Routing in packet-switched communication networks

Hasan Pirkul and Ali Amiri

This paper addresses the problem of selecting a route for every pair of communicating nodes in a packet-switched communication network in order to minimize the average delay encountered by messages. A mathematical programming formulation of the problem is presented. An efficient solution procedure based on a Lagrangean relaxation of the problem is developed. Unlike most previous approaches where the best route for a communicating node pair is restricted to a set of prespecified candidate routes, our approach considers all possible routes for every communicating node pair. Our approach can also be used to generate good feasible routing schemes, whereas other approaches fail to generate feasible schemes beyond moderate levels of traffic. Extensive computational results across a variety of networks are reported. These results indicate that the solution procedure is effective for a wide range of traffic loads.

Keywords: packet-switched communication network, routing problem

During the last two decades the movement of information has become a critical success factor for organizations as globalization and decentralization became more and more pervasive. One of the strategic aspects of managing information for decentralized organizations is the proper design and operation of a telecommunication network. However, it has been pointed out[1] that in many cases, computer networks are characterized by excess line capacity costs and/or queuing and transmission delays of messages. Given a particular topology and link capacities, the queuing and transmission delays can be significantly reduced through proper routing schemes. Here we study the routing problem in packet-switched communication networks.

Most of the commercially available networks such as DATAPAC[2,3], TELENET[4], TRANPAC[5] and TYMNET[6,7] have adopted static or semi-dynamic routing schemes. Even in designing networks which use adaptive routing, a fixed routing is assumed because network configurations optimized with fixed routing are (near) optimal for adaptive routing operations[8]. In static or semi-dynamic routing methods, routes are specified at system generation or session initiation for each source/destination pair. Routing policy can be either bifurcated or nonbifurcated. In bifurcated routing, messages between an origin node and destination node can be sent through several routes. In nonbifurcated routing, there is only one route over which messages between a communicating node pair are sent.

Route selection is a significant factor in determining response time experienced by network users, and has a major effect on efficient utilization of network resources such as node buffers and link capacities. In this paper, we discuss the problem of selecting the optimal set of routes for all communicating node pairs in a network with static, nonbifurcated routing. This routing scheme is commonly used in many network architectures[9]. Given the network topology, link capacities and traffic requirements between communicating node pairs, the problem can be further described as that of selecting, for every source/destination pair, one route over which all messages for that pair of nodes will be routed. The objective is to minimize the average queueing and transmission delay encountered by messages at the network nodes. This delay is due to the finite transmission capacities of links and the resultant queueing at intermediate nodes.

Early research efforts on routing on computer networks have been devoted to developing heuristic sub-optimal algorithms[10–13]. With advances in mathematical programming techniques, improved routing algorithms have been developed. Frank and Chou[10], Cantor and Gerla[14] and Bertsekas[15] studied the problem of optimal route selection in networks with bifurcated routing. Some of the techniques they used are the gradient

College of Business, The Ohio State University, 1775 College Road, Columbus, OH 43210-1399, USA
Paper received: 9 March 1992; revised paper received: 20 February 1993

0140-3664/94/050307-10

projection algorithm, flow deviation and the extremal flows methods. Courtois and Semal[16] developed a heuristic based on a modified version of the flow deviation method to solve the nonbifurcated routing problems.

Gavish and Hantler[9], Narasimhan *et al.*[17] and Tcha and Maruyama[18] have all used solution procedures based on mathematical programming techniques to solve the nonbifurcated routing problem. Gavish and Hantler[9] used Lagrangean relaxation to obtain lower bounds as well as feasible solutions to minimize average message delays. They reported results of computational experiments with reasonable gaps between lower bounds and feasible solutions. Narasimhan *et al.*[17] presented a new formulation for the same problem which led to a new relaxation that was shown to be capable of yielding lower bounds better than those reported by Gavish and Hantler[9]. They report computational experiments that confirm their claims. Tcha and Maruyama[18] discussed a related problem of minimizing the maximum link utilization factor. They described a linear programming bound, outlined a technique to obtain feasible solutions, and then applied them to small problem instances with up to 34 links and 95 communicating node pairs.

Pirkul and Narasimhan[19] extended their model to include reliability considerations. For each communicating node pair a primary and secondary route must be selected from among a set of predetermined candidate routes. The model captures situations where a single link or node failure would divert traffic to the appropriate secondary routes. A mathematical programming formulation was presented and an effective solution procedure based on Lagrangean relaxation of the problem was developed.

These studies[9,17–19] share one shortcoming. In all of them, a set of prespecified candidate routes is assumed to be given for every communicating node pair. Obviously, the quality of the solutions obtained by these methods depends heavily on the choice of the candidate route set generated before the procedure is applied. Gavish and Altinkemer[20] extended the algorithm of Gavish and Hantler[9] to overcome this shortcoming by considering all possible routes for every communicating node pair. This routing scheme is used as an integral part of a procedure to solve the routing and capacity assignment problem. One drawback of their algorithm is that when link utilization exceeds moderate levels the procedure frequently terminates without a feasible routing scheme. In this paper, we present a new formulation of the routing problem and discuss an improved heuristic for generating feasible solutions for even heavily loaded networks. With this new method the gaps between the lower bounds and the feasible solutions measuring the solution quality are generally very small.

Other researchers have also studied the routing and capacity assignment problem[21–24]. Different routing schemes are used as integral parts of their solution procedures. The quality of these procedures can be significantly improved by incorporating some of the results presented in this paper.

PROBLEM FORMULATION

As noted earlier, response time (defined as an average source-to-destination packet delay) is an important factor in the performance of packet-switched communication networks. In these networks, messages of different sizes between pairs of communicating nodes arrive at random intervals. Packets of these messages travel over the network forming queues at intermediate nodes waiting for an outgoing channel to become available. Thus it is possible to model packet-switched communication networks as networks of queues[25].

To formulate the problem of minimizing the average end-to-end delay in the packet-switched communication network, we assume that the network topology, the capacities of the links and the traffic requirements between each pair of communicating nodes are given. We also make the usual assumptions which are used in modelling the queuing phenomena in packet-switched communication networks. Specifically, we assume that nodes have infinite buffers to store messages waiting for transmission links, that the arrival process of messages to the network follows a Poisson distribution, and that message lengths follow an exponential distribution. We further assume that there is no message processing delay at the nodes, that there is only a single class of service for each communicating node pair, and that the propagation delay in the links is negligible. We should point out that as the transmission rate goes up, propagation delay becomes increasingly a larger component of the total delay. Therefore the assumption of negligible propagation delay might not be realistic for very high bandwidth networks. However, many networks are designed for transmission delays of around 250–500 ms. In these networks, using an average distance of 2000 miles, the propagation delay would account for only 11 ms and can be considered negligible. The packet-switched communication network is modelled as a network of independent M/M/1 queues[25,26] in which links are treated as servers with service rates proportional to the link capacities. The customers are messages whose waiting areas are the network nodes.

We use the following notation:

- N set of nodes in the network
- E set of undirected links in the network
- M set of communicating node pairs
- A^m message arrival rate for communicating node pair $m \in M$
- $O(m)$ source node for communicating node pair $m \in M$
- $D(m)$ destination node for communicating node pair $m \in M$

$\frac{1}{\mu}$ average message length

C_{ij} capacity of link $(i, j) \in E$

The decision variables are:

$$Y_{ij}^m = \begin{cases} 1 & \text{if the route for communicating node pair } m \text{ traverses link } (i,j) \text{ in the direction of } i \text{ to } j \\ 0 & \text{otherwise} \end{cases}$$

X_{ij}^m = flow of communicating node pair m on link (i,j)

If ρ is defined as:

$\rho = \frac{1}{\mu} \sum_{m \in M} A^m$, then the problem can be formulated as:

Problem P

$$Z_P = \text{Min } \frac{1}{\rho} \sum_{(i,j) \in E} \frac{\sum_{m \in M} X_{ij}^m}{C_{ij} - \sum_{m \in M} X_{ij}^m} \tag{1}$$

subject to:

$$\sum_{j \in N} Y_{ij}^m - \sum_{j \in N} Y_{ji}^m = \begin{cases} 1 & \text{if } i = O(m) \\ -1 & \text{if } i = D(m) \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in N \text{ and } m \in M \tag{2}$$

$$\frac{1}{\mu} A^m (Y_{ij}^m + Y_{ji}^m) \leqslant X_{ij}^m \quad \forall (i,j) \in E \text{ and } m \in M \tag{3}$$

$$\frac{1}{\mu} \sum_{m \in M} A^m (Y_{ij}^m + Y_{ji}^m) \leqslant \sum_{m \in M} X_{ij}^m \quad \forall (i,j) \in E \tag{4}$$

$$X_{ij}^m \leqslant \frac{1}{\mu} A^m \quad \forall (i,j) \in E \text{ and } m \in M \tag{5}$$

$$\sum_{m \in M} X_{ij}^m \leqslant C_{ij} \quad \forall (i,j) \in E \tag{6}$$

$$X_{ij}^m \geqslant 0 \quad \forall (i,j) \in E \text{ and } m \in M \tag{7}$$

$$Y_{ij}^m \in (0, 1) \quad \forall (i,j) \in E \text{ and } m \in M \tag{8}$$

In this formulation, the objective function minimizes the average queuing delay for messages. The queuing delay in link (i,j) is $1/(\mu C_{ij} - \lambda_{ij})$ where λ_{ij} is the arrival rate of messages to link (i,j). The average end-to-end delay in the network can be estimated as the weighted sum of the expected delays of the links in the network. Constraint set (2) contains the flow conservation equations which define a route (path) for each communicating node pair m. This route is used to send messages between $O(m)$ and $D(m)$. Constraint set (3) links together the X_{ij}^m and Y_{ij}^m variables. They ensure that the flow for communicating node pair m on link (i,j) is at least equal to the traffic requirement for that pair if its assigned route uses link (i,j); constraints in set (3) hold as equalities at the optimum. Constraints in set (4) can be seen as the aggregate form of the constraints in set (3). Even though these constraints are redundant in problem P, they are helpful in obtaining better lower bounds in the Lagrangean relaxation suggested in the next section. Constraint set (5) guarantees that the flow for communicating node pair m on arc (i,j) does not exceed its traffic requirement. Constraint set (6) represents the capacity constraints on the links. Constraint set (7) restricts the X_{ij}^m variables to be nonnegative, and constraint set (8) enforces integrality conditions on Y_{ij}^m.

By allowing the best route for each communicating node pair to be chosen from the set of all possible routes, our solution method based on the above formulation eliminates a shortcoming that the other methods suffer from[9, 17] which is the theoretical possibility of generating lower bound values that are higher than the optimal solution to the original routing problem when all possible routes are considered.

Our formulation is similar to that used by Gavish and Altinkemer[20]. It can be seen as a disaggregate formulation of the routing problem solved therein. If we were to drop constraint sets (3) and (5) and substitute the terms $\sum_{m \in M} X_{ij}^m$ by variables X_{ij}, representing the total flows on links (i,j), we would obtain the formulation in Gavish and Altinkemer. The variable set X_{ij}^m together with constraint sets (3) and (5) is introduced to represent flows between each pair of communicating nodes separately. This disaggregate formulation leads to better lower bounds and feasible solutions at the expense of added computational effort, as shown later. Other researchers have also observed a similar effect regarding the disaggregation of flows within the context of other classes of problems[27 29].

LAGRANGEAN RELAXATION OF THE PROBLEM

Problem P is a combinatorial optimization problem with a nonlinear objective function. Problems studies in Gavish and Hantler[9] and Narasimhan *et al.*[17] are special cases of problem P, and are known to be NP-complete. Since problem P is a nonlinear mixed integer programming problem, and since a typical problem contains thousands of variables and constraints, it is difficult to solve this problem to optimality. We propose, instead, a composite upper and lower bounding procedure based on a Lagrangean relaxation of the problem. Consider the Lagrangean relaxation of problem P obtained by dualizing constraint set (3) and (4) using nonnegative multipliers α_{ij}^m and β_{ij} for all $(i,j) \in E$ and $m \in M$, respectively.

Problem L

$$Z_L = \text{Min}\ \frac{1}{\rho} \sum_{(i,j)\in E} \frac{\sum_{m\in M} X_{ij}^m}{C_{ij} - \sum_{m\in M} X_{ij}^m} - \sum_{(i,j)\in E} \sum_{m\in M} (\alpha_{ij}^m + \beta_{ij}) X_{ij}^m + \frac{1}{\mu} \sum_{m\in M} \sum_{(i,j)\in E} A^m (\alpha_{ij}^m + \beta_{ij})(Y_{ij}^m + Y_{ji}^m) \quad (9)$$

subject to constraint sets (2), (5)–(8).

Problem L can be decomposed into two subproblems:

Problem L1

$$Z_{L1} = \text{Min}\ \frac{1}{\rho} \sum_{(i,j)\in E} \frac{\sum_{m\in M} X_{ij}^m}{C_{ij} - \sum_{m\in M} X_{ij}^m} - \sum_{(i,j)\in E} \sum_{m\in M} (\alpha_{ij}^m + \beta_{ij}) X_{ij}^m \quad (10)$$

subject to constraint sets (5)–(7), and:

Problem L2

$$Z_{L2} = \text{Min}\ \frac{1}{\mu} \sum_{m\in M} \sum_{(i,j)\in E} A^m (\alpha_{ij}^m + \beta_{ij})(Y_{ij}^m + Y_{ji}^m) \quad (11)$$

subject to constraint set (2) and (8).

Problem $L1$ can be further decomposed into $|E|$ subproblems (one for each link) as follows:

$$\text{Min}\ \frac{1}{\rho} \frac{\sum_{m\in M} X_{ij}^m}{C_{ij} - \sum_{m\in M} X_{ij}^m} - \sum_{m\in M} (\alpha_{ij}^m + \beta_{ij}) X_{ij}^m \quad (12)$$

subject to:

$$X_{ij}^m \leqslant \frac{1}{\mu} A^m \quad \forall m \in M \quad (13)$$

$$\sum_{m\in M} X_{i,j}^m \leqslant C_{ij} \quad (14)$$

$$X_{ij}^m \geqslant 0 \quad \forall m \in M \quad (15)$$

Similarly, problem $L2$ can be further decomposed into $|M|$ subproblems (one for each sourced/destination pair) as follows:

$$\text{Min}\ \frac{1}{\mu} \sum_{(i,j)\in E} A^m(\alpha_{ij}^m + \beta_{ij})(Y_{ij}^m + Y_{ji}^m) \quad (16)$$

subject to:

$$\sum_{j\in N} Y_{ij}^m - \sum_{j\in N} Y_{ji}^m = \begin{cases} 1 & \text{if } i = O(m) \\ -1 & \text{if } i = D(m) \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in N \quad (17)$$

$$Y_{ij}^m \in (0,1) \quad \forall (i,jJ) \in E \quad (18)$$

We note that the Lagrangean problem (L) does not satisfy the integrality property[30,31] because the linear programming relaxation of (L) does not necessarily have integer solution. Thus, our relaxation of problem (P) can theoretically give a lower bound which is at least as good as, and possibly better than, the linear programming relaxation of (P).

SOLUTION TECHNIQUES

This section presents the methods used to solve the Lagrangean subproblems and a heuristic solution procedure to obtain feasible solutions to problem (P). The subgradient optimization procedure used to generate the Lagrangean multipliers is also described in this section.

Solution of Problem *L1*

For any given vector of multipliers α_{ij}^m and β_{ij}, each of the $|E|$ subproblems of $L1$ is a continuous knapsack problem with nonlinear objective function that can be solved using the following greedy type procedure:

Procedure-LAG1

Step 1: Reorder the X_{ij}^m variables by sorting them in nonincreasing order of α_{ij}^m; assume that the variables are reindexed in this order. Let $m = 0$.

Step 2: Let $m = m + 1$ and set

$$X_{ij}^m = \begin{cases} X_0 & \text{if } \alpha_{ij}^m + \beta_{ij} > 0 \text{ and } X_0 > 0 \\ 0 & \text{otherwise} \end{cases}$$

where

$$X_0 = \min\left\{ \frac{1}{\mu} A^m, (C_{ij} - S) - \left[\frac{C_{ij}/\rho}{\alpha_{ij}^m + \beta_{ij}}\right]^{1/2} \right\}$$

and:

$$S = \sum_{k<m} X_{ij}^k$$

Step 3: If $m = |M|$ stop; If $X_{ij}^m < \frac{1}{\mu} A^m$ then stop and set $X_{ij}^k = 0$ for $k = m+1, \ldots, |M|$. Otherwise go to Step 2.

The value assigned to X_{ij}^m in Step 2 is determined in such a way that it decreases the objective function of the subproblem the most.

The subproblems of problem *L1* are different from those corresponding to the routing problem discussed in Gavish and Altinkemer[20]. In the former, the total flow passing through each link depends on the individual communicating node pairs, because the objective function of each subproblem contains terms related to the communicating node pairs and constraint set (5) is included in the model. In the latter, the total flow is determined independently of the communicating node pairs because no disaggregate constraints similar to those in sets (3) and (5) are used in the formulation of the routing problem. It is this level of disaggregation included in our formulation that leads to better feasible solutions and tighter lower bounds, but requires more computational time.

Solution of Problem *L2*

Each subproblem of problem *L2* can be solved as a shortest path problem from $O(m)$ to $D(m)$ with $(\alpha_{ij}^m + \beta_{ij})$ as the nonnegative costs on the links. In our study, Dijkstra's algorithm[32] is used for solving them. These subproblems are similar to those in Gavish and Hantler[9], with one major difference. In Gavish and Hantler[9] the 'lengths' of the links are the same for all communicating node pairs, hence all the subproblems can be solved with one run of Floyd's algorithm[32] which finds shortest paths between all pairs of nodes simultaneously. In our subproblems, the 'length' of a link depends on the commodity flowing through it. This requires that shortest paths be determined separately for each communicating node pair. This is a direct result of the disaggregated treatment of the flows. Even though it is more time consuming, it results in better solutions and helps solve problems that could not be solved using the aggregated flow formulation of the problem.

Complexity of solving Problem *L*

As discussed in the previous section, problem *L1* of the Lagrangean problem *L* can be separated into $|E|$ subproblems (one for each link). Procedure-LAG1 is used to solve each subproblem independently. The main operation in Procedure-LAG1 is the sorting of the communicating node pairs with complexity $O\{|M|\log|M|\}$. Therefore, the complexity of solving problem *L1* is $O\{|E||M|\log|M|\}$. Similarly, problem *L2* of the Lagrangean problem *L* can be separated in $|M|$ subproblems (one for each communicating node pair). Each subproblem is a shortest path problem, and can be solved in $O\{|N|^2\}$ using Dijkstra's algorithm. Thus, the complexity of solving problem *L2* is $O\{|M||N|^2\}$. Therefore, the complexity of solving problem *L* is $O\{|M|\max(|E|\log|M|, |N|^2)\}$.

Subgradient optimization algorithm

If we let $Z_L(\alpha, \beta)$ be the value of the Lagrangean function with a multiplier vector (α, β), then the best bound using this relaxation is derived by calculating:

$$Z_L(\alpha*, \beta*) = \underset{(\alpha, \beta)}{\text{Max}} \{Z_L(\alpha, \beta)\}$$

For simplicity of presentation, we let ω represent (α, β) in the rest of this section. The challenging issue in generating good bounds using Lagrangean relaxation is to find a good set of multipliers. This is in general known to be a very difficult task[31]. In practice, a good but not necessarily optimal set of multipliers is often derived using iterative methods such as the subgradient optimization method and various multiplier adjustment methods, known as *ascent* (*descent*) methods[30].

In this study, we use the subgradient optimization method to search for 'good' multipliers. The subgradient method is a modified version of the gradient method in which subgradients replace gradients[33]. Given an initial multiplier vector ω^0, a sequence of multipliers is generated by updating the vector at the iteration k using the formula:

$$\omega^{k+1} = \omega^k + t_k(AX^k - b)$$

where ω^{k+1} and ω^k are the multiplier vectors at iterations $k + 1$ and k, respectively, X^k is the optimal solution to the Lagrangean Problem L with multiplier vector ω^k, t_k is a positive scalar step size, and $AX^k \leqslant b$ is the set of constraints being relaxed. It is well known that lim sup $Z_L(\omega^k)$ converges to $Z_L(\omega*)$ if $t_k \to 0$ and $\sum_{k=0}^{\infty} t_k \to \infty$[34]. Since in general these conditions are very difficult to satisfy, this method is always used as a heuristic. In this study, we used the following step size that has been found to perform satisfactorily in practice:

$$t_k = \lambda_k(Z_f - Z_L(\omega^k)) / \|AX^k - b\|^2$$

where Z_f is the value of the best feasible solution found so far, and λ_k is scalar satisfying $0 \leqslant \lambda_k \leqslant 2$ and $\|\cdot\|$ stands for the Euclidian norm. This scalar is set to 2 at the beginning of the algorithm, and is halved whenever the bound does not improve in 20 consecutive iterations. The algorithm is terminated after a specified number of iterations unless an optimal solution is reached before that point. The algorithm is also terminated if the gap between the best lower bound and the best feasible solution found is less than 0.1% of the best lower bound, or the best lower bound does not improve in 100 consecutive iterations by at least 0.01%.

Initial multipliers can be set equal to zero. Here we started the procedure by setting $\beta_{ij} = 1/(TC_{ij})$ and $\alpha_{ij}^m = 0$ for all (i,j) and m. With these initial values, $Z_{L1} = 0$ and $Z_{L2} > 0$ and therefore $Z_L > 0$. This is clearly better than setting the initial multipliers $\alpha_{ij}^m = 0$ and $\beta_{ij} = 0$ for all (i,j) and m.

Heuristic solution methods

In this section we introduce a heuristic solution procedure for solving problem P. This is a two phase procedure which first generates an initial routing schedule (possibly infeasible), and then improves this schedule by reallocating traffic from overloaded links that are lightly utilized. This procedure is used to generate a feasible solution which is used as the starting solution for the subgradient optimization procedure. Additionally, at every iteration of the subgradient optimization procedure, feasibility of the Lagrangean solution is checked. In fact, the improvement phase of the heuristic procedure can be used at every iteration of the subgradient procedure to either convert an infeasible solution to a feasible solution or improve a feasible solution. We have chosen not to do this as it is found to be very time consuming. Instead, the improvement phase is applied at the termination of the subgradient search to the best feasible solution generated during the search. In our computational experiments this step typically leads to significant improvement in solution quality, as will be seen in the computational results reported in the next section.

The heuristic used to generate an initial feasible solution can be described as follows:

Procedure-Init

Step 1: For each communicating node pair determine a route with the minimum number of links.

Step 2: If for every link the total flow on the link does not exceed the link capacity, then a feasible solution is at hand (stop); otherwise go to Step 3.

Step 3: Pick the most violated link (i,j). Among all communicating node pairs using that link, reroute the traffic requirement of the communicating node pair k which, using the 'alternative path' would decrease the following total cost function the most:

$$\sum_{(i,j)\in E} cost(i,j)$$

where

$$cost(i,j) = \begin{cases} \dfrac{1}{\rho}\dfrac{\sum_{m\in M} X_{ij}^m}{C_{ij} - \sum_{m\in M} X_{ij}^m} & \text{if } \sum_{m\in M} X_{ij}^m < C_{ij} \\[2ex] \mathscr{M}\dfrac{\sum_{m\in M} X_{ij}^m}{C_{ij}} & \text{otherwise} \end{cases}$$

($\mathscr{M}$ is a large positive number.)

cost (i,j) represents the contribution of link (i,j) to the objective function value when the flow on link (i,j) does not exceed its capacity. However, when the capacity of link (i,j) is exceeded, $cost(i,j)$ represents a penalty term proportional to the violation of the capacity restriction of link (i,j).

The artificial cost function $\sum_{(i,j)\in E} cost$ (i,j) decreases when the capacity utilization of that link (i,j) and possibly of other links on the original route for commodity k decreases. The 'alternative path' is the path from $O(k)$ to $D(k)$ such that the most utilized link on the path has a lower utilization than the most utilized link on any other path from $O(k)$ to $D(k)$. The problem of finding such a path is known as the *bottleneck shortest path problem*, and can be solved using a modified version of Dijkstra's algorithm[32]. If the traffic requirement of no communicating node pair can be rerouted, then the algorithm stops, since no initial feasible solution can be obtained; otherwise, repeat Step 3 until no link capacity is violated.

Even though in Step 3 of Procedure-Init the 'improvement measure' is strictly used to find a single feasible solution, it can be used to improve a given feasible solution. Indeed, at the termination of the subgradient optimization procedure the best feasible solution is further improved utilizing Step 3 of Procedure-Init with minor modifications. The only modifications necessary to Step 3 are: (1) to consider 'the most utilized link' rather than 'the most violated link' and (2) to change the stopping criterion to 'repeat step 3 until no further improvement can be effected'.

OPTIMIZATION PROCEDURE APPLICATION

The solution procedures presented in the previous section were coded in Pascal. A number of computational experiments were performed using IBM-3081D running under MVS/XA 2.1.7.

A variety of previously used problems were utilized in the tests. We studied the four topologies shown in *Figures 1–4*, *viz.* ARPA, OCT, USA and RING. These networks and traffic parameters are similar to those tested by Gavish and Hantler[9], Narasinham *et al.*[17] and Gavish and Altinkemer[20]. In all four networks each node communicates with every other node. In the ARPA network (*Figure 1*) there were 420 communicating node pairs with four messages per second being sent along the chosen route. The corresponding values were 650 and one for OCT (*Figure 2*), and 650 and four for USA (*Figure 3*) and 992 and one for RING (*Figure 4*).

Tables 1–4 summarize the results of the computational tests for ARPA, OCT, USA and RING, respectively. The results with our solution procedures as well as with the procedure described in Gavish and Altinkemer[20] are reported. In these tests the original coding of

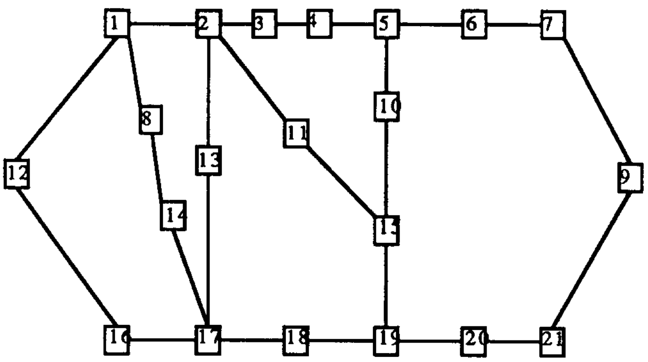

Figure 1 ARPA network. ——: 50 Kbits/s

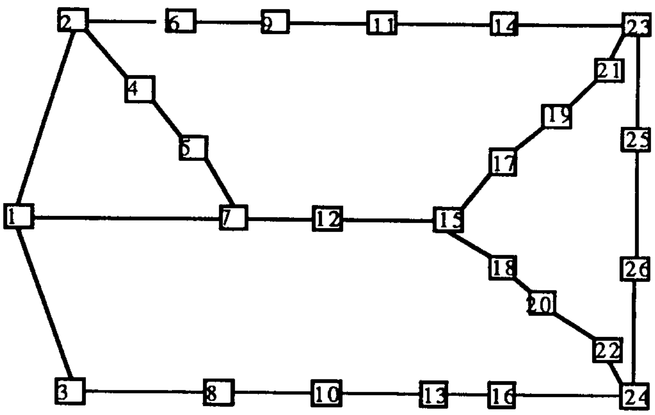

Figure 2 OCT network. ——: 50 Kbits/s

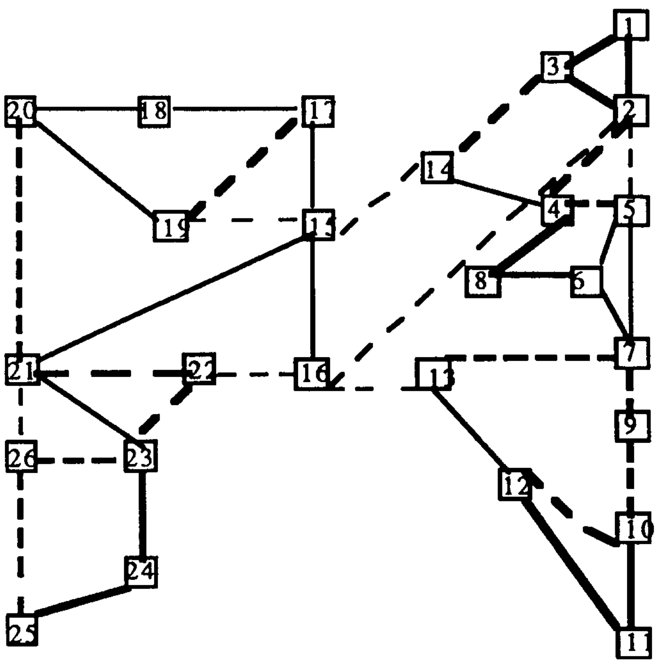

Figure 3 USA network. ——: 50 Kbits/s; ——: 25 Kbits/s; - -: 15.2 Kbits/s; - -: 9.6 Kbits/s

this procedure provided by the authors is used. The results of the experiments are described by providing the values of average message length, best feasible solution (upper bound), best Lagrangean bound (lower bound), the 'gap' between the upper and lower bounds, and the maximum and average percentage link utilizations. The mean message length is measured in bits and the lower and upper bounds in milliseconds.

In all the cases, our solution procedure produced better feasible solutions and smaller 'gaps' between the upper and lower bounds than those produced by the procedure reported by Gavish and Altinkemer[20]. The improvement in the feasible solutions obtained by our procedure is between 0.1% and 13.5%. But more importantly, the procedure reported in Gavish and Altinkemer[20] does not generate feasible solutions beyond moderate levels of traffic, whereas our procedure finds good feasible solutions even for heavily loaded networks (to be exact, it did not locate feasible solutions beyond average link utilization of 51% for ARPA, 71.4% for OCT, 34.5% for USA and 24.2% for RING networks). This improved effectiveness is obtained at the expense of increased computational time. For the test problems one iteration of the procedure reported in Gavish and Altinkemer[20] takes between 0.05 and 0.2 s, whereas our procedure takes between 1 and 5 s.

These results clearly have significant implications not only for solving the routing problem, but also the capacity assignment problem. Since the routing problem is a subproblem in the overall solution procedure for the capacity assignment problem, the ability to find routing schedules with higher link utilization levels implies that better capacity assignments can be made.

CONCLUSION

In this paper, we studied the nonbifurcated static routing problem in packet-switched communication networks. In this problem, a route for every pair of communicating nodes is to be identified to minimize the mean delay faced by messages. A mathematical programming formulation of the problem is presented. The new approach developed in this paper incorporates significant improvements over previous methods proposed[9,17,20]. Two methods[9,17] suffer from the theoretical limitation that lower bounds generated might be higher than the optimal solution values. This limitation is the result of restricting the best routes for communicating node pairs to be chosen from a predetermined set of candidate routes. The new approach eliminates this limitation by considering all possible routes for every communicating node pair. Even though the method described in Gavish and Altinkemer[20] overcomes this limitation, it fails to generate feasible routing schemes when link utilization exceeds moderate levels. The new approach based on Lagrangean relaxation generates good feasible solutions for even heavily loaded networks. This approach has been tested on the same networks used by other researchers[9,17,20]. In all the cases, our solution approach produced better feasible solutions and smaller gaps between the upper

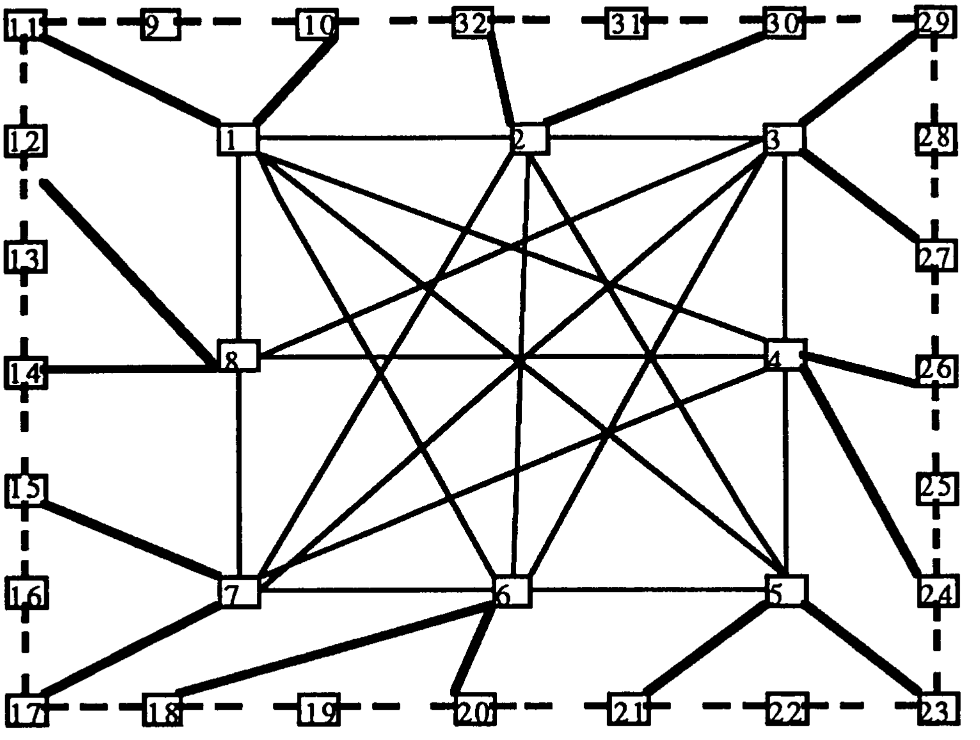

Figure 4 RING network. - -: 35 Kbits/s; ——: 33 Kbits/s; ——: 30 Kbits/s

Table 1 Summary of computational results for ARPA network

						Gavish and Altinkemer's Method				
Message length	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization
75	8.53	8.53	0.02	52.80	33.40	8.54	8.53	0.18	54.00	33.41
100	14.87	14.86	0.11	65.60	45.20	15.06	14.86	1.37	68.80	44.86
112	19.65	19.61	0.20	71.70	50.80	20.02	19.61	2.10	75.30	50.57
113	20.11	20.09	0.10	74.10	51.20	20.53	20.01	2.61	75.90	51.03
125	27.45	27.41	0.13	80.00	56.80	no feasible solution				
130	31.80	31.72	0.25	83.20	59.10	no feasible solution				
140	45.43	45.01	0.94	89.60	63.80	no feasible solution				

[a]Gap = 100* (Upper Bound - Lower Bound)/Lower Bound

Table 2 Summary of computational results for OCT network

						Gavish and Altinkemer's Method				
Message length	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization
225	36.25	36.24	0.02	57.60	44.00	36.28	36.23	0.16	56.70	44.90
250	44.64	44.62	0.04	62.00	49.00	44.66	44.61	0.10	63.00	49.00
275	55.11	55.09	0.03	68.20	54.10	55.13	55.08	0.10	68.20	54.12
300	68.61	68.57	0.06	73.20	59.00	68.68	68.57	0.16	72.00	59.03
325	86.98	86.82	0.18	78.00	63.90	87.03	86.81	0.25	79.30	63.97
350	113.43	113.23	0.18	84.00	68.90	113.71	113.21	0.44	84.00	68.94
360	127.78	127.58	0.16	85.00	70.90	128.26	127.51	0.59	86.40	70.92
362	131.23	130.76	0.36	85.40	71.30	131.56	130.76	0.61	86.90	71.40
375	156.25	155.58	0.44	88.50	74.00	no feasible solution				
400	242.12	240.75	0.57	92.80	79.30	no feasible solution				

[a]Gap = 100* (Upper Bound - Lower Bound)/Lower Bound

and lower bounds than those produced by the procedure proposed by Gavish and Altinkemer. The improvement in the feasible solutions obtained by our procedure is between 0.1% and 13.5%. These results show that our procedure is very effective for the variety of networks used in this study.

The approach described in this paper is not applicable to packet-switched communication networks with reli-

Table 3 Summary of computational results for USA network

						Gavish and Altinkemer's Method				
Message length	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization
25	5.25	5.24	0.11	48.00	23.30	5.33	5.24	1.71	48.30	23.70
30	6.94	6.93	0.17	55.70	28.20	7.15	6.86	4.12	57.20	29.30
31	7.33	7.31	0.16	56.50	29.20	7.63	7.08	7.81	58.10	30.30
32	7.75	7.71	0.47	59.40	30.30	8.10	6.90	17.49	60.40	31.20
33	8.16	8.13	0.38	60.20	31.30	9.42	2.17	333.69	76.95	34.50
35	9.08	9.03	0.60	63.80	33.30	no feasible solution				
40	11.90	11.71	1.56	71.70	38.50	no feasible solution				
45	15.74	15.47	1.71	80.60	43.50	no feasible solution				
50	22.09	21.56	2.46	89.60	48.80	no feasible solution				

[a]Gap = 100* (Upper Bound - Lower Bound)/Lower Bound

Table 4 Summary of computational results for RING network

						Gavish and Altinkemer's Method				
Message length	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization	Upper bound	Lower bound	Gap[a]	Maximum % link utilization	Average % link utilization
150	17.73	17.61	0.66	33.60	22.30	17.86	17.61	1.39	33.91	22.30
160	19.34	19.19	0.79	36.80	23.80	19.50	19.18	1.64	37.10	23.90
161	19.52	19.51	0.05	38.30	24.10	19.68	19.34	1.76	38.40	24.15
162	19.67	19.51	0.84	38.30	24.10	19.84	19.51	1.71	38.42	24.20
200	26.45	26.21	0.92	43.60	29.80	no feasible solution				
250	37.51	37.11	1.06	54.50	37.30	no feasible solution				
300	52.38	51.55	1.60	63.60	45.00	no feasible solution				
325	61.90	60.71	1.96	67.00	48.90	no feasible solution				
350	73.24	71.64	2.23	72.10	52.90	no feasible solution				
375	87.28	84.92	2.78	68.60	57.40	no feasible solution				

[a]Gap = 100* (Upper Bound - Lower Bound)/Lower Bound

able network components. One extension is to incorporate reliability issues in the routing problem. The new model should capture the possibilities of link or node failures. We have been successful in developing a model which considers the possibility of a single link failure and allows routes for communicating node pairs to be chosen from among all possible routes. Another extension is to design simultaneously the local access network and the packet-switched communication network. More research is needed to solve this difficult problem.

REFERENCES

1 LeBlanc, L and Simmons, R V 'Continuous models for capacity design of large packet-switched telecommunication networks', *ORSA J. Comput.*, Vol 1 No 4 (1989) pp 271–286

2 Cahin, P M 'Datapac network protocols', *Proc. 3rd Int. Conf. on Comput. Commun.*, Toronto, Canada (1976) pp 150–155

3 McGibbon, C H, Gibbs, H and Young, S C K 'DATAPAC – Initial experience with a commercial packet network', *Proc. 4th Int. Conf. Comput. Commun.*, Kyoto, Japan (1978) pp 103–108

4 *TELENET Communication Corporation – Packet Switching Network*, Auerbach, New York (1978)

5 Despres, R and Pichon, G 'The TRANSPAC network status report and perspectives', *Proc. Online Conf. on Data Networks – Development and Use*, London, UK (1980) pp 209–232

6 Rajaram, A 'Routing in TYMNET', *Proc. Euro. Computation Congress* (1978)

7 Tymes, L R W 'Routing and flow control in TYMNET', *IEEE Trans. Commun.*, Vol 29 No 8 (1981) pp 392–398

8 Gerla, M 'Deterministic and adaptive routing policies in packet switched networks', *ACM-IEEE 3rd Data Commun. Symp.*, Florida (November 13–15 1973)

9 Gavish, B and Hantler, S 'An algorithm for optimal route selection in SNA networks', *IEEE Trans. Commun.*, Vol 31 No 10 (1983) pp 1154–1161

10 Frank, H and Chou, W 'Routing in computer networks', *Networks*, Vol 1 (1971) pp 99–122

11 Fratta, L, Gerla, M and Kleinrock, L 'The flow deviation method – An approach to store-and-forward communication network design', *Networks*, Vol 3 (1973) pp 97–133

12 Gerla, M *The Design of Store-and-Forward (S/F) Networks for Computer Communications*, PhD Thesis, University of California, Los Angeles (1973)

13 Maruyama, K, Fratta, L and Tang, D T 'Flow assignment algorithm for computer communication network design with different classes of packets', *Proc. Symposium on Computer Networks: Trends and Applications*, Gaithersburg, MD (November 17 1976)

14 Cantor, D G and Gerla, M 'The optimal routing of messages in a computer network via mathematical programming', *IEEE Comput. Conf. Proc.*, San Francisco, CA (September 1972)

15 Bertsekas, D P 'A class of optimal routing algorithms for communications networks,' *Proc. Int. Conf. Circuits Comput.*, Atlanta, GA (November 1980)

16 Courtois, P J and Semal, P 'An algorithm for the optimization of nonbifurcated flows in computer communication networks', *Perf. Eval. J.*, Vol 1 (1981) pp 139–152

17 Narasimhan, S, Pirkul, H and De, P 'Route selection in backbone data communication networks', *Comput. Networks & ISDN Syst.*, Vol 15 No 15 (1988) pp 121–133

18 Tcha, D and Maruyama, K 'On the selection of primary paths for a communication network', *Comput. Networks & ISDN Syst.*, Vol 9 (1985) pp 257–265

19 Pirkul, H and Narasimhan, S 'Primary and secondary route selection in backbone data communication networks', Working Paper, College of Business Ohio State University (1987)

20 Gavish, B and Altinkemer, K 'Backbone network design tools with economic tradeoffs', *ORSA J. Comput.*, Vol 2 No 3 (1990) pp 226–252

21 Gershet, H and Weihmayer, R 'Joint optimization of data network design and facility location', *IEEE J. Selected Areas in Commun.*, Vol 8 No 9 (1990) pp 149–152

22 Monma, C and Shallcross, D 'Methods for designing communication networks with certain two-connected survivability constraints', *Operat. Res.*, Vol 37 (1989) pp 531–541

23 Ng, T and Hoang, D 'Joint optimization of capacity and flow assignment in a packet-switched communications network', *IEEE Trans. Commun.*, Vol 35 No 2 (1987) pp 202–209

24 Rosenberg, E 'A nonlinear programming heuristic for computing optimal link capacity capacities in a multi-hour alternate routing communications network', *Operat. Res.*, Vol 35 (1987) pp 354–364

25 Kleinrock, L *Communications Nets: Stochastic Message Flow and Delay*, Dover, New York (1964)

26 Kleinrock, L *Queuing Systems*, Vols 1/2, Wiley-Interscience, New York (1975, 1976)

27 Balakrishnan, A, Magnanti, T L and Wong, R T 'A dual-ascent procedure for large scale uncapacitated network design,' *Operat. Res.*, Vol 37 (1989) pp 716–740

28 Wong, R T 'Integer programming formulations of the Traveling Salesman Problem', *Proc. IEEE Int. Conf. Circuits and Comput.* (1980) pp 149–152

29 Wong, R T 'A dual based procedure for solving Steiner tree problems on a directed graph', *Math. Program.*, Vol 28 (1984) pp 271–287

30 Bazaraa, M S and Goode, J J 'A survey of various tactics for generating Lagrangean multipliers in the context of Lagrangean duality', *Euro. J. Operat. Res.*, Vol 3 (1979) pp 322–328

31 Gavish, B 'On obtaining the 'best' multipliers for a Lagrangean relaxation for integer programming', *Comput. Operat. Res.*, Vol 5 (1978) pp 55–71

32 Syslo, M M, Deo, N and Kowalik, J S *Discrete Optimization Algorithms with Pascal Programs*, Prentice-Hall, Englewood Cliffs, NJ (1983)

33 Held, M., Wolfe, P. and Crowder, H P, 'Validation of subgradient optimization', *Math. Prog.*, Vol 5 (1974) pp 62–68

34 Poljack, B T 'A general method of solving extremum problems', *Soviet Math. Doklady*, Vol 8 (1967) pp 593–597