Optimal Routing in a Packet-Switched Computer Network

DAVID G. CANTOR, SENIOR MEMBER, IEEE, AND MARIO GERLA

Abstract—The problem of finding optimal routes in a packet-switched computer network can be formulated as a nonlinear multicommodity flow problem.

The application of traditional mathematical programming techniques to the solution of the routing problem for reasonably large networks is computationally inefficient. Satisfactory results have been recently obtained with various heuristic techniques; however, such techniques are nonoptimal and subject to several limitations.

The purpose of this paper is to present a method, based on decomposition techniques, which is exact and, at the same time, computationally very efficient. Such a method, originally developed for a computer network application, can be extended to a variety of convex multicommodity flow problems.

Index Terms—Computer networks, decomposition method, mathematical programming, minimum cost flow, multicommodity flow, packet switching, routing.

I. INTRODUCTION

THE optimal routing problem in a computer network consists of the determination of the optimal routing policy, i.e., the set of routes on which packets have to be transmitted in order to optimize a well-defined objective function (e.g., delay, cost, throughput etc.).[1] Under appropriate assumptions, the optimal routing problem can be formulated as a nonlinear multicommodity flow problem [1].

General techniques for solving multicommodity problems can be found in the mathematical programming literature [2], [3]; however, the straightforward application of these techniques to the routing problem in computer networks proves to be computationally cumbersome. In fact, the algorithms for the determination of optimal topology and channel capacities in a computer network require hundreds of optimal routing computations; therefore, an extremely fast routing technique has to be used. For that reason, considerable effort has been spent in developing heuristic techniques [1], [4]. Quite satisfactory results have been obtained and computational efficiency has been greatly improved; however, all of these techniques are affected by various limitations.

Manuscript received April 23, 1973; revised February 1, 1974. This work was supported in part by the Advanced Research Projects Agency of the Department of Defense under Contract DAHC15-73-C-0368, and by the U. S. Atomic Energy Commission under Contract AT(11-1) Gen 10 Project 14.

D. G. Cantor is with the Department of Mathematics, University of California, Los Angeles, Calif.

M. Gerla is with the Network Analysis Corporation, Glen Cove, N. Y.

[1] One can distinguish between routing policies which are determined *a priori* and are time invariant (deterministic policies), and policies which vary in time, according to load and queue fluctuations (adaptive policies) [5]. Here we restrict the analysis to deterministic policies.

In this paper the problem is approached via mathematical programming. The constraint equations are first investigated and some interesting properties are recognized. Taking advantage of these properties, a decomposition method is applied, in a greatly simplified form, and an algorithm for the exact solution is presented. The algorithm is shown to be computationally competitive with the existing heuristic techniques.

II. THE ROUTING PROBLEM

Consider a packet-switched (also referred to as store-and-forward) computer communication network [5]. In such a network, messages are segmented into packets, and each packet traveling from source N_i to destination N_j is "stored" in a queue at each intermediate node N_k, while awaiting transmission, and is sent "forward" to N_l, the next node in the route from N_i to N_j, when channel (k,l) is free. Thus, at each node there are several queues, one for each output channel. Packet flow requirements between nodes arise at random times and packets are of random length; therefore, channel flows, queue lengths, and packet delay are random variables.

Under appropriate assumptions,[2] it is possible to relate the average delay T of a packet traveling from source to destination (the average is over time and over all pairs of nodes) to the average flows in the channels. The result of the analysis is [5]

$$T = (1/\gamma) \sum_{i=1}^{NA} f_i/(C_i - f_i) \tag{1}$$

where

- T total average delay per packet [seconds/packet].
- NN number of nodes, NA = number of arcs.
- r_{ij} average packet rate from source i to destination j [packet/second].
- γ $= \sum_{i=1}^{NN} \sum_{j=1}^{NN} r_{ij}$ = total packet arrival rate from external sources (throughput) [packet/second].
- f_i total bit rate[3] on channel i [bits/second].
- C_i capacity of channel i [bits/second].

The expression of T becomes more complicated when more details are included in the model [6], [7]; the method proposed here applies also to those more general models.

[2] Assumptions: Poisson arrivals at nodes, exponential distribution of packet length, independence of arrival processes at different nodes, independence assumption of service times at successive nodes [5].

[3] f_i is given by the contribution of all packets transmitted over channel i.