

SOLUTION OF ASSIGNMENT PROBLEMS BY DIRECTED GRAPHS

DANIEL B. SUITS
University of Michigan

A NUMBER of linear programming problems of importance to agriculture belong to the general class called assignment problems: i.e., those in which specified requirements (in different uses, at different locations, etc.) are to be optimally satisfied by the assignment of amounts of a homogeneous resource available for the purpose (from different processes, at different locations etc.) The transportation problems, in which deficiencies at specified locations are to be satisfied by the shipment of surpluses from specified locations in such fashion as to minimize transportation cost, is a prototype of the assignment problem.

Several methods for the solution of such problems have been discussed.¹ In this paper a new procedure will be illustrated in which the directed graph is used as the central tool. Although the direct application of this method is limited to relatively small problems,² as a pedagogical device the graph is without equal. The graph provides the student with a concrete expression of the problem, and gives insight into the steps of the solution entirely lacking in abstract algebraic methods. In the following presentation the pedagogical aspects of the method will be emphasized.

A directed graph is defined as any collection of points, some of which are connected by directed lines which we may, for simplicity, call "arrows." Each of the figures of Figure 1 is a directed graph. A consists of four points with arrows pointing out from point (1) to each of the others; the three points of B are connected by two arrows emanating from point (3), the other going from (1) to (2). Although four of the five points of C are connected by arrows, the fifth point is isolated.

In general any number of points and any pattern of interconnecting arrows is permissible in a graph. Moreover the only thing that matters is the identification of the points and the arrows. The physical representation, e.g. the selection of lengths and shapes for the arrows, the layout of the graph on the page, etc., may be selected to suit convenience. For example, Graph D of Figure 1 is the same as Graph A.

¹ See, for example, Milton M. Snodgrass and Charles E. French, "Simplified Presentation of 'Transportation-Problem Procedure' in Linear Programming," *Journal of Farm Economics*, Vol. 39, No. 1, January, 1957, pp. 40-51, and the literature there cited.

² A problem involving N locations will require N(N-1) entries in Table I. (See below.) Although half of these are simply copied with a change in sign, the burden of computation still increases approximately as the square of the number of locations.

The study of graphs and their properties involves an extensive mathematics.³ For the present purpose, however, it is only necessary to distinguish two properties of graphs. A graph is called *connected* when, for every pair of points in the graph, it is always possible to find a path of arrows leading from one to the other. It is *not* required that this path coincide with the directions of the arrows over which it passes, nor does it matter how many intermediate points are passed through on the trip. If we think of the arrows simply as members that connect the points together, a connected graph is any one that can be picked up in a single piece. In Figure 1, graphs A, B, and D are connected; graph C is not.

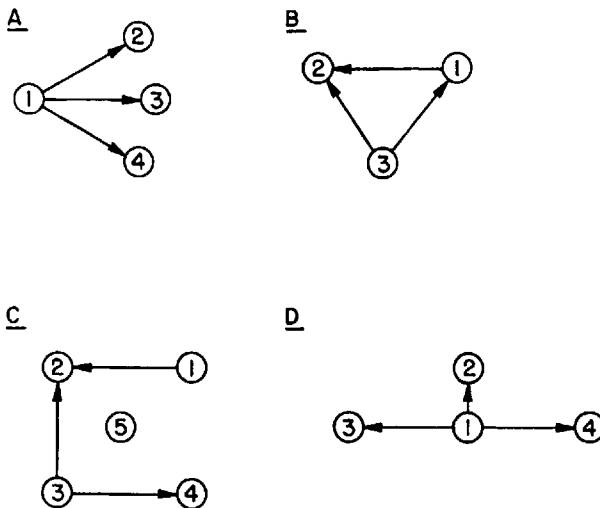


FIG. 1

A graph is *cyclical*, or contains a cycle, whenever there exists a closed path of arrows permitting a round trip which can be followed without passing over the same arrow twice. B is a cyclical graph; the others of Figure 1 are not. Note that the cycle need not follow all arrows in the same direction.

The great advantage of using the graph is precisely that it can be represented by points and lines on paper. Discussion in the text, however, and any algebraic manipulation require an alternative notation. For this purpose we will denote a point by enclosing its number in parentheses: e.g. (4). An arrow will be represented by A_{ij} , the first subscript designat-

³ An excellent introduction to graph theory with particular emphasis on applications in social sciences will be found in Frank Harary and Robert Z. Norman, *Graph Theory as a Model in Social Science* (Research Center for Group Dynamics Monograph No. 2), Institute for Social Research, University of Michigan, Ann Arbor, 1953.

ing the origin of the arrow, the second the point at which the tip rests: A_{52} is the arrow extending from (5) to (2). Finally, it is convenient to refer to the direction of a given path over an arrow. If the direction of a path coincides with the direction of the arrow, the arrow is said to be traversed in a positive direction. The arrow is traversed negatively when the direction of the path is opposed to the direction of the arrow.

The nature of the transportation problem is generally known and we need make only a few remarks about it. We are given a set of output (export, surplus, or source) points, each endowed with a specified amount of homogeneous output; also a set of input (import, deficit, or disposition) points, each with a specified requirement of the homogeneous commodity as input. We are also given a list of possible point-to-point shipping paths and the unit cost of shipping along each. We are required to assign quantities of goods to the available shipping paths in such fashion (a) that the requirements of the input points are satisfied and (b) that the total shipping cost is a minimum.

To facilitate comparison of the graphical method with other techniques we will illustrate the procedure with the same example used by Snodgrass and French. Their data for the problem are as follows:⁴

Point	Available Output	Required Input
(1)	-	20
(2)	-	10
(3)	5	-
(4)	25	-

Unit Shipping Costs				
From/To	(1)	(2)	(3)	(4)
(1)	-	2	4	6
(2)	2	-	6	2
(3)	4	6	-	4
(4)	6	2	4	-

The first step in the graphical procedure is to set up a format such as Table I to receive calculations. In column (1) we list all available assignments (routes) and in column (2) record the cost of each route as given in the shipping-cost table. The remaining columns will be ignored for the moment. We obtain our first basis or tentative solution as follows: on a sheet of paper set down the output points on the left and the input points on the right, noting the quantities available or required at each point. The order in which the points in each set are placed is immaterial. We now construct a graph on these points by connecting an arrow from the top output point to the top input point. (See Figure 2.) This arrow

⁴ Snodgrass & French, *loc. cit.*, p. 42. The shipping cost matrix used happens to be symmetrical. In general this need not be so, and the symmetry plays no role in the solution.

TABLE 1

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Route	Direct unit cost	Basis 1		Basis 2		Basis 3	
		Equiv. cost	Net saving	Equiv. cost	Net saving	Equiv. cost	Net saving
A 12	2	-4	-	-4	-	-2	-
A 13	4	-10	-	-4	-	-4	-
A 14	6	-6	-	-6	-	-4	-
A 21	2	4	2	4	2	2	0
A 23	6	-6	-	0	-	-2	-
A 24	2	-2	-	-2	-	-2	-
A 31	4	10	6	4	0	4	0
A 32	6	6	0	0	-	2	-4
A 34	4	4	0	-2	-	0	-
A 41	6	6	0	6	0	4	-2
A 42	2	2	0	2	0	2	0
A 43	4	-4	-	2	-2	0	-

represents the use of this output point to supply the input requirements of the top input point. If there is more than enough output for this purpose, connected the top output point to the next input point, and continue making connections until the top output is exhausted. The amount

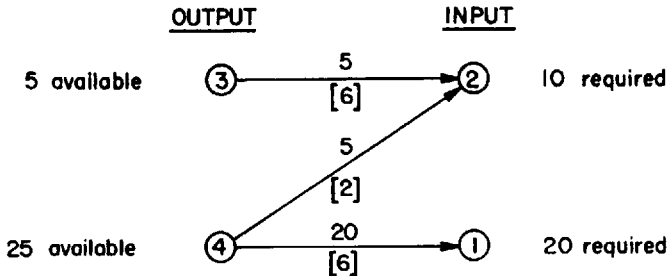


FIG. 2

being assigned to each arrow is marked above it and the unit cost is recorded underneath in brackets. These will be required later in the solution. We then follow with connections from the next output point etc., until all outputs are assigned to inputs. In this way we form a basis for the solution. That is, we obtain a program that will supply the input as required, although not in general at minimum cost. At this point the program is represented by a directed graph that is (a) connected and (b) without cycles.⁵

⁵ The graph will have no cycles because of the nature of its construction. Sometimes, however, the graph is not connected, i.e. some proper subset of the output points will precisely satisfy some proper subset of the input requirements. This is degeneracy. It is avoidable in a number of ways, one of which will be noted in our concluding remarks. We will assume for the present that proper steps have been taken to preclude its occurrence.

Since the amounts shipped and the unit costs have been recorded, the total shipping cost is readily calculated from the graph; $(5 \times 6) + (5 \times 2) + (20 \times 6) = 160$. In order to determine if this cost can be reduced, we now ask whether or not we could gain by utilizing some route not included in the basis. For example, what would happen if we sent a unit over A_{34} ? In the first place if a unit is sent over A_{34} , only 4 units could be sent over A_{32} . Therefore an additional unit will have to be sent over A_{42} in order to keep (2) supplied. Another way of expressing this is to say that if the route A_{34} is to be included in the shipping program, for every unit sent over it, we must remove a unit from route A_{32} and "remove a negative unit" (i.e. *add* a unit) to A_{42} .

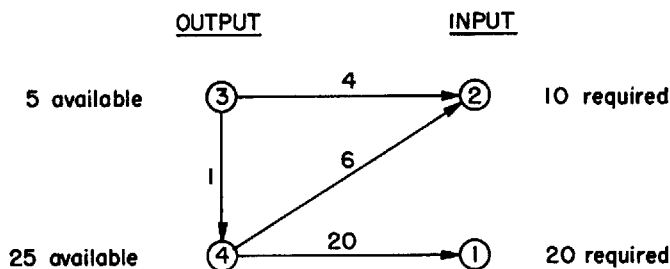


FIG. 2-A

Figure 2-A shows the shipping program modified to include one unit routed over A_{34} . The amount sent over A_{32} has been decreased by one and the amount moving over A_{42} is decreased by minus one. The program still exactly supplies the required inputs from the given outputs. The relationship among A_{34} , A_{32} and A_{42} is expressed as $A_{34} = A_{32} - A_{42}$. This expression means that shipping a unit over A_{34} is the equivalent, so far as accomplishing the program is concerned, of shipping an extra unit via A_{32} and one less unit via A_{42} . $A_{32} - A_{42}$ may be called the basis equivalent of A_{34} .

Whenever required, the unique equivalent of any route A_{ij} may be written down at a glance. We merely follow the path of arrows from (i) to (j), writing down the arrows traversed positively with a plus sign and those traversed negatively with a minus sign.⁶ Thus in Figure 2, $A_{21} = -A_{42} + A_{41}$; $A_{13} = -A_{41} + A_{42} - A_{32}$; $A_{24} = -A_{42}$, etc.⁷

⁶That such a path always exists is guaranteed by the fact that the graph is connected. That the path is unique follows from the fact that the graph contains no cycles.

⁷In an expression like $A_{24} = -A_{42}$ we can see most clearly the meaning of equivalence of routes. The expression means that if, in a program already involving shipments from (4) to (2), we include shipments from (2) to (4), every unit moved over the latter route must be sent right back again over the former in order to keep the requirements of (2) supplied.

Understanding the nature of equivalent routes can also be approached in this way: In a connected graph with no cycles, the insertion of a new arrow will always complete one cycle. The equivalent of the inserted arrow is then the remainder of the cyclic path.

Now that an equivalent for each A_{1j} can be found, we may ask if it would pay to insert A_{1j} in the program and remove its equivalent. To determine this we must compare the cost of A_{1j} with the cost of its equivalent. If the former is smaller than the latter, a gain can be had from the substitution. The cost of A_{1j} is already recorded in Table 1. We need only compute and record the cost of its equivalent. It follows from the foregoing that this is readily done by inspection of the graph. We follow the path of arrows from (i) to (j), summing the costs: those associated with arrows traversed positively enter the total positively, those with a negative path enter the total negatively. The result is the unit cost of the basis equivalent and indicates the gross amount that could be saved if the alternate route were followed. The cost of the basis equivalent of A_{34} is thus $6 - 2 = 4$. I.e., 4 per unit is the gross saving associated with route A_{34} . This is not, of course, the *net* unit saving, since the direct cost of A_{34} must be paid before any saving can be realized. The net unit-cost saving from any A_{1j} is the unit cost of the basis equivalent minus the unit cost of A_{1j} . Thus the net saving associated with the introduction into the program of one unit shipped via A_{34} is $4 - 4 = 0$, and there is no net gain to be had from the substitution. When the unit cost of the basis equivalent is negative there is no possibility of positive net saving, since even if the unit cost of A_{1j} were zero, its use would entail an increase in cost.⁸

These costs are quickly computed from the graph, but they need not all be independently calculated. The basis equivalent of A_{j1} is clearly the negative of the basis equivalent of A_{1j} , and if the cost of one has been computed, the other may be immediately set down with merely a sign change. Our next step is to compute column (3) of Table 1 by inspection of the graph, and, where required, obtain column (4) by subtracting the entry in column (2) from the entry in column (3).⁹

Inspection of column (4) indicates that a net saving of 2 per unit would be associated with introducing route A_{21} in the program; a net saving

⁸ If we like we can think of any route that enters in a basis equivalent with a negative sign as complementary to the given route. Thus, since $A_{34} = A_{33} - A_{23}$, A_{23} is complementary to A_{34} . Negative costs of basis equivalents are always associated with such complementary routes, arising because an increase in the use of A_{34} requires an increase in the use of A_{23} which, despite the saving from reduction of shipping along A_{34} , may well entail an increase in cost even though the unit cost of A_{34} is small.

⁹ Where column (3), the cost of the basis equivalent is a negative number, the net saving will always be negative and no calculation need be made in column (4).

of 6 per unit with the introduction of A_{31} in the program, and that no other substitutions would be profitable. Which substitution should be made? In reaching this decision the experience and judgment of the programmer is of considerable help. In the absence of other criteria a good general policy is to make the substitution with the largest saving in unit cost. Thus we will elect to substitute A_{31} in the program in place of its basis equivalent. Since we gain 6 in net saving for every unit sent over A_{31} , we want to divert as many units as possible to that route. What are the limitations? In order to determine this we inspect the graph and note the basis equivalent of $A_{31} = A_{32} - A_{42} + A_{41}$. This says that diverting a unit to A_{31} permits us to remove a unit from A_{32} and A_{41} provided that we add (remove a negative unit) a unit to A_{42} . We can remove no more units from A_{32} or A_{41} than are already moving over these routes. Inspection of the graph shows that 5 units are moving over A_{32} and 20 units over A_{41} . We can divert 5 units to A_{31} , since we can reduce the traffic on A_{32} by five units, as required. But we clearly cannot reduce the traffic on A_{32} by more than this, and 5 is the maximum number of units we can divert to A_{31} . This gives us our second basis: 5 units shipped over A_{31} , 15 units over A_{41} and 10 units over A_{42} .¹⁰

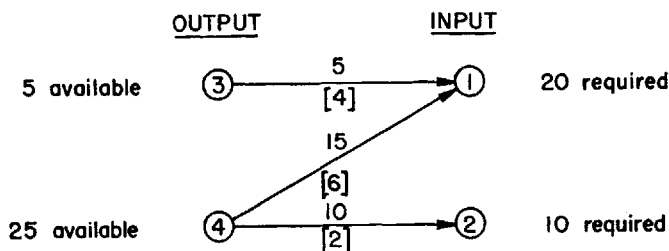


FIG. 3

A new graph (Figure 3) is constructed embodying the new program. Note that we have exercised our privilege of rearranging the points in a more convenient configuration. Computation of the cost of the new program

$$(5 \times 4) + (15 \times 6) + (10 \times 2) = 130$$

* The advantage of the graph is that it makes this operation of substitution a common sense procedure. Once this is grasped, the algebraic treatment can be approached with confidence. In the algebra of the graph the substitution is treated as follows: The original program was $5A_{32} + 5A_{42} + 20A_{41}$. In order to save cost we will add θA_{31} to the program and subtract its equivalent $\theta(A_{32} - A_{42} + A_{41})$, so the new program is $\theta A_{31} + (5 - \theta) A_{32} + (5 + \theta) A_{42} + (20 - \theta) A_{41}$. θ is now maximized subject to the condition that no coefficient in the new program be negative. Max. $\theta = 5$ and the new basis becomes $5A_{31} + 10A_{42} + 15 A_{41}$.

verifies that we have made a net saving of 6 per unit for five units diverted to A_{31} . The process of evaluating equivalents is now repeated with the new graph, and the results entered in columns (5) and (6) of Table I. The only profitable substitution is now $A_{21} = -A_{42} + A_{41}$. The graph shows that 15 units of A_{21} can be substituted into the program and the new program becomes $15 A_{21} + 25 A_{42} + 5A_{31}$, giving a total net saving of 30. The new program is shown in Figure 4.

The evaluation process is repeated in columns 7 and 8 of Table 1. No profitable substitutions remain, and program 3 is a least-cost program, the total transportation cost having been reduced to 100.

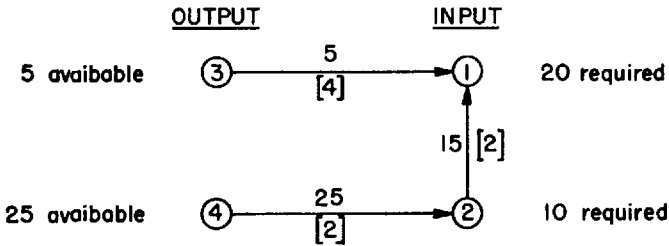


FIG. 4

Concluding Remarks

The graphic technique is merely one method of solving the assignment problem and as such it is subject to the general considerations pertinent to any such solution. We have supposed that total a available output is exactly equal to required input. In general this need not be so. If output exceeds requirements, we convert the problem to one of equality by setting up a dummy disposal point with input requirements equal to the excess supply. A unit "shipped" to this point is actually assigned to disposal (storage, dumping, etc.) at the point from which it is shipped. The unit cost of this assignment may be set as appropriate. In order to avoid nonsense solutions, however, no routes out of the disposal point should be permitted.

One interesting by-product of the graphical technique is the illustration of the problem of degeneracy. In terms of the graph, a problem degenerates whenever it is possible to represent a program by a non-connected graph. The problem shown in Figure 5 is degenerate. The effect of degeneracy is seen as soon as we try to determine equivalent costs of certain routes. For example we cannot calculate an equivalent cost for A_{35} or A_{16} or for any route that would connect the graph, since there is no alternative path to follow. The problem of degeneracy can

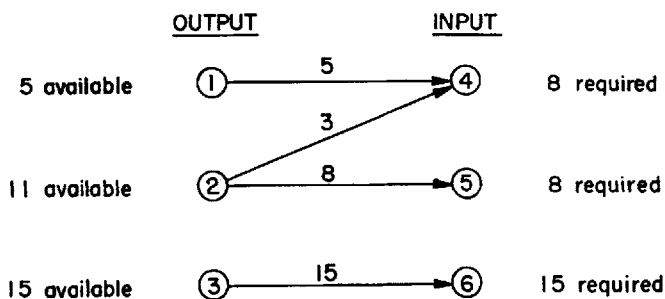


FIG. 5

always be avoided by changing the quantities of the problem very slightly. One way to accomplish this is to divide the last significant digit in the measurement of quantities (outputs and inputs) by $2m$ (where m is the number of output points) and add d , the first significant digit in the answer, to the supply of each of the output points and $(m \times d)$ to the requirements at an arbitrarily selected input point. The resulting problem always requires a connected graph for its solution.¹¹ When the optimum solution has been reached in terms of these adjusted quantities, the an-

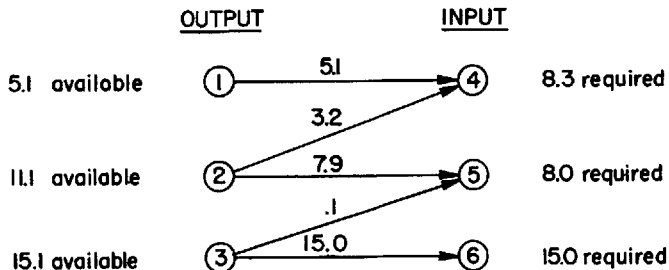


FIG. 6

swers are rounded to the original accuracy and the result is the optimum program for the original problem. This may be applied to Figure 5; $m = 3$. The last significant digit in the quantities is 1. $1/6 = .16$, the first significant digit of which is $.1$. Hence $.1$ is added to each supply and $.3$ to input requirements at, say, (4). The result is a connected graph as shown in Figure 6.

¹¹ Proof of this proposition is due to A. Orden. See T. C. Koopmans (Editor), *Activity Analysis of Production and Allocation*, New York: John Wiley & Sons, 1951, p. 366, footnote 3