INTERFACES
Vol. 8, No. 4
August 1978

IMPROVED COMPUTER-BASED PLANNING TECHNIQUES, PART I*#

Fred Glover,† John Hultz,‡ Darwin Klingman§

ABSTRACT. This is Part 1 of a two-part series. It shows how advances in network solution techniques have brought about improved computer-based planning. We give examples of practical problems and their network models that have resulted in dramatic cost savings for OR/MS practitioners. Not only are the models more convenient (easier to visualize and communicate) than customary LP models, but the new advances enable them to be solved more than 100 times faster than LP models.

Part 2 (coming in the November issue) will show how related models have led to further cost savings in practical applications.

Introduction

The growth of the computer industry has had a profound influence on many areas, with the most dramatic impact probably in the area of management science. This is evidenced by the evolution of computer-based planning models. This evolution has resulted in new developments in two technical areas: network computer implementation techniques and network formulation (NETFORM) techniques.

The first emerged from recent research on new solution algorithms and implementation techniques for solving network problems [2], [3], [5], [11], [13], [14], [17], [19], [23]. This development has dramatically reduced the cost of solving linear and mixed integer network type problems, without requiring any changes in computer hardware or compilers. For example, the cost of solving network problems with 2400 equations and 500,000 arcs on an IBM 360/65 has been reduced from a conservative estimate of $10,000 in 1968 to $300 in 1976 by these advances.

The second technical advance consists of new modeling techniques designed to handle a multitude of problems that arise in applications of scheduling, routing, resource allocation, production, inventory management, facilities location, distribution planning, and other areas. These new modeling techniques [12], [15], [16], [18] are

---

*This two-part paper is a condensed version of an original paper delivered at the SAHRE XLVIII Conference in Houston, March 6–11, 1977. The full version of this paper may be obtained by writing Fred Glover or Darwin Klingman.

#This research was partly supported by ONR Contract N00014-76-C-0383 with Decision Analysis and Research Institute and by Project NR047-021, ONR Contracts N00014-75-C-0616 and N00014-75-C-0569 with the Center for Cybernetic Studies, The University of Texas at Austin. Reproduction in whole or in part is permitted for any purpose of the United States Government.

†Professor of Management Science, College of Business, University of Colorado, Boulder, CO 80302

‡Senior Systems Analyst, Analysis, Research and Computation, Inc., Box 4067, Austin, TX 78767

§Professor of Operations Research and Computer Sciences and Director of Computer Science Research, Center for Cybernetic Studies, The University of Texas at Austin, BEB 608, Austin, TX 78712

mathematically and symbolically linked to network structures and are called NETFORM (network formulation). A major attribute of NETFORM is that it allows users to think of their problems graphically. The pictorial aspect of this technology has proven to be extremely valuable in both communicating and refining problem interrelationships. It does this without the use of obscure mathematical notation and computer jargon. Thus, it protects the nontechnical person against technical legerdemain and exaggerated claims of model "realism." Another powerful attribute of this technology is that it often yields a model which can be solved as a sequence of linear network problems.

The purpose of this paper (Part 1 of a two part series) is to briefly describe and demonstrate the power of each of these techniques when used in *concert* to model and solve real-world applications. We argue on this basis of *practical* experience that these advances overcome many of the conceptual design and computational difficulties of previous optimizing procedures, and consequently provide the flexibility required of truly useful decision planning tools.

In this paper we present examples of how practical management science problems can be viewed graphically, by focusing on fundamental model constructions and applications of pure and generalized network problems. We also indicate the computational advantages of the network-related models by reporting comparisons of network computer codes with a commercial linear programming code. In Part II, which will appear in a subsequent issue, we introduce the very general and powerful NETFORM constructions which dramatically extend the applications of network-related models, and provide detailed discussions of three real-world NETFORM applications. These applications include computational comparisons of algorithms based on the NETFORM models and the mixed integer programming methods (commercial and otherwise).

Pure network models

Pure network problems actually embody a group of distinct model types. This group includes shortest path, assignment, transportation, and transshipment problems. For the sake of brevity we will focus attention on the most general of these model types.

Transshipment problem

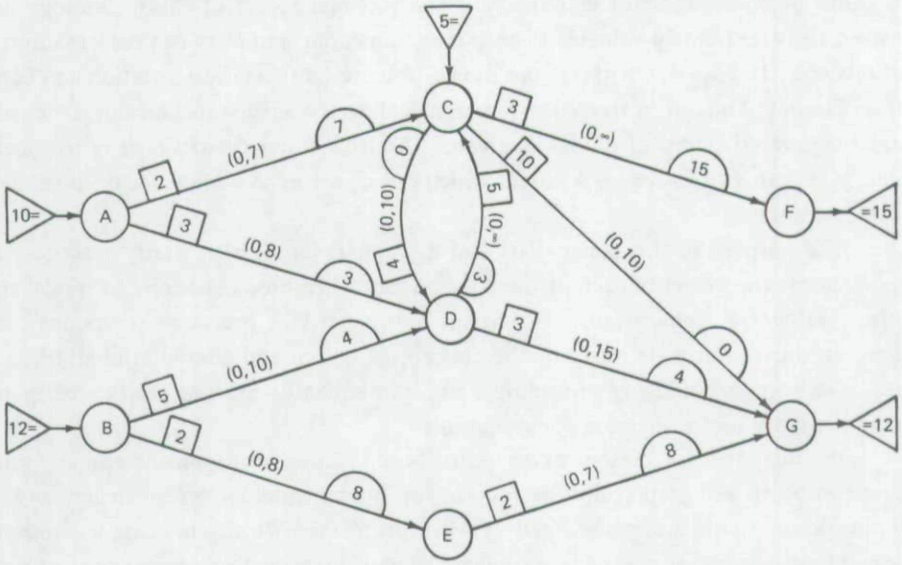
The most general pure network structure that appears in numerous applications—either directly, or as a subproblem—is the *transshipment model.* A transshipment problem, sometimes also called the minimum cost flow problem, seeks the minimum cost (or maximum profit) way to sent goods from origins to destinations along admissible routes. It extends the "classical" transportation or distribution problem by introducing intermediate (transshipment) nodes to provide junctions or "way stations" through which goods may be shipped en route from origins to destinations.

An illustration of this model for a cash flow problem is depicted in Figure 1.

The *arrows* shown in Figure 1 are called *arcs* and the *circles* are called *nodes.* In this cash flow network, the nodes may be thought of as corresponding to subsidiaries in different locations. The *supplies* and *demands*—which are shown in the triangles leading into a node for a supply and out of a node for a demand—may be thought of

FIGURE 1

Capacitated Transshipment Cash Flow Problem



as representing excess or deficit cash positions. Thus, by the indicated supplies and demands, before sending any flows nodes A, B, and C have excess funds, nodes D and E have no funds, and nodes F and G have deficit funds. The squares enclose the arc costs (per unit of flow) and the semi-circles enclose the values of flows in a current cash distribution. The arcs indicate the ways to transfer cash from one subsidiary to another. For instance, the arc from node A to node C indicates that it is possible to transfer funds from subsidiary A to subsidiary C. The absence of an arc between a pair of subsidiaries indicates that it is not possible to transfer funds directly between them (though it may be possible to transfer funds indirectly by means of a sequence of arcs). Each arc has a lower bound and an upper bound which appear within the parentheses on the arc. For example, Figure 1 indicates that the arc from node A to node C has a lower bound of 0 and an upper bound of 7. The quantity in the square also indicates that this arc has a cost of 2.

The objective in the transshipment problem is to determine how much to ship along each arc (route) within the limits stipulated by the bounds in order to: (1) satisfy all supplies and demands; and (2) minimize total cost. By satisfying supplies and demands we mean that the total flow into the node minus the total flow out must equal its demand, and the total flow out of the node minus the total flow in must equal its supply. For all other nodes the flow into the node must equal the flow out. The numbers in the semi-circles on the arcs in Figure 1 illustrate a solution which satisfies these *node equations* and the bound requirements.

A number of practical cash management problems have recently been modeled as transshipment problems using constructions of the sort illustrated. These models include sources of funds in addition to cash (such as maturing accounts and notes receivable, sales of securities, borrowing, etc.) and uses of funds other than a single "investment." A generalized network model, to be discussed subsequently, makes it possible to further incorporate discount, interest, and other financial considerations directly into the model. We will now indicate other practical applications of pure network models.

Applications of pure network problems

Numerous mathematical optimization problems and major components of many additional problems can often be modelled as pure network problems. For example, inventory maintenance problems [9], [24], [26] typically exhibit an underlying network structure. A cousin of the inventory maintenance problem is the so-called PERT/CPM problem, which seeks the best way to sequence a complex set of interdependent activities. The PERT/CPM framework, which constitutes one of the simplest network model forms, has been used in a variety of practical applications (including construction of the Polaris submarine) and has been reported to save enormous dollar costs and greatly speed the completion of complex projects.

Problems involving the effective management of resources also often exhibit network structures and are becoming increasingly important in government and industry. Direct network formulations of water resource management problems, for example, are finding use in a number of states. In these, canals, river reaches, and pipelines take the role of arcs while reservoirs and pumping stations take the role of nodes. Planning over time frequently looms as a major consideration in these applications.

The Texas Water Development Board and the government of Poland, for example, use a succession of simulations of alternative "supply configurations," and solve the resulting network for each simulation run. (The step of finding the optimum solution to each network problem is used to determine the best response to meet demands for water use, given a particular supply configuration.) Roughly 500 such runs are made each month. The feasibility and cost-effectiveness of such runs of course owe heavily to the efficiency of solving the underlying networks.

The problem of determining flows and heads in a general pipeline system (such as a municipal water system) with reservoirs, pumps, gate and check valves, given fixed inputs and withdrawals has been recently shown in [21] to be equivalent to a convex transshipment problem under the assumption of convex head losses. Such problems are easily solved as ordinary transshipment problems using a piecewise linear approximation of the convex function. Since the convexity requirements are usually satisfied for real pipe networks, this is an example of another class of real-world problems which can now be handled by network procedures with far greater effectiveness than by the procedures applied to these problems in the past.

Another important instance of the use of network models occurs in man-power promotion and assignment problems. AT&T has developed such models in order to guarantee acceptable hiring and promotion policies in accordance with HEW rules and regulations.

Many nonlinear problems involve network subproblems. One of the most basic and prevalent forms of nonlinear problems is the fixed-charge network problem, whose major offshoots include the extremely important genre known as "location" problems. The nonlinear element of a fixed-charge network is the fixed-charge arc which has the following special property: whenever the arc is "used" (i.e., permitted to transmit flow), a charge is incurred that is independent of the amount of flow across the arc. Fixed-charge networks have been used to model problems of plant and warehouse location, equipment purchasing and leasing, personnel hiring and offshore oil drilling platform location, among others.

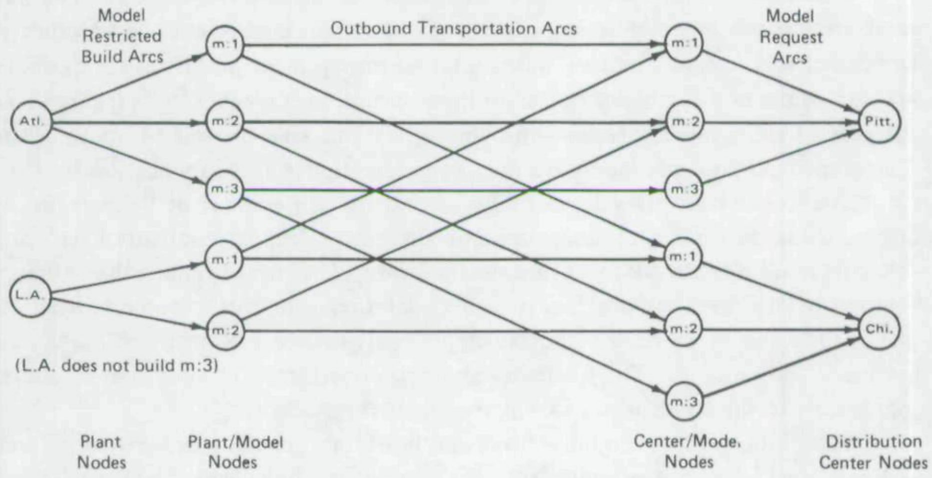
To provide a fuller appreciation of the ingredients of such models, we will now discuss in detail an example from an important practical application.

Production planning and distribution application

A major U. S. car manufacturer has developed and implemented a transshipment model for production planning and distribution decisions. This model is noteworthy for demonstrating the value of networks in interactive decision making. Figure 2 illustrates this application.

FIGURE 2

Production Planning and Distribution



The problem, in simplified form, is to determine the number of cars of each of three models (m:1, m:2, and m:3) to produce at the Atlanta and Los Angeles plants (represented by the "Atl." and "L.A." nodes), and then to determine how many of each of these car models to ship from each plant to the distribution centers in Pittsburgh and Chicago (represented by the "Pitt." and "Chi." nodes). The objective is to identify a production-distribution plan that minimizes total cost.

Bounded supplies are associated with the Atl. and L.A. nodes, indicating the least and most that can be produced at these plants. In addition, upper and lower bounds are placed on the various arcs emanating from these two nodes to control the minimum and maximum number of each particular car model that can be produced at these plants. Similar bounds (capacity restrictions) can also be placed on other arcs. For instance, if there is a limit on the number of m:1 cars that can be shipped from Atlanta to Pittsburgh, then this appears as an upper bound restriction on the "top-center" arc in the network. Finally, the number of each particular model required at Pittsburgh and Chicago is handled by placing bounds on the "far right" arcs.

For example, if exactly 4,000 m:3-type cars are required in Chicago, then 4,000 becomes both the lower and upper bound on the m:3-Chi. arc.

An interesting feature of this model is not only that it coordinates the production and distribution decision, but that it handles a multi-commodity problem in a "single-commodity" framework. That is, the three models, m:1, m:2, and m:3, are distinct commodities being shipped through the network, but their identities never get mixed or confused, as could be possible in some network models. This illustrates the importance of getting the "right" network formulation.

The typical size of this problem for a particular division (e.g., Pontiac, Ford, Dodge, etc.) is 1,200 nodes and 4,000 arcs. The company initially used a version of the SHARE out-of-kilter code [22] to solve these problems. The solution time ranged from 10 to 20 minutes on an IBM 370/145 and required 150K bytes of computer memory. Using the transshipment codes of [1], [14] the problem was solved in less than 20 seconds on the company's IBM 370/145, using only 80K bytes of computer memory, thus making it possible to solve such problems in an on-line computer mode. In fact, due to the nature of the decision making environment of this application, the company has developed an on-line real-time production planning and distribution system which is linked to a graphics display terminal and an English language input processor. This system is currently being used at several administrative levels within the corporation hierarchy for planning purposes.

Today, by using the interactive on-line network system and utilizing visual displays, plant executives are able to discuss their goals and assumptions in a very short time. Answers to "what if" questions are quickly obtained and evaluated. In fact, the executives typically are able to evaluate 150–200 production plans each quarter with the aid of the network system.

Generalized network

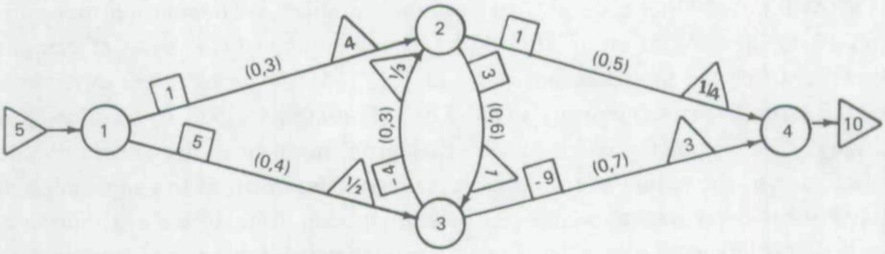
The generalized network (GN) problem represents a class of LP problems that heretofore has received only a small portion of the attention it has deserved. Recently, however, with the identification of many new generalized network applications and with the emergence of computer codes able to solve these problems efficiently, generalized networks are rivaling or even surpassing pure networks in their practical significance. Generalized networks include pure networks as a special case. Practical settings in which GN problems arise include more advanced forms of resource allocation, production, distribution, scheduling, capital budgeting, as well as other problem types which will be elaborated on subsequently.

As previously noted, the most effective procedures for modeling and communicating pure network problems are based on viewing these problems as directed graphs. A generalized network problem can also be represented as a directed graph. However, there is an important distinction between arcs in pure network problems and arcs in GN problems. An arc of a generalized network has a *multiplier* associated with it. This situation is illustrated in Figure 3. Arc multipliers are shown in Figure 3 within triangles. As in the pure network case, costs are shown within the squares and bounds are shown within parentheses.

Flow across an arc in a generalized network problem is acted upon by the multiplier so that the amount starting out on an arc will not necessarily be the amount arriving at the opposite end. Specifically, the flow entering the arc is multiplied by the value of the multiplier to produce the quantity of flow leaving the arc. For example, if 3 units start on the arc from node 1 to node 2 in Figure 3, the multiplier of 4 will cause 12 units to arrive at node 2. Likewise, 8 units starting on the arc from node 2 to node 4 will result in − 2 units arriving at node 4 since the multiplier in this case is − 1/4. It is important to keep in mind that *the arc's cost, lower bound, and upper bound refer only to the units of flow entering the arc.*

FIGURE 3

Generalized Network



Applications of generalized networks

As already indicated, generalized networks can successfully model many problems that have no pure network equivalent. This is made possible by the ability to interpret arc multipliers in two ways. First, multipliers can be viewed as modifying the amount of flow of some particular item. By means of flow modification, generalized networks can model such situations as evaporation, seepage, deterioration, breeding, interest rates, sewage treatment, purification processes with varying efficiencies, machine efficiency, and structural strength design. Second, it is possible to interpret the multiplier process as transforming one type of item into another. This interpretation provides a way to model such processes as manufacturing, production, fuel to energy conversions, blending, crew scheduling, manpower to job requirements, and currency exchanges. (See [6], [12], [15], [16] for more detailed discussions.) The following applications (12], [15]) illustrate some practical uses of generalized networks.

A complete water distribution system with losses has been modeled by Bhaumik [4] as a generalized network problem. This model is primarily concerned with the movement of water through canals to various reservoirs. However, the model must also consider the retention of water over several time periods. The multipliers in this case represent the loss effect due to both evaporation and seepage.

Kim [20] has utilized generalized networks to represent copper refining processes. The electrolytic refining procedure, in this case, is modeled by a large d-c electrical network. The arcs are current paths with the multipliers representing the appropriate resistances. In this way, Kim analyzes the effect of short circuits in the refining process.

Charnes and Cooper [6] identify applications of generalized networks for both plastic-limit analysis and warehouse funds-flow models. In plastic-limit analysis, the network is generated by forming the equations for horizontal and vertical equilibrium and by employing a coupling technique. The warehouse funds-flow model is actually a multi-time period model. The arcs are used to represent sales, production, and the inventory holding of both products and cash. The multipliers are introduced to facilitate the conversions between cash and products.

A cash management problem has been modeled as a generalized network by Crum [7]. This model for the multi-national firm incorporates transfer pricing, receivables and payables, collections, dividend payments, interest payments, royalties, and management fees. The arcs represent possible cash flow patterns and the multipliers represent costs, savings, liquidity changes, and exchange rates. Other applications include machine loading problems [6], [8], [26], blending problems [6], [26], the caterer problem [8], [26], and scheduling problems such as production and distribution problems, crew scheduling, aircraft scheduling, and manpower training [6], [8], [26].

Comparison results

In this section we take a closer look at the advances in the realm of solution methods. Better insight into their practical significance is provided by results of empirical tests of the new methods against a leading example of the state-of-the-art in solution systems that does not incorporate the network technology. In particular, we report computational comparisons of the new network codes against APEX-III on a wide array of network problems of varying structures.

These results are not biased by variations in computer hardware: all problems were solved on the same machine. Further, an attempt was made to execute the the codes during comparable time periods. Even with these safeguards, minor differences between two solution times should be statistically ignored and the focus should be on order of magnitude differences. For this reason, the times reported are for large problems so that timing variations become less significant.

Table I contains solution times on 12 network problems using APEX-III on a CYBER-74. The first set of problems consists of assignment problems and the reported network solution times were obtained using the AP-AB code of [3]. The solution times indicate that the new AP-AB code is roughly 200 times faster than APEX-III on assignment problems.

The network code times reported on the transportation and transshipment problems were obtained using the ARC-II code of [2]. Again the new method network solution times are substantially superior (on the order of 130 times faster than APEX-III).

The fourth set of solution times are for generalized network problems. The network code times refer to the NETG code [12]. The relative superiority of network code times to APEX-III is smaller for generalized networks than for pure networks. The code NETG is on the order of 30 times faster than APEX-III on generalized networks; nevertheless, this superiority is dramatic, especially in terms of computer costs for solving such problems.

In addition to improving solution speed, the network processing techniques have the noteworthy advantage of requiring less computer memory to solve a problem. This allows larger problems to be solved without resorting to external storage devices, which can incur significant cost increases due to lengthened computer run times.

Further, the reduced memory requirements enable many computer-based decision systems that would otherwise be excluded from this option to be used in an interactive real-time processing environment.

TABLE I
(times are in billing units)

PROBLEM TYPE	no. of equations	no. of variables	LP Code solution times	cost	Network Code solution times	cost
Assignment	400	1500	231.85	$ 41.73	1.16	$ .21
	400	2250	336.37	$ 60.55	1.34	$ .24
Transportation	200	1300	105.68	$ 19.02	0.94	$ .17
	200	1500	124.53	$ 22.42	1.07	$ .19
	200	2000	164.94	$ 29.69	1.21	$ .22
Transshipment	400	1306	174.83	$ 31.47	1.51	$ .27
	1000	2900	833.63	$150.05	5.28	$ .95
Generalized Networks	250	4000	453.02	$ 81.54	16.65	$3.00
	250	4000	742.61	$133.67	14.74	$2.65
	500	5000	1044.34*	$187.98*	22.55	$4.06
	1000	6000	1633.64*	$294.06*	50.22	$9.04

*Time and cost after 10,000 iterations. Optimal solution not found by LP code.

Yet another important advantage of the network codes is their portability. All of these codes are written in standard FORTRAN IV. Several beneficial consequences result. For example, this portability feature allows easy transfer of the network component of a computer-based decision system to a new computer. It also greatly facilitates imbedding the code as a subroutine within a larger system

A final computational advantage is reduced round-off error. This not only yields improved accuracy and solution reliability, but reduces (or eliminates) the need for reinversion, contributing again to improved efficiency. Taken together, the impressive array of advantages of the network solution codes makes it clear why their use in industry and government applications is rapidly increasing.

Acknowledgement

We are especially grateful to Bud Curtin and Gene Woolsey for comments and suggestions that have improved this paper.

REFERENCES

[1] Analysis, Research, and Computation, Inc., "Development and Computational Testing on a Capacitated Primal Simplex Transshipment Code," ARC Technical Research Report, P. O. Box 4067, Austin, Texas 78765

[2] Barr, R., Glover, F. and Klingman, D., "Enhancements of Spanning Tree Labeling Procedures for Network Optimization," Research Report CCS 262, Center for Cybernetic Studies, The University of Texas at Austin, 1976.

[3] Barr, R., Glover, F. and Klingman, D., "The Alternating Basis Algorithm for Assignment Problems," *Mathematical Programming,* Vol. 13, No. 1, 1–13, August 1977.

[4] Bhaumik, G., *Optimum Operating Policies of a Water Distribution System with Losses,* unpublished dissertation, The University of Texas at Austin, August 1973.

[5] Bradley, G., Brown, G. and Graves, G., "Design and Implementation of Large Scale Primal Transshipment Algorithms," Technical Report NPS55BZBW76091, Naval Postgraduate School, Monterey, California, 1976.

[6] Charnes, A. and Cooper, W., *Management Models and Industrial Applications of Linear Programming,* Vols. I and II, Wiley, New York, 1961.

[7] Crum, R., "Cash Management in the Multinational Firm: A Constrained Generalized Network Approach," Working Paper, The University of Florida, Gainesville, Florida, 1976.

[8] Dantzig, G., *Linear Programming and Extensions,* Princeton University Press, Princeton, New Jersey, 1963.

[9] Evans, J., "A Single Commodity Network Model for a Class of Multicommodity Dynamic Planning Problems," Working Paper, University of Cincinnati, 1975.

[10] Gilliam, G. and Turner, J., "A Profile Analysis Network Model to Reduce the Size of Microdata Files," Working Paper, Office of Tax Analysis, Office of the Secretary of the Treasury, Washington, D. C., 1974.

[11] Gilsinn, J. and Witzgall, C., "A Performance Comparison of Labeling Algorithms for Calculating Shortest Path Trees," NBS Technical Note 772, U. S. Department of Commerce, 1973.

[12] Glover, F., Hultz, J., Klingman, D. and Stutz, J., "Generalized Networks: A Fundamental Computer Based Planning Tool," to appear in *Management Science.*

[13] ———, Karney, D. and Klingman, D., "The Augmented Predecessor Index Method for Locating Stepping Stone Paths and Assigning Dual Prices in Distribution Problems," *Transportation Science,* Vol. 6, No. 2 (1972), 171–179.

[14] ———, ——— and ———, "Implementation and Computational Study on Start Procedures and Basis Change Criteria for a Primal Network Code," *Networks,* Vol. 4, No. 3 (1974), 191–212.

[15] ——— and Klingman, D., "Network Application in Industry and Government," *AIIE Transactions,* Vol. 9, No. 4 (1977), 363–376.

[16] ———, ——— and McMillan, C., "The NETFORM Concept," *Proceedings of the ACM,* 1977.

[17] ———, ——— and Stutz, J., "The Augmented Threaded Index Method for Network Optimization," *INFOR,* Vol. 12, No. 3 (1974), 293–298.

[18] ——— and Mulvey, J., "Equivalence of the 0-1 Integer Programming Problem to Discrete Generalized and Pure Networks," MSRS 75-19, University of Colorado, Boulder, 1975.

[19] Helgason, R., Kennington, J. and Lall, H., "Primal Simplex Network Codes: State-of-the-Art Implementation Technology," Technical Report IEOR 76014, Department of Industrial Engineering and Operations Research, Southern Methodist University, Dallas, 1976.

[20] Kim, Y., "An Optimal Computational Approach to the Analysis of a Generalized Network of Copper Refining Process," Presented at the Joint ORSA/TIMS/AIIE Conference, Atlantic City, New Jersey, 1972.

[21] Collins, M., Cooper, L. and Kennington, J., "Solving the Pipe Network Analysis Problem Using Optimization Techniques," Technical Report IEOR 76008, Southern Methodist University, Dallas, 1976.

[22] "Out-of-Kilter Network Routine," SHARE Distribution 3536, SHARE Distribution Agency, Hawthorne, New York, 1967.

[23] Srinivasan, V. and Thompson, G., "Accelerated Algorithms for Labeling and Relabeling of Trees with Applications for Distribution Problems," *JACM,* Vol. 19, No. 4 (1972), 712–726.

[24] Taha, H., *Operations Research,* The Macmillan Company, New York, 1971.

[25] Tavis, L., Crum, R. and Klingman, D., "Implementation of Large-Scale Financial Planning Models: Solution Efficient Transformations," Research Report CCS 267, Center for Cybernetic Studies, The University of Texas at Austin, 1976.

[26] Wagner, H., *Principles of Operations Research,* Prentice-Hall, Englewood Cliffs, New Jersey, 1969.