REPRINTED FROM

NAVAL RESEARCH LOGISTICS QUARTERLY

OFFICE OF NAVAL RESEARCH

NAVSO P-1278

VOL. 13, NO. 1

MARCH 1966

MINIMUM-COST FLOWS IN CONVEX-COST NETWORKS*

T. C. Hu

University of California
Berkeley, California

ABSTRACT

An algorithm is given for solving minimum-cost flow problems where the shipping cost over an arc is a convex function of the number of units shipped along that arc. This provides a unified way of looking at many seemingly unrelated problems in different areas. In particular, it is shown how problems associated with electrical networks, with increasing the capacity of a network under a fixed budget, with Laplace equations, and with the Max-Flow Min-Cut Theorem may all be formulated into minimum-cost flow problems in convex-cost networks.

INTRODUCTION

Consider a connected network consisting of nodes N_i and arcs A_{ij} leading from N_i to N_j. Among the nodes N_i, there is a special node N_s called the *source*, and a special node N_t called the *sink*. The flow from N_i to N_j in the arc A_{ij} is denoted by x_{ij}. If it is possible to have flow from N_i to N_j or from N_j to N_i, then the arc A_{ij} is called an *undirected* arc. We consider the following problem:

$$\text{Min } z = \Sigma\, c_{ij}(x_{ij}) \tag{1}$$

subject to

$$\sum_i x_{ij} - \sum_k x_{jk} = \begin{cases} -v & \text{for } j = s \\ 0 & \text{for } j \neq s,t, \\ v & \text{for } j = t, \end{cases} \tag{2}$$

where $c_{ij}(x_{ij})$ are non-negative convex functions of x_{ij}, $(c_{ij}(0) = 0)$ and the arc flows x_{ij} are required to be positive integers or zero. The parameter v, which is required to be a non-negative integer, represents the total flow from source 'o sink. Note that Eqs. (2) express the conservation of flow at nodes other than the source and the sink. $\sum_j x_{sj}$ is the outflow of N_s and $\sum_j x_{jt}$ the inflow to the sink. The objective function (1) is a sum of convex functions (not necessarily stri^tly convex), and is thus convex.

*This manuscript is based on RAND Report RM-4265-PR. Views or conclusions contained in this paper should not be interpreted as representing the official opinion or policy of the U.S. Air Force. The completion of this manuscript was done when the author was at the University of California, supported by The Office of Naval Research, The Army Research Office, and The National Science Foundation.

We shall discuss, under Applications, how this problem is related to problems of finding maximum flow in a network with arc-capacity restrictions, problems in the synthesis of traffic or communication systems, electrical-network problems, and certain boundary-value problems. Similar work has been done in this area (see [1,3,7,11]). In [3], linear cost functions $c_{ij}(x_{ij})$ are considered; the method of [1] deals with bipartite networks and starts with a feasible solution.

The algorithm presented in this paper deals with general convex-cost functions in an arbitrary network and gives a feasible solution which is optimal for the parameter v. In spirit, it is closely related to that of [1] and [3].

A set of non-negative x_{ij} satisfying (2) is called a *flow* with value v. A flow which minimizes (1) for fixed v is called an *optimal flow* corresponding to v. Since the cost of shipping the flow along an arc is a convex function of the amount of flow shipped, the cost of shipping one additional unit of flow along the arc will depend on how much flow already exists on the arc. Following Beale [1], we define the so-called "up-cost" of an arc as follows. For an arc A_{ij} with $x_{ij} \geqq 0$ in the arc, the up-cost of that arc is the cost of sending one additional unit of flow from N_i to N_j, i.e.,

$$u_{ij}(x_{ij}) = c_{ij}(x_{ij} + 1) - c_{ij}(x_{ij}) \quad \text{for} \quad x_{ij} \geq 0. \tag{3}$$

Suppose we want to send one unit of flow from N_s to N_t in which we have to send one unit of flow from N_j to N_i along the arc A_{ij} where there already exists $x_{ij} \geq 0$ in the arc; then this one unit of flow from N_j to N_i will cancel one unit of x_{ij}, hence the cost of sending one unit of flow from N_j to N_i is actually negative.

We shall call this the "down-cost" of an arc; i.e., the cost of sending one unit of flow from N_j to N_i. In symbols,

$$d_{ji}(x_{ij}) = -c_{ij}(x_{ij}) + c_{ij}(x_{ij} - 1) \quad \text{for} \quad x_{ij} \geq 1. \tag{4}$$

We shall assume throughout the paper that $c_{ij}(0) = 0$; then it follows from the convexity of the cost functions that

$$c_{ij}(0) + [c_{ij}(x_{ij} + 1) - c_{ij}(0)]x_{ij}/(x_{ij} + 1) \geq c_{ij}(x_{ij}) \quad \text{for} \quad x_{ij} > 0.$$

Since $c_{ij}(0) = 0$, we have

$$c_{ij}(x_{ij} + 1) \geq (x_{ij} + 1)\, c_{ij}(x_{ij})/x_{ij}.$$

This means the up-cost of an arc is always positive. Similar reasoning shows that the down-cost of an arc is always negative. Furthermore, for any two non-negative integers a and b with $a < b$, we have

$$u_{ij}(a) \leq u_{ij}(b), \tag{5}$$

$$|d_{ji}(a)| \leq |d_{ji}(b)|. \tag{6}$$

For a given network with directed and undirected arcs, let a flow with value v_1 be given and denote its arc flows by $x_{ij}^{(1)}$. Let another flow with value v_2 be given and denote its arc flows by $x_{ij}^{(2)}$. If we superpose the two flows, then we get a flow with value $v_1 + v_2$. Note that if A_{ij} is a directed arc and having arc flow $x_{ij} > 0$, then it is possible to superpose a flow with arc flow $x_{ji} > 0$ provided $|x_{ij}| > |x_{ji}|$

$$x_{ij}^{(3)} = x_{ij}^{(1)} + x_{ij}^{(2)} ,$$

if $x_{ij}^{(1)}$ and $x_{ij}^{(2)}$ are of the same directions, and

$$x_{ij}^{(3)} = | x_{ij}^{(1)} | - | x_{ij}^{(2)} | ,$$

if $x_{ij}^{(1)}$ and $x_{ij}^{(2)}$ are of the opposite directions and $x_{ij}^{(1)}$ is of the greater magnitude. We say that two flow patterns are conformal if and only if

$$x_{ij}^{(3)} = x_{ij}^{(1)} + x_{ij}^{(2)}$$

for all arcs.

A particular flow, called a "path flow," is a flow with $x_{s1} = x_{12} = \ldots = x_{nt} = 1$, and $x_{ij} \equiv 0$ otherwise. If the cost of a flow with value v is known and we superpose a path flow on this given flow, the resulting flow has value v + 1. The total cost of the resulting flow with value v + 1 is the sum of the cost of the flow with value v, plus the sum of u_{ij} and d_{ij} used in the path flow. u_{ij} is used if the arc flow of the path flow is of the same direction as that of the arc flow of the flow with value v, and d_{ij} is used if the two arc flows are of opposite direction. The sum of u_{ij} and d_{ij} used in the path flow is called the *incremental cost* of the path flow.

ALGORITHM

The algorithm for solving the minimum-cost flow problem in convex-cost networks can be simply described as follows.

Starting with all $x_{ij} \equiv 0$, send one unit of flow from N_s to N_t along the path whose incremental cost relative to the existing flow is minimum. (This can be done by any of the existing shortest-path methods with u_{ij} and d_{ji} as the lengths; see, for example [6,11].) In the beginning, only the u_{ij} are relevant since there are no positive arc flows which could be potentially cancelled. Redefine the u_{ij} and d_{ji} based on the new flow pattern obtained, and send one additional unit of flow along the path with minimum incremental cost. The process of using the minimum incremental cost path is repeated until the total outflow of N_s is v (or the total inflow of N_t is v).

Many proofs are known for the case where the objective function (1) is a linear function. In Beale [1], an algorithm is given for a bipartite network with convex costs, and it starts with a feasible solution. It is easy to convert the existing proofs and ideas into the case of an arbitrary network and convex-cost functions, and to show that at every successive stage of the algorithm the flow pattern is optimum for the corresponding parameter v.

Let us give one example to illustrate the algorithm. Consider Figure 1. The cost function of each undirected arc is $c_{ij} x_{ij}^2$, with c_{ij} written beside the arc and $c_{ij} = c_{ji}$. Assume a given flow pattern as shown in Figure 2.

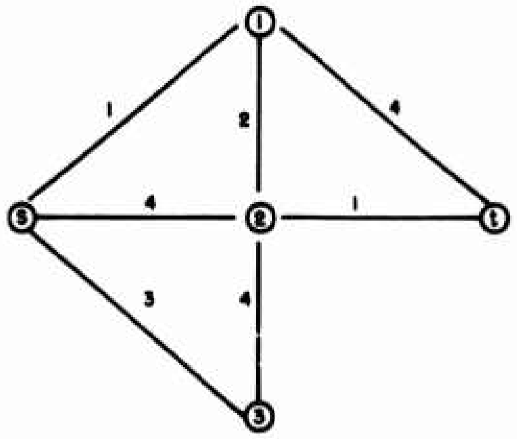

Figure 1

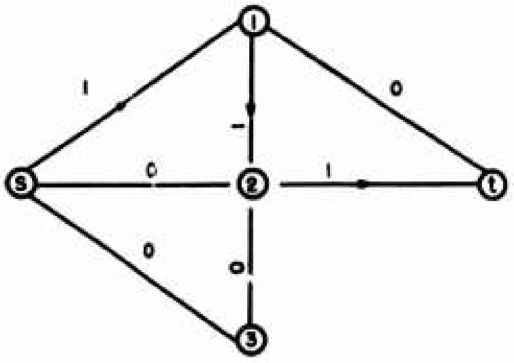

Figure 2

Then the up-costs and down-costs of every arc are calculated from (3) and (4) with the result shown in Figure 3, where the first number beside an arc is the up-cost, the second number is the down-cost, and the directions of up-costs are the same as in the flow pattern of Figure 2. For example, the up-cost of A_{12} is $2 \times 2^2 - 2 \times 1^2 = 6$. If an arc has no flow, like A_{s2}, the cost of flow from both directions will be $4 \times 1^2 - 4 \times 0^2 = 4$. If we want to send one additional unit flow with minimum incremental cost, we should use the arcs A_{s2}, A_{21}, and A_{1t} with total incremental cost $4 + (-2) + 4 = 6$, with the resulting flow pattern as shown in Figure 4.

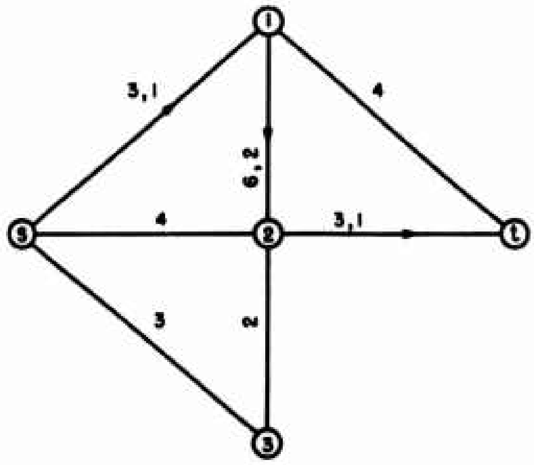

Figure 3

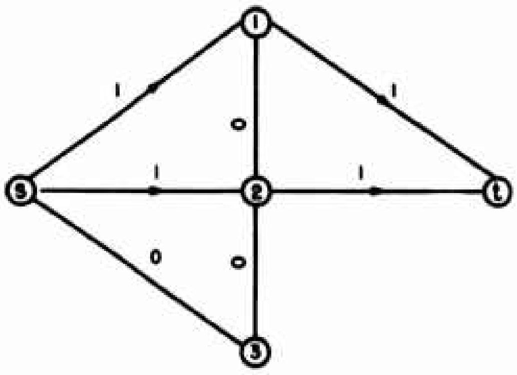

Figure 4

APPLICATIONS

Maximum-Flow Min-Cut Theorem

The problem of finding maximum flow through a network with b_{ij} as the capacities of arcs can be formulated as follows: max v subject to

(7)
$$\sum_i x_{ij} - \sum_k x_{jk} = \begin{cases} -v & \text{for } j = s, \\ 0 & \text{for } j \neq s, t, \\ v & \text{for } j = t, \end{cases}$$

and

(8)
$$0 \le x_{ij} \le b_{ij} \quad \text{for all } i, j.$$

This problem can be formulated as a minimum-cost flow in convex-cost networks as in (1) and (2). The cost function of an arc is shown in Figure 5. The up-cost when $x_{ij} > b_{ij}$ is not defined as it is not permitted by (8), and the up-costs and down-costs are easily seen to be

$$u_{ij}(x_{ij}) = 0 \quad \text{if } x_{ij} < b_{ij},$$
$$d_{ji}(x_{ij}) = 0 \quad \text{if } x_{ij} > 0,$$
$$u_{ij}(x_{ij}) = \infty \quad \text{if } x_{ij} = b_{ij}.$$

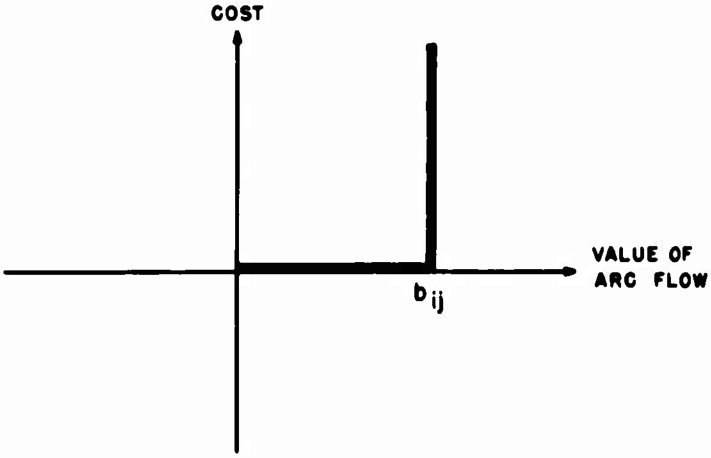

Figure 5

The v in (2) is taken as a parameter. The maximum flow v is obtained when the value of the objective function z becomes infinity for v + 1. This means when the value in (2) is v, there is no arc in the network with infinity cost, and when the value is v + 1, at least one arc is with infinity cost. Since we are minimizing z, there must be a set of arcs which form a cut in the flow with value v in which all $x_{ij} = b_{ij}$. Hence, the value v is the maximum-flow value. A proof as well as an algorithm is shown in Busacker and Gowen [3]. We can generalize the approach used in [3] to solve the following case:

(9)
$$\min z = \sum c_{ij}(x_{ij})$$

subject to

(10)
$$\sum_i x_{ij} - \sum_k x_{jk} = \begin{cases} -v & j = s, \\ 0 & j \neq j, t, \\ v & j = t, \end{cases}$$

(11)
$$0 \leq x_{ij} \leq b_{ij},$$

where in (9) z is a sum of any convex-cost functions. The algorithm of always choosing the path flow with minimum incremental cost works for the case when the objective function is a linear function (see, for example [3]), and its validity does not depend on how many arcs connect the two nodes. We shall transform Eqs. (9), (10), and (11) into a linear case as follows.

Consider an arc b_{ij} as a set of arcs, each with unit capacity. Index those arcs with positive integers $1, 2, \ldots, p$. The cost of the k-th arc from N_i to N_j is

$$c_{ij}(k) = c_{ij}(k-1) \text{ if } x_{ij} = 0,$$

and the cost is ∞ if $x_{ij} = 1$. It follows from the convexity of $c_{ij}(x_{ij})$ that the up-costs of arcs from N_i to N_j are always monotonically increasing with the index of the arcs, while the down-cost of the k-th arc is negative if it has flow. Assume that $x_{ij} > 0$; then if we want to send additional flow from N_i to N_j, we always use the arc with smallest index if that arc is not saturated; if we want to send flow from N_j to N_i, we always use the saturated arc with largest index. Then (9) becomes a set of linear cost functions of those unit-capacity arcs, since in no case would we use an arc with infinity cost.

Increasing the Capacity of a Network

This problem solved in [9] can be stated as follows. A network with capacity b_{ij} is given. Now, we want to increase the capacity or construct new arcs such that the maximum flow from N_s to N_t is increased. The cost of increasing or constructing a unit capacity from N_i to N_j is c_{ij}. The problem is to find $\max z = v$ subject to $c = \Sigma\, c_{ij}\, y_{ij}$

$$\sum_i x_{ij} - \sum_k x_{jk} = \begin{cases} -v & \text{for } j = s, \\ 0 & \text{for } j \neq s, t, \\ v & \text{for } j = t, \end{cases}$$

$$0 \leq x_{ij} \leq b_{ij} + y_{ij};$$

i.e., with a given budget, we want to maximize the flow from N_s to N_t by allocating y_{ij} appropriately.*

*The cost function on y_{ij} is like Figure 5 with the vertical line replaced by an inclined arc with slope c_{ij}.

This problem can also be solved by the algorithm of minimal incremental cost path. It follows from (3) and (4) that the up-cost and down-cost are:

$$\begin{aligned} u_{ij}(x_{ij}) &= 0 && \text{if } x_{ij} < b_{ij}, \\ u_{ij}(x_{ij}) &= c_{ij} && \text{if } x_{ij} \geq b_{ij}, \\ d_{ji}(x_{ij}) &= 0 && \text{if } 0 < x_{ij} \leq b_{ij}, \\ d_{ji}(x_{ij}) &= -c_{ij} && \text{if } x_{ij} > b_{ij}. \end{aligned}$$

Then the solution is to always send from N_s to N_t one unit of flow along the minimal incremental cost path (since we can consider the problem as $\min \Sigma\, c_{ij}\, y_{ij}$ and treat v as a parameter), until the total amount of money used up is c. Then

$$\begin{aligned} y_{ij} &= x_{ij} - b_{ij} && \text{if } x_{ij} > b_{ij} \\ y_{ij} &= 0 && \text{if } x_{ij} \leq b_{ij}. \end{aligned}$$

Electrical Network

Consider a passive electrical network with one current-input source at N_s and one current-output source at N_t. From Ohm's law, the electrical current x_{ij} from N_i to N_j is proportional to the potential difference ϕ_{ij} and inversely proportional to the resistance r_{ij} of that arc; i.e.,

$$x_{ij} = \frac{\phi_{ij}}{r_{ij}}.$$

The work done by that arc is $x_{ij}\ \phi_{ij} = r_{ij}\, x_{ij}^2$. To solve an electrical network of the preceding type, we can solve the simultaneous equations given by Kirchoff's node law and Kirchoff's loop law. Alternatively, we can regard Kirchoff's node law as linear constraints of the currents x_{ij}, and minimize the total work done. This, then, becomes the following quadratic programming problem (see [5]): $\min z = \Sigma\, r_{ij}\, x_{ij}^2$ subject to

$$\sum_i x_{ij} - \sum_k x_{jk} = \begin{cases} -v & \text{for } j = s, \\ 0 & \text{for } j \neq s, t, \\ v & \text{for } j = t. \end{cases}$$

This problem can again be handled by the minimum incremental cost-path algorithm by defining costs of arcs as follows. Let

$$C_{ij}(x_{ij}) = r_{ij}\, x_{ij}^2$$

thus

$$u_{ij}(x_{ij}) = r_{ij}(2x_{ij} + 1) \quad \text{if } x_{ij} \geq 0,$$

$$d_{ji}(x_{ij}) = -r_{ij}(2x_{ij} - 1) \quad \text{if } x_{ij} \geq 1.$$

Laplace Equations

The consideration of a network problem as a boundary-value problem was done in [2]. Let us consider the Laplace equation in a region G, where ϕ is a function of two real variables.

(12) $$\nabla^2 \phi = 0$$

with the normal derivative prescribed on the boundary of G. If we use difference equations to replace (12) and use a uniform grid, then the value of ϕ_0 at a point is the average value of its four neighbors (see Figures 6 and 7); i.e.,

(13) $$4\phi_0 - \phi_N - \phi_S - \phi_E - \phi_W = 0.$$

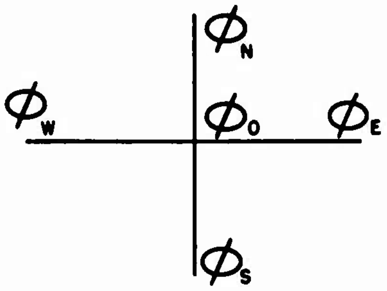

Figure 6

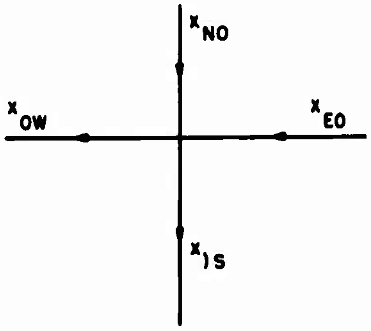

Figure 7

By rewriting (13) and letting $x_{E0} = \phi_E - \phi_0$, and so forth, we have

(14) $$x_{E0} - x_{0W} + x_{N0} - x_{0S} = 0.$$

Equation (14) then can be considered as the conservation-of-flow equation, with x_{E0} the arc flow from node E to node 0. The boundary condition of prescribing $\partial\phi/\partial n$ is then interpreted as the condition of inflow and outflow at sources and sinks in a network.* The Dirichlet principle (see, for example [8]) for solving a Laplace equation can then be regarded as that of minimizing a quadratic objective function,

(15) $$\min z = \Sigma x_{ij}^2,$$

subject to Eq. (14) at interior points of region G and satisfying the boundary condition $x_{ij} = \partial\phi/\partial n$ at the boundary of G.

*The multiple source and sinks can be easily transformed into one source and one sink. See page 15 of [8].

We have the objective function

$$z = \Sigma x_{ij}^2 ,$$

thus

$$u_{ij}(x_{ij}) = 2x_{ij} + 1 \quad \text{if} \ x_{ij} \geq 0,$$

$$d_{ji}(x_{ij}) = -2x_{ij} + 1 \quad \text{if} \ x_{ij} \geq 1.$$

Then the Laplace equation can be solved by min incremental cost path from sources to sinks, as done previously. Special examples can be given to show that this approach is better. Details of this algorithm and numerical examples will be shown in [10].

ACKNOWLEDGMENT

The author wishes to thank Dr. A. J. Hoffman for valuable suggestions.

BIBLIOGRAPHY

[1] Beale, E. M. L., "An Algorithm for Solving the Transportation Problems when the Shipping Cost Over Each Route is Convex," Naval Research Logistics Quarterly, 6, 43-56 (Mar. 1959).

[2] Birkhoff, G. and Diaz, J. B., "Non-linear Network Problems," Quarterly of Applied Mathematics, Vol. 8, No. 4 (Jan. 1956).

[3] Busacker, R. G. and Gowen, P. J., "A Procedure for Determining a Family of Minimal-Cost Network Flow Patterns," Operations Research Office, Technical Paper 15 (1961).

[4] Courant, R., Dirichlet's Principle, Conformal Mapping, and Minimal Surface (Interscience, New York, 1950).

[5] Dennis, J. B., Mathematical Programming and Electrical Networks (John Wiley & Sons, Inc., New York, 1959).

[6] Ford, L. R., Jr., "Network Flow Theory," The RAND Corporation, P-923 (July 1956).

[7] Ford, L. R., Jr. and Fulkerson, D. R., "Constructing Maximal Dynamic Flows from Static Flows," Operations Research, 6, 419-433 (1958).

[8] Ford, L. R., Jr. and Fulkerson, D. R., Flows in Networks (Princeton University Press, Princeton, N. J., 1962).

[9] Fulkerson, D. R., "Increasing the Capacity of a Network: The Parametric Budget Problems," Management Science, 5, 472-483 (1959).

[10] Hu, T. C., "Laplace's Equation and Network Flows," to appear in Operations Research.

[11] Moore, E. F., "The Shortest Path Through a Maze," Proceedings of an International Symposium on the Theory of Switching - Part II; The Computation Lab of Harvard University, Annals, 30, 285-292 (1959).

* * *