Letters to the Editor

THE MAXIMUM CAPACITY ROUTE PROBLEM

T. C. Hu

IBM Research Center, Yorktown, New York

(Received May 3, 1961)

THE maximum capacity route problem raised by POLLACK[1] can be formulated as follows. An n-node network consists of nodes $N_i(i=1, \cdots, n)$ and arcs A_{ij} leading from N_i to N_j. Each arc has associated with it a nonnegative number c_{ij}, called the arc capacity, which denotes the maximum amount of flow that can pass through the arc from N_i to N_j. The problem is to find a route from N_i to N_j such that

$$\min\ (c_{ia}, c_{ab}, \cdots, c_{dj}) \text{ is maximum,}$$

where $c_{ia}, c_{ab}, \cdots$ denotes arc capacities of the arcs which form a route from N_i to N_j.

Now we consider the problem of finding maximum capacity routes between all pairs of nodes in a network. This can be divided into two cases; (1) in which $c_{ij}=c_{ji}$, (2) $c_{ij}\neq c_{ji}$.

When $c_{ij}=c_{ji}$, let us define a maximum spanning tree of the network as a spanning tree with the sum of c_{ij} associated with the arcs in the tree a maximum. This problem was solved by KRUSKAL[2] and PRIM,[3] and very simple algorithms for constructing the tree can be found in references 2 and 3 or Appendix II of reference 4. For any arc A_{ij} not in the maximum spanning tree, we have

$$c_{ij} \leqq \min\ (c_{ia}, c_{ab}, \cdots, c_{dj}),$$

where $A_{ia}, A_{ab}, \cdots$, are arcs in the tree which form a unique route from N_i to N_j. So the whole problem can be summarized as follows. *For a network with $c_{ij}=c_{ji}$, maximum capacity routes between all pairs of nodes can be found in a subset of arcs in the network. This subset consists of n-1 arcs which form a maximum spanning tree.*†

When $c_{ij}\neq c_{ji}$, we consider the n-node network consisting of all $n(n$-1) arcs. If a route from N_a to N_q with c_{ij} associated with its arcs has the following property

$$c_{ip}\leqq\min(c_{ij}, c_{jk}, \cdots, c_{0p}),$$

where $A_{ij}, A_{jk}, \cdots$ are arcs in the route and A_{ip} is any arc that is not in the route, we say that the A_{ip} is dominated. Clearly, a dominated arc can be omitted with-

† If c_{ij} = min (c_{ia}, … , c_{dj}) we choose the maximum capacity route that uses arcs in the subset. This will not occur if all capacities are different.

out reducing capacities of any maximum capacity route. Now we establish a similar result when $c_{ij} \neq c_{ji}$.

For a n-node directed network, when $n(n\text{-}1)$ maximum capacity routes are to be found, all routes can be found in a subset of the original $n(n\text{-}1)$ arcs. This subset contains at least n arcs and contains at most $(n\text{-}1)$ $(\frac{1}{2}n+1)$ arcs.

Proof: Assume that the subset of arcs contains $(n\text{-}1)$ arcs or less. Since it takes at least n arcs to form a strongly connected graph[5] containing n nodes, there must be a route from N_i to N_j that uses an arc not in the subset. On the other hand, if n arcs with the largest c_{ij} form a circuit, all other arcs are dominated, and all maximum capacity routes use only the n arcs.

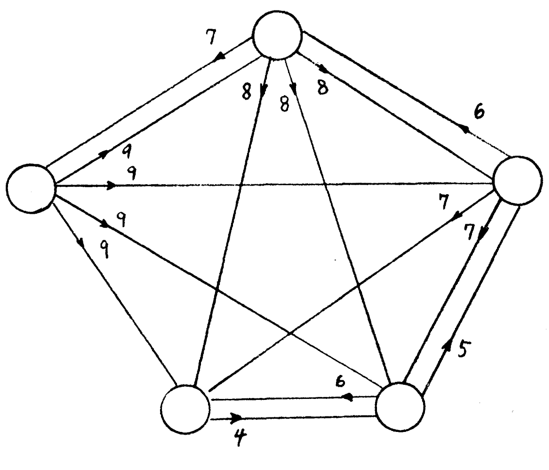
To prove that there are at most $(n\text{-}1)$ $(\frac{1}{2}n+1)$ arcs in the subset D, let us select the subset of arcs A_{ij} by their magnitude, i.e., first select the arc with the largest c_{ij}, then the arc with the next largest c_{ij}, as each arc is selected as a member of D, all arcs which become dominated are then eliminated from the network before the next arc is selected. This is continued until all A_{ij} are either selected or eliminated. In other words, when the selection process is completed, for any A_{kl} not in the subset, there is a route formed by the arcs in the subset with minimum capacity $\geqq c_{kl}$. The selected A_{ij} will form a graph called the dominating net D.

Two arcs A_{ij} and A_{ji} in the dominating net D are said to form a pair. If the A_{ij} are chosen as described, there are at most $(n\text{-}1)$ pairs in D. Assume that there are n or more pairs. Since there are at most $(n\text{-}1)$ pairs in a tree, n or more pairs of them must form a loop L. Choose min c_{pq}, where $N_p \in L$, $N_q \in L$. Then this A_{pq} is dominated, as there is a route $A_{pa}, A_{ab}, \cdots, A_{dq}$ with minimum capacity $\geqq c_{pq}$. So A_{pq} does not belong to D, a contradiction.

Consider all $n(n\text{-}1)$ arcs between n nodes. They will form $\frac{1}{2}n(n\text{-}1)$ pairs if all are selected. If the dominating subset has no pair at all, it contains at most $\frac{1}{2}n(n\text{-}1)$ arcs, i.e., only one of the A_{ij}, A_{ji} is selected. A dominating subset with $(n\text{-}1)$ pairs then contains at most

$$\tfrac{1}{2}n(n\text{-}1)+n\text{-}1=(n\text{-}1)[\tfrac{1}{2}n+1] \text{ arcs.}$$

The following is a dominating net D which consists of $(5-1)[\frac{5}{2}+1]=14$ arcs.



Although the algorithms given yield the maximum capacity route between all node pairs, it is also true that the algorithms can be halted at any time. The routes found thus far are the correct routes for those nodes that are connected. Therefore, even if one does not want the routes between all the node pairs, the algorithms given can still be used and the procedure halted when the node pairs of interest are connected.

REFERENCES

1. M. Pollack, "The Maximum Capacity Route Through Network," *Opns. Res.* **8,** 733–736 (1960).
2. J. B. Kruskal, Jr., "On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem," *Proc. Am. Math. Soc.* **7,** 48–50 (1956).
3. R. C. Prim, "Shortest Connection Networks and Some Generalizations," *Bell Systems Tech. J.* **36,** 1389–1401 (1957).
4. R. E. Gomory and T. C. Hu, "An Application of Generalized Linear Programming to Network Flows,"*IBM Research Report* RC-319, September, 1960.
5. C. Berge, *Theorie des Graphes*, Dunod, Paris, 1958.

NOTE ON A PAPER BY HANSSMANN

Anthony Mediate and Stanley Zionts

United States Steel Corporation, Pittsburgh, Pennsylvania

(Received May 26, 1961)

IN THE interesting paper "Determination of Optimal Capacities of Service for Facilities with a Linear Measure of Inefficiency,"* Friedrich Hanssmann solved the following problem:

> A service facility F_i always employs an 'effort' x_i to perform a service. x_i is called its capacity. If a customer desires an effort $x<x_i$, there will be a wasted effort x_i-x. If a specified number, m, of facilities with different capacities $x_1, \cdots, x_m$ is to be installed, what are the capacities $x_1, \cdots, x_m$ that will minimize the wasted effort?

The purpose of this note is to present a corrected version of Table I, the iteration of the example considered, and another observation. Mr. Hanssmann was very fortunate in having the errors of computation in the first two iterations lead him directly to the correct solution. The interpolated solutions do converge quite rapidly; however, an additional iteration is involved.

We have also observed that the equation following Fig. 2 is in error, although the entry in Table II has been calculated by the use of the correct formula.

It follows from Hanssmann's equation (2) that

$$E=[1-F(x_{n\text{-}1})]x_n+[F(x_{n\text{-}1})-F(x_{n\text{-}2})]x_{n-1}+\cdots+[F(x_2)-F(x_1)]x_2+[F(x_1)]x_1-\bar{x}.$$

* *Opns. Res.* **5,** 713–717 (1957).