problem

1. The algorithm can be started with any *feasible* (trial) solution. (This corresponds to finding a flow of value $v = n$.) It is frequently advantageous to try to choose a "good" solution using some heuristic rule, *e.g.* the "Least-Cost Rule" given by Dantzig in [4] (p. 309).

Let X be a trial solution and $A'(X)$ an associated matrix with elements

$$\begin{aligned} a'_{ij} &= -a_{ij}, \quad \text{if} \quad x_{ij} > 0, \\ &= a_{ij}, \quad \text{if} \quad x_{ij} = 0, \end{aligned} \tag{3.1}$$

where $i = 1, \cdots, n$ and $j = n + 1, \cdots, 2n$.

Further, let $D(X)$ be the $(2n) \times (2n)$ matrix with rows and columns each corresponding to both the men and the jobs of the problem and with elements.

$$\begin{aligned} d_{ij} &= \infty, && i, j = 1, \cdots, n \quad \text{or} \quad i, j = n + 1, \cdots, 2n, \\ &= a_{ij}, && i = 1, \cdots, n, \quad j = n + 1, \cdots, 2n, \\ &= -a_{ji}, && x_{ji} > 0 \quad i = n + 1, \cdots, 2n; j = 1, \cdots, n \\ &= \infty, && x_{ji} = 0 \quad \text{``} \qquad \text{``} \quad \text{``} \qquad \text{``}. \end{aligned} \tag{3.2}$$

2. Test for the existence of a negative cycle as in step 3 of the algorithm for minimal cost flows using D as described in (3.2) above as the cost matrix.

If all $d_{ii} \geqq 0$, the algorithm terminates and the trial solution is optimal. Otherwise, let d_{rr} be the first negative element encountered and use r as the initial point for the cycle tracing routine next.

3. Because of the special bipartite form of the network interpretation of the problem (the source and sink can be dropped since their associated edges have zero costs) it is convenient to do the cycle tracing routine using the matrix $A'(X)$. This is done by means of a series of alternate row and column labelings on A' with each successive row (column) label giving the cost associated with following a traceable route from the initial point to the column (row) from which the current cost is measured.

Let (M_i, d_i) be a label assigned to column j of A'. Then the implication is that there is a directed route from the initial point to M_i and a directed edge to (job) column j such that the total cost is d_i. The row labels (J_j, d_j) are defined similarly.

A convenient notational device is to write $a_{\overline{ij}}$ if $a'_{ji} \leqq 0$ and $x_{ij} > 0$.

Let J_r be the initial point.[1] Label column r $(-, 0)$.

1. Label each row $(J_r, | a'_{ir} |)$.

2. Label each column $j \neq r$, $(M_i, d_i + a_{\overline{ij}})$.

[1] If a row vertex M_r is the indicated initial point, the labeling routine can be followed as written by interchanging the rows and columns of A'.

3. Label each row $(\{J_k\}, \min_j (d_j + | a'_{ij} |))$ where each J_k represents a column at which the minimum is attained.

4. Label each column $(M_i , d_i + a^{-}_{ij})$.

5. Continue steps 3 and 4 until the initial point's label becomes negative.

The negative cost cycle, C, can be identified by tracing back from the initial point according to the succession of adjacent row and column labels until the initial point is encountered for the second time or some other vertex is encountered twice indicating a negative cycle not involving the initial point. Then, an improved trial solution X' is given by

$$(3.3) \qquad \begin{aligned} x'_{ij} &= x_{ij}, && \text{if} \quad (M_i , J_j) \not\varepsilon\, C, \\ &= 0, && \text{if} \quad | M_i , J_j | \;\varepsilon\; C \quad \text{and} \quad x_{ij} = 1, \\ &= 1, && \text{if} \quad | M_i , J_j | \;\varepsilon\; C \quad \text{and} \quad x_{ij} = 0, \end{aligned}$$

where the notation $| M_i , J_j |$ indicates that either (M_i, J_j) or (J_j , M_i) is an element of C.

Return to step 2.

Example. Suppose (using a least-cost rule) the first trial solution is $x_{17} = x_{25} = x_{38} = x_{46} = 1$, then $A'(X)$ and $D(X)$ are given below.

$$A'(X) = \begin{array}{c|c|c|c|c|} & J_5 & J_6 & J_7 & J_8 \\ \hline M_1 & 2 & 3 & -1 & 1 \\ \hline M_2 & -5 & 8 & 3 & 2 \\ \hline M_3 & 4 & 9 & 5 & -1 \\ \hline M_4 & 8 & -7 & 8 & 4 \\ \hline \end{array}$$

$$D(X) = \begin{array}{c|c|c|c|c||c|c|c|c|} & M_1 & M_2 & M_3 & M_4 & J_5 & J_6 & J_7 & J_8 \\ \hline M_1 & & & & & 2 & 3 & 1 & 1 \\ \hline M_2 & & & & & 5 & 8 & 3 & 2 \\ \hline M_3 & & & & & 4 & 9 & 5 & 1 \\ \hline M_4 & & & & & 8 & 7 & 8 & 4 \\ \hline\hline J_5 & & -5 & & & & & & \\ \hline J_6 & & & & -7 & & & & \\ \hline J_7 & -1 & & & & & & & \\ \hline J_8 & & & -1 & & & & & \\ \hline \end{array}$$

(Note: All unmarked entries have values $d_{ij} = \infty$.)

The matrix multiplication routine (step 2), at pivot step $k = 5$, gives

	M_1	M_2	M_3	M_4	J_5	J_6	J_7	J_8
M_1		-3			2	3	0	-1
M_2		0			5	8	3	2
M_3		-1			4	7	2	1
M_4		3			8	7	6	4
J_5		-5			0	3	-2	-3
J_6		-4		-7	1	0	-1	-2
J_7	-1	-4			1	2	−①	-2
J_8		-1			4	7	2	1

Since $d_{77} < 0$, J_7 is the initial point for the cycle tracing routine. The appropriate labels on $A'(X)$ are shown below.

Initial point

	J_5	J_6	J_7	J_8		
M_1	2*	3	-1*	1	$J_7, 1$	$J_5, 0$
M_2	-5*	8	3*	2	$J_7, 3$	$J_5, J_7, 3$
M_3	4	9	5	-1	$J_7, 5$	$J_5, 2$
M_4	8	-7	8	4	$J_7, 8$	$J_5, 6$
	$M_2, -2$	$M_4, 1$	$-, 0$ $M_1, -1$	$M_3, 4$		

Stop

The elements of C are (J_7, M_1), (M_1, J_5), (J_5, M_2), (M_2, J_7); these are indicated by asterisks (*) above. The new trial solution is, from (3.2) $x'_{15} = x'_{27} = x'_{46} = x'_{38} = 1$. Its total cost is 13.

The example is continued. $D(X')$ is

	M_1	M_2	M_3	M_4	J_5	J_6	J_7	J_8
M_1					2	3	1	1
M_2					5	8	3	2
M_3					4	9	5	1
M_4					8	7	8	4
J_5	-2							
J_6				-7				
J_7		-3						
J_8			-1					

and the matrix multiplication routine yields

	M_1	M_2	M_3	M_4	J_5	J_6	J_7	J_8
M_1	0	−2	−1	−4	3	3	1	0
M_2	3	0	1	−1	5	6	4	2
M_3	2	0	0	−2	4	5	3	2
M_4	5	3	3	0	7	8	6	4
J_5	−2	−4	−3	−6	0	1	−1	−2
J_6	−2	−4	−4	−7	0	0	−1	−3
J_7	0	−3	−2	−4	2	3	0	−1
J_8	1	−1	−1	−3	3	4	2	0

Since all elements on the main diagonal are non-negative, X' is an optimal solution.

4. An Algorithm for the Transportation Problem

The method given here for the transportation problem is, with slight alteration, the same as that suggested for the assignment problem. The reason for this computational similarity is, of course, the well-known relationship between the two problems: each is a special case of the other.

We suppose that the transportation problem involves shipments of a single commodity from n plants $P_1, \cdots, P_n$, with capacities $c_1, \cdots, c_n$ to m warehouses $W_{n+1}, \cdots, W_{n+m}$ with requirements $r_{n+1}, \cdots, r_{n+m}$. If a_{ij} represents the cost of shipping a unit from P_i to W_j, and x_{ij} the quantity shipped from P_i to W_j, the problem is to minimize the total cost

$$(4.1) \qquad Q(X) = \sum_{j=n+1}^{n+m} \sum_{i=1}^{n} x_{ij} a_{ij}$$

constrained by

$$(4.2) \qquad \begin{aligned} \sum_{i=1}^{n} x_{ij} &= r_j, & j &= n+1, \cdots, n+m; \\ \sum_{j=n+1}^{n+m} x_{ij} &= c_i, & i &= 1, \cdots, n; \\ x_{ij} &= 0, 1, \cdots, \end{aligned}$$

where we assume that the c_i's and r_j's are positive integers, and $\sum c_i = \sum r_j$.

We omit the network formulation of the problem and go directly to the computational procedure. The terminology is the same as that used for the assignment problem except that we speak of plants and warehouses instead of men and jobs.

1. *Trial Solutions:* Although the computational procedure can be written so that one can work with any trial solution satisfying (4.2), it is convenient to restrict ourselves to the "basic feasible solutions", possibly degenerate, which

are used in Dantzig's adaptation of the simplex method for transportation problems. These solutions contain, at most, $m + n - 1$ positive entries. They also have the property that no cycles can be formed by their positive entries. These two properties may also be justified by simple combinatorial arguments [6].

Suppose X is a trial solution (i.e., it satisfies (4.2) and the above). We again define the associated matrix $A'(X)$ by

$$\text{(4.3)} \qquad \begin{aligned} a'_{ij} &= -a_{ij}, \quad \text{if} \quad x_{ij} > 0, \\ &= a_{ij}, \quad \text{if} \quad x_{ij} = 0, \end{aligned}$$

and write a_{ij}^{-} if $a'_{ij} \leqq 0$ and $x_{ij} > 0$.

Let $D(X)$ be the $(m + n) \times (m + n)$ matrix with elements

$$\text{(4.4)} \qquad \begin{aligned} d_{ij} &= \infty, & & i, j = 1, \cdots, n \quad \text{or} \quad i, j = n + 1, \cdots, n + m \\ &= a_{ij}, & & i = 1, \cdots, n; \quad j = n + 1, \cdots, n + m \\ &\left.\begin{aligned} &= -a_{ji}, \quad x_{ji} > 0 \\ &= \infty, \quad\ \ x_{ji} = 0 \end{aligned}\right\} & & \begin{aligned} i &= n + 1, \cdots, n + m; \\ j &= 1, \cdots, n \end{aligned} \end{aligned}$$

2. *Matrix Multiplication Routine:* Terminating with an initial point corresponding to first negative main diagonal element, or, if no such element is negative, with the conclusion that the trial solution is optimal.

3. *Cycle Tracing Routine:* Using the matrix $A'(X)$; suppose W_r is the initial point.[2] Label its column $(-, 0)$.

1. Label each row $(W_r, |a'_{ir}|)$
2. Label each column $j \neq r$ $(\{P_k\}, \min_i\{d_i + a_{ij}^{-}\})$ where each P_k is a row at which the minimum is attained.
3. Label each row $(\{W_k\}, \min_j\{d_j + |a'_{ij}|\})$
4. Label each column $(\{P_k\}, \min_i\{d_i + a_{ij}^{-}\})$
5. Continue steps 3 and 4 until the initial point's label becomes negative.

The negative cost cycle is then obtained by tracing backwards from the initial point, discarding any zero cost sub-cycles which may be encountered. As indicated earlier, a negative cycle, not involving the initial point, may be discovered before a return to the initial point. This may be used to find an improved solution.

Let $X(C)$ be the entries of X whose subscripts correspond to those of C, i.e., if $|P_i, W_j| \varepsilon C$ then $x_{ij} \varepsilon X(C)$, and index these entries (with superscripts) as follows: assign the index 1 to any entry whose value is zero. (There is at least one such entry by virtue of our use of "basic feasible" trial solutions.) Now, following the cycle (in either direction) continue with successive positive integers: $2, 3, \cdots, k$ until all elements in $X(C)$ have been indexed. Note that the index k is an even number.

[2] If a row vertex P_r is the initial point, the labeling routine can be followed with the rows and columns of A' interchanged.

A new improved trial solution X' is defined by

$$(4.5)\qquad \begin{aligned} x'_{ij} &= x_{ij}\,, && \text{if } x_{ij} \not\in X(C), \\ &= x_{ij} + \delta, && \text{if } x_{ij}^{(t)} \varepsilon\, X(C) \text{ and } t \text{ is odd}, \\ &= x_{ij} - \delta, && \text{if } x_{ij}^{(t)} \varepsilon\, X(C) \text{ and } t \text{ is even}, \end{aligned}$$

where δ is the value of the smallest even indexed element of $X(C)$. It is easy to see that X' is also a "basic feasible" solution.

Return to step 2.

Example. Consider the following transportation array [1].

	W_4	W_5	W_6	W_7	W_8	
P_1	3	6	3	1	1	4
P_2	2	4	3	2	7	5
P_3	1	1	2	1	2	6
	2	2	3	4	4	

and the trial solution given below:

	W_4	W_5	W_6	W_7	W_8
P_1				2	2
P_2			3		2
P_3	2	2		2	

Here we have attempted to find a "good" trial solution by trying to ship as much as possible along the cheapest routes.

Steps 1–2. $D(X)$ is given below.

	P_1	P_2	P_3	W_4	W_5	W_6	W_7	W_8
P_1				3	6	3	1	1
P_2				2	4	3	2	7
P_3				1	1	2	1	2
W_4			-1					
W_5			-1					
W_6		-3						
W_7	-1		-1					
W_8	-1	-7						

The matrix multiplication routine yields, at pivot step $k = 4$

	P_1	P_2	P_3	W_4	W_5	W_6	W_7	W_8
P_1				3	3	3	1	1
P_2				2	2	3	2	3
P_3				1	1	2	1	2
W_4			−1	0	0	1	0	1
W_5			−1	0	0	1	0	1
W_6		−3		−1	−1	0	−1	0
W_7	−1		−1	0	0	1	0	1
W_8	−1	−7		−5	−5	−4	−5	−④

Since $d_{88} < 0$, W_8 is the initial point.

Step 3.

					↓Initial point		
	W_4	W_5	W_6	W_7	W_8		
P_1	3	6	3	−1*	−1*	W_8, 1	W_7, W_8, 1
P_2	2	4	−3	2*	−7*	W_8, 7	W_7, 2
P_3	−1	−1	2	−1	2	W_8, 2	W_7, 1
	P_3, 1	P_3, 1	P_2, 4	P_1, 0	−, 0		
					P_2, −5	← Stop	

Now, C is located by tracing back from the initial point: W_8 to P_2 to W_7 to P_1 to W_8. Thus $C = (W_8P_2), (P_2W_7), (W_7P_1), (P_1W_8)$, asterisked above.

Continuing,

$$\delta = 2 \text{ and } X' \text{ is}$$

	W_4	W_5	W_6	W_7	W_8
P_1				$2 - 2 = 0^{(2)}$	$2 + 2 = 4^{(3)}$
P_2			3	$0 + 2 = 2^{(1)}$	$2 - 2 = 0^{(4)}$
P_3	2	2		2	

with the index numbers shown. The total cost is 23.

Now, returning to step 2 with $D(X')$ as shown below

	P_1	P_2	P_3	W_4	W_5	W_6	W_7	W_8
P_1				3	6	3	1	1
P_2				2	4	3	2	7
P_3				1	1	2	1	2
W_4			−1					
W_5			−1					
W_6		−3						
W_7		−2	−1					
W_8	−1							

the matrix multiplication routine gives us

	P_1	P_2	P_3	W_4	W_5	W_6	W_7	W_8
P_1	0	−1	0	1	3	2	1	2
P_2	2	0	1	2	4	3	3	3
P_3	1	−1	0	1	3	2	2	2
W_4	0	−2	−1	0	2	1	1	1
W_5	0	−2	−1	0	0	1	1	1
W_6	−1	−3	−2	−1	1	0	0	0
W_7	0	−2	−1	0	2	1	0	1
W_8	−1	−2	−1	0	2	1	0	0

Since no main diagonal elements are negative, the trial solution X' is optimal.

5. Convex Cost Flows

As indicated by Hu [9], the primal-dual type algorithms for minimal (linear) cost flow problems can be adapted to handle cases in which the edge cost functions are piece-wise linear convex. The basic notion, used here also, is to use the (changing) marginal costs associated with possible unit flow alterations in the associated graph. In our case, this means that cyclical flow changes will involve only one unit of flow (i.e., $\delta = 1$) and the edge costs for the associated graph will depend somewhat more on the current flow values on each edge than they did in the linear cost problem.

If we represent each edge cost function by b_{ij}, the problem is to find the minimum of

$$(5.1) \qquad Q(X) = \textstyle\sum_{(i,j)\varepsilon E} b_{ij}(x_{ij})$$

constrained by (1.1) and (1.2).

All we need to do is to note that the network $G(X)$ associated with a feasible flow X, is defined as before, except that the revised (marginal) edge costs b'_{ij} are given by

$$b'_{ij} = [b_{ij}(x_{ij} + 1) - b_{ij}(x_{ij})], \quad \text{if} \quad x_{ij} < k_{ij} \quad \text{and} \quad x_{ji} = 0;$$

and

$$b'_{ji} = -[b_{ij}(x_{ij}) - b_{ij}(x_{ij} - 1)], \quad \text{if} \quad x_{ij} > 0.$$

This, plus our previous observation that $\delta = 1$, enables the use of the minimal cost flow routine. Similar alterations can be made to handle the transportation problem with convex costs.

References

1. Balinski, M. L. and Gomory, R. E., "A Primal Method for the Assignment and Transportation Problems," *Management Science*, Vol. 10, No. 3, 1964, pp. 578–593.
2. Beale, E. M. L., "An Algorithm for Solving the Transportation Problem When the Shipping Cost Over Each Route is Convex", *Naval Research Logistics Quarterly*, Vol. 6, No. 1, 1959, pp. 43–56.
3. Busacker, R. G. and Saaty, T. L., *Finite Graphs and Networks*, McGraw-Hill, New York, N.Y., 1965.
4. Dantzig, G. B., *Linear Programming and Extensions*, Princeton University Press, Princeton, N.J., 1963.
5. Flood, M. M., "An Alternative Proof of a Theorem of Konig as an Algorithm for the Hitchcock Distribution Problem", in *Combinatorial Analysis*, American Mathematical Society, Providence, Rhode Island (1960) pp. 299–307.
6. ——, "On the Hitchcock Distribution Problem", *Pacific Journal of Mathematics*, Vol. 3, No. 2 (1953) pp. 369–386.
7. Ford, L. R., Jr. and Fulkerson, D. R., *Flows in Networks*, Princeton University Press, Princeton, N.J., 1962.
8. Hoffman, A. J. and Markowitz, H. M., "A Note on Shortest Path, Assignment, and Transportation Problems", *Naval Research Logistics Quarterly*, Vol. 10, No. 4 (1963) pp. 375–379.
9. Hu, T. C., "Minimum Convex-Cost Flows in Networks", *Naval Research Logistics Quarterly*, Vol. 13, No. 1 (1966).
10. ——, "A Decomposition Algorithm for Shortest Paths in a Network", *IBM Research Center* Report, RC 1562, Feb. 1966.
11. ——, "Revised Matrix Algorithm for Shortest Paths", *IBM Research Center* Report, RC 1478, Sept. 1965.
12. Jewell, W. S., "Optimal Flow Through Networks With Gains", *Operations Research*, Vol. 10, No. 4 (1962) pp. 476–499.
13. Kuhn, H. W., "The Hungarian Method for the Assignment Problem", *Naval Research Logistics Quarterly*, Vol. 2, Nos. 1 and 2 (1955) pp. 83–97.
14. Menon, V. V., "The Minimum Cost Flow Problem With Convex Costs", *Naval Research Logistics Quarterly*, Vol. 12, No. 2 (1965) pp. 163–172.
15. Munkres, J., "Algorithms for the Assignment and Transportation Problems", *Journal of the Society for Industrial and Applied Mathematics*, Vol. 5, No. 1 (1957) pp. 32–38.
16. Murchland, J. D., "A New Method for Finding All Elementary Paths in a Complete Directed Graph", LSE—TNT—22, *London School of Economics*, Oct. 1965.