Network Programming Methods for Loss Networks

R. J. Gibbens and F. P. Kelly

(Invited Paper)

Abstract—**This paper describes how some of the insights available from the stochastic analysis of dynamic routing may be incorporated into the classical mathematical programming approach to the design of networks. In particular, we present the results of a number of numerical investigations into network architectures for circuit-switched communication networks. Our investigations use recent theoretical results integrating network flow optimization and Markov decision processes to provide performance bounds for dynamic routing strategies. Following a tutorial introduction of the above mentioned topics we develop a sequence of network examples. Our first examples are familiar ones, such as symmetric fully connected networks and networks with moderate amounts of asymmetry, and we describe how network programming methods complement earlier work on dynamic routing. We then consider a variety of example networks which have a more sparse collection of links. These examples indicate the potential applicability of the methods to a variety of areas, including studies of the design, performance and resilience of future communication networks.**

I. Introduction

MATHEMATICAL programming and network flow approaches to the design of communication networks have a long and distinguished history (for reviews and further references see [3], [8], [14]). Typical problems include the design at minimum cost of a network able to support a given multicommodity flow, or a collection of nonsimultaneous multicommodity flows, perhaps after one or more link or node failures within the network, and the design of routing patterns to make best use of a given network. The network programming approach is able to deal with large and complex networks, and concepts such as a cut set and a shadow price provide important insights. Deterministic flows are often treated, or random flows are represented to a first approximation by an assumption of a fixed pattern of routing and the use of a simple formula, such as Erlang's formula or the M/M/1 delay formula, to assess loss or delay at resources of the network [3], [7], [18].

Modern dynamic routing schemes [1] create more subtle interactions between random traffic flows and the network topology that may not be well modeled by the above first approximation. For example dynamic routing schemes may cause distinct resources within the network to act as a single pooled resource, able to cope with the aggregate random fluctuation arising from several traffic flows [21], [22], [27]. There now exist several approximations and asymptotics for the stochastic analysis of dynamic routing schemes (see, for example, [16], [19], [26], [31], [34]) but these lose much of the simplicity and generality of the network programming approach. Our aim in this paper is to show that at least some of the insights available from stochastic analysis may be incorporated into the classical network programming approach, without losing the latter's tractability or conceptual simplicity. Conversely we expect the network programming approach to assist in the development of dynamic routing schemes for sparse and irregular network topologies.

In Section II we describe our basic network model, and review the performance bounds of [20]. These bounds are obtained from a network flow synthesis of various Markov decision processes, one for each resource of the network. Following this we develop a sequence of network examples, illustrating how with an appropriate choice of the resources modeled it is possible to capture the essential aspects of good dynamic routing schemes within the network programming formalism.

Our first network examples, taken from [12], [33], are familiar ones, such as symmetric fully connected networks and networks with moderate amounts of asymmetry. Dynamic routing in such networks is now well understood [10], [17], [29], [30], and we show how known results are reflected in the behavior of the dual variables of the extended network programming approach. We also discuss a network where one node is substantially overloaded, and how bounds may be improved by including vertex constraints to model the limited capacity emanating from each node.

Next, in Section IV, we consider a network where links correspond to the edges of a cube. This example illustrates how our methods extend to more sparsely connected networks, where it becomes important to model the limited capacity of certain cut sets. Finally, in Section V, we indicate how our methodology extends to irregular network structures, through a discussion of a network with random topology.

In this paper we concentrate on modeling loss networks where the route of an accepted demand is fixed for the duration of the demand. The network bounds of Section II may also be developed for queueing networks or for loss networks where demands may be rerouted while still in progress [20], but we leave a closer examination to another study. Our examples of loss networks are deliberately simplified, to expose the fundamental arguments, but we believe the methods will be of use in a variety of more realistic settings, including

Manuscript received September 30, 1994; revised April 1, 1995. R. J. Gibbens' work was supported by a Royal Society University Research Fellowship. Computing work was supported by the EPSRC under Grant GR/J371896.

The authors are with the Statistical Laboratory, University of Cambridge, Cambridge CB2 1SB U.K.

IEEE Log Number 9413107.

studies of the design, performance and resilience of future networks [5], [15]. An early instance is the recent investigation [13], where the joint use of bounds for any routing scheme and simulations of particular routing schemes facilitated a systematic comparison of the performance of different network architectures under various overload and failure conditions.

A different approach to the integration over a network of single resource Markov decision processes, based on an application of the policy improvement lemma to an initial fixed routing scheme, has been developed by Ott and Krishnan ([32]; see also [28], [36]). An overview, containing a discussion of several points of contact between this and our approach, is provided by Key [23], [24].

Under Poisson and exponential modeling assumptions the bound used in this paper becomes the solution to a linear program: successive refinements of the set of resources modeled corresponds to the addition of further variables and constraints. Other linear programs that bound the performance of stochastic networks are developed in [4], [25] and the papers referenced therein. Of course any Markov decision process may be formulated as a linear program [35], even the process capturing the full stochastic dynamic routing problem, although the resulting linear program may be of vast size. To clarify the relationship between these various linear programs remains a challenge.

II. Network Bounds

We begin by describing our model of a loss network. Let I label the set of possible demands on a network, and also the set of network resources. Assume that demands of type i arrive at the network at rate ν_i, for $i \in I$. Let C_i be the *capacity* of resource i. We shall refer to a resource as a *link*, interpret C_i as the number of *circuits* on link i, and refer to a demand labeled i as a *call* of type i. Suppose that an arriving call of type i may potentially be routed directly, on link i, or alternatively, on a route $r \in R(i)$. Here a route r identifies a subset of I, and $R(i)$ is the set of alternative routes potentially available to a call of type i. Label routes so that the sets $R(i), i \in I$, are disjoint: let $R = \bigcup_{i \in I} R(i)$, and let $i(r)$ be the unique element in I such that $r \in R(i(r))$. If an arriving call of type i is sent to route $r \in \{\{i\}\} \cup R(i)$ then it uses one circuit from each link $j \in r$ for the holding period of the call. A call may only be sent to a route r with at least one free circuit on each link $j \in r$. The call may also be discarded, and it *must* be discarded if there are no routes with spare capacity available. The holding period of a call is arbitrarily distributed with unit mean, and is unaffected by the route used to carry the call.

Suppose that acceptance of a call of type i generates a reward of w_i. Then the average reward per unit time is bounded above by the value attained in the following maximum flow problem

$$\text{maximize} \qquad \sum_i w_i f_i \tag{1}$$

subject to

$$f_i = x_i + \sum_{r \in R(i)} y_r \le \nu_i \qquad i \in I \tag{2}$$

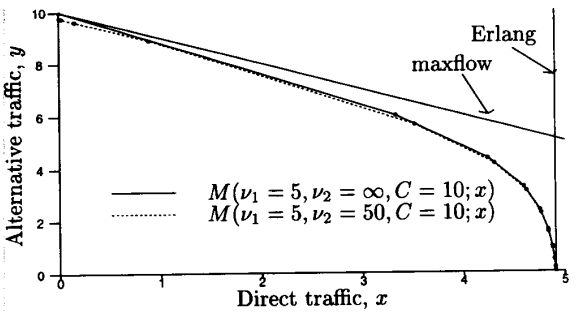

Fig. 1. The "maxflow" bound is constraint (3), while the "Erlang" bound is constraint (6). Two examples of the M function, defined by (9), are shown with the choices $\nu_2 = 50$ and $\nu_2 = \infty$: observe that there is little difference between these two functions.

$$x_i + \sum_{r \ni i} y_r \le C_i \qquad i \in I \tag{3}$$

over

$$x_i \ge 0, \quad f_i \qquad i \in I \tag{4}$$

$$y_r \ge 0, \qquad r \in R. \tag{5}$$

The variables x_i, y_r represent mean flows along routes $\{i\}$, r, respectively, and thus, for example, inequality (3) follows since the total mean flow on link i cannot exceed the capacity of that link. For a formal proof of this intuitively obvious result see [10], [34]. The bound may sometimes be achieved for deterministic arrivals and holding periods, although even in this case certain integer and packing constraints must be satisfied. When arrivals and holding periods are random the constraint (3) may often be considerably tightened: for example, if arrivals of calls of type i form a Poisson process, then necessarily

$$x_i \le \nu_i (1 - E(\nu_i, C_i)) \tag{6}$$

where

$$E(\nu, C) = \frac{\nu^C}{C!} \left[\sum_{j=1}^{C} \frac{\nu^j}{j!} \right]^{-1} \tag{7}$$

is *Erlang's formula* for the proportion of lost calls at a link of capacity C circuits, offered a single Poisson stream of calls of rate ν.

In Fig. 1 we illustrate the two constraints (3) and (6), labeled "maxflow" and "Erlang", respectively, where the horizontal axis measures the direct flow $x = x_i$, and the vertical axis measures the net alternatively routed traffic $y = \sum_{r \ni i} y_r$. Observe that neither constraint dominates the other. The two constraints may be tightened further, to yield a single composite constraint dominating both, as follows.

Consider a single link, of capacity C circuits, offered two streams of traffic. Suppose that acceptance of a type 1 call generates a reward w_1, and acceptance of a type 2 call generates a reward w_2, where $w_1, w_2 \ge 0$. Suppose that the two streams of arriving traffic form independent Poisson processes of rates ν_1 and ν_2, respectively, and that accepted calls have holding times which are exponentially distributed

with unit mean, independently of earlier arrival and holding times. When a call arrives a decision is made to accept or reject the call, where the decision can depend on the type of the call.

Let $W(\nu_1, \nu_2, C; w_1, w_2)$ be the maximal expected reward per unit time over all policies. Observe that W is a convex function of (w_1, w_2), since it can be expressed as a supremum of linear functions of (w_1, w_2)

$$W(\nu_1, \nu_2, C; w_1, w_2) = \sup_{\pi} \{w_1 x(\pi) + w_2 y(\pi)\} \tag{8}$$

where $x(\pi)$ and $y(\pi)$ are the mean acceptance rates of calls of types 1 and 2, respectively under a policy π. Let

$$M(\nu_1, \nu_2, C; x) = \sup_{\pi} \{y(\pi) : x(\pi) \geq x\} \tag{9}$$

the maximal mean acceptance rate of calls of type 2, subject to the requirement that the mean acceptance rate of calls of type 1 must be at least x. M is a concave function of x – indeed W and M are conjugate functions [20].

The policy achieving the supremum in (8) is a *trunk reservation policy* which operates as follows. If $w_1 \geq w_2 \geq 0$, accept type 1 calls provided the link is not full and accept type 2 calls provided the number of spare circuits is above a certain integer t, say, and reject type 2 calls whenever t or fewer circuits are free. The parameter t is called the *trunk reservation parameter*. Restricting to trunk reservation policies it is straightforward to analyze the resulting birth and death process describing the performance of the single link system to obtain the following expressions for the mean acceptance rates of the two call types parameterized by the value of t

$$x(\nu_1, \nu_2, C, t) = \tag{10}$$

$$\nu_1 G(\nu_1, \nu_2, C, t) \left[\sum_{k=0}^{C-t-1} \frac{(\nu_1 + \nu_2)^k}{k!} + \right. \tag{11}$$

$$\left. (\nu_1 + \nu_2)^{C-t} \sum_{k=C-t}^{C-1} \frac{\nu_1^{k-C+t}}{k!} \right] \tag{12}$$

and

$$y(\nu_1, \nu_2, C, t) = \nu_2 G(\nu_1, \nu_2, C, t) \left[\sum_{k=0}^{C-t-1} \frac{(\nu_1 + \nu_2)^k}{k!} \right] \tag{13}$$

where the normalization constant is given by

$$G(\nu_1, \nu_2, C, t) = \tag{14}$$

$$\left[\sum_{k=0}^{C-t-1} \frac{(\nu_1 + \nu_2)^k}{k!} + (\nu_1 + \nu_2)^{C-t} \sum_{k=C-t}^{C} \frac{\nu_1^{k-C+t}}{k!} \right]^{-1}. \tag{15}$$

These expressions may be numerically computed with little difficulty under all practical choices of the parameters. Indeed, efficient numerical expressions can nowadays be written in just a few lines of a high level language such as S [2] or similar languages. Hence, the maximal expected reward per unit time is given by a maximization over the choice of parameter t

$$W(\nu_1, \nu_2, C; w_1, w_2) = \tag{16}$$

$$\max_{-C \leq t \leq C} \{w_1 x(\nu_1, \nu_2, C, t) + w_2 y(\nu_1, \nu_2, C, t)\} \tag{17}$$

where negative values of t refer to a parameter $|t|$ used against type 1 calls, and will be appropriate if $w_2 > w_1 \geq 0$. Similarly, M is a decreasing, piece-wise linear and concave function of x (Fig. 1 shows examples of the function M) given by the convex hull of the $2C + 1$ points

$$\{(x(\nu_1, \nu_2, C, t), y(\nu_1, \nu_2, C, t)), t = -C, -C+1, \ldots, C\}.$$

Throughout we let a maximum over an empty set be $-\infty$; thus $M = -\infty$ for $x > \nu_1(1 - E(\nu_1, C))$.

Return now to the network model, and suppose that the different arrival streams of rate ν_i are independent Poisson processes for $i \in I$. For each $i \in I$ define

$$M_i(x) = M(\nu_i, \sum_{j: i \in R(j)} \nu_j, C_i; x) \tag{18}$$

$$W_i(w_1, w_2) = W(\nu_i, \sum_{j: i \in R(j)} \nu_j, C_i; w_1, w_2). \tag{19}$$

Here the expression $\sum_{j: i \in R(j)} \nu_j$ gives an upper bound on the total amount of traffic that could possibly overflow onto alternative routes which use link i. Then [20] the expected reward per unit time is bounded above by the value attained in the following maximum flow problem

$$\text{maximize} \quad \sum_i w_i f_i \tag{20}$$

subject to

$$f_i = x_i + \sum_{r \in R(i)} y_r \leq \nu_i \qquad i \in I \tag{21}$$

$$\sum_{r \ni i} y_r \leq M_i(x_i) \qquad i \in I \tag{22}$$

over

$$x_i \geq 0, \quad f_i \qquad i \in I \tag{23}$$

$$y_r \geq 0 \qquad r \in R \tag{24}$$

or, equivalently, the value attained in the dual minimum cost problem

$$\text{minimize} \quad \sum_i [W_i(w_i - s_i, c_i) + s_i \nu_i] \tag{25}$$

subject to

$$w_{i(r)} - \sum_{j \in r} c_j \leq s_{i(r)} \qquad r \in R \tag{26}$$

over

$$c_i, s_i \geq 0 \qquad i \in I. \tag{27}$$

The primal problem is perhaps the easier to interpret: inequality (22) captures the insight that link i cannot achieve higher acceptance rates as part of a network than it could if the rest of the network were transparent. A derivation of the dual problem is instructive. Consider a feasible choice of

$\mathbf{c}, \mathbf{s}$; that is, a collection $c_i, s_i, i \in I$, satisfying (26) and (27). Suppose that a call of type i is charged an amount s_i by the network when it is offered to the network, and in addition is charged an amount $w_i - s_i$ by link i if it is carried directly on link i, or an amount c_j at each link $j \in r$ if it is carried on alternative route r. The maximum revenue that can possibly be collected thus cannot exceed the objective function (25), for any feasible choice of $\mathbf{c}, \mathbf{s}$. Since $\mathbf{c}, \mathbf{s}$ satisfy (26) each routed call of type i pays at least w_i, whether the call is routed directly or alternatively, and the bound follows.

It is interesting to interpret the complementary slackness conditions that interrelate the solutions to problems (20) and (25) in terms of the charges used in the preceding derivation. The main condition shows that if the charges along route r are too high (that is $\sum_{j \in r} c_j > w_{i(r)} - s_{i(r)}$) then the route is not used (that is $y_r = 0$). The interpretation of the remaining complementary slackness conditions is more familiar: the charge c_i at link i is zero if there is spare capacity at this link, as indicated by slackness in the constraint (22); while the charge s_i on an offered call of type i is zero if calls of type i are being lost, as indicated by slackness in the constraint (21). We might term s_i the *surplus value* of an additional call of type i, and c_j the *implied cost* of using a circuit from link j: if a call of type i is accepted on route r it will earn w_i directly but at an implied cost of c_j for each circuit used from link j, leaving a surplus value of $s_i = w_i - \sum_{j \in r} c_j$.

The function $M_i(x_i)$ is concave and piecewise linear in x_i, for $i \in I$, and so both the primal and dual problems (20) and (25) may be written as linear programming problems. Note that if the constraint (22) is written as a collection of linear constraints (see Fig. 1), then at most two of these constraints can be satisfied with equality. Note also that constraint (22) dominates both the "maxflow" constraint (3) and the "Erlang" constraint (6). In our later examples the second argument of the function (18) will usually be large. For simplicity we shall often weaken the bound slightly, and substitute infinity for this second argument. We see, in Fig. 1, that this relaxation mainly affects the function M when x is small.

Finally we note that our formulation of the primal and dual pair (20) and (25) has deliberately suppressed the dependence of the function M_i on the variables $(\nu_k, k \in I)$. This leads to the simplicity of the above formulation and interpretation, and of the numerical results to follow. To determine explicitly the total dependence of the optimal value function, ϕ, say, on the parameter ν_i, we must calculate

$$\frac{\partial \phi}{\partial \nu_i} = s_i + \sum_k \frac{\partial}{\partial \nu_i} W\left(\nu_k, \sum_{j:k \in R(j)} \nu_j, C_k; w_k - s_k, c_k\right). \tag{28}$$

This is not difficult, using for example the representation (17).

III. Fully Connected Networks

The first and simplest network architecture we consider is the *symmetric* network on n nodes. This network has received much attention and there is now a considerable literature and understanding of the behavior of dynamic routing strategies [10], [11], [30], [31]. Accordingly, it is a natural example to investigate with the methodology reviewed in Section II.

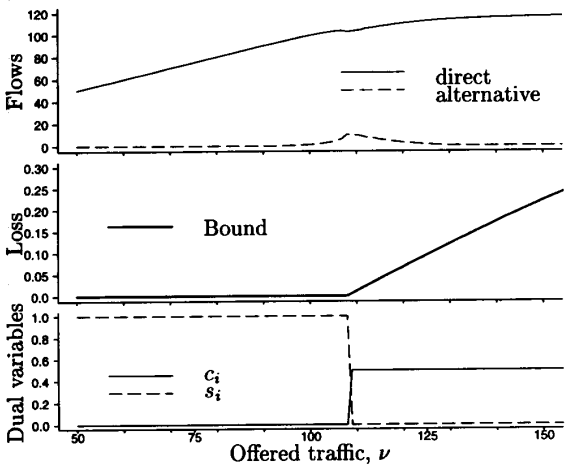

Fig. 2. Symmetric network with $C = 120$. The first panel shows the direct flow x and alternative flow $2(n-2)y$ through a typical link. The second panel shows the bound on the proportion of lost calls. The third panel shows the dual variables c_i and s_i, and their sudden change at around $\nu = 108$.

A. Symmetric Network

Consider a symmetric network with n nodes where I is the set of $n(n-1)/2$ edges of the complete graph on n nodes, $\nu_i = \nu$, $C_i = C$, $w_i = 1$, $i \in I$, and $R(i)$ is the set of $(n-2)$ two-link paths connecting the node pair identified by i. The primal problem (20) becomes the following optimization problem

$$\text{maximize} \quad \binom{n}{2} f \tag{29}$$

subject to

$$f = x + (n-2)y \le \nu \tag{30}$$

$$2(n-2)y \le M(x) \tag{31}$$

over

$$x, y \ge 0. \tag{32}$$

Here, by symmetry and convexity, we may take there to be just two types of flows. The flow x is the traffic carried on a direct route and the flow y is the traffic carried on a two link alternative route. More fully the function $M(x)$ might be

$$M(x) = M(\nu, 2(n-2)\nu, C; x) \tag{33}$$

but for simplicity we shall relax the constraint slightly by using

$$M(x) = M(\nu, \infty, C; x). \tag{34}$$

Fig. 2 presents the results of our numerical investigations for the symmetric network when the offered traffic ν is allowed to vary and the link capacities are held fixed at $C = 120$ and $n = 12$. The figure is divided into three panels. In the top panel direct flow, x, and alternative flow, $2(n-2)y$, through

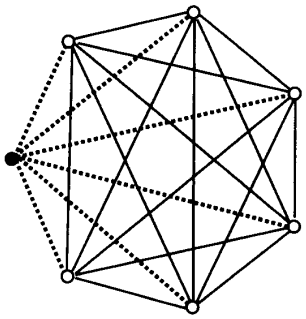

Fig. 3. Overload node network.

a link are shown, while the second panel shows the bound on the proportion of calls lost. When ν is small the bound on loss is zero. In this regime many solutions achieve the optimum: for definiteness we shall here and henceforth select from amongst such solutions one that maximizes the amount of directly routed traffic. As ν increases a smaller proportion of the offered traffic should be carried directly and up to 4.7% is carried alternatively. As ν increases further y reduces, and very little alternative routing is used. Above a threshold value of around 108 loss starts to occur and the bound on the loss probability becomes positive.

How well can dynamic routing schemes perform relative to the bound of Fig. 2? This question is, for the symmetric network, fairly well understood [10], [16], [17], [19], [29], [30], [31]. In particular, Hunt and Laws [17] have established that as the number of nodes n increases the proportion of calls lost under an optimal dynamic routing scheme approaches the loss rate shown in Fig. 2. This conclusion had been supported by several simulation studies: analytical and simulation evidence for particular routing schemes is discussed by [31] for the case of 12 nodes and capacity 120 circuits.

It is interesting to note, from Fig. 2, that the implied costs c_j take the value 0 or $\frac{1}{2}$, depending on whether the proportion of flow lost, as shown in panel 1, is zero or positive. This is a common feature of the solution of the dual problem (25)–(27) for a fully connected network, and does *not* require that all capacities, or all offered traffics, be identical. When a solution to the primal problem (20)–(24) allows positive flow y_r on all routes $r \in R$, the constraints (26) from the dual problem will all be tight, implying $c_i = \frac{1}{2}$, $i \in I$, when $w_i = 1$, $i \in I$. In [13] a network is considered with $n = 24$ nodes and with C of the order of 120 or 600 circuits. Several asymmetric traffic patterns and failure conditions are considered, all of which have $c_i = \frac{1}{2}$, $i \in I$. Simulations of a dynamic routing scheme are also considered, and compare reasonably well with the bound [13, Figs. 4 and 6].

Next we consider a simple example where a solution to (20)–(24) does *not* allow positive flows on all routes $r \in R$, and hence where implied costs may vary over different links.

B. Overloaded Node Network

In this network one node has its traffic to all other nodes multiplied by a factor $(1 + \epsilon)$ relative to all the other traffics,

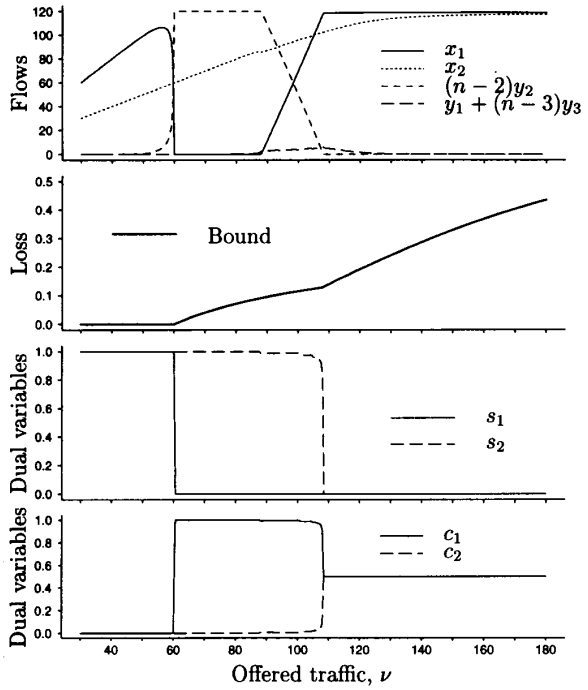

Fig. 4. Overloaded node network with $C = 120$, $n = 12$, and $\epsilon = 1$. The top panel shows: the direct flow on an overload link, x_1; the direct flow on an nonoverloaded link, x_2; the alternative flow for an overloaded stream, $(n-2)y_2$; and the alternative flow for a nonoverloaded stream, $y_1+(n-3)y_3$.

which have a common value ν. All link capacities are the same with $C_i = C$. Fig. 3 shows an illustration of an overloaded node network for the case where $n = 7$. The solid circle represents the overloaded node. We refer to the links from this node, represented by dashed lines, as the *overloaded links*: the dashed lines are direct routes for the *overloaded traffic streams*. We refer to the other streams as *nonoverloaded* and their direct routes are the links shown as solid lines. Again by symmetry there are only a small number of distinct flow types which we label x_1, x_2, y_1, y_2 and y_3. Flows x_1 and x_2 refer to the traffics carried on direct routes which are overloaded and nonoverloaded, respectively. The flows y_1, y_2 and y_3 are the traffics carried on two link alternative routes which contain 0, 1 or 2 nonoverloaded links, respectively.

There are two types of each of the dual variables s and c. We label s_1 the surplus value for the overloaded streams and s_2 the surplus value for the nonoverloaded streams. The implied cost c_1 refers to an overloaded link, while the implied cost c_2 refers to a nonoverloaded link.

The primal problem (20) becomes the following [12], [33] linear program

$$\text{maximize} \qquad (n-1)f_1 + \binom{n-1}{2} f_2 \tag{35}$$

subject to

$$f_1 = x_1 + (n-2)y_2 \le (1+\epsilon)\nu \tag{36}$$

$$f_2 = x_2 + y_1 + (n-3)y_3 \le \nu \tag{37}$$

$$(n-2)y_1 + (n-2)y_2 \le M_1(x_1) \quad (38)$$

$$2y_2 + 2(n-3)y_3 \le M_2(x_2) \quad (39)$$

over

$$x_1, x_2 \ge 0 \quad (40)$$

$$y_1, y_2, y_3 \ge 0\,. \quad (41)$$

Here

$$M_1(x_1) = M((1+\epsilon)\nu, \infty, C; x_1) \quad (42)$$

$$M_2(x_2) = M(\nu, \infty, C; x_2)\,. \quad (43)$$

Fig. 4 shows our numerical results for the overloaded node network when $n = 12$ and $\epsilon = 1$, so that overloaded streams have double the offered load relative to the nonoverloaded streams. The top panel shows several of the interesting flows. Alternative routing can be seen to take place on routes through one or two nonoverloaded links but not on routes through two overloaded links. The lower panels show the existence of two threshold values: one at 60 and another at 108.

The second panel shows the charges s_1 and s_2. The charge s_1 drops to 0 at the threshold value of 60 when the overloaded node first saturates and loss of traffic on the overloaded streams first occurs. Note that $\epsilon = 1$ so that when $\nu = 60$ each overloaded stream has an offered traffic of 120 matching the link capacity. The charge s_2 behaves in a similar manner to that shown in the symmetric network dropping to 0 at around 108 when loss first occurs from the nonoverloaded streams.

The third panel shows the link charges c_1 and c_2. The charge c_1 first increases from 0 to 1 and then drops back to a value of $\frac{1}{2}$. The charge c_2 increases directly from 0 to $\frac{1}{2}$ in an analogous manner to that of the symmetric network. The intermediate traffic region between the two traffic thresholds where c_1 is 1 while c_2 is still $\frac{1}{2}$ simultaneously achieves two ends: it prevents alternative traffic from using two overloaded links, while enabling alternative routing to continue amongst routes through one or two nonoverloaded links.

At first sight the large flows y_2, on alternative routes from the overloaded node, may seem surprising. On further consideration, however, this result is indicative of the behavior of a good dynamic routing scheme. When links from the overloaded node are nearly saturated, but links elsewhere have spare capacity, then an arriving call involving the overloaded node will probably not be able to be routed directly, but should nevertheless be accepted if a free circuit can be found anywhere out from the overloaded node.

In Fig. 5 we show how the bound varies with the choice of the overload parameter ϵ. Observe the existence of two thresholds for positive values of ϵ.

C. Vertex Constraints

When the number of nodes n is small, additional constraints may have a pronounced effect on the performance of dynamic routing schemes. In this section we illustrate this point by extending the analysis of Section III-A to include constraints corresponding to the total capacity into and out of each vertex. First we develop further the general model of Section II.

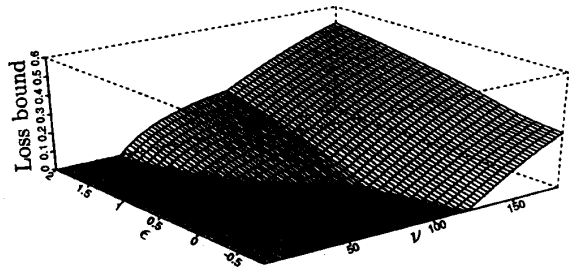

Fig. 5. Overloaded node network with $n = 12$ nodes and varying ϵ. Observe the existence of two threshold values for positive ϵ.

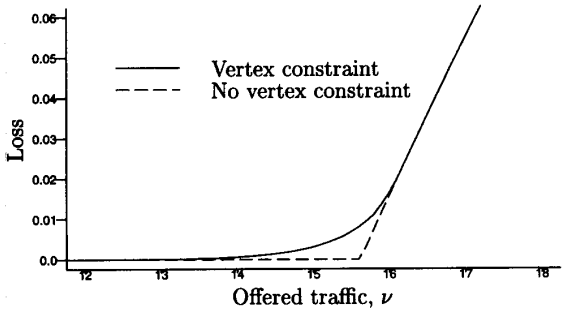

Fig. 6. Comparison of bounds with and without vertex constraint.

Write $v \in r$ if route r passes through vertex v using this vertex as a tandem, but v is not an end vertex of the path r. Write $v \in i$ if vertex v is at an end of link i. Consider the flow of traffic into and out of vertex v. The amount of tandem traffic is

$$2\sum_{r \ni v} y_r \quad (44)$$

since each call that uses vertex v as a tandem vertex occupies a circuit into, *and* a circuit out of, vertex v. Hence under any dynamic routing scheme

$$2\sum_{r \ni v} y_r \le M_v(f_v) \qquad v \in V \quad (45)$$

where

$$\sum_{j \ni v} f_j = f_v \quad (46)$$

and

$$M_v(x) = M\left(\sum_{j \ni v} \nu_j, \infty, \sum_{j \ni v} C_j; x\right). \quad (47)$$

We can improve the bound provided by problem (20) by appending to that problem the additional constraints (45) and (46). The corresponding dual problem can be written in the form

$$\text{minimize} \quad \sum_i [W_i(d_i, c_i) + s_i\nu_i] + \sum_v W_v(d_v, c_v) \quad (48)$$

subject to

$$w_i = s_i + d_i + \sum_{v \in i} d_v \qquad i \in I \tag{49}$$

$$d_{i(r)} \leq \sum_{j \in r} c_j + 2 \sum_{v \in r} c_v \qquad r \in R \tag{50}$$

over

$$c_i, s_i \geq 0\,, \quad d_i\,, \qquad i \in I \tag{51}$$

$$c_v \geq 0\,, \quad d_v\,, \qquad v \in V\,. \tag{52}$$

It is as if the vertex v charges an amount d_v for the use of a single circuit out of vertex v by a call terminating at vertex v, and an amount $2c_v$ for the use of two circuits out of the vertex v by a call using vertex v as a tandem node.

Consider again the symmetric networks of Section III-A. Let $n = 5$, $C = 20$ and add the constraints (45) and (46) to the primal problem, to give

$$\text{maximize} \qquad 10f \tag{53}$$

subject to

$$f = x + 3y \leq \nu \tag{54}$$

$$6y \leq M(x) \tag{55}$$

$$12y \leq M_v(f_v) \tag{56}$$

$$4f = f_v \tag{57}$$

over

$$x, y \geq 0\,. \tag{58}$$

Here M is again given by equation (34), while the vertex constraint has

$$M_v(f_v) = M(4\nu, \infty, 4C; f_v)\,. \tag{59}$$

In Fig. 6 we compare bounds of problem (53)–(58) with the bound produced if the vertex constraints (56) and (57) are omitted. The vertex constraint produces a loss of 0.1% or more for $\nu \geq 14.4$, while the model without the vertex constraint produces no loss until $\nu \geq 15.7$.

The dual variables c_i, s_i and d_v are illustrated in Fig. 7. Note that as ν increases through $\nu = 9$ the surplus value s_i drops to zero, and the charge d_v jumps to $\frac{1}{2}$: as ν further increases through $\nu = 15$ the charge d_v drops back to zero, while the charge c_i jumps to $\frac{1}{2}$. Over the range $\nu \in (9, 15)$ the dominant dual variable is d_v, representing the impact of stochastic effects on the limited total capacity out of each vertex. For $\nu > 16$, dominance reverts to the link constraints (54) and (55), as in Section III-A.

The reader will have noted that our vertex constraint (45)–(46) corresponds to the identification of a node on a route as just another resource of the network: sometimes a more severe constraint than (45)–(46) may be appropriate, if, for example, the processing power of a node imposes a tighter constraint on traffic through the node than that implied by the total link capacity emanating from the node.

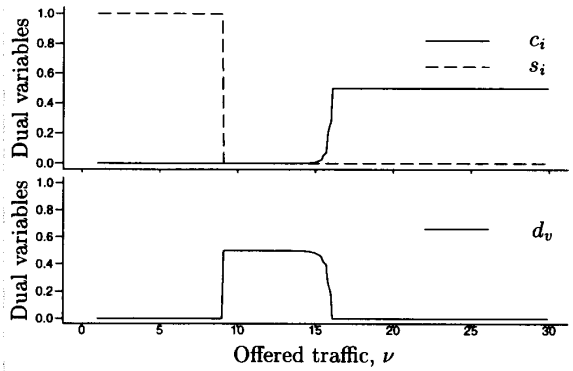

Fig. 7. Vertex constraints. Over the range $\nu \in (9, 15)$ the dominant dual variable is d_v: the total capacity out of a vertex is the major constraint on performance.

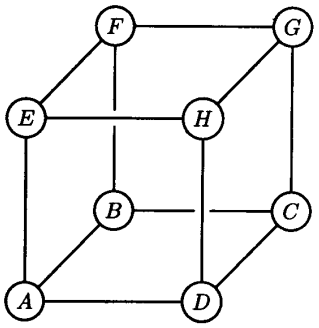

Fig. 8. Cube network. Each link has capacity C and the offered traffic between each pair of nodes is ν.

IV. Cube Network

Next we indicate how our methods extend to more sparsely connected networks. Note that the analysis of Section II applies directly when the network is not fully connected: we simply set $C_i = 0$ for the missing links. For instance, in the cube network illustrated in Fig. 8, we may let $C_i = C$ as i ranges over the twelve edges of the cube, and $C_i = 0$ for all other node pairs. In contrast we may set $w_i = 1$, $\nu_i = \nu$ for all 28 possible node pairs, so that calls arrive at rate ν between *any* two of the 8 nodes. Next we must identify those routes which are allowed. We allow the shortest routes, and, if that identifies a unique route (perhaps the direct route), then all the next shortest routes. Thus if i identifies an edge of the cube, we let $R(i)$ be the set of two routes, each of length 3, that may serve as alternatives. If i labels two nodes a distance 2 apart, let $R(i)$ be the set of two routes each of length 2, that connect these two nodes. Finally if i labels two nodes a distance 3 apart, let $R(i)$ be the set of six routes, each of length 3, that connect these two nodes.

When a network is sparsely connected then certain cut constraints may dominate, for example the cut set of four edges separating one face of the cube from the opposite face. Before considering this example in more detail, we develop further the general model of Section II.

Let $J \subset I$ be such that

$$j \in J, r \in R(j) \Rightarrow |r \cap J| = 1$$

where $|r \cap J|$ is the number of links from J on route r. Thus J forms a cut, in that every call from the set J requires to use a single resource from J however it is routed. Let

$$M_J(x) = M\left(\sum_{j\in J}\nu_j, \infty, \sum_{j\in J} C_j; x\right) \tag{60}$$

and consider the flow of traffic across the cut J. Under any dynamic routing scheme

$$\sum_{k\notin J}\sum_{r\in R(k)} |r \cap J| y_r \le M_J(f_J) \tag{61}$$

where

$$\sum_{j\in J} f_j = f_J\,. \tag{62}$$

We can improve the bound provided by problem (20) by appending to that problem the additional constraints (61) and (62).

Consider now the cube network of Fig. 8, and introduce a cut constraint for each of the three cut sets separating a face of the cube from the opposite face. Then

$$M_J(f_J) = M(16\nu, \infty, 4C; f_J) \tag{63}$$

since the cut J separating a face of the cube from the oppposite face separates 16 node pairs each offering traffic at rate ν, and has a total capacity of $4C$ circuits. Let f_1, f_2, f_3, respectively represent the carried flow between a pair of end-points a distance 1, 2 or 3 apart. Then the primal problem becomes

$$\text{maximize} \qquad 12f_1 + 12f_2 + 4f_3 \tag{64}$$

subject to

$$f_1 = x_1 + 2y_1 \le \nu \tag{65}$$
$$f_2 = 2y_2 \le \nu \tag{66}$$
$$f_3 = 6y_3 \le \nu \tag{67}$$
$$6y_1 + 4y_2 + 6y_3 \le M(x_1) \tag{68}$$
$$16y_1 \le M_J(f_J) \tag{69}$$
$$4f_1 + 8f_2 + 4f_3 = f_J \tag{70}$$

over

$$x_1, y_1, y_2, y_3 \ge 0\,. \tag{71}$$

In Fig. 9 we describe some aspects of the solution to the linear program (64), as ν varies. The dual variables s_1, s_2, s_3, c, c_J, d_J correspond to constraints (65)–(70), respectively. Observe that the dual variable d_J is significant over the range $\nu \in (2.2, 4.8)$, which we deduce is the range of traffic over which the cut constraints (69)–(70) force some loss.

Observe that over the range $\nu \in (2.2, 4.8)$ the surplus value s_1 is greater than s_2, while for $\nu \in (4.8, 6.3)$ the order is interchanged. Calls of type 1 can be carried directly on single-linked routes, but require 3-link routes if they are routed alternatively. In contrast calls of type 2 have no direct route, but have a choice of 2-link alternative routes. The network finds the former type of call easier to carry when $\nu \in (2.2, 4.8)$ but not when $\nu \in (4.8, 6.3)$.

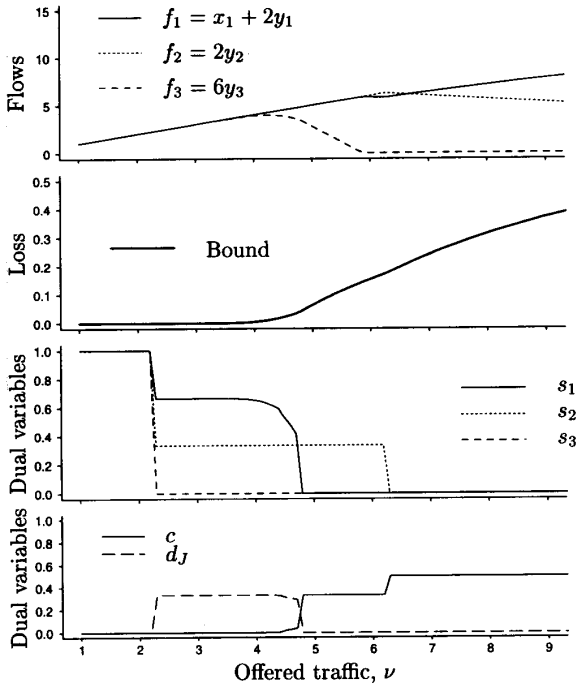

Fig. 9. Cube network ($C = 20$). Over the range $\nu \in (2.2, 4.8)$ the dual variable d_J is significant, and the additional cut constraints are a constraint on performance.

For the cube network we have used the three cut constraints whose inclusion has most effect on the performance bound. For a general network it may not be clear in advance which are the important cut constraints. In [9], [10] all of the possible cut constraints in a particular network are included as Erlang bounds: these bounds correspond to the vertical line in Fig. 1. In [9], [10] max-flow bounds were also used, corresponding to the diagonal line in Fig. 1. It was found that neither type of bound dominated uniformly over networks or traffic conditions, an observation explained by Fig. 1. Of course since M is bounded by both the diagonal and vertical lines of Fig. 1, the bound of this paper dominates *both* the Erlang and the max-flow bounds of [9], [10].

V. Random Network

In our earlier network examples we have used symmetries to reduce the number of distinct flows and dual variables that are needed to describe solutions. This has simplified the presentation of examples, but our methodology extends readily to irregular network structures, as we illustrate in this Section. Indeed a major part of our motivation is the expectation that the network programming approach will assist in the development of dynamic routing schemes for more sparse and irregular network architectures.

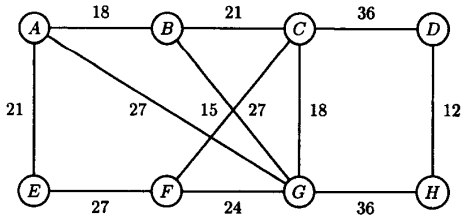

Fig. 10. Random network. Links are labeled with their capacity. The offered traffic between each pair of nodes is ν.

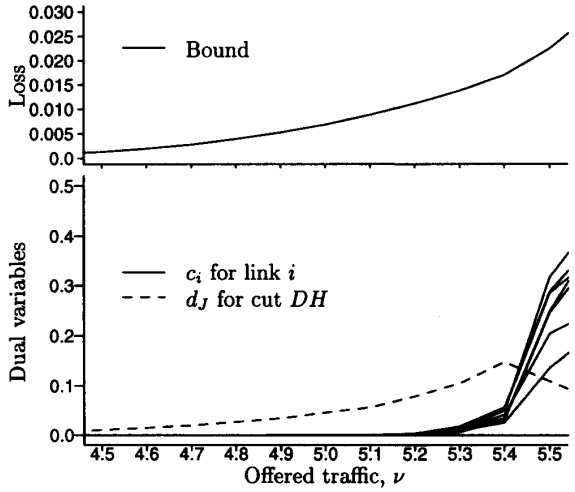

Fig. 11. Dual variables c_i for each link i and d_J for the DH cut in the random network of Fig. 10. The dual variable d_J for the DH cut has a significant effect as ν increases up to 5.5.

The topology we consider in this Section, illustrated in Fig. 10, was constructed by selecting a random graph with eight nodes and twelve edges, conditional on each node being the endpoint of at least two edges, and the graph being connected. We let $w_i = 1$, $\nu_i = \nu$ for all 28 possible node pairs. Allowed routes are constructed as follows. Between any node pair we allow the shortest routes, and, if that identifies a unique route (perhaps the direct route), we also allow all the next shortest routes. Thus between nodes D and H we allow the route D–C–G–H as well as the direct route D–H and between nodes C and H we allow the routes C–D–H and C–G–H. Capacities are constructed as follows. Nominally assign traffic to routes by splitting a flow ν between two nodes equally over all the shortest routes between these two nodes; repeat this for each pair of nodes. Let the capacity of a link be such that it equals the aggregate traffic nominally assigned to it when $\nu = 6$. (Note that if this procedure were applied to the cube network of Section IV, then the capacity of each link would be $C = 24$.) The capacities thus assigned to the links of the random network are shown in Fig. 10.

There are three cuts in this network whose capacity equals the nominal traffic when $\nu = 6$, the cuts isolating the node pairs (A, E), (E, F) and (D, H), respectively. In Fig. 11 we show the results obtained by augmenting the problem (20) with the additional contraints (61)–(62) for each of these three cuts. The bound on overall loss increases to about 2% as ν increases to 5.5. The dual variable d_{DH} is significant over this range of traffic, while the dual variables d_{AE} and d_{EF} remain at zero: thus the cut DH has a significant effect on the overall loss probability, while the cuts AE and EF are dominated by the cut DH and the various single link constraints (21)–(22). As ν increases above 5.4, the effect of the cut DH begins to diminish, while the implied costs c_i corresponding to the single link constraints (22) continue to increase, behavior familiar from the analysis of the cube network in Section IV.

The important constraints are, of course, a consequence of the particular topology and traffic patterns used in this example. In a general network we might find the dominant constraints to be some combination of single link, vertex and cut constraints, depending on the network's size, connectivity, asymmetry and degree of overload.

VI. Conclusion

We have described how the classical network programming approach to the design and analysis of communication networks may be extended to represent some of the resource pooling features of dynamic routing schemes. Conversely the extended network programming approach gives insight into the qualitative behavior that should be expected of good dynamic routing schemes.

We have restricted attention in this paper to relatively simple examples of loss networks, where the route of an accepted demand is fixed for the duration of the demand, but the methodology developed in [20] is fairly robust to the precise model specification. In particular, it is described there how performance bounds may be calculated for networks where demands may be rerouted while in progress, and have differing holding periods and relative worths. We believe that such generalizations may be useful in the analysis of future multiservice networks.

Acknowledgment

The authors are grateful to P. Reichl for his participation in the earlier stages of this investigation, and for his contributions reported in [12], [33].

References

[1] "Advanced traffic control methods for circuit switched telecommunication networks," *IEEE Commun. Mag.*, vol. 28, Oct. 1990.
[2] R. A. Becker, J. M. Chambers, and A. R. Wilks, *The New S Language*. Pacific Grove, CA: Wadsworth and Brooks/Cole, 1988.
[3] D. Bertsekas and R. Gallager, *Data Networks*. Englewood, Cliffs, NJ: Prentice-Hall, 1992, 2nd ed.
[4] D. Bertsimas, I. C. Paschalidis, and J. H. Tsitsiklis, "Optimization of multiclass queueing networks: polyhedral and nonlinear characterizations of achievable performance," *Annals Appl. Prob.*, vol. 4, no. 1, pp. 43–75, 1994.
[5] G. N. Brown, W. D. Grover, J. B. Slevinsky, and M. H. MacGregor, "Mesh/arc networking: An architecture for efficient survivable self-healing networks," in *Proc. ICC*. 1994.
[6] R. L. Franks and R.W. Rishel, "Optimum network call-carrying capacity," *Bell Syst. Tech. J.*, vol. 52, pp. 1195–1214, 1973.
[7] R. G. Gallager, "A minimum delay routing algorithm using distributed computation," *IEEE Trans. Commun.*, vol. 25, pp. 73–85, 1977.
[8] M. Gerla and L. Kleinrock, "On the topological design of distributed computer networks," *IEEE Trans. Commun.*, vol. 25, pp. 48–60, 1977.

[9] R. J. Gibbens, "Dynamic Routing in Circuit-Switched Networks: The Dynamic Alternative Routing Strategy," Ph.D. thesis, Univ. Cambridge, July 1988.
[10] R. J. Gibbens and F. P. Kelly, "Dynamic routing in fully connected networks," *IMA J. Math. Contr. Inform.*, vol. 7, pp. 77–111, 1990.
[11] R. J. Gibbens, F. P. Kelly, and P. B. Key, "Dynamic alternative routing—modeling and behavior," in *12th Int. Teletraffic Congr.* Turin, Italy: North-Holland, 1988.
[12] R. J. Gibbens and P. Reichl, "Performance bounds applied to loss networks," in *Complex Stochastic Systems and Engineering*. D. M. Titterington, Ed. London, England: Oxford University Press, 1995.
[13] R. J. Gibbens, F. P. Kelly, and S. R. E. Turner, "Dynamic routing in multiparented networks," *IEEE/ACM Trans. Networking*, vol. 1, pp. 261–270, Apr. 1993.
[14] M. Gondran and M. Minoux, *Graphs and algorithms*. Chichester, England: Wiley, 1984.
[15] M. Herzberg and S. J. Bye, "Bandwidth management in reconfigurable networks," *Australian Teletraffic Res.*, vol. 27, pp. 57–70, 1993.
[16] P. J. Hunt and C. N. Laws, "Least busy alternative in queueing and loss networks," *Probab. Eng. Inform. Sci.*, vol. 6, pp. 439–456, 1992.
[17] ——, "Asymptotically optimal loss network control," *Math. Oper. Res.*, vol. 18, pp. 880–900, 1993.
[18] F. P. Kelly, "Routing in circuit-switched networks: optimization, shadow prices and decentralisation," *Adv. Appl. Probab.*, vol. 20, pp. 112–144, 1988.
[19] ——, "Loss networks," *Annals Appl. Probab*, vol. 1, pp. 319–378, 1991.
[20] ——, "Bounds on the performance of dynamic routing schemes for highly connected networks," *Math. Oper. Res.*, vol. 19, pp. 1–20, 1994.
[21] ——, "Dynamic routing in stochastic networks," in *Stochastic Networks*, F. P. Kelly and R. J. Williams, Eds. New York: Springer-Verlag, 1995, pp. 170–187.
[22] F. P. Kelly and C. N. Laws, "Dynamic routing in open queueing networks: Brownian models, cut constraints and resource pooling," *Queue. Syst.*, vol. 13, pp. 47–86, 1993.
[23] P. B. Key, "Optimal control and trunk reservation in loss networks," *Prob. Eng. Inform. Sci.*, vol. 4, pp. 203–242, 1990.
[24] ——, "Some control issues in telecommunication networks," in *Probability, Statistics and Optimization*, F. P. Kelly, Ed. Chichester, England: Wiley, 1994, pp. 383–395.
[25] P. R. Kumar, "Scheduling queueing networks: stability, performance analysis and design," in *Stochastic Networks*, F. P. Kelly and R. J. Williams, Eds. New York: Springer-Verlag, 1995, pp. 21–70.
[26] H. J. Kushner, "Control of trunk line systems in heavy traffic," Div. Appl. Sci., Brown Univ., 1992.
[27] C. N. Laws, "Resource pooling in queueing networks with dynamic routing," *Adv. Appl. Probab.*, vol. 24, pp. 699–726, 1992.
[28] V. G. Lazerev and S. M. Starobinets, "The use of dynamic programming for optimization of control in networks of commutation channels," *Eng. Cybernet*, vol. 15, pp. 107–117, 1977.
[29] V. Marbukh, "An asymptotic study of a large fully connected communication network with reroutes," *Problemy Peredachi Informatsii*, vol. 3, pp. 89–95, 1981.
[30] D. Mitra and R. J. Gibbens, "State-dependent routing on symmetric loss networks with trunk reservations, II: Analysis and asymptotics," *Annals Oper. Res.*, vol. 35, pp. 3–30, 1992. Special Issue on Stochastic Modeling of Telecommunication Systems.
[31] D. Mitra, R. J. Gibbens, and B. D. Huang, "State-dependent routing on symmetric loss networks with trunk reservations, I.," *IEEE Trans. Commun.*, vol. 41, pp. 400–411, Feb. 1993.
[32] T. J. Ott and K. R. Krishnan, "Separable routing: A scheme for state-dependent routing of circuit-switched telephone traffic," *Annals Oper. Res.*, vol. 35, pp. 43–68, 1992.
[33] P. Reichl, Eine allgemeine untere "Schranke für die Verlustrate in nichtsymmetrischen Netzwerken," Diplomarbeit, 1993, Inst. Angewandte Math., Statistik, Tech. Univ, München.
[34] K. W. Ross, Multiservice loss models for broadband telecommunication networks. Draft manuscript, 1995.
[35] S. M. Ross, *Introduction to Stochastic Dynamic Programming*. New York: Academic, 1983.
[36] P. Whittle, *Systems in Stochastic Equilibrium*. Chichester, England: Wiley, 1986.

R. J. Gibbens received the B.A. degree in mathematics, the Diploma in mathematical statistics, and the Ph.D. degree from the University of Cambridge, Cambridge, England, in 1983, 1984, and 1988, respectively.

From 1988 to 1993, he worked as a Research Associate, and since 1993 he has held a Royal Society University Research Fellowship in the Statistical Laboratory at the University of Cambridge. His research interests are in the area of mathematical modeling of telecommunication systems, especially the design of dynamic routing schemes. He is a co-inventor of the Dynamic Alternative Routing (DAR) strategy.

F. P. Kelly received the B.Sc. degree from the University of Durham in 1971, and the Ph.D. degree from the University of Cambridge in 1976.

He is now Professor of the Mathematics of Systems at the University of Cambridge. His main research interests are in random processes, networks and optimization, and especially in applications to the design and control of communication networks. He is a co-inventor of the Dynamic Alternative Routing (DAR) strategy.

Dr. Kelly has been awarded the Guy Medal in Silver of the Royal Statistical Society and the Lanchester Prize of the Operations Research Society of America. He is a Fellow of the Royal Society.

A New Degree of Freedom in ATM Network Dimensioning: Optimizing the Logical Configuration

András Faragó, Søren Blaabjerg, László Ast, Géza Gordos, and Tamás Henk

Abstract— **A mathematical model is presented that provides a well-defined formulation of the *logical configuration problem* of ATM networks with the objective of maximizing the total expected network revenue, given the physical network parameters and the traffic requirements of each virtual subnetwork. A two-phase solution procedure is developed in which the decision variables are the logical link capacities that specify the logical decomposition into virtual subnetworks, and the load sharing parameters. The first phase of the solution finds a global optimum in a rougher model. The second phase uses this as an initial point for a gradient-based hill climbing that applies the partial derivatives of the network revenue function obtained in a more refined model.**

I. Introduction

IT is expected that in large ATM networks, the carriers of future B-ISDN, a new degree of freedom appears in the design, dimensioning and management of the network: On top of the physical infrastructure a number of *logical* or *virtual subnetworks* can coexist, sharing the same physical transmission and switching capacities.

The simplest and most well-known example is the standardized concept of the *virtual path* that can be regarded as a very special virtual subnetwork. More complex examples arise, however, when *virtual leased networks* and *virtual LAN's* are considered.

Another reason for configuring virtual subnetworks comes from the fact, gradually recognized in the last couple of years, that it is not at all easy to integrate services with *very different demands* to e.g., bandwidth, grade of service (GoS) or congestion control functions. In some cases it turns out to be easier to support different services by offering separate logical networks, and limiting the degree of integration to only partial, rather than complete, sharing of physical transmission and switching resources. For example, delay sensitive and loss sensitive service classes can be managed and switched easier if the two groups are handled separately in different logical subnetworks, rather than all mixed on a complete sharing basis. Moreover, in this way they can be safely handled on call level without going down to cell level, as e.g., in priority queues. Of course, within a virtual subnetwork statistical multiplexing, priority queuing and other mechanisms can still be applied among service classes that already have not too different nature.

Since the virtual subnetworks share the same given physical capacities, therefore, there is a trade-off between their quality: GoS parameters, call blocking probabilities, etc., in one of the subnetworks can be improved only at the price of degrading some others. Moreover, the overall quality is also affected by the distribution of traffic among different routes even within a single subnetwork. It is a highly nontrivial task how to find the logical configuration, that is, the partition into virtual subnetworks along with the appropriate load sharing, such that given demands and constraints are satisfied and the overall network performance is optimized.

At first glance it might appear that partitioning, as opposed to complete sharing, is a serious reduction of the full flexibility of ATM. This is, however, not necessarily the case if the "partitioning" is viewed on a more general level. To explain this, let us mention the elegant and simple *complete sharing* multiplexing scheme of J. Roberts [10], called virtual spacing. In this queuing discipline various rates are assigned to different "streams" of traffic (a stream is like a logical link in our treatment) and it is guaranteed that each stream can forward cells at least at the specified rate. Thus, this discipline can realize complete resource sharing on the cell level with attractive simplicity. On the other hand, the scheme assumes that the assigned rates are already given and nothing is told about how to set these rates, beyond the requirement that their sum cannot exceed the physical capacity on any given link. Our approach can be applied to the problem of setting these rates, as well, if the blocking measures are chosen appropriately.

Thus, on a conceptual level, we can say: the complete sharing schemes, e.g., priority queuing, virtual spacing, etc., tell us how to *implement* resource sharing at the cell level, while our approach seeks for the *call scale* characteristics (e.g., how to assign rates to various streams) that is then to be realized on the cell level. In this sense our approach complements, rather than excludes, the complete sharing approaches.

In this paper a mathematical model is presented to provide a well-defined formulation of the above problem. The chosen objective is to maximize the *total network revenue,* i.e., the weighted version of the total expected carried traffic, given the physical network parameters and the traffic requirements of each virtual subnetwork. The decision variables are the logical link capacities that specify the decomposition into virtual subnetworks, and the load sharing parameters. The latter tell us how to share the load among routes that connect the same endpoints.

Manuscript received September 30, 1994; revised April 1, 1995.

A. Faragó, L. Ast, G. Gordos, and T. Henk are with the Department of Telecommunications and Telematics, Technical University of Budapest, XI. Stoczek u. 2., Budapest, Hungary H-1111.

S. Blaabjerg is with Ellemtel Telecommunication Systems Laboratories, S-22370 Lund, Sweden.

IEEE Log Number 9413106.