Technical Report

A NETWORK PROGRAMMING MODEL WITH NONLINEAR COST FUNCTIONS

by

Shishir K. Mukherjee

INDIAN INSTITUTE OF MANAGEMENT
AHMEDABAD

A NETWORK PROGRAMMING MODEL WITH NONLINEAR COST FUNCTIONS

by

Shishir K. Mukherjee

March 1974

Indian Institute of Management
Ahmedabad

No. 27

To

Chairman (Research)
IIMA

Technical Report

Title of the report ..A NETWORK PROGRAMMING MODEL.......COST FUNCTIONS

Name of the Author ...Shishir K. Mukherjee.......................

Under which area do you like to be classified? ....

ABSTRACT (within 250 words)

In this paper, a network programming model with nonlinear cost functions is described along with computer-based algorithms for the solution of the network model. The network model described is general as it can be applied for the optimization of various physical, economic and social systems, e.g. water supply and wastewater system, traffice and transportation system, solid waste handling system, natural gas and petroleum pipelines, communication systems, etc.

Realistic nonlinear cost functions reflecting economies-of-scale are use for flow through the arcs of the system network. These nonlinear cost functions are usually nonconvex in nature, and various complexities are encountered in minimization problems involving these functions, specifically the difficulty ob obtaining globally optimum solutions.

Two algorithms utilizing approximating iterative schemes developed during an ongoing research project are described in this paper. These algorithms have been applied with success in water supply planning and wastewater treatment and disposal system optimization.

Please indicate restrictions if any that the author wishes to place upon this note ....

See attached paper .

Date 18/3/74....

S.K.M.
Signature of the Author

A NETWORK PROGRAMMING MODEL WITH NONLINEAR COST FUNCTIONS*

Shishir K. Mukherjee

In this paper, a network programming model with nonlinear cost functions is described along with computer-based algorithms for the solution of the network model. The network model described is general as it can be applied for the optimization of various physical, economic and social systems, e.g., water supply and wastewater system, traffic and transportation system, solid waste handling system, natural gas and petroleum pipelines, communication systems, etc.

Realistic nonlinear cost functions reflecting economies-of-scale are used for flow through the arcs of the system network. These nonlinear cost functions are usually non-convex in nature, and various complexities are encountered in minimization problems involving these functions, specifically the difficulty of obtaining globally optimum solutions.

Two algorithms utilizing approximating iterative schemes developed during an ongoing research project are described in this paper. These algorithms have been applied with success in water supply planning and wastewater treatment and disposal system optimization.

Introduction

A large number of optimization problems involving physical and economic systems can be successfully described and solved by optimizing the distribution of flow through a conceptual network as has been evidenced in literature; see [3] for a bibliography. A network consists in general of a collection of elements called nodes or vertices some of which are connected by branches or arcs. The nodes may be divided into three categories - 'sources' at which flow is generated, 'sinks' at which flow is consumed

* Presented at the Fourth Annual Convention of Operational Research Society of India, IIT, Madras, March 2-4, 1972.

and 'intermediate' nodes or junctions at which flow is conserved. The arcs are usually associated with minimum and maximum capacities of flow (in one or both directions) of some commodity per unit time.

The Network Model consists in representing the functioning of a physical or economic system by the concept of flow through a network of nodes and arcs, and determining an optimum design and/or operating plan for the system by finding a minimum cost flow through the system network. A wide variety of systems involving production, transportation, storage, processing and consumption can be represented by the network model. Usually the production centres become the source nodes and the consumption centres become the sink nodes; alternatively the flow enters the system boundary through the source nodes and leaves the system boundary through the sink nodes. Activities such as transportation, storage, processing, etc. become the arcs of the network, their capacity being limited by the plant, equipment or transportation capacity and the unit cost is determined by the process cost. Junction points, nodal transfer points, etc. are described by intermediate nodes.

The flow of one commodity of prime importance is considered through the network. The flow of multiple commodities through network would require more sophisticated modelling techniques. We assume that there is no loss or gain of flow in the network. The case of networks with gains have been considered by Jewell [5] by using arc multipliers. The model described in this paper can be generalized to consider losses or gains during flow through the arcs of the network by using a special algorithm for finding minimum-cost flow.

Some of the specific systems that can be represented by the Network Model are: Urban Water Supply/Waste Treatment and Disposal System, Solid Waste collection, Treatment and Disposal System, Natural Gas and Petroleum Distribution System, Electric Power Transmission System, Communication Systems, General Transportation and Warehousing System, Postal Delivery System, etc.

One of the common problems connected with flows in networks is to determine the maximal flow through a network with specified arc capacities. It is also plausible to assume a cost of flow in each arc of the network such cost being incurred for the creation of the facility (e.g. pipeline highway) represented by the arc and/or its operation (e.g. pumping cost, freight rate). An interesting problem with wide-ranging applications is to find the distribution of flow that minimizes the total cost of transporting a given amount of flow through the network, if feasible. The well-known transportation and transhipment problems fall in this category.

The costs may be linear or nonlinear functions of flow based on physical and economic laws applicable to the situation. The linear cost case has been described extensively in literature and several algorithms are available [1, 3]. A method for convex cost flow problems based on a shortest path algorithm is descibed by Hu [4]. Another algorithm for the convex cost case is descibed by Fillet et al [2] which finds a negative loop in a related network. In a large number of applications, the cost functions are concave reflecting either 'fixed charge' or economies-of-scale or both. Zanguill [10] has given solution methods for certain specially structured networks with concave costs. Sanderson [9] describes a 'branch and bound' algorithm for general piecewise linear nonconvex cost functions. Narayanamurthy [8] also describes a 'branch and bound' method for monotone networks with piecewise linear nonconvex costs based on a method described by Minty [6] for monotone networks.

In this paper, two algorithms utilizing iterative schemes are described for continuous nonlinear cost functions. These algorithms were developed for application to optimal design of water supply, waste-water treatment and disposal systems [7]. The cost functions encountered were generally nonlinear and concave reflecting decreasing marginal cost of facilities with increasing capacity. Minimum concave cost network flow problem is mathematically difficult to solve the main problem being that an algorithm may converge to a solution which may be a local minima but not a global one. Techniques of nonlinear programming will usually converge to such a local minima with no assurance of its being the global one or with no indication regarding existence of other local minima in the solution set.

The first algorithm described in this paper has shown good convergence property but has no device to stop convergence at a local minima. In the second algorithm a lower bounding scheme is developed for the globally optimum solution and the algorithm is directed to converge within a specified range of this value.

Mathematical Formulation of Network Flow Problem

The mathematical formulation of the minimum cost network flow problem for a closed network is given below. The network is called closed in the sense that there is a no-cost return arc of very large capacity connecting the sink node to the source node and the flow in the network is considered as a circulation. In this formulation source, sink and intermediate nodes lose their distinctions and flow must be conserved at all nodes of the network. Only one source and one sink node is considered here as multiple source and sink networks could be easily converted into this type by the addition of appropriate dummy arcs and nodes.

$$\text{Minimize } C = \sum_{(i,j) \in A} c_{ij} (f_{ij}) f_{ij} \tag{1}$$

$$\text{Subject to } l_{ij} \leq f_{ij} \leq u_{ij} \text{ for all arcs } (i,j) \in A \tag{2}$$

$$\text{and } \sum_{i \in N} f_{ij} \quad \sum_{k \in N} f_{jk} = 0 \text{ for all nodes } j \in N \tag{3}$$

where

N is the collection of all nodes in the network

A is the collection of all arcs in the network

(i,j) is a directed arc starting at node i and ending at node j.

f_{ij} is the flow in arc (i,j), $f_{ij} \geq 0$

l_{ij} is the lower bound on flow in arc (i,j), $l_{ij} \geq 0$

u_{ij} is the upper bound on flow in arc (i,j), $u_{ij} \geq l_{ij} \geq 0$

$c_{ij}(f_{ij})$ is the unit cost of flow in arc (i,j) corresponding to a flow f_{ij}

C is the total cost of flow in the network.

Equation (1) states that the total cost of flow over all arcs of the network must be minimized, subject to the arc capacity constraints expressed by equation (2). Equation (3) is the flow conservation constraint.

Usually there are a large number of feasible flow patterns in the network satisfying the arc capacity constraints. The constraints on the network flow due to the demands at sink nodes and the availabilities at the source nodes are also expressed through arc capacities. A highly powerful computational method for solving minimum cost flow problems in networks with constant unit arc costs (representing linear cost functions) is provided by the Ford-Fulkerson "Out-of-kilter" algorithm (3). If equation (1) above is replaced by equation (4) below, where c_{ij} is a constant for each arc (i,j) A. Equations (2), (3) and (4) will represent a minimum cost-flow problem which could be easily solved by the application of the "Out-of-kilter" algorithm even for very large networks consisting of thousands of arcs and nodes.

$$\text{Minimize } C = \sum_{(i,j) \in A} c_{ij} f_{ij} \tag{4}$$

The basic procedure of 'out-of-kilter' algorithm to solve minimum cost network flow problems is explained below. It is clear that equation (2) to (4) define a Linear Programme. Dual variables may be defined π_i for each $i \in N$ for the constraints in equation (3) along with a_{ij} corresponding to the constraints $f_{ij} \leq u_{ij}$ and b_{ij} corresponding to the constraints $l_{ij} \leq f_{ij}$ for each $(i,j) \in A$. The variables π_i are often called 'node prices.' Then the dual linear programming problem corresponding to the minimum cost flow problem with linear cost becomes

$$\text{Maximize} \sum_{(i,j) \in A} (-u_{ij} \; a_{ij} + l_{ij} \; c_{ij}) \tag{6}$$

$$\text{subject to} \quad \pi_i - \pi_j - a_{ij} + b_{ij} = c_{ij} \quad \text{for all } (i,j) \in A \tag{7}$$

$$a_{ij} \geq 0, \quad b_{ij} \geq 0 \quad \text{for all } (i,j) \in A \tag{8}$$

The optimality conditions for complimentary slackness for all arcs $(i,j) \in A$ yield

$$c_{ij} + \pi_i - \pi_j > 0 \quad \text{implies} \quad f_{ij} = l_{ij} \tag{9}$$

$$c_{ij} + \pi_i - \pi_j < 0 \quad \text{implies} \quad f_{ij} = u_{ij} \tag{10}$$

$$c_{ij} + \pi_i - \pi_j = 0 \quad \text{implies} \quad l_{ij} \leq f_{ij} \leq u_{ij} \tag{11}$$

The relationships (9) to (11) represents the necessary and sufficient conditions to be satisfied by a flow solution of the minimum cost flow problem represented by equations (2) to (4) to be optimal.

The out-of-kilter algorithm (3) starting with an arbitrary set of node prices π_i and any set of flows f_{ij} satisfying equation (3) builds flow circulations and/or changes node prices in the network with the objective of fulfilling the optimality criteria expressed in relationships (9) to (11) for each arc in the network. Any arc for which any of those relationships are not satisfied is said to be 'out-of-kilter,' its 'kilter number' specifying the amount of violation. The algorithm uses a labelling routine to send a flow circulation (breakthrough) through that arc in the forward or reverse direction designed to reduce the kilter number of the particular arc while not increasing the 'kilter number' of any other arc in the process. If such labelling is not possible (non-breakthrough) then some of the node prices are changed based on the condition of arcs connecting labelled and unlabelled nodes. This may lead to another flow circulation or breakthrough. This process is continued until all arcs are put in 'kilter' or it is learned that certain arcs cannot all be in kilter simultaneously indicating that the problem is infeasible. If integer value are used for all the variables, costs and flow bounds, it is easily proven that the algorithm terminates in a finite number of steps [3].

Non-linear Network Programming Model

The conceptual system network is developed from the specification and properties of the system under study. Engineering and scientific judgement is needed to design the conceptual system network so that most of the alternative system designs are considered and a solution is technically feasible and implementable. Cost functions are developed for flow through each arc of the conceptual network based on the process that arc represents. Such cost functions become an integral part of the network model for a particular system. The cost functions quite commonly encountered by the author in studying several physical systems had a common property. It was observed that unit costs of flow when plotted against flow on a log-log paper result in linear or piece-wise linear functions. The basic equation representing the unit cost functions is thus derived from the values at two points on the same linear segment of the graph and is represented by

$$C(f) = a \cdot f^{-b}$$

where $b = \dfrac{\log(c_1/c_2)}{f_2/f_2}$; $a = \dfrac{c_1}{f_1^{-b}}$

$c(f)$ = unit cost corresponding to a flow f

f = flow rate

c_1 = unit cost corresponding to a flow rate f_1

c_2 = unit cost corresponding to a flow rate f_2

A representative cost function is shown in Figure 1.

More than two breakpoints are included if there are more than one linear segment in the log-log plot of the cost function. The cost functions described above are concave and usually monotone nondecreasing. The method described in this paper should be applicable to general class of continuous monotone nondecreasing functions.

In certain applications it may be possible to approximate the continuous nonlinear cost functions by piecewise linear cost functions. Using this approach an enlarged network can be constructed with multiple arcs connecting a pair of nodes each such arc having a cost corresponding to the linear segment it represents. In the case of convex costs, a direct application of 'out-of-kilter algorithm will solve the problem whereas in the non-convex case a branch and bound method [9] could be used.

A Nonlinear Network Programming Model was developed which can accommodate continuous nonlinear cost functions. This model uses a linear minimum cost network flow programme (out-of-kilter algorithm) in an iterative pattern, so that unit arc costs (which are dependent on arc flow) are corrected for the next iteration after each flow assignment by out-of-kilter algorithm. The computation is terminated when the total costs do not change beyond a specified small percentage between successive iterations; the termination and iterative rules are slightly different for the two algorithms proposed for the nonlinear network programming model as described in the next two sections. But for both algorithms it was found that often the same optimal solution is repeated in successive iterations when the solution converges indicating a range for arc unit costs within which the optimal solution does not change due to small changes in the unit arc costs.

The model basically consists of three computer programmes controlled through an executive routine. The Preprocessing Programme stores the nonlinear cost functions for each arc of the network and computes the unit cost as a function of flow rate for each arc of the network from basic input data regarding the costs of the process represented by the arc. From initially assumed flow values in the input data unit arc costs are computed and these are used by the 'out-of-kilter' network programme. When new flow values are generated, new unit arc costs are computed and these cost values replace the old ones.

The optimization programme, essentially an "out-of-kilter" algorithm, solves the linearized problem with given unit arc cost data and arrives at a least-cost flow pattern. The Recosting Programme is used to correct the error introduced by linearizing the problem around a given flow assignment. It takes the least-cost solution from the optimization programme and based on this reassigns flows in the input data for the preprocessing programme. The whole computational process is then repeated with the preprocessing programme computing new unit arc costs for the newly assigned flows and the network programme finding a new least-cost flow solution.

The total system cost is computed after each iteration and a check is made using a specific termination rule based on percentage error with total cost. In all cases where the programme was used, convergence to a solution has been obtained within a few (4-5) iterations of the recosting programme.

Network Model Algorithm A

The iterative scheme for Algorithm A are described below. The underlying logic is quite simple. Large values of assumed flow are used at the beginning for computing unit costs which are corrected in subsequent iterations based on the optimal solution of the linearized problem. The unit arc costs in arcs having zero flow arc are kept low so that they could become candidates during subsequent iterations. However, once any arc has a positive flow, its unit cost is corrected to the corresponding value at the next iteration. Though the convergence property of the algorithm is quite fast there is no guarantee that the solution obtained is globally optimum. No indication is available regarding the existence of other minima, no bounds are available for the globally optimum solution. To obtain sufficient assurance that a globally optimum solution has been obtained, the problem can be solved starting at several random beginning solutions to check if the same solution is obtained.

Iterative Scheme for Algorithm A

Loop 1

Step 1 : Enter with assumed flows (1) for each arc. These flow arc chosen so that for each arc with concave cost functions, the maximum possible flow is used consistent with the physical constraints of the network.

Step 2 : Computer linear arc costs (1) using the preprocessing programme and assumed flows (1).

Step 3 : Enter optimization programme (out-of-Kilter solutions) with zero or some feasible flow and find the optimum solution with total cost (1) and optimal flow (1

Step 1: Enter with assumed flow (2) which is computed from assumed flow (1) and optimal flow (1). For the arcs for which optimal flow (1) is zero, set assumed flow (2) equal to assumed flow (1). For the arcs for which optimal flow (1) is positive, set assumed flow (2) equal to optimal flow (1).

Step 2: Computer linear arc costs (2) using the preprocessing programme and assumed flow (2).

Step 3: Enter optimization programme with optimal flow (1) and linear arc costs (2); find the optimal solution with total cost (2) and optimal flow (2).

Step 4: Compute 100% x $\frac{\text{Total cost (2) - Total cost (1)}}{\text{Total cost (2)}}$

If the absolute value of this percentage is less than the specified termination value (1=2%), terminate. Otherwise enter Loop 3.

Loop 3 ...n (where n is the maximum allowable loop). These loops are similar to Loop 2.

Network Model Algorithm B

The iterative Algorithm B described in the following steps is a improvement over the Algorithm A though they are quite similar. The advantage of this algorithm is that for any particular case of the network model with given boundary conditions it provides a lower bound value of the total cost at each loop or iteration. The obtained solution is compared to the lower bound and improvements are made at the next loop to tighten the lower bound. Thus one always knows the maximum error from the globally optimum solution and in subsequent iterations this error is reduced. With luck, the lower bound solution and the model solution may be identical in which case the model solution is a global optima. The lower bound solutions are always globally optimum solutions as they use linear (convex) envelopes of the concave arc cost curves. The behaviour of the solution total costs and the lower bound costs is schematically shown in Figure 2.

Iterative Scheme for Algorithm B

Loop 1

Step 1 : Enter with assumed flows (1) for each arc. These flow values are chosen so that for each arc with concave cost functions the maximum possible flow is used. These flow values are obtained by examining the physical network and various solution modes. One should not put arbitrarily large flow values, as then the lower bound on cost will be too much lower than the globally optimal solution. But care should be taken that these flow values are large enough so that for any arc no subsequent solution flow values do not exceed these assumed flow values.

Step 2 : Compute linear arc costs (1) using the preprocessing programme and assumed flows (1).

Step 3 : Enter optimization programme (out-of-kilter network optimization subroutine) with zero or some feasible flow and find the optimum solution with total cost (1) and optimal flow (1). Total cost (1) is the lower bound cost (1).

Step 4 : Enter preprocessing programme with optimum flow (1) and compute true arc costs (1). Compute corrected total cost (1) by multiplying optimal flow (1) by true arc costs (1).

Step 5 : By definition, corrected cost (1) Total cost (1).

$$\text{Compute } 100\% \times \frac{\text{Corrected Total Cost (1)} - \text{Total Cost (1)}}{\text{Total cost (1)}}$$

If this percentage is less than the specified termination value (1% is a good guess), terminate. Otherwise enter Loop 2.

Loop 2.

Step 1 : Enter with assumed flow (2) which is computed from assumed flow (1) and optimal flow (1). For the arcs for which optimal flow (2) is zero, set assumed flow (2) equal to assumed flow (1). For the arcs for which optimal flow (1) is positive, assumed flow (2) is equal to optimal flow (1).

Step 2 : Compute linear arc costs (2) using the preprocessing programme and assumed flow (2).

Step 3: Enter optimization programme with optimal flow (1) and linear arc costs (2); find the optimal solution with total cost (2) and optimal flow (2). Total cost (2) is the lower bound cost (2) and it can be easily proved that lower bound cost (2) lower bound cost (1).

Step 4 : Enter preprocessing programme with optimum flow (2) and compute true arc costs (2). Compute corrected total cost (2).

Step 5 : Compute percent error. If this error is more than termination value and optimal flow (2) is not equal to optimal flow (1) for all arcs, go to Loop 3; otherwise terminate.

Loop 3 ... n (where n is the maximum allowable loop) These loops are similar to Loop 2.

Conclusions

The Nonlinear Network Model and the associated algorithms discussed here have proved successful in the solution of several optimization studies of physical systems formulated as network flow problems. A research project is being continued to improve upon the model and develop algorithms with better convergence properties towards globally optimal solutions. Experimentation with nonlinear cost functions of different forms are needed with special emphasis on non-convex cost function to study the convergence properties of these algorithms. Theoretical investigation in the working of these algorithms are also being continued. The development of an efficient algorithm for network flow problem with non-convex cost functions will enhance the application of this highly powerful operations research technique to various complex physical, economic and social systems which could be conceputally modelled by networks.

References

1. Busacker, R.G., and Satty, T.L. Finite Graphs and Networks: An Introduction with Applications, McGraw-Hill, N.Y. 1965.

2. Fillet, A., Giraud, P., and Sakarovitch, M., "An Algorithm to Find Minimum Cost Flow in a Network," Operations Research Center, University of California, Berkeley, ORC 70-37, November 1970.

3. Ford, L.R., and Fulkerson, D.R., Flows in Networks, Princeton University Press, N.J. 1962.

4. Hu, T.C., "Minimum-Cost Flows in Convex-Cost Networks," Naval Research Logistics Quarterly, V. 13, No. 1, March 1966, pp. 1-9.

5. Jewell, William S., "Optimal Flow through Networks with Gains," Operations Research, Vol. 10, No. 4, July-August 1962, pp. 476-499.

6. Minty, G.J., "Monotone Networks," Proc. Royal Society A, V. 257, pp. 194-212.

7. Mukherjee, S.K., and Weddle, C.L., "A Network Programming Model for Water Supply System including Desalination and Wastewater Renovation," presented at XVII International Conference of the Institute of Management Sciences, London, July 1-3, 1970.

8. Narayanamurthy, S.G., "Monotone Networks - Application to Transportation-type Problems-I and II, "OPSEARCH, V. 6, No. 3 and 4, September and December 1969, pp. 145-170 and 262-282.

9. Sanderson, R.D., "Enumerative Algorithms for Solving a Class of Network Synthesis Problems," Operations Research Center, University of California.

10. Zanguill, W.I., "Minimum Concave Cost Flows in Certain Networks," Management Science: Theory, Vol. 14, No. 7, March 1968, pp. 429-450.