A Unified Approach to Network Survivability for Teletraffic Networks: Models, Algorithms and Analysis

D. Medhi, *Member, IEEE*

Abstract— **In this paper, we address the problem of network survivability by presenting a unified approach where the wide-area circuit-switched teletraffic network and the underlying transmission facility network are considered simultaneously. We assume the backbone circuit-switched teletraffic network to be nonhierarchical with dynamic call routing capabilities. The transmission facility network is considered to be sparse (as is observed for emerging fiber optic networks) and is assumed to be two-arc connected. Our approach addresses the network survivability objective by considering two grade-of-service parameters: one for the traffic network under normal operating condition and the other for affected part of the network under a network failure. We present unified mathematical models and develop heuristic algorithms. We then present computational results to demonstrate the effectiveness of the unified approach.**

I. Introduction

THE planning process for telecommunications networks can be categorized into the following phases: topological design, traffic routing and dimensioning in the switched traffic network (network synthesis), and the circuit routing design in the transmission facility network [9]. Traditionally the output of the topological optimization becomes input to the network synthesis problem, and in turn, the output of the network synthesis becomes input to facility (circuit) routing design. Although there is a loose coupling between these two processes and the topological design [9] of an iterative nature, the telecommunications switched traffic network and the underlying transmission facility network are, in practice, designed independently. (Here, the traffic network refers to the logical network where different services [e.g. voice, data, video] are offered, and the transmission facility network refers to the physical network through which the connectivity for the traffic network is provided. For brevity, we will refer to the transmission facility network as the facility network). The design issue for the traffic network is primarily limited to providing optimal trunk capacity subject to an acceptable level of blocking for the network in normal state (for example, see [1]). Similarly, the circuit routing design for the facility network is based on minimum-cost routing and other variations [7], [24], [25], [26], [28], [29] (See [7] for more references). However, as the facility network rapidly changes towards a fiber optic-based network, the graph of the network is becoming sparse. As a result, a single transmission link in such a network can carry a significant amount of traffic. Failure of such a link can cause major disruption of services. Thus, it is becoming imperative that in such an environment, the network is to be designed for survivability, i.e., so that a certain acceptable percentage of the traffic can still be carried immediately after a failure.

Recently, several researchers have addressed the issue of designing various networks for survivability [3], [6], [8], [13], [19], [26], [27]. These studies address the design in terms of either the traffic network or the facility network. However, since the traffic network exists purely at the logical level, design of a survivable traffic network may not be adequately addressed without incorporating the connectivity aspect of the facility network. For example, consider the case of two logically diverse traffic routes — one direct and another switched via a tandem — between a pair of switching nodes. Although they are logically diverse, they may actually use the same transmission facility, thereby ruling out the ability to carry traffic in either of the two routes in case of an intermediate transmission facility link failure between these two end nodes. Recently, Ash, Chang and Medhi [2] presented a robust traffic design method for dynamic routing networks. In their work, models to design for survivability are presented for nonhierarchical teletraffic networks with dynamic (call) routing capabilities (which use at most two traffic links for connecting a call) (For a survey on dynamic call routing, see [9], [10]). This work incorporated the underlying facility network for survivable design of the traffic network by *implicitly* assuming diverse facility routes.

Here, we present a unified approach to network survivability by considering the wide-area circuit-switched traffic network and the transmission facility network simultaneously. This work is considered for backbone nonhierarchical teletraffic networks with dynamic call routing capabilities which use at most two traffic links for connecting a call. (Call routing is not to be confused with circuit routing). The facility network is considered to be sparse (as is observed for emerging fiber optic networks) with the assumption that the graph of this network is two-arc connected [5, p. 445]. Our approach addresses network survivability by considering both the traffic network and the facility network *explicitly* in an integrated model by addressing traffic routing and dimensioning for the traffic network,

Paper approved by I. Cidon, the Editor for Network Algorithms of the IEEE Communications Society. Manuscript received March 1, 1992; revised February 1993, August 16, 1993. A part of this paper was presented at the IEEE ICC'92, Chicago, IL, June 1992. This work was supported in part by the Sprint Corporation and the Missouri Department of Economic Development.
The author is with the Computer Science Telecommunications Program, University of Missouri–Kansas City, Kansas City, MO 64110 (email: dmedhi@cstp.umkc.edu).

and circuit routing for the facility network. For brevity, we refer to our approach as survivable teletraffic network design, or simply as survivable design. Note that our work does not consider the *fundamental* facility network design and planning [20] (for example, no facility capacity expansion), rather we consider existing facility network in this integrated model. We present models when no trunks are already in the network ("desert" model) and when there are trunks already in the network, but the design is on additional trunk augmentation to satisfy given requirements ("incremental" model). In both these cases, the capacity of the physical links is assumed to be given. For these models, we then present heuristic algorithms. A prototype tool has been developed to implement the algorithms. We present computational results on example networks extracted from actual networks. We present results on the effectiveness of survivable design by doing network simulation and comparing with networks designed under present modes of operation.

The remainder of this paper is organized as follows. In Section II, we describe unified network models combining the switched traffic network and the transmission facility network. We present heuristic algorithms for the design models in section III. In section IV, we give computational results on network designed without and with survivability objective and present simulation results to show the effectiveness of survivable design.

II. Unified Network Models

We consider here the unified design of a nonhierarchical dynamic routing backbone traffic network together with a sparse facility network to carry traffic both for normal network and affected network (for a transmission link failure) while minimizing total cost. The level of traffic to be carried can be provided through two grade-of-service (GOS) parameters: acceptable level of blocking under normal condition (normal GOS, or nGOS) and acceptable level of blocking under failure condition (failure GOS, or fGOS). Explicit incorporation of the facility network is considered here to address survivable network design by combining traffic network routing and dimensioning together with circuit routing for the facility network. For the traffic network, we are given a set of traffic switching nodes, the traffic matrix between the switching nodes for different load periods (hours) during the day, unit cost of trunks on each traffic link, normal GOS and failure GOS. For the facility network, we assume that we have the set of facility nodes (cross-connect), the set of transmission links, the maximum capacity on these links, and the unit cost of circuit for different transmission paths in the network. We assume that the facility network is two-arc connected. This makes it possible to have at least one transmission path available between two facility nodes in case of failure of a transmission link. We further assume that the circuit layout remains static during the course of a day while the call routing is dynamic, varying as the traffic changes from one instance to another during the day. Note that the switching node sites may not necessarily coincide with the facility node sites. A traffic link (also known as trunk group), which is different from a facility link, connects two switching nodes while a facility link connects two facility nodes. For clarity, we refer to a traffic link as a t-link and a facility link as an f-link. A traffic path consists of at most two t-links connecting a demand pair either directly or via another switch. A transmission path is the physical path of a logical t-link consisting of f-links connected by a chain. Clearly, a traffic path is different from a transmission facility path. In Fig. 1(a) and Fig. 1(b), we illustrate an example showing the logical traffic links between three nodes, considered to be a part of a larger network. We assume that facility nodes are co-located with switching nodes for A, V and B. For example, a call between traffic nodes A and B can be connected using the direct t-link A–B or switched via node V; in the latter case, the call uses trunks from the t-links A–V and V–B. Fig. 1(b) depicts the underlying facility network corresponding to Fig. 1(a) showing the f-links. Here the trunks between nodes A and B may be circuit-routed using the facility paths A–F_2–F_3–B, A–F_1–V–B or A–F_2–V–B or a combination of them. Similarly, group A–V can use A–F_1–V, A–F_2–V, and group V–B can use V–B, V–F_2–F_3–B. This example shows the difference between the traffic network and the facility network as well as the difference between call routing and circuit routing.

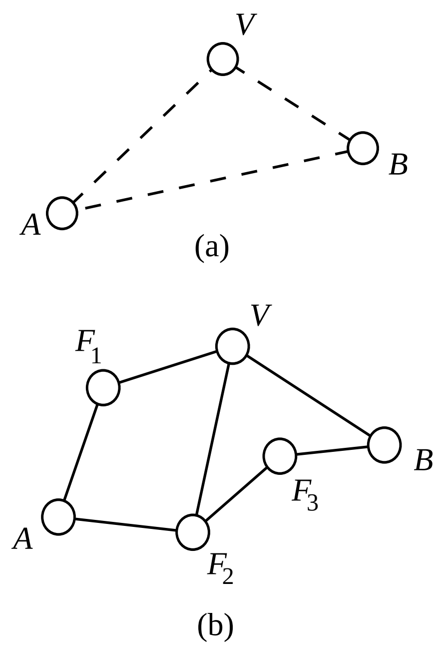

Fig. 1. (a) Partial view of a traffic network, (b) Underlying transmission facility network

The network can be in different states due to different types of failure. In our formulation, we only consider failure states due to f-link failures assuming that the failures take place one at a time. Thus, we consider the set of possible states of the network to include the normal network (i.e., when the network is intact) and the states of the network due to failure of each f-link (separately). For the normal state, the problem is considered in the full graph of the network, while for the affected network states, the problem is considered in the subgraph consisting of the graph derived from the original network minus the failed f-link. Consequently, different sets of candidate traffic and transmission facility paths are generated for each of the states. Due to restrictions on possible paths at various states, we consider an arc-path formulation approach. We can write flow equations for given requirements for each of these states. Requirements are based on how much load is to be carried under both normal and failure situations for given blocking levels (GOS). To present the mathematical formulations for the problem, we first define notation. For clarity, the notation has been classified under the traffic network and the facility network.

$\mathcal{E}$ Set of states (denoted by σ) of the network to be considered [$\sigma = 0 (\in \mathcal{E})$ is the normal state of the network]

Traffic network:

- $\mathcal{K}$ Set of traffic node pairs
- $\mathcal{L}$ Set of traffic links (trunk groups)
- $\mathcal{H}$ Set of traffic load periods (hours)
- $\mathcal{J}_k^{\sigma h}$ Set of traffic paths for node pair $k \in \mathcal{K}$ in hour $h \in \mathcal{H}$ in network state $\sigma \in \mathcal{E}$
- c_i Unit cost of trunk on traffic link $i \in \mathcal{L}$
- y_i^σ Number of trunks needed for traffic link $i \in \mathcal{L}$ in state σ (variable)
- y_i Maximum number of trunks needed for traffic link $i \in \mathcal{L}$ (variable)
- $r_{kj}^{\sigma h}$ Path variable in traffic network – amount of flow on traffic path j for node pair k in hour h in network state σ (variable)
- $\delta_{kj}^{\sigma ih}$ Entries for arc-path incidence matrix for traffic network in network state σ; 1 if traffic path j for node pair k uses link i in hour h in network state σ, 0 otherwise
- a_k^h Traffic offered load (in erlangs) for node pair k in hour h
- $B_k^{\sigma h}$ Blocking level for node pair k in hour h in state σ (to specify nGOS and fGOS)
- $v_{kj}^{\sigma h}$ Upper bound corresponding to the traffic path variable $r_{kj}^{\sigma h}$
- θ Diversity bound parameter ($0 < \theta \leq 1$). Derived parameter value θ_{ij}^σ is defined as θ for diverse facility paths in $\sigma = 0$ or for unaffected facility path j when $\sigma \neq 0$

Facility network:

- $\mathcal{F}$ Set of facility links in the physical network
- $\mathcal{L}_i$ Set of facility paths for traffic link i
- s_{ij} Maximum circuit flow on facility path $j \in \mathcal{L}_i$ for traffic link $i \in \mathcal{L}$ (variable)
- s_{ij}^σ Circuit flow on facility path j for traffic link i in state σ (variable)
- $\Delta_{ij}^{\sigma \ell}$ Incidence matrix for the facility network– 1 if the facility path j for the traffic link i uses the f-link ℓ in the network state σ
- u_ℓ^σ Maximum capacity on f-link $\ell \in \mathcal{F}$ available in network state σ
- f_{ij} Unit cost of circuit on facility path j for traffic link i

A. Desert model

First we present a model when no trunks are already in the network, though the capacity on physical links are assumed. We refer to this optimization model as the "desert" model or Model-A. The goal here is to minimize total trunk and circuit routing cost so as to design a network for a given traffic survivability objective (through nGOS and fGOS). Specifically, the network under normal operating condition is to satisfy nGOS; in case of a failure, the unaffected pairs are to satisfy nGOS and directly affected pairs are to satisfy fGOS.

Desert model (Model-A):

$$\min_{\{y_i, y_i^\sigma, r_{kj}^{\sigma h}, s_{ij}, s_{ij}^\sigma\}} \quad \sum_{i \in \mathcal{L}} c_i y_i + \sum_{i \in \mathcal{L}} \sum_{j \in \mathcal{L}_i} f_{ij} s_{ij} \qquad (A1)$$

subject to

$$\sum_{j \in \mathcal{J}_k^{\sigma h}} r_{kj}^{\sigma h} = vt(\sigma, a_k^h, B_k^{\sigma h}), \quad k \in \mathcal{K}; h \in \mathcal{H}; \sigma \in \mathcal{E} \qquad (A2)$$

$$\sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_k^{\sigma h}} \delta_{kj}^{\sigma ih} r_{kj}^{\sigma h} \leq y_i^\sigma, \qquad i \in \mathcal{L}; h \in \mathcal{H}; \sigma \in \mathcal{E} \qquad (A3)$$

$$\sum_{j \in \mathcal{L}_i} s_{ij}^\sigma = y_i^\sigma, \qquad i \in \mathcal{L}; \sigma \in \mathcal{E} \qquad (A4)$$

$$\sum_{i \in \mathcal{L}} \sum_{j \in \mathcal{L}_i} \Delta_{ij}^{\sigma \ell} s_{ij}^\sigma \leq u_\ell^\sigma, \qquad \ell \in \mathcal{F}; \ \sigma \in \mathcal{E} \qquad (A5)$$

$$s_{ij}^\sigma \leq \theta_{ij}^\sigma y_i^\sigma, \qquad j \in \mathcal{L}_i; i \in \mathcal{L}; h \in \mathcal{H}; \sigma \in \mathcal{E} \qquad (A6)$$

$$0 \leq y_i^\sigma \leq y_i, \qquad i \in \mathcal{L}; \ \sigma \in \mathcal{E} \qquad (A7)$$

$$0 \leq s_{ij}^\sigma \leq s_{ij}, \qquad i \in \mathcal{L}; \ \sigma \in \mathcal{E} \qquad (A8)$$

$$0 \leq r_{kj}^{\sigma h} \leq v_{kj}^{\sigma h}, \qquad j \in \mathcal{J}_k^{\sigma h}; k \in \mathcal{K}; h \in \mathcal{H}; \sigma \in \mathcal{E} \qquad (A9)$$

In Model-A, expression ($A1$) represents the total cost due to trunking and circuit routing. Equation ($A2$) refers to satisfying traffic demand on various traffic paths at different load set periods during a day (this reflects variation of a traffic during a day). The function, vt, is used to compute the initial virtual trunk requirement for given traffic load in erlangs, the state of the network, and survivability level for that state of the network. The concept of a virtual trunk is used since a call may require one-link or two-link trunks for completion (discussed later). ($A3$) shows the trunk capacity obtained for each traffic link based on the traffic flow in a particular state. Equation ($A4$) enforces flow requirements of each traffic link on facility paths. ($A5$) assures that requirements on f-links do not go over upper limits. Constraints ($A6$) enforces imposition of any diversity requirement on facility paths. Constraints ($A7$) & ($A8$) define maximum over all states for the variables y and s, while ($A9$) gives bounds on traffic path variables. Finally, left parts of ($A7$), ($A8$), ($A9$) indicate that the variables are non-negative in value. In this and the subsequent models, we assume that the variables are continuous (a post-processing procedure can be used to obtain integral values;

see §IV.A). Note that for a fully inter-connected traffic network, the number of traffic links is equal to the number of traffic node pairs [i.e., $\#(\mathcal{L}) = \#(\mathcal{K})$; here # denotes the cardinality of a set]. Note also that total number of switching nodes may differ from total number of facility nodes; this is due to the possible existence of facility hubs (without any associated switches) and/or due to more than one switches using a facility node for incoming/outgoing trunks (for example, two switches in a big city may be connected to one facility node).

The desert model combines traffic routing and dimensioning with circuit routing for survivable teletraffic network design. Note that for the traffic network routing and dimensioning, the unified algorithm (UA) for DNHR as described in [1] can be employed. Instead, following [2], the concept of virtual trunk is used to initially approximate trunk requirements for given offered load (in erlangs) and blocking requirements. This approach is found to be a reasonable approximation for large networks as noted in [2]. Another major aspect to be noted in the desert model is that the circuit routing problem is explicitly modeled for addressing survivability. Finally, by incorporating traffic variation during a day, this design models ensures that GOS objectives are met no matter what time of the day a failure occurs.

B. Incremental model

The desert model is the basic model for the unified network design. Here we present an extension of the desert model to be referred as the incremental model. We use the term incremental model in the sense that it assumes a given network in the beginning of a planning cycle and seeks an optimal design based on this network. Thus, we start with a network which has an initial number of trunks from a previous planning period/cycle and know to which physical routes (and quantity) these initial trunks are routed. Thus, the goal for the current planning cycle is to optimize based on this initial network.

Variables defined in the desert model are used in the incremental model. However, the following have different meaning:

- t_i *Initial* number of trunks on link i already in the network
- t_i^σ Number of trunks available on link i in the network in state σ
- y_i^σ *Incremental* number of trunks needed for traffic link i in state σ (variable)
- y_i Maximum *incremental* number of trunks needed for traffic link i over all states (variable)
- s_{ij}^σ *Additional* circuits flow on path j for the pair i in state σ (variable)
- s_{ij} *Additional* circuits flow on path j for the pair i (variable)

Since y_i^σ is now the incremental trunk capacity, equation $(B4)$ refers to circuit routing for any additional trunk demands. Thus, final s_{ij} together with the initial number of circuits routed on the transmission paths give the total number of circuits on the transmission paths at the end of a planning run.

Incremental model (Model-B):

$$\min_{\{y_i, y_i^\sigma, r_{kj}^{\sigma h}, s_{ij}, s_{ij}^\sigma\}} \sum_{i\in\mathcal{L}} c_i y_i + \sum_{i\in\mathcal{L}}\sum_{j\in\mathcal{L}_i} f_{ij} s_{ij} \qquad (B1)$$

subject to

$$\sum_{j\in\mathcal{J}_k^{\sigma h}} r_{kj}^{\sigma h} = vt(\sigma, a_k^h, B_k^{\sigma h}), \quad k\in\mathcal{K}; h\in\mathcal{H}; \sigma\in\mathcal{E} \qquad (B2)$$

$$\sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{J}_k^{\sigma h}} \delta_{kj}^{\sigma i h} r_{kj}^{\sigma h} \le t_i^\sigma + y_i^\sigma, \quad i\in\mathcal{L}; h\in\mathcal{H}; \sigma\in\mathcal{E} \qquad (B3)$$

$$\sum_{j\in\mathcal{L}_i} s_{ij}^\sigma = y_i^\sigma, \qquad i\in\mathcal{L}; \sigma\in\mathcal{E} \qquad (B4)$$

$$\sum_{i\in\mathcal{L}}\sum_{j\in\mathcal{L}_i} \Delta_{ij}^{\sigma\ell} s_{ij}^\sigma \le u_\ell^\sigma, \qquad \ell\in\mathcal{F};\ \sigma\in\mathcal{E} \qquad (B5)$$

$$s_{ij}^\sigma \le \theta_{ij}^\sigma y_i^\sigma, \qquad j\in\mathcal{L}_i; i\in\mathcal{L}; h\in\mathcal{H}; \sigma\in\mathcal{E} \qquad (B6)$$

$$0 \le y_i^\sigma \le y_i, \qquad i\in\mathcal{L};\ \sigma\in\mathcal{E} \qquad (B7)$$

$$0 \le s_{ij}^\sigma \le s_{ij}, \qquad i\in\mathcal{L};\ \sigma\in\mathcal{E} \qquad (B8)$$

$$0 \le r_{kj}^{\sigma h} \le v_{kj}^{\sigma h}, \qquad j\in\mathcal{J}_k^{\sigma h}; k\in\mathcal{K}; h\in\mathcal{H}; \sigma\in\mathcal{E} \qquad (B9)$$

C. Virtual trunk (VT) calculation

For a given offered load, virtual trunk requirement is computed based on the offered load, network state and the GOS level. The load to be carried in a dynamic routing environment can be carried in two ways: on a direct (one t-link) traffic path and on two t-link traffic paths. If we assume a blocking b_d of the load to be carried on the direct path, then the overflow traffic can use two t-link paths to complete the requirements. This overflow traffic shares trunks from other traffic pairs. Conceptually, although it uses two t-links, the overflow traffic can be visualized as being carried on a shared virtual trunk group. Thus, the total requirement over the direct link and alternate two t-link paths is the total virtual trunk required for the offered load to be carried. For a normal network, the GOS used is the nGOS; this is reflected by B_k^{0h} (for normal state, $\sigma = 0$) for the demand pair k in hour h. For affected pairs in failure states, the blocking level $B_k^{\sigma h}$ (for failure state $\sigma \ne 0$) uses fGOS. Let $B(c, a)$ be the well-known Erlang blocking formula (see, for example, [9]) for trunk c and offered load a defined by

$$B(c, a) = \frac{a^c/c!}{\sum_{k=0}^{c}(a^k/k!)}.$$

We denote the number of trunks necessary to carry offered load a at a particular blocking level b by the inverse function $B^{-1}(a, b)$. Let the average occupancy of a trunk group be ρ. Finally, let the carried load be $a' = a(1 - B_k^{0h})$. Then the vt demand is approximated by the following formula:

$$vt(\sigma, a, B_k^{\sigma h}) = \begin{cases} B^{-1}(a', b_d) + \frac{a' \cdot b_d}{\rho}, & \text{if } \sigma = 0 \text{ or pair } k \text{ is unaffected when } \sigma \neq 0; \\ B^{-1}(a, B_k^{\sigma h}), & \text{otherwise.} \end{cases} \quad (1)$$

The approximation used for the normal state has been used in other works [2], [21].

III. Heuristic Algorithms

The models presented are large-scale optimization models due to the variables involved for traffic routing, circuit (facility) routing and trunk dimensioning. To give some idea, we first consider the size of a problem. Consider a network with N_t switching nodes, N_f facility nodes. Let $F = \#(\mathcal{F})$ and $H = \#(\mathcal{H})$. Then for an H-traffic load period, fully interconnected traffic network (i.e., number of traffic pairs = number of traffic links = $N_t(N_t - 1)/2$) with average number of candidate traffic paths per pair = p_t and average number of candidate facility paths per pair = p_f, the total number of constraints for the linear program of Model-A is $N_t(N_t - 1)(F + 1)(H + 1 + p_f) + F(F + 1)$ [not counting pure bound constraints], and the number of variables is $N_t(N_t - 1)((F + 1)(Hp_t + 1 + p_f) + p_f + 1)/2$. If we set p_t to be $\lfloor N_t/2 \rfloor$ and p_f to be $\lfloor N_f/2 \rfloor$, then number of constraints and variables become $N_t(N_t - 1)(F + 1)(H + 1 + \lfloor N_f/2 \rfloor) + F(F + 1)$, and $N_t(N_t - 1)((F + 1)(H\lfloor N_t/2 \rfloor + 1 + \lfloor N_f/2 \rfloor) + \lfloor N_f/2 \rfloor + 1)/2$, respectively. For three example networks (see Table I), we show the problem sizes with Model-A in Table II. From this table, it is clear that the problems size becomes unmanageable even for a fairly small problem. In the following section, we present a heuristic algorithm for Model-A where the survivable design problem is considered at each state separately to arrive at a more manageable optimization model at each state.

Table I
Size of Example Networks

Example Network	Switching Node (N_t)	Facility Node (N_f)	Facility Link (F)
EN-1	7	10	14
EN-2	10	18	27
EN-3	15	23	33

Table II
Problem Size for Three Examples with Model-A
(for $H = 3$, $p_t = \lfloor N_t/2 \rfloor$, $p_f = \lfloor N_f/2 \rfloor$, $\#(\mathcal{L}) = \#(\mathcal{K})$)

Example Network	Number of constraints	Number of variables
EN-1	5,880	5,796
EN-2	33,516	31,950
EN-3	108,222	119,070

A. Heuristic Algorithm for Model-A

In this approach, we start with the network in the normal state. We first obtain the optimal trunk requirements in the switched network part and do an optimal circuit routing for these requirements. The network under the normal state is designed for nGOS, assuming no failure. Once this initial network is obtained, then we address the issue of facility link failures. Failure of a facility link can affect several traffic paths and traffic node pairs, especially in a sparse physical network environment. Thus, a mapping from logical network (switched traffic) to the physical network (transmission facility) and back is required. From the initial design for the network under normal conditions, we use a heuristic algorithm based on the concept of a constraint set generation approach to address facility link failures. (This approach is somewhat different from the constraint generation approach used by other researchers in the context of various network design problems [15].) In our approach, the *constraint set generation* process is as follows: For each of the failure states (corresponding to each transmission facility link) considered one at a time, we generate a set of constraints which enforces the requirements for that state of the network and then solves the traffic and facility network design problem on the subgraph. Once the problem for a state is solved, the procedure updates any augmented trunk capacity and associated circuit layout and then moves to the next state. It may noted that any augmentation at a state is never reduced in any subsequent states. The process is continued until all the states are done. Thus, all the states are considered one at a time to solve the entire model.

We now discuss the design problem at each state. At each state of the network (normal or one of the failure states) we solve the problem of the traffic routing and dimensioning together with circuit routing. Note that at each failure state, we are operating on a subgraph deduced from the initial graph by deleting the failed f-link which in turn affects multiple t-links. At each failure state, candidate traffic paths that contain any affected t-links are marked as unsuitable for alternate call routing. Accordingly, the set of (traffic and facility) paths are different at each state. For brevity, we drop the state index σ from the model below and assume that the VT capacity is already computed using (1) and is an input to this stage. For clarity, we list notation.

- $\mathcal{J}_k^h$ Set of acceptable traffic paths for node pair $k \in \mathcal{K}$ in hour $h \in \mathcal{H}$
- t_i Available trunks on traffic link i
- y_i Number of trunks needed for traffic link $i \in \mathcal{L}$ (variable)
- r_{kj}^h Path variable in traffic network – amount of flow on traffic path j for node pair k in hour h (variable)
- δ_{kj}^{ih} Entries for arc-path incidence matrix for traffic network ; 1 if traffic path j for node pair k uses link i in hour h, 0 otherwise
- $(vt)_k^h$ Virtual trunk demand for node pair k in hour h computed using (1)

v_{kj}^h Upper bound corresponding to the traffic path variable r_{kj}^h
$\mathcal{F}$ Set of f-links in the facility network
$\mathcal{L}_i$ Set of acceptable facility paths for traffic link i
s_{ij} circuit flow on facility path $j \in \mathcal{L}_i$ for traffic link $i \in \mathcal{L}$ (variable)
Δ_{ij}^ℓ Incidence matrix for the facility network– 1 if the facility path j for the traffic link i uses the f-link ℓ
u_ℓ Available capacity on facility link $\ell \in \mathcal{F}$
f_{ij} Unit cost of circuit on facility path j for traffic link i
θ_{ij} Diversity bound parameter on path j for traffic link i

For this state of the network, the combined traffic and facility network design problem can be formulated as follows:

Model-C:

$$\min_{y_i, r_{kj}^h, s_{ij}} \quad \sum_{i \in \mathcal{L}} c_i y_i + \sum_{i \in \mathcal{L}} \sum_{j \in \mathcal{L}_i} f_{ij} s_{ij} \qquad (C1)$$

subject to

$$\sum_{j \in \mathcal{J}_k^h} r_{kj}^h = (vt)_k^h, \quad k \in \mathcal{K}; h \in \mathcal{H} \qquad (C2)$$

$$\sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_k^h} \delta_{kj}^{ih} r_{kj}^h \le t_i + y_i, \qquad i \in \mathcal{L}; h \in \mathcal{H} \qquad (C3)$$

$$\sum_{j \in \mathcal{L}_i} s_{ij} = y_i, \qquad i \in \mathcal{L} \qquad (C4)$$

$$\sum_{i \in \mathcal{L}} \sum_{j \in \mathcal{L}_i} \Delta_{ij}^\ell s_{ij} \le u_\ell, \qquad \ell \in \mathcal{F} \qquad (C5)$$

$$0 \le s_{ij} \le \theta_{ij} y_i, \qquad j \in \mathcal{L}_i; i \in \mathcal{L} \qquad (C6)$$

$$0 \le r_{kj}^h \le v_{kj}^h, \qquad j \in \mathcal{J}_k^h; k \in \mathcal{K}; h \in \mathcal{H} \qquad (C7)$$

$$y_i \ge 0, \qquad i \in \mathcal{L} \qquad (C8)$$

Various constraints in Model-C can be interpreted similar to Model-A. We now discuss the size of Model-C. In this model (using the same notation as in the beginning of this section and assuming that the traffic network is fully interconnected), the number of constraints is $N_t(N_t - 1)(2H + 1 + \lfloor N_f/2 \rfloor)/2 + F$ and the number of variables $N_t(N_t - 1)(H\lfloor N_t/2 \rfloor + 1 + \lfloor N_f/2 \rfloor)/2$. In Table III, we give the size of optimization model-C for the same three example networks (Table I); compare their problem sizes to the ones with Model-A in Table II. The size of problems in Model-C is more manageable than the size of the entire problem as given in Model-A although the heuristic procedure requires solving Model-C for each state at a time.

Finally, Model-C can be solved by another approximation based on the following observation: 1) taking advantage of the natural relation between the traffic network and the facility network, one can solve the traffic network part first and then solve the facility network part, 2) in each failure state, only a fraction of the traffic pairs requires trunk augmentation which results in solution of the facility design (circuit routing) part only for these pairs. Thus, instead of Model-C, the following model can be solved for the traffic network part:

TABLE III
PROBLEM SIZE FOR THREE EXAMPLES WITH MODEL-C (IN EACH STATE)
($H = 3$, $p_t = \lfloor N_t/2 \rfloor$, $p_f = \lfloor N_f/2 \rfloor$, $\#(\mathcal{L}) = \#(\mathcal{K})$)

Example Network	Number of constraints	Number of variables
EN-1	266	378
EN-2	747	1,125
EN-3	1,923	3,465

Model-D:

$$\min_{y_i, r_{kj}^h} \quad \sum_{i \in \mathcal{L}} c_i y_i \qquad (D1)$$

subject to

$$\sum_{j \in \mathcal{J}_k^h} r_{kj}^h = (vt)_k^h, \quad k \in \mathcal{K}; h \in \mathcal{H} \qquad (D2)$$

$$\sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_k^h} \delta_{kj}^{ih} r_{kj}^h \le t_i + y_i, \quad i \in \mathcal{L}; h \in \mathcal{H} \qquad (D3)$$

$$0 \le r_{kj}^h \le v_{kj}^h, \quad j \in \mathcal{J}_k^h; k \in \mathcal{K}; h \in \mathcal{H} \qquad (D4)$$

$$y_i \ge 0, \qquad i \in \mathcal{L}. \qquad (D5)$$

Let the set for which the optimal solution $y_i > 0$ be $\hat{\mathcal{L}}$ ($\subseteq \mathcal{L}$), and let the solution be denoted by $\hat{y}_i, i \in \hat{\mathcal{L}}$. Then the following model can be solved for the facility network design part given $\hat{y}_i$:

Model-E:

$$\min_{s_{ij}} \sum_{i \in \hat{\mathcal{L}}} \sum_{j \in \hat{\mathcal{L}}_i} f_{ij} s_{ij} \qquad (E1)$$

subject to

$$\sum_{j \in \hat{\mathcal{L}}_i} s_{ij} = \hat{y}_i, \qquad i \in \hat{\mathcal{L}} \qquad (E2)$$

$$\sum_{i \in \hat{\mathcal{L}}} \sum_{j \in \hat{\mathcal{L}}_i} \Delta_{ij}^\ell s_{ij} \le u_\ell, \qquad \ell \in \mathcal{F} \qquad (E3)$$

$$0 \le s_{ij} \le \theta_{ij} \hat{y}_i, \qquad j \in \hat{\mathcal{L}}_i; i \in \hat{\mathcal{L}}. \qquad (E4)$$

Model-D/E approximation for Model-C requires solving a problem with $N_t(N_t - 1)H$ constraints and $N_t(N_t - 1)(H\lfloor N_t/2 \rfloor + 1)/2$ variables for the (fully interconnected) traffic network part, and then to solve a problem with $\gamma N_t(N_t - 1)/2 + F$ constraints and $\gamma N_t\,(N_t - 1)\lfloor N_f/2 \rfloor/2$ variables for the facility network part, where γ is the fraction of pairs that require circuit routing. From computational experimentations, we have found that for some failure states, there is no trunk augmentation at all (usually when the trunk lost is not that much). This means there is no need to solve the facility network part as $\hat{\mathcal{L}} = \emptyset$ for these states. Among the failure states in which there are trunk augmentation, we have found that, on average, about 11% of the traffic pairs require circuit layout. In Table IV, we give the size of problems for Model D/E for the same three example networks (Table I). Given these observation, we propose solving Model-D/E (instead of Model-C) at each network state.

TABLE IV
PROBLEM SIZE FOR THREE EXAMPLES WITH MODEL-D/E (IN EACH STATE)
($H = 3$, $p_t = \lfloor N_t/2 \rfloor$, $p_f = \lfloor N_f/2 \rfloor$, $\#(\mathcal{L}) = \#(\mathcal{K})$)
(*ASSUMING 10% OF TRAFFIC PAIRS REQUIRE CIRCUIT ROUTING DURING A FAILURE STATE (IN THE CASE OF TRUNK AUGMENTATION))

	Model-D (traffic part)		Model-E (facility part)	
	No. of constraints	No. of variables	No. of constraints	No. of variables
EN-1	126	273	35	105
			16*	10*
EN-2	270	720	72	405
			32*	45*
EN-3	630	2,300	138	1,135
			44*	121*

The heuristic algorithm, HA-A, is presented in Fig. 2 in a systematic manner. In HA-A, two t-link traffic paths can be generated based on distance or cost requirements. Also, any switching node that is not desirable as via node for traffic routing can be taken into consideration in the traffic path generation procedure. Similarly, facility paths can be generated based on a k-shortest path algorithm [12]. For designing either at normal or at one of the failure states (i.e., Step 3.1 or Step 5.3.1), virtual trunk (VT) quantity is computed based on the offered load (in erlangs) using (1). The acceptable GOS (nGOS or fGOS) for each of the states is as described in the previous section. To summarize, we have arrived at this heuristic algorithm by making two approximations to the original model (Model-A): a) by handling each failure state sequentially and b) by separating the design problem at each failure state into two subproblems, one for the traffic network (Model-D) and the other for the facility network (Model-E). Note that in the algorithmic description, the notation $(a, \{b\}) \leftarrow DoFunction(c, \{d\})$ means that procedure *DoFunction* takes c and a set of data $\{d\}$ as input producing a and a set of data $\{b\}$ as output.

B. Algorithm for Model-B

The algorithm for solving the incremental model is similar to the desert model. However, for Model-B, we need to provide an initial network to the model. The initial network provides information about the present number of trunks in the traffic network and the present circuit routing of these trunks in the facility network. Thus, this changes mainly Step 0 where we now additionally require existing network with trunk capacity and circuit routing as input. Also, additional facility paths may be generated in step 2 if needed. For clarity, we call this algorithm HA-B. We have listed the steps of HA-B that are changed from HA-A in Fig. 3. Also, note that in HA-B, steps 3.3 and steps 3.4 are done if $\#(\hat{\mathcal{L}}) > 0$.

IV. COMPUTATIONAL STUDIES

The present mode of operation (PMO) for telecommunications networks is usually independent design of the traffic network and the facility network; the traffic network is designed primarily for GOS under a normal operating condition with little or no information concerning reliability; for the facility network, a minimum cost circuit layout is applied without or with diversity requirement. We have done two sets of computational work here: in the first set, we have obtained design results based on algorithm HA-A and compared with two scenarios for PMO in terms of cost; and in the second set, we have obtained results on impact of a facility link failure using a call-by-call traffic simulator to see the effectiveness of survivable design compared to present modes of design.

A. Design Results

A prototype tool, UTAFNET (for Unified Traffic and Facility NETwork design), has been developed implementing heuristic algorithms HA-A and HA-B. In this prototype some additional issues are addressed. For example, enumeration of failure states (step 4) can be ordered in various ways. In our implementation, we have the following two: 1) one based on computing flow on facility links from the normal network design part and then sorting these links in descending order of flow; and 2) the other according to the order the facility links are provided in the input procedure (for example, we use alphanumeric names of f-links as the order in the input file). For brevity, we refer to these two rules as $ord = y$ and $ord = n$, respectively. Additionally, if a user wants to study the failure scenario for a specific f-link or a subset of all the possible f-links, this can be done by providing the appropriate directive using a spec file. The concept of fictitious facility paths (one for each pair) is also provided; the benefit is that if all the candidate facility paths generated (before the design phase) for a particular pair is affected due to an f-link failure, then the design procedure can still load the required trunks on the fictitious path, thereby letting the user know that an additional path is needed to route circuits. We have also added a simple integerization routine right before outputing design to obtain integral solution. (This may be replaced by a more sophisticated modularization routine if trunks are required to be a multiple of certain units such as DS1 (24 voice channels)). Linear programs of Model-D and Model-E are solved using MINOS [17]. Note that the incremental model can be useful for studying different network scenarios for strategic planning since this allows the user the flexibility to start with an initial network.

We have considered three example networks to study survivable design (these are the same three examples for which problem sizes have been discussed in the previous section.)

Algorithm : HA-A

<u>Step</u> *0: Input parameters and data for traffic and facility network.*

<u>Step</u> *1:* $\{\mathcal{J}_k^h\}_{k\in\mathcal{K},h\in\mathcal{H}} \longleftarrow \mathit{GenerateTrafficPaths}\left(N_t,\mathcal{K},\mathcal{L},\mathcal{H}\right)$

<u>Step</u> *2:* $\{\mathcal{L}_i\}_{i\in\mathcal{L}} \longleftarrow \mathit{GenerateFacilityPaths}\left(N_f,\mathcal{L},\mathcal{F}\right)$

<u>Step</u> *3: /* Normal network design (note:* $\sigma=0$*) */*

<u>Step</u> *3.1: /* Compute VT */*

$$\{vt_k^h\}_{k\in\mathcal{K},h\in\mathcal{H}} \longleftarrow \mathit{ComputeVT}\left(0,\{a_k^h\},\ nGOS,\ fGOS\right)$$

<u>Step</u> *3.2: /* Do traffic routing and dimensioning */*

$$\left(\hat{\mathcal{L}},\{y_i^{\text{aug}}\}_{i\in\hat{\mathcal{L}}},\ cost_t\right) \longleftarrow \mathit{SolveModelD}\left(0,\mathcal{L},\mathcal{K},\mathcal{H},\{\mathcal{J}_k^h\},\{c_i\},\{vt_k^h\},\{0\}_{i\in\mathcal{L}}\right)$$

<u>Step</u> *3.3: /* Do circuit layout */*

$$\left(\{s_{ij}^{\text{aug}}\}_{j\in\hat{\mathcal{L}}_i,\,i\in\hat{\mathcal{L}}},cost_f\right) \longleftarrow \mathit{SolveModelE}\left(0,\hat{\mathcal{L}},\mathcal{F},\{f_{ij}\},\{y_i^{\text{aug}}\}_{i\in\hat{\mathcal{L}}},\{u_\ell\},\theta\right)$$

<u>Step</u> *3.4: /* Update */*

$$y_i^{\text{now}} \longleftarrow y_i^{\text{aug}},\quad i\in\hat{\mathcal{L}};\qquad s_{ij}^{\text{now}} \longleftarrow s_{ij}^{\text{aug}},\quad j\in\hat{\mathcal{L}}_i, i\in\hat{\mathcal{L}}$$

$$u_\ell \longleftarrow u_\ell + \sum_{i\in\hat{\mathcal{L}}}\sum_{j\in\hat{\mathcal{L}}_i}\Delta_{ij}^\ell s_{ij}^{\text{aug}},\quad \ell\in\mathcal{F};\qquad cost_{\text{now}} \longleftarrow cost_t + cost_f$$

<u>Step</u> *4: Enumerate failure states in* $\mathcal{E}_f$

<u>Step</u> *5: For each* $\sigma\in\mathcal{E}_f$ *do*

<u>Step</u> *5.1: Identify acceptable sets* $\mathcal{L}^\sigma,\mathcal{J}_k^{h\sigma},\mathcal{L}_i^\sigma$ *based on failure state*

<u>Step</u> *5.2: /* Adjust availability */*

$$y_i^{\text{avail}} \longleftarrow y_i^{\text{now}} - \sum_{j\in\mathcal{L}_i\setminus\mathcal{L}_i^\sigma} s_{ij},\quad i\in\mathcal{L};\qquad u_\ell^{\text{avail}} \longleftarrow u_\ell,\ \ \ell\in\mathcal{F}\setminus\{\sigma\};\quad u_\sigma^{\text{avail}} \longleftarrow 0$$

<u>Step</u> *5.3: /* Design for this failure state */*

<u>Step</u> *5.3.1: /* Compute VT in this failure state */*

$$\{vt_k^h\}_{k\in\mathcal{K},h\in\mathcal{H}} \longleftarrow \mathit{ComputeVT}\left(\sigma,\{a_k^h\},\ nGOS,\ fGOS\right)$$

<u>Step</u> *5.3.2: /* Do traffic routing and trunk augmentation, if any */*

$$\left(\hat{\mathcal{L}}^\sigma,\ \{y_i^{\text{aug}}\}_{i\in\hat{\mathcal{L}}^\sigma},\ cost_t\right) \longleftarrow \mathit{SolveModelD}\left(\sigma,\mathcal{L}^\sigma,\mathcal{K},\mathcal{H},\{\mathcal{J}_k^{h\sigma}\},\{c_i\},\{vt_k^h\},\{y_i^{\text{avail}}\}_{i\in\mathcal{L}}\right)$$

<u>Step</u> *5.3.3: If (* $\#(\hat{\mathcal{L}}^\sigma)>0$ *) then /* there are augmentation */*

<u>Step</u> *5.3.3.1: /* Do circuit layout */*

$$\left(\{s_{ij}^{\text{aug}}\}_{j\in\hat{\mathcal{L}}_i^\sigma,i\in\hat{\mathcal{L}}^\sigma},cost_f\right) \longleftarrow \mathit{SolveModelE}\left(\sigma,\hat{\mathcal{L}}^\sigma,\mathcal{F},\{f_{ij}\},\{y_i^{\text{aug}}\}_{i\in\hat{\mathcal{L}}^\sigma},\{u_\ell^{\text{avail}}\},\theta\right)$$

<u>Step</u> *5.3.3.2: /* Update */*

$$y_i^{\text{now}} \longleftarrow y_i^{\text{now}} + y_i^{\text{aug}},\quad i\in\hat{\mathcal{L}}^\sigma;\qquad s_{ij}^{\text{now}} \longleftarrow s_{ij}^{\text{now}} + s_{ij}^{\text{aug}},\quad j\in\hat{\mathcal{L}}_i^\sigma, i\in\hat{\mathcal{L}}^\sigma$$

$$u_\ell \longleftarrow u_\ell + \sum_{i\in\hat{\mathcal{L}}^\sigma}\sum_{j\in\hat{\mathcal{L}}_i^\sigma}\Delta_{ij}^\ell s_{ij}^{\text{aug}},\quad \ell\in\mathcal{F};\qquad cost_{\text{now}} \longleftarrow cost_{\text{now}} + cost_t + cost_f$$

*EndIf /** <u>Step</u> *5.3.3 */*

<u>Step</u> *6: OutputDesignResults:* $\{y_i^{\text{now}}\},\{s_{ij}^{\text{now}}\},cost_{\text{now}}$

Fig. 2. Algorithm HA-A

Parts of **Algorithm : HA-B**

<u>Step</u> *0.1: Also input initial trunking* $\{y_i^{\text{now}}\}$*, and layout* $\{\mathcal{L}_i\}_{i\in\mathcal{L}},\{s_{ij}^{\text{now}}\}$.

<u>Step</u> *2:* $\{\mathcal{L}_i\}_{i\in\mathcal{L}} \longleftarrow \mathit{GenerateAdditionalFacilityPathsIfNeeded}\left(N_f,\mathcal{L},\mathcal{F},\{\mathcal{L}_i\}_{i\in\mathcal{L}}\right)$

<u>Step</u> *3.2: /* Do traffic routing and dimensioning */*

$$\left(\hat{\mathcal{L}},\{y_i^{\text{aug}}\}_{i\in\hat{\mathcal{L}}},\ cost_t\right) \longleftarrow \mathit{SolveModelD}\left(0,\mathcal{L},\mathcal{K},\mathcal{H},\{\mathcal{J}_k^h\},\{c_i\},\{vt_k^h\},\{y_i^{\text{now}}\}_{i\in\mathcal{L}}\right)$$

Part of <u>Step</u> *3.4: /* Update* u_ℓ *and* $cost_{\text{now}}$ *as in HA-A */*

$$y_i^{\text{now}} \longleftarrow y_i^{\text{now}} + y_i^{\text{aug}},\quad i\in\hat{\mathcal{L}}^\sigma;\qquad s_{ij}^{\text{now}} \longleftarrow s_{ij}^{\text{now}} + s_{ij}^{\text{aug}},\quad j\in\hat{\mathcal{L}}_i^\sigma, i\in\hat{\mathcal{L}}^\sigma$$

Fig. 3. Parts of Algorithm HA-B

The data for these networks are extracted from an actual public switched voice network spanning the continental US. Examples EN-1, EN-2 and EN-3 (discussed earlier in the context of the size of problems) are extracted from this network by considering various subsets of switching nodes, facility nodes and facility links (see Table I). The facility network for EN-2 which is also used in our simulation study (discussed later) is shown in Fig. 4. (Some more discussion on EN-2 data can be found in [14]). We assume that the unit trunk cost (c_i) to have two components: termination cost at switch of \$300 per port and an airline distance cost of \$0.05 per mile between two switching nodes. For each of these example networks, three different load periods of traffic data are considered to reflect variation of traffic during the day; they are for morning, early afternoon and late afternoon. For the facility network, we assume the unit transmission path cost (f_{ij}) to be distance-based with the cost of \$0.25 per trunk-mile. We assume that there is no initial trunking/circuit layout in the network.

We have considered two scenarios for present mode of operation as observed in the case of actual networks: 1) the traffic network is designed for a given GOS under normal operating condition (nGOS) and the facility network is designed based on minimum cost routing subject to link capacity constraints, and 2) the traffic network is designed as in scenario one but the facility network is designed based on minimum cost routing subject to link capacity constraints *and* additional constraints that demand between two switching nodes are split on two or more physically diverse transmission paths. For brevity, we refer to these two scenarios as PMO-1 and PMO-2, respectively. (We like to note that although PMO-1 has been the norm for a long time, PMO-2 is becoming more prevalent in the recent years.) We obtain results for PMO-1 by setting θ to 1.0 and running UTAFNET *without* steps 4 and 5 of heuristic HA-A. Similarly, we have obtained results for PMO-2 by setting θ to 0.5 and using UTAFNET *without* steps 4 and 5 of heuristic HA-A. Note that, here, $\theta = 0.5$ implies that not more than 50% of the trunk demands between two switching nodes can be circuit-routed on a facility path connecting the nodes. Additionally, we ensure that, of the generated facility paths, any link diverse paths are not allowed to have flow more than $100\,\theta$ % of trunk requirements between two switching nodes; this can be accomplished by appropriately setting θ_{ij}. In Table V, we report the trunk required and the total cost for PMO-1 and PMO-2 for an nGOS of 1%. Note that trunk required under PMO-1 and PMO-2 are the same as the traffic network design rule is the same; the only difference is in the facility network layout as PMO-1 does not have any diverse layout requirement whereas PMO-2 has. This table gives us some perspective on the cost of diversity.

Now we discuss survivable network design. We have primarily used two heuristic rules ($ord = y$ and $ord = n$) for enumeration of the failure states due to failure of an f-link (step 4 of HA-A). These two rules give us some indication on the impact on design results if the failures states are considered in different orders. In Table VI, we present trunk requirements and total costs using these two rules for the three example networks. These results are obtained starting with θ to be 0.5, and using nGOS to be 1% and fGOS to be 50%. For ease of reference, we call this design SDSN-α. The cost for SDSN-α is on average about 16% more than PMO-2. We observe that for the scenarios considered here the order of f-link does not appear to have significant impact on the design cost of the network, the difference between them is less than 2%. While for EN-1 and EN-2, using $ord = y$ results in less cost than using $ord = n$, it is the other way for EN-3. Note that $ord = y$ and $ord = n$ are heuristic rules only; for some problems, it may be possible to arrange the failure states in a way that may result in noticeably higher cost than these two heuristic rules. For example, when we used a third rule (where the failure states are ordered in ascending order of link flow), we found that the cost for EN-2 is 6.89% higher than the cost with rule $ord = y$. However, the cost for EN-1 and EN-3 with the third rule was less than 0.2% higher than the cost with rule $ord = y$.

TABLE V
TRUNKING AND COST FOR THREE NETWORKS UNDER PMO-1 AND PMO-2

	Trunks (PMO–1/ PMO–2)	Cost (\$) PMO–1	Cost (\$) PMO–2
EN-1	1,932	1,445,211	1,618,996
EN-2	3,863	3,920,768	4,423,443
EN-3	6,008	5,873,876	6,660,032

TABLE VI
TRUNKING AND COST WITH SURVIVABLE DESIGN SDSN–α WITH TWO DIFFERENT ORDER FOR FAILURE STATES

	SDSN–α $ord = y$		SDSN–α $ord = n$	
	Trunks	Cost	Trunks	Cost
EN-1	2,281	1,904,399	2,278	1,911,259
EN-2	4,241	4,910,500	4,295	4,996,516
EN-3	6,934	8,003,182	6,835	7,890,613

In Table VII, we have reported another set of design results for the three networks. Note that in HA-A as described, we have set the requirement that at each failure state, unaffected traffic pairs follow nGOS (see (1)). This has been followed in obtaining results SDSN-α presented in Table VI. For the sake of fairness, it may be desirable that when a failure occurs in the network, the *non* affected traffic pairs have a GOS higher than nGOS (but lower than failure GOS) since affected pairs try to maintain fGOS. To reflect this case, we have computed design results by initially setting nGOS at 5%; for each failure state, we set fGOS at 50% for affected pairs. To ensure that GOS under normal circumstances is maintained at 1% GOS, we have invoked the operation done in step 3 of HA-A one more time after step 5 is over using 1% GOS. The results are reported in Table VII and we refer to this design as SDSN-β. Note that the cost of design SDSN-β is on average 2.5% lower than SDSN-α; trunk requirement is 2.8% lower with SDSN-β than with SDSN-α. Similarly, using this approach, other design results can be obtained by setting different acceptable values for nGOS.

TABLE VII
TRUNKING AND COST WITH SURVIVABLE DESIGN SDSN–β WITH TWO DIFFERENT ORDER FOR FAILURE STATES

	SDSN–β			
	$ord = y$		$ord = n$	
	Trunks	Cost	Trunks	Cost
EN-1	2,207	1,845,023	2,206	1,853,795
EN-2	4,120	4,794,095	4,192	4,902,267
EN-3	6,748	7,814,886	6,643	7,692,969

In Table VIII, we report CPU time taken for running PMO-1, PMO-2 and SDSN-α (with $ord = y$ and $ord = n$) on a NeXTstation (reported performance: 15 Dhrystone MIPS, 2 MFLOPS DP LINPACK [18]). This time includes time to do I/O for input of various files and design output. For the example networks, the survivable design takes about five times more computational time than present modes of operation. This is certainly dependent on number of failure states in the network. In any case, the total computational time can be reduced if, instead of considering all failure states, only a subset of failure states (e.g. f-links that are likely to fail due to where they are located) are considered. Note that at each failure state, Model-D is solved — Model-E is then solved if there is trunk augmentation dictated by Model-D. For the study networks, we have observed that up to 70% of the failure states require augmentation, and when there is augmentation required, only about 11% of the traffic pairs require circuit layout (requirement to solve Model-E). Note that the bulk of the time is spent on solving Model-D in steps 3.2 and 5.3.2 of HA-A (shown as percentage of total CPU time in the table).

B. Failure Analysis

To observe the impact of a failure on a network with present modes of operation and with survivable design, we have used a call-by-call dynamic routing traffic simulator. We first briefly describe this simulator.

1) Dynamic Call Routing Traffic Simulator: This simulator is written in CSIM, a process-oriented simulation language based on the C programming language [22], [23]. The dynamic call routing scheme used in the simulation uses at-most two t-links to complete a call. In this scheme, the routing table is updated for each switching pair at a regular interval based on the free trunk capacity available in the network. To describe how the routing table is computed, we introduce the following notation:

N := Total number of switching nodes in the network
t_{ij} := Total number of trunks on the traffic link (trunkgroup) between node i and node j
o_{ij} := Total number of busy trunks on the traffic link (trunkgroup) between node i and node j at the time of computing the routing table
r_{ij} := Number of reserved trunks [1] for the direct traffic between node i and node j.

The number of free trunks available, d_{ij}^k, via an alternate switching node k for a call for the pair i–j is then calculated as:

$$d_{ij}^k := \min \{t_{ik} - o_{ik} - r_{ik}, t_{kj} - o_{kj} - r_{kj}\}.$$

For each switching node pair i–j, we sort $\{ d_{ij}^k \mid k \in N, k \neq i, k \neq j \}$ in descending order. The first m ($\leq N-2$)alternate nodes (i.e. the alternate nodes with most free capacity) are then included in the routing table. An arriving call between switching nodes i and j first tries on the direct traffic link i–j. If there is a free trunk, the call is connected on that trunk. If there are no free trunks on the direct t-link, then the call first tries through the first alternate via node (say, k') as given in the routing table; if it cannot find any free trunks on this alternate route then the call is crankbacked (see [1]) and tried via the next alternate via node as given in the routing table. If the call cannot find any free trunks after trying all the m alternate nodes, then the call is blocked. (Some more discussion about this routing can be found in [14]).

The main inputs to the simulator are: the length of the simulation run, when to fail the network, how long the failure will last, how often to update the routing table, how often to collect traffic statistics, the traffic data and the number of trunks required, and the status of the trunks in case of a failure. The simulator does the following: based on traffic offered load and mean call holding time, it generates a call. We assume call arrival to be Poissonian and the mean call holding time to be exponential. This call is first tried on the direct trunkgroup and then alternately up to two traffic link routes based on the call routing scheme described earlier — if it does not succeed the call is blocked; if it does succeed, the call is held for an exponential amount of time, in which case appropriate trunks are occupied. At the end of the holding time, the call is released and the associated trunk (two trunks if a two t-link call) is freed up. In case of a failure, it determines which trunks are affected, and which active calls are affected. All active calls on affected trunks are "aborted" as soon as the failure occurs. For our study, we have set the mean call holding time to be 180 seconds, the traffic statistics collection interval to be every 300 seconds and routing update interval to be ten seconds, and trunk reservation to be uniformly 5 % of the trunks for direct traffic for each trunk group. For a particular load period, we start collecting statistics after three

[1]Trunk Reservation is a control mechanism which is assigned through the parameter r_{ij} for the trunk group for the switching node pair i–j. It means direct routed calls may always be carried on the direct trunkgroup, while alternate routed calls may be carried only if at least r_{ij} free trunks are available. Its importance has already been addressed by other researchers (see, for example, [4], [11], [16]).

hours of simulation (to ignore any initial transient behavior); we continue simulation run for another hour (collecting statistics) at which point the failure of an f-link occurs (if we are doing a failure study); the run is further continued for another hour (collecting statistics). For results described below, for each case we did ten replications using different seeds.

2) Simulation Results: We conducted our simulation study using example network EN-2. This network has ten switching nodes, eighteen facility nodes and twenty seven f-links (Fig. 4). The traffic network is fully-interconnected under normal circumstances. Total offered loads in the three load periods are 2684.80 erlangs, 2826.16 erlangs and 3224.08 erlangs; for ease of reference, we call them as load-1, load-2, and load-3, respectively. For the study of this network, we have set the maximum number of alternate routes to 4 ($m = 4$) in the simulator. First, we did network simulation using trunks for PMO-1/2 when there is no failure in the network. In this case, 95 % confidence intervals for network blocking for load-1, load-2 and load-3 are $0.046 \pm 0.014\%$, $0.641 \pm 0.098\%$, and $0.616 \pm 0.078\%$, respectively. (All the results report below are also at a 95% confidence interval).

For failure study, we choose two f-links in the network: 4–15 and 1–3. In Table IX, we show the number of trunk groups directly affected and the total number of trunks that fail due to each of these two failures. These numbers are shown for PMO-1, PMO-2, SDSN-α ($ord = y$), SDSN-β ($ord = y$). Recall that PMO-1 and PMO-2 have the same number of trunks and that they have different circuit layouts in the facility network. Recall that SDSN-β is designed to provide 50% GOS for affected traffic pairs and 5% for non affected traffic pairs in case of a failure whereas SDSN-α is designed to provide 50% GOS for affected traffic pairs and 1% for non affected traffic pairs in case of a failure. It can be easily seen that a significant number of trunks are affected due to each of these f-link failures. Note that more trunk groups are affected under PMO-2, SDSN-α and SDSN-β than PMO-1; this does not necessarily mean that more trunks are affected (see Table IX, f-link 1–3). Due to different traffic loads at different time of the day, all affected trunks may not have active calls when there is a failure in the network. In Table X, we report from simulation of each load and failure case, the number of calls aborted at a 95% confidence interval. Note that more trunks lost do not necessarily mean that more active calls are aborted (failure 4–15). We further note that although total offered load in load-1 is less than in load-2, the total number of calls aborted is more in load-1 than in load-2 in three cases.

In Table XI, we report overall network blocking *after* failure. Introduction of diversity (PMO-2) in the facility network can significantly reduce overall blocking compared to the case where the facility network is designed based on straight minimum cost routing (PMO-1); in the case of f-link failure 1–3, the network blocking difference is as much as 20%. This also gives us some idea on cost/benefit trade off between PMO-1 and PMO-2; note that PMO-2 costs about 13% more than PMO-1. (More detailed results on impact of an f-link failure on traffic networks designed *only* to nGOS, but with various circuit layout policies, can be found in [14]).

Both survivable designs SDSN-α and SDSN-β further reduce overall network blocking (after failure) compared to PMO-2. While the cost of SDSN-β is 8% more than PMO-2, the network blocking (for the cases we considered) with SDSN-β is about 3 to 5% lower than PMO-2. Difference in blocking between SDSN-α and SDSN-β is most often less than 1% while the cost of SDSN-β is about 3% lower than SDSN-α.

Finally, we consider the impact on pairwise blocking due to a failure. In Table XII, we report number of pairs (along with its percentage compared to total number of traffic pairs) that have blocking over 50% after a failure (we also show the number of pairs for which it is inconclusive at 95% confidence interval whether the blocking is more than 50%). Additionally, we report the maximum (pairwise) blocking faced by a traffic pair. This table shows us the difference between network designed without and with a survivability objective. With PMO-1, we observe as high as 40% of the traffic pairs have blocking over 50% (f-link failure 1–3); maximum pairwise blocking is as high as 99.22 $\pm$ 0.13%. It may be noted that under PMO-1, when a trunk group is affected due to an f-link failure, usually all its trunks are lost. For affected traffic pairs, percentage of calls that can be alternate routed using dynamic routing vary depending on the location of the failure and the offered load in the network; in some instances, a significant amount of traffic can be alternate routed. For example, we observed that if f-link failure 4–15 occurs in load period-1, the lowest blocking among the directly affected traffic pairs is $25.0 \pm 2.2\%$. However, if the same failure occurs in load period 2 and load period 3, the lowest blocking among the directly affected pairs is $70.0 \pm 1.7\%$ and $88.0 \pm 1.4\%$, respectively. However, if we consider the f-link failure 1–3, we observe that the lowest blocking among directly affected traffic pairs for load-1, load-2 and load-3 are $79.2 \pm 1.7\%$, $71.1 \pm 1.7\%$, $87.9 \pm 1.5\%$, respectively.

We report number of traffic pair that have blocking over 50% for PMO-2 in Table XII also. Although trunk group diversity has improved the impact (compared to PMO-1), we observe that as high as 13.3% of the traffic pairs can have blocking over 50% and that maximum pairwise blocking can be as high as $64.7 \pm 1.4\%$. With survivable design SDSN-α (same table), we observe that one traffic pair has blocking over 50%. With SDSN-β, there are no pairs with more than 50 % blocking; however, for a few pairs it is inconclusive (if blocking is over 50%) at 95% confidence interval. On closer look, we have found that the pairs (with more than 50% blocking or inconclusive under SDSN-α and SDSN-β) have offered load less than 30 erlangs each whereas the pairs (with more than 50% blocking or inconclusive under PMO-2) have offered load more than 60 erlangs each and as high as 100 erlangs in some cases. From these results, we can infer that both SDSN-α

TABLE VIII
TOTAL CPU TIME (IN MIN.) ON A NeXTSTATION FOR PMO–1, PMO–2, SDSN–α
([1] – PERCENTAGE OF TOTAL TIME SPENT SOLVING MODEL-D IN STEPS 3.2 AND 5.3.2 OF HA-A)

	PMO–1 time	PMO–2 time	SDSN–α ($ord = y$) time	SDSN–α ($ord = y$) ModelD[1] (%)	SDSN–α ($ord = n$) time	SDSN–α ($ord = n$) ModelD[1] (%)
EN-1	0.04	0.04	0.21	73.9%	0.24	74.2%
EN-2	0.84	0.84	3.69	93.7%	3.77	93.9%
EN-3	8.72	8.85	41.95	98.4%	46.00	98.6%

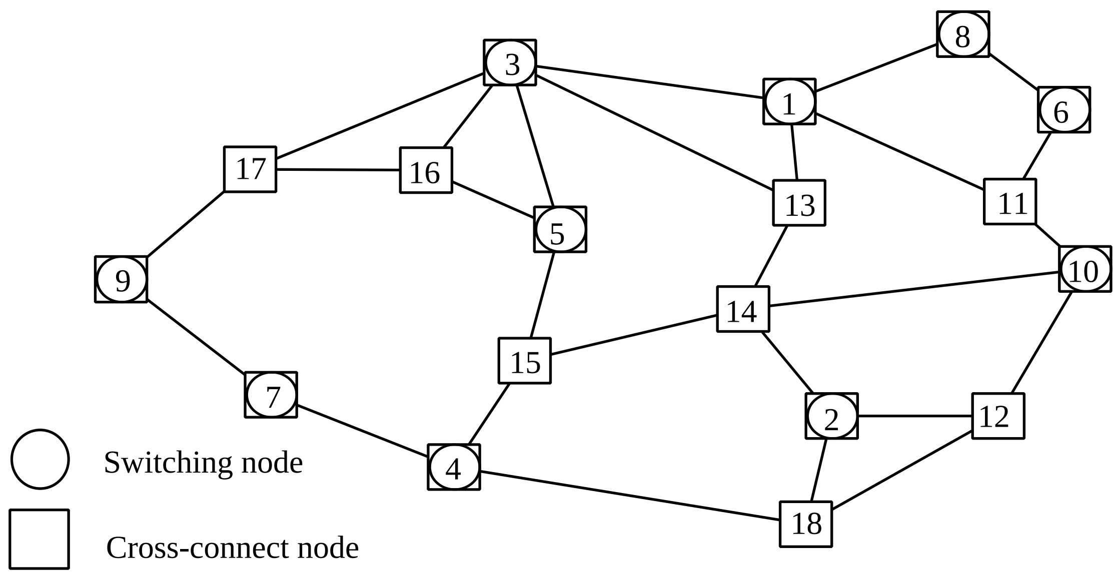

Fig. 4. 10-switching node, 18-cross connect node network (EN-2)

TABLE IX
NUMBER OF TRUNK GROUPS (TG) AFFECTED AND TRUNKS (TK) LOST DUE TO A FAILURE

f-link	PMO–1 TG	PMO–1 tk	PMO–2 TG	PMO–2 tk	SDSN–α TG	SDSN–α tk	SDSN–β TG	SDSN–β tk
4–15	12	977	23	1006	23	1101	23	1073
1–3	18	1313	20	716	21	813	21	809

TABLE X
NUMBER OF ABORTED CALLS

f-link	Load Period	PMO–1	PMO–2	SDSN–α	SDSN–β
4–15	1	703 ± 15	578 ± 27	569 ± 26	563 ± 27
	2	752 ± 9	813 ± 12	795 ± 16	798 ± 17
	3	864 ± 11	828 ± 20	839 ± 30	832 ± 21
1–3	1	1025 ± 23	572 ± 21	580 ± 27	566 ± 19
	2	951 ± 16	499 ± 13	565 ± 21	584 ± 10
	3	1095 ± 19	580 ± 14	640 ± 20	655 ± 18

TABLE XI
NETWORK BLOCKING (IN %) AFTER A FAILURE

f-link	Load Period	PMO–1	PMO–2	SDSN–α	SDSN–β
4–15	1	19.72 ± 0.16	10.36 ± 0.17	5.46 ± 0.10	6.27 ± 0.12
	2	28.57 ± 0.18	25.33 ± 0.20	19.93 ± 0.26	20.43 ± 0.23
	3	28.20 ± 0.19	24.57 ± 0.21	19.43 ± 0.25	19.97 ± 0.20
1–3	1	37.29 ± 0.11	14.91 ± 0.15	10.81 ± 0.16	11.91 ± 0.17
	2	35.30 ± 0.14	14.48 ± 0.25	11.05 ± 0.25	11.76 ± 0.28
	3	36.44 ± 0.15	18.05 ± 0.20	15.00 ± 0.21	15.71 ± 0.20

TABLE XII
TRAFFIC PAIRS WITH BLOCKING OVER 50% AND MAXIMUM PAIRWISE BLOCKING AFTER A FAILURE
([1] – PAIRS FOR WHICH IT COULD NOT BE CONCLUDED IF BLOCKING IS OVER 50% AT 95% CONFIDENCE INTERVAL)
([2] – IN PERCENTAGE OF TOTAL TRAFFIC PAIRS)

f-link	Load Period	Design	Traffic pairs with blocking over 50% No.	over 50% in p.c.[2]	inconclusive[1] No.	inconclusive[1] in p.c.[2]	Maximum Pairwise Blocking (in %)
4–15	1	PMO-1	11	24.4	0	0.0	93.3 ± 0.5
		PMO-2	1	2.2	1	2.2	55.5 ± 1.0
		SDSN-α	0	0.0	0	0.0	44.7 ± 2.3
		SDSN-β	0	0.0	0	0.0	45.9 ± 1.7
	2	PMO-1	12	26.7	0	0.0	98.8 ± 0.2
		PMO-2	6	13.3	2	4.4	59.0 ± 1.4
		SDSN-α	0	0.0	0	0.0	47.0 ± 0.6
		SDSN-β	0	0.0	1	2.2	47.5 ± 2.9
	3	PMO-1	12	26.7	0	0.0	97.0 ± 0.3
		PMO-2	5	11.1	5	11.1	64.7 ± 1.4
		SDSN-α	1	2.2	0	0.0	51.4 ± 1.4
		SDSN-β	0	0.0	3	6.7	51.5 ± 1.9
1–3	1	PMO-1	18	40.0	0	0.0	99.2 ± 0.1
		PMO-2	0	0.0	0	0.0	46.9 ± 0.9
		SDSN-α	0	0.0	0	0.0	45.0 ± 0.9
		SDSN-β	0	0.0	0	0.0	46.0 ± 1.0
	2	PMO-1	18	40.0	0	0.0	98.7 ± 0.2
		PMO-2	3	6.7	0	0.0	55.7 ± 1.7
		SDSN-α	0	0.0	0	0.0	41.2 ± 1.1
		SDSN-β	0	0.0	0	0.0	42.3 ± 1.2
	3	PMO-1	18	40.0	0	0.0	98.8 ± 0.2
		PMO-2	3	6.7	1	2.2	53.5 ± 1.1
		SDSN-α	0	0.0	0	0.0	42.5 ± 1.2
		SDSN-β	0	0.0	0	0.0	44.6 ± 1.7

and SDSN-β essentially meet the design goal of providing 50% GOS for affected traffic pairs in case a failure occurs.

V. Discussion

We have presented here mathematical models and heuristic algorithms for designing a survivable dynamic routing teletraffic network and described a unified definition by explicitly addressing the traffic and the facility network. We have reported cost of survivable design for realistic networks compared to present modes of network design. It should be kept in mind that the results obtained are topology dependent. Nevertheless, the trade off between present modes of design and survivable design can be assessed by considering the cost against the performance of the network in the event of a failure.

We observed that, for the test networks considered here, the computational time for survivable design is about five times more than present modes of design (Table VIII). This is certainly dependent on number of failure states and computational time (trunk augmentation and/or circuit routing) required for solving Model-D/E at each of these failure states. Our observation has been that bulk of the computational time is spent on solving Model-D (see Table VIII). To reduce computational time, a faster method to solve D (then using MINOS) by exploiting the structure of the problem can be explored.

We also observed that by providing trunk diversity alone in the transmission network, one may not necessarily be able to obtain a desired level of performance in the traffic network for various services. A survivable design approach (considering the traffic and the facility network in a unified framework) such as the one presented in this work is desirable if one wants to meet a specific GOS for the traffic network in the event of a failure.

The survivable teletraffic network design problem is a complex problem. As a matter of fact, the facility network is much more complex than the view we have taken here. We do not address multiplex bundling for the facility network which is itself a complex problem [7], or nonlinearity of the cost functions. Nevertheless, our view shows an interrelationship between the traffic and the facility network, which has not been addressed before.

Acknowledgement

We wish to thank the Sprint Corporation for providing us with the network data used in this study. The author benefited from discussion with G. R. Ash, F. Chang and J. Pearce. We used a subroutine version of MINOS as a part of UTAFNET; this subroutine version was provided to us by R. Maier. Constructive comments by the anonymous referees and J. G. Morris were very helpful in improving the content and the presentation of the paper.

References

[1] G. R. Ash, R. H. Cardwell and R. P. Murray, "Design and Optimization of Networks with Dynamic routing," *Bell System Technical Journal*, vol. 60, no. 8, pp. 1787-1820, 1981.

[2] G. R. Ash, F. Chang and D. Medhi, "Robust Traffic Design for Dynamic Routing Networks," in *Proceedings of IEEE Conference on Computer Communications (INFOCOM'91)*, Bal Harbour, Florida, pp. 508-514, April 1991.

[3] Y. K. Agarwal, "An Algorithm for Designing Survivable Networks," *AT&T Technical Journal*, vol. 68, no. 3, pp. 64-76, 1989.

[4] J. M. Akinpelu, "The Overload Performance of Engineered Networks with Nonhierarchical and Hierarchical Routing," *AT&T Bell Labs Tech. Journal*, vol. 63, no. 7, pp. 1261-1281, 1984.

[5] D. Bertsekas and R. Gallager, *Data Networks - Second Edition*, Prentice-Hall, Englewood Cliffs, NJ, 1992.

[6] R. H. Cardwell, C. L. Monma and T.-H. Wu, "Computer-Aided Design Procedure for Survivable Fiber Optic Networks," *IEEE J. on Selected Areas in Comm.*, vol. 7, no. 8, pp. 1188-1197, 1989.

[7] R. D. Doverspike, "Algorithms for Multiplex Bundling in a Telecommunications Network," *Operations Research*, vol. 39, pp. 925-944, 1991.

[8] B. Gavish, P. Trudeau, M. Dror, M. Gendreau and L. Mason, "Fiber Optic Circuit Network Design under Reliability Constraints," *IEEE J on Selected Areas in Comm.*, vol. 8, pp. 1181-1187, 1989.

[9] A. Girard, *Routing and Dimensioning in Circuit-Switched Networks*, Addison-Wesley, Readings, Mass., 1990.

[10] B. R. Hurley, C. J. R. Seidl and W. F. Sewell, "A Survey of Dynamic Routing Methods for Circuit-Switched Traffic," *IEEE Communications Magazine*, vol. 25, no. 9, pp. 13-21, 1987.

[11] R. S. Krupp, "Stabilization of Alternate Routing Networks," in *Proceedings of IEEE International Conference on Communications (ICC'82)*, Philadelphia, PA, pp 31.2.1-31.2.5, June 1982.

[12] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

[13] D. Medhi, "Diverse Routing for Survivability in a Fiber-based Sparse Network," in *Proceedings of IEEE International Conference on Communications (ICC'91)*, Denver, Colorado, pp. 672-676, 1991.

[14] D. Medhi and S. Sankarappan, "Impact of a Transmission Facility Link Failure on Dynamic Call Routing Circuit-Switched Networks under Various Circuit Layout Policies," *Journal of Network and Systems Management*, vol. 1, no. 2, pp. 143-169, 1993.

[15] M. Minoux, "Network Synthesis and Optimum Network Design Problems: Models, Solution Methods and Applications," *Networks*, 19, pp. 313-360, 1989.

[16] D. Mitra and J. B. Seery, "Comparative Evaluations of Randomized and Dynamic Routing Strategies for Circuit-Switched Networks," *IEEE Trans. Comm.*, vol. 39, pp. 102-115, 1991.

[17] B. A. Murtagh and M. A. Saunders, "MINOS 5.0 User's Guide," Techical Report # SOL 83-20, Department of Operations Research, Standford University, Stanford, CA, December 1983.

[18] *NeXT User's Reference*, NeXT Computer, Inc., Redwood City, CA, 1991.

[19] T. Oda and Y. Watanabe, "Optimization of International ISDNs with Dynamic Routing under Reliability Constraint," *Trans. of the IEICE*, vol. E 73, no. 2, pp. 212-219, 1990.

[20] R. F. Rey, ed., *Engineering and Operations in the Bell System*, Bell Telephone Laboratories, 1983.

[21] E. Rosenberg, "A Nonlinear Programming Heuristic for Computing Optimal Link Capacities in a Multi-hour Alternate Routing Communications Network," *Operations Research*, vol. 35, pp. 354-367, 1987.

[22] H. D. Schwetman, "CSIM: A C-based, Process-Oriented Simulation Language," in *Proceedings of the 1986 Winter Simulation Conference*, pp. 387-396, 1986.

[23] H. D. Schwetman, *CSIM Reference Manual (Revision 14)*, Microelectronics and Computer Technology Corporation, Austin, TX, March 13, 1990.

[24] S. Sen, R. D. Doverspike and M. S. Dunatunga, "Unified Facilities Optimizer," Technical Report, Systems and Industrial Engineering Department, University of Arizona, Tucson, Arizona, January, 1989.

[25] Y. K. Tsai, "Inter-Office Transmission Network Planning with the Network Planning System (NPS)," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM'86)*, Houston, Texas, pp. 25.6.1-25.6.6, 1986.

[26] T.-H. Wu, D. Kolar and R. Cardwell, "Survivable Network Architectures for Broad-Band Fiber Optic Networks: Model and performance comparison," *IEEE J. of Lightwave Technology*, vol. 6, pp. 1698-1709, 1988.

[27] T.-H. Wu, *Fiber Network Service Survivability*, Artech House, Boston, Mass., 1992.

[28] B. Yaged, "Minimum Cost Routing for Static Routing Models," *Networks*, vol. 1, pp. 139-172, 1971.

[29] B. Yaged, "Minimum Cost Routing for Dynamic Routing Models," *Networks*, vol. 3, pp. 193-224, 1973.

Deepankar "Deep" Medhi (M'89) has B.Sc. (Hons.) in Mathematics from Gauhati University (Cotton College), Assam, India, M.Sc in Mathematics from the University of Delhi, India, M.S. and Ph.D. in Computer Sciences from the University of Wisconsin–Madison in 1981, 1983, 1985 and 1987, respectively. He is currently an Assistant Professor in the Computer Science Telecommunications Program and a member in its Center for TeleComputing Research at the University of Missouri–Kansas City (UMKC). Prior to joining UMKC, he was a member of the technical staff at the AT&T Bell Laboratories, Holmdel, New Jersey from 1987 to 1989. His current research interests are in survivable network design, network routing and performance, broadband network design, large-scale optimization algorithms and parallel computing, network management. He is a member of ACM, ORSA and SIAM.