



ELSEVIER

Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Computer Communications 26 (2003) 2025–2036

computer
communications

www.elsevier.com/locate/comcom

Prioritized traffic restoration in connection oriented QoS based networks

Bjørn Jæger^{a,*}, David Tipper^{b,1}

^aDepartment of Informatics, Molde University College, P.O. Box 2110, Britvegen 2, 6402, Molde, Norway

^bDepartment of Information Science and Telecommunications, University of Pittsburgh, Pittsburgh, PA 15260, USA

Received 27 June 2002; revised 12 May 2003; accepted 14 May 2003

Abstract

Connection oriented quality-of-service (QoS) based networks offer several service classes each designed to handle connections supporting various types of applications. A network failure will typically result in a large number of disrupted connections in each service class, all of which need to be restored simultaneously. In this paper we propose a distributed fault recovery priority scheme for dynamic restoration of connections. The proposed scheme exploits available network resources immediately after a failure together with properties of typical applications carried by the connections in order to give the applications sufficient network performance. The scheme is formulated within the context of switched virtual circuit routing in ATM networks, but it is applicable to ATM virtual path restoration as well and to other class based QoS networks like DiffServ MPLS based networks. The priority based restoration scheme is formulated as a centralized path flow integer programming optimization problem serving as a comparison benchmark for suboptimal distributed schemes. We propose a distributed heuristic scheme and give numerical results illustrating the quality of the proposed distributed scheme in comparison to the centralized optimal approach. Our numerical results show that the proposed distributed scheme can significantly improve utilization of network resources and enhance network performance after a failure.

© 2003 Elsevier B.V. All rights reserved.

Keywords: Fault recovery; Survivability; Priority; Restoration

1. Introduction

With the widespread use of telecommunications networks and society's growing dependence upon the exchange of information, the need for reliable communication service has become increasingly important. Several highly publicized outages have illustrated that disruption of communication services is very costly to businesses, governments and the general public. Note that typical causes of service outages are equipment failures, software bugs, accidents (e.g. fiber cable cut), incorrect maintenance, natural disasters and sabotage. Since many of the causes of service outages are outside the control of network service providers, there has been growing interest in the analysis and design of networks that are survivable. Providing survivability in a network consists of provisioning sufficient spare capacity and diversity in the network along with implementing traffic

restoration mechanisms to enable fault recovery via rerouting of traffic in the event of a failure. The topic of traffic restoration in connection oriented wide area networks has received attention of late with work on the physical layer [1–5], logical layer [6–9] and virtual circuit layer [10–13] appearing in the literature.

In Refs. [1,2] the restoration problem is studied for SONET-technology and WDM, respectively. This work focus on providing spare capacity in the forms of rings ensuring short restoration time. In Refs. [3,4] the authors study the capacity assignment problem for fault recovery using digital cross connect systems focusing on high efficiency. In Ref. [5] a ring-to-mesh evolution scheme for restoration of connections is given showing how mesh networks can realize ring-like speed with mesh-like efficiency. The associated traffic restoration procedure in Refs. [1–5] consists of determining the routes around a ring or the cross connect route through the network.

At the logical layer several restoration algorithms have been proposed [6–9]. These algorithms differ by how the locus of rerouting and the rerouting scheme are specified. The locus of rerouting determines which of the network

* Corresponding author. Tel.: +47-71214217; fax: +47-71214100.

E-mail addresses: bjorn.jager@himolde.no (B. Jæger), tipper@tele.pitt.edu (D. Tipper).

¹ Tel.: +1-412-624-9421; fax: +1-412-624-2788. Supported in part by NSF grant NSR-9506652.

nodes in the route of a failed path is responsible for rerouting. For example, the node adjacent to a failed link can be responsible for rerouting, or the source node of the path can decide a new route. The responsible node must pick a new route for the failed path. Several rerouting schemes have been proposed ranging from the pre-selection of disjoint backup routes to a dynamic search for routes. Suggestions for how to utilize pre-selected backups include methods for reserving capacity along the backups, and methods for conducting a dynamic search for capacity over a group of pre-selected backups.

Restoration at the virtual circuit layer has been focused on routing after a failure [10–13]. In Ref. [10] the effects of locus of reroute, reroute timing among affected source nodes and retry policies are investigated. In Ref. [11] the authors consider several routing algorithms and how they perform during the transient congestion period immediately after a failure. A pre-planned source routing fault recovery scheme is suggested in Ref. [12]. In Ref. [13] we perform an initial investigation of prioritized dynamic restoration of virtual circuits. This paper extends the work of Jæger and Tipper [13] by introducing a centralized path flow optimization problem formulation together with multiple workload combinations for centralized and distributed implementations.

Our scheme is targeted at virtual circuit restoration, hence we present some arguments for why restoration at the virtual circuit layer is needed. First, virtual circuit layer restoration is a cost efficient way to provide restoration capabilities in cases where restoration at lower layers is considered to be too costly either because extra equipment is needed or because the virtual circuits span networks of several providers with different restoration capabilities. Second, virtual circuit layer restoration can restore connections that cannot be restored by lower layers since lower layers restore bundles of virtual circuits as one unit and there might not be sufficient spare capacity for restoration of the unit as a whole. Third, failures originating at the virtual circuit layer can only be restored at the virtual circuit layer or layers above, and fourth, restoration priorities for individual virtual circuits are possible. Therefore a multi-layer fault recovery procedure is needed wherein a procedure for virtual circuit restoration is possible in case lower layer restoration is not available, or cannot address a particular failure.

One aspect of fault recovery at the logical- and the virtual circuit layer that has been ignored is the ordering of virtual paths and virtual circuits for rerouting. To illustrate this problem we consider an arbitrary wide area ATM network that uses source node routing of the Virtual Circuits (VCs). In source node routing, each network node maintains a database of the network topology, and determines the route through the network for all VCs originating at the node. The PNNI routing scheme developed by the ATM Forum for switched VC routing within a peer group adopts this approach [18]. In the event of a failure we assume that all

the virtual circuits using the failed device are disrupted and need to be reconnected if possible. The source nodes for the VCs that were traversing the failed device are responsible for restoration of the affected virtual circuits, as is discussed in the ATM Forum. A failure will typically result in *several* nodes being sources for affected virtual circuits with each having *many* virtual circuits to *simultaneously restore*, possibly in the order of tens of thousands. The way in which the VCs are processed will determine in part, if a connection is restored, the duration of the restoration process for each connection, and the quality-of-service (QoS) received by applications during restoration and immediately after the restoration.

After a failure the reconnection of the virtual circuits takes place only after a time delay that consists of the time to detect the failure and set alarms, the time to disseminate knowledge of the failure to the fault recovery mechanism, and the time to reroute and setup the connection. During the restoration time cells will be lost from the disrupted VCs and depending on the application they may require retransmission from the source. Note that a dropped cell may trigger a packet retransmission at a higher level in the protocol stack causing the number of retransmitted cells to be larger than the number dropped. The dropped cells in need of retransmission create a backlog at the traffic source that may result in congestion in the network. In Ref. [23] it is shown that the dominant factor on network performance immediately after a failure is the transient congestion caused by retransmission of lost cells. Further, in Ref. [23] it is pointed out that since many VCs need to be restored simultaneously, congestion control schemes are not entirely effective in preventing congestion. The importance of transient congestion after failures has been noted in other contexts as well (e.g. SS7 networks) [26–28]. Therefore traffic restoration techniques should incorporate techniques to reduce the transient congestion.

In this paper we propose a distributed fault recovery priority scheme for dynamic restoration of connections. The proposed scheme attempts to give the application sufficient network performance by exploiting available network resources after a failure, while considering the nature of applications in terms of whether they retransmit lost data or not, how long an acceptable disruption period is, and the amount of bandwidth required. The scheme involves both assignment of priorities among ATM service categories, and a strategy for ordering and rerouting VCs within each service category. This is based in part on our work on fault recovery routing [11,21], where it is shown that rerouting has a significant impact on network congestion after a failure. Further, we formulate a path flow Integer Programming (IP) optimization problem for multi-service category based networks such as ATM. Solution of the optimization problem provides a priority connection acceptance or rejection decision and the selection of restoration paths for restored connections. In this problem formulation we show how to impose either absolute- or relative-priorities among

the service categories. Absolute priorities are obtained by decomposing the IP-problem by service category, where as, relative priorities are obtained by selecting appropriate weight factors and link cost formulas for the routing algorithms. This optimal approach requires a centralized node with a global knowledge of the network to perform the optimization. Centralized control is not recommended in the ATM standards due to scalability and reachability problems [14]. Also, the problem is NP hard, which means that exact solutions can only be found for small networks. However, finding the optimal results for appropriately selected networks is useful as a benchmark to compare with the results found by our heuristic distributed scheme. Numerical results are presented for a default distributed scheme using no priorities, our proposed distributed priority scheme, and the centralized optimization based priority scheme. The numerical results clearly show how priority based restoration can improve network performance after failures. It should be noted, that while both the distributed and optimization based schemes are formulated within the context of switched VCs, they are applicable to virtual path (VP) restoration as well and to other class based QoS networks like DiffServ or IntServ MPLS based networks.

The rest of the paper is organized as follows. In Section 2. We present a discussion of ATM service categories and retransmission of lost cells. Section 3 discusses the timing of the fault recovery process. In Section 4 the ordering of VCs for processing at a source node is discussed. Section 5 presents the formulation of the priority restoration problem as an integer programming optimization problem, and in Section 6 we describe the proposed distributed priority VC processing scheme. Section 7 presents numerical results comparing the performance of the schemes. Lastly in Section 8 we summarize our conclusions.

2. Service categories and cell retransmission

One of the primary benefits of ATM networks is that they can provide users with a guaranteed network performance. To do this, the user must inform the network upon connection setup of the requirements of the connection by specifying a set of parameters describing the traffic *presented to the network*, and a set of parameters describing the QoS *required from the network*. The traffic descriptors specified by the connection are the peak-cell-rate (PCR), sustained-cell-rate (SCR) and maximum burst size (MBS). The QoS parameters requested from the network by the users include cell loss rate (CLR), cell transfer delay (CTD), and cell delay variation (CDV) as appropriate. ATM networks offer a specific set of service categories each with a mix of QoS parameters. Upon connection setup the user specifies the traffic requirements for the connection and selects a category which supports the QoS requirements needed. A setup message is sent from source to destination containing traffic and QoS-parameters causing resources to

be allocated along the path. An ATM connection that is set up with specified traffic descriptors constitutes a traffic contract between the user and the network. The network offers the type of QoS guarantee appropriate to the service category, as long as the user keeps the traffic on the connection within the range defined by the traffic parameters. The service categories supported by ATM networks are defined in Ref. [16], and specified in detail by the ATM Forum [17] as Constant-Bit-Rate (CBR), real-time Variable Bit Rate (rt-VBR), non-real-time Variable Bit Rate (nrt-VBR), Available Bit Rate (ABR), and Unspecified Bit Rate (UBR).

In case of a network failure all cells are lost for the duration of the restoration period causing service disruption ranging from ms to several seconds and beyond depending of the type of failure and the restoration strategy chosen. For some applications the lost cells must be retransmitted causing an additional load on the network. Studies have shown [23] that this additional load is the reason for buffer overflow at both the source node switch and at the buffers of other nodes along the path to the destination. Hence, it is important to consider how the various applications supported by each service category handle retransmission of dropped cells.

CBR and rt-VBR applications. The CBR and rt-VBR categories both have identical timing requirements (CTD and CDV), and are suited for real-time applications. Under normal operation it is assumed that the CLR is sufficiently small to be ignored by the applications. Applications in these categories are not likely to retransmit lost cells since they will be of little or no value for the applications. The rt-VBR category differ from CBR by requiring applications to specify SCR and MBS to allow statistical multiplexing (see Table 1).

nrt-VBR applications. The nrt-VBR category has CTD guarantees but no guarantees regarding CDV making it suitable for response-time-critical data applications. For these applications all cells are vital, and lost cells are retransmitted. Under normal operation this category guarantees a low CLR, which means that retransmission of these cells does not place any significant additional burden on the network. In case of a failure all cells lost during the restoration period must be retransmitted putting an extra load on the network. Since this additional traffic load is caused by a network failure (i.e. the user is not at fault), it behoves network provider to carry this traffic in a timely fashion minimizing the impact on applications.

ABR applications. Under normal operation, the CLR is low since the sources adjust cell flow in response to network congestion as indicated by control information. Lost cells are normally retransmitted causing no significant extra burden on the network. Upon a network failure, the sources are notified of the network unavailability by the missing control-cells. The application is then expected to temporarily stop transmitting until the restoration procedure is completed.

Table 1
Traffic parameters, QoS parameters and retransmission policy for ATM

Service category	Traffic parameters			QoS parameters			Retransmission likely
	PCR	SCR	MBS	CTD	CDV	CLR	
CBR	Yes	n/a	n/a	Yes	Yes	Yes	No
rt-VBR	Yes	Yes	Yes	Yes	Yes	Yes	No
Nrt-VBR	Yes	Yes	Yes	Yes	No	Yes	Yes
ABR	Yes	No	No	No	No	No	Yes
UBR	Yes	No	No	No	No	No	Yes

UBR applications. The UBR category has no QoS requirements. Both under normal and failure operation lost cells are expected to be valuable for the application and retransmitted. However, since the application itself typically is running in a private environment, which normally is not a multi-service environment, all retransmissions are handled by the applications within that environment. In case applications use the UBR category in environments supporting multi-services, the network will not give any QoS guarantees for these applications, and it will try to serve their traffic in a best effort manner. This will cause a large number of cells to be dropped in the event of a failure since there is no traffic contract and no feedback mechanisms.

Table 1 summarizes the ATM service categories, traffic descriptors appropriate to each category, QoS parameters, and whether retransmission of dropped cells is likely to occur.

3. Timing considerations

A network failure causes a service outage until the failed component is fixed or until the traffic is restored around the failure. How long outage time that is acceptable varies with the type of application, the protocol used, and the protocol settings. Some applications within telemedicine, real-time control systems and military systems depend upon uninterruptible service and need 100% restoration as quickly as possible. Other applications such as telephony and data transactions need 100% restoration but can tolerate outage periods of several seconds. Another group of applications is the one used largely for entertainment purposes that can tolerate outage times in the order of minutes. A detailed study of the impact of various outage times on different types of services in *existing* networks with segregated transport of each service type has been performed by Sosnosky [22]. The four major restoration time targets found were: (1) 50 ms to < 200 ms; potential voiceband disconnects, trigger changeover of signaling links, (2) 200 ms to < 2 sec; may drop voiceband calls depending on channel bank vintage, (3) 2 sec to < 10 sec; call-dropping (all circuit switched services), potential packet disconnects (X.25), potential data session timeouts, and

(4) 10 sec to < 5 min; packet (X.25) disconnects, data session timeouts. Restoration times above 5 min are termed undesirable due to the social and business impacts it will cause. Thus, the required restoration time varies for each type of service in existing networks. ATM restoration schemes should restore connections within the same thresholds for similar services to provide service continuity when integrated with existing networks. CBR and rt-VBR being suitable for real-time applications like voice and video with CTD and CDV requirement should be restored in a short time (within 2 s). The same applies to nrt-VBR connections since they are suitable for response-time critical applications with CTD requirements. ABR and UBR connections being mainly intended for data applications with limited QoS requirements would accept longer outage times.

In studying the timing of fault recovery it is useful to consider a single VC traversing multiple switches in an ATM network. The access switch next to the calling party is denoted as the source switch, and the last switch of a connection is denoted as the destination switch since it is next to the called party. Consider a link between two intermediate switches along the VC path. We denote the neighboring switches of the link as the upstream and downstream switches, where the upstream switch is in the direction of source switch, and the downstream switch is the one closest to the destination switch.

The general reconnection mechanism specified by the ATM Forum works as follows: when the downstream switch of a failed link detects the failure, it sends a failure message downstream to the destination switch. The destination switch sends a failure message back to the corresponding source switch of the connection via a signaling connection. The source switch then release resources allocated to failed connections, finds alternative routes, and sets up new connections along the routes. An estimate of the time to complete the restoration can be found by considering each step of the restoration process for a single VC, namely: *Detection* of a failure, *Notification* of the failure to nodes responsible for restoration, *Identification* of failed connections, *Selection* of new paths for failed connections, and *Establishment* of connections along the new paths (DeNISE) [4,15]. We briefly discuss each in turn below.

Detection Time. The detection time varies considerably depending on the type of failure that occurs (e.g. fiber cut, line card, software, etc.). For the specific case of a physical layer failure, the failure is detected by the downstream switch and reported to the local ATM Layer Management. The reporting of a physical layer failure to the ATM layer occurs within an *unspecified vendor specific* time period [19] with a typical value reported from measurement studies of 1 s. There is a variation in the detection time at the physical layer as well. For example, the ATM Forum calls for detection of loss-of-signal within 100 μ s and for loss-of-frame within 3 ms.

Notification Time. Upon detecting the link failure the local ATM Layer Management notifies the other switches along the path downstream towards the destination switch by using the dedicated Fault Management operation-and-maintenance (OAM) cells called alarm-indication-signal (AIS) cells. According to ATM Forum specifications, within 250 μ s after the destination switch has received the AIS cells, the switch generates a remote-defect-indication (RDI) OAM cell that is sent upstream to the source switch. This notification is used to trigger the dynamic rerouting process in each source switch and other failure dependent operations. Typically the RDI signal must persist for a certain vendor specific time before the source switch considers a failure to have occurred. Note that each disrupted VC within a source node receives the RDI-cells at different points of time since the distance from the source to the destination generally varies for each VC, and the signaling might use connections of varying length and speed.

Identification Time. The time to identify disrupted connections within a source node is insignificant since AIS and RDI cells are sent for each individual disrupted VC.

Path Selection Time. When a source switch is notified of a disrupted VC it releases all resources allocated to the disrupted VC and tries to select a new route for the VC. In ATM PNNI routing each switch maintains a topology database containing the cost of using each link in the network. Also, to speed up the routing computation each node typically maintains a pre-computed set of alternative paths between itself and the other network nodes. The set of alternative paths is determined from the topology of the network and is usually restricted by a hop count limit. The link cost database is updated periodically with each node notifying the other nodes of its current cost using a flooding approach. In addition, asynchronous updating of the link cost occurs whenever link utilization changes by an amount exceeding a predefined threshold. Based on the cost of each individual link the path can be calculated as the sum of the link cost on the route. The ATM Forum does not specify any standard routing algorithm, thus it is up to the vendor to select which algorithm to use for each class or network situation.

In Refs. [11,21] we studied the behavior of four different routing algorithms for fault recovery within a connection

oriented packet switched network and their effects on network performance in terms of call blocking, demand restored, route selection and the transient network congestion. The routing schemes considered were as follows.

1. *Minimum Delay (MD)* routing use a link cost proportional to the derivative of the link queuing delay based on an M/M/1 type model as discussed in Ref. [24].

$$M_\ell = \frac{B_\ell}{(R_\ell)^2}, \quad \forall \ell \in L \quad (1)$$

B_ℓ is the total capacity of link ℓ of which a residual capacity R_ℓ is available after the failure for restoration purposes. L denote the ordered set of directed links l in the network.

2. *Minimum Hop (MH)* routing uses a load independent constant link cost equal to the hop count [11]. The MH approach results in the number of nodes directly handling rerouted connections being minimized since the restoration paths are as short as possible. Thus MH attempts to isolate and restrict the physical area of the congestion occurring due to rerouting.

$$M_\ell = 1, \quad \forall \ell \in L \quad (2)$$

3. *Load Distribution Among Paths (LDAP)* uses the negative of the residual capacity as the link cost, resulting in selecting the route with the maximum residual bandwidth [11]. The LDAP approach distributes the potential congestion among the various links in the network in order not to degrade the performance seriously at any particular link.

$$M_\ell = -R_\ell \quad \forall \ell \in L \quad (3)$$

4. *Load Distribution among Primary Links (LDPL)* is an alternate approach to load distribution that focuses on balancing the load on all links leading out of a source node, such links are called primary links. The link cost on the outgoing links of a source node is defined as inversely proportional to the residual capacity on the link, downstream link costs are a load independent cost as in the minimum hop scheme. The cost formula proposed for links out of source node \hat{s}_k and the remaining network links [11] is defined as

$$M_\ell = \frac{B_\ell}{R_\ell} \text{ for } \ell : \hat{s}_l = \tilde{s}_k \text{ and } M_\ell = 1, \quad \forall \ell : \hat{s}_l \neq \tilde{s}_k \quad (4)$$

where \tilde{s}_k is the source node of each link ℓ . It was shown in Refs. [11,21] that each algorithm has advantages over the others depending on the network load and which performance parameters (length of congestion period, % demand restored, call blocking, etc.) one want to emphasize. The computational time used by a source node to select new paths is expected to be on the order of μ s per VC, based on the measurements given in Ref. [20].

Establishment of VC Time (setup time). The re-establishment of a VC is accomplished using the same mechanisms as when setting up a new connection. A setup message containing the destination address and the desired traffic and QoS-parameters is sent along the minimum cost path. The Connection Admission Control (CAC) procedure at each switch along the path allocates resources for the call, forwards the message to the next switch if sufficient resources were available, or sends a crank-back message back towards the source switch in order to try to find another route. Upon receiving a crank-back message, each intermediate switch releases previously allocated resources. The call will be rejected if no alternative route with sufficient resources can be found. If the network and the destination accept the call, a connection acceptance message is sent back to the source and the VC begins working. One can see that the VC setup time is a largely a function of the number of hops traversed.

4. Ordering of connections

An interesting issue arises in fault recovery routing that does not occur in normal network operation, namely that each source switch will typically have many VCs within each service category that needs to be rerouted all at once. Since after a failure the total resources of the network are reduced, the network might not be able to restore all disrupted connections. Further, the diversity among service categories and the applications they support motivates a scheduling scheme among categories and connections. Consequently the network must decide which of the connections to disconnect, and for the ones that can be restored the order of restoration must be determined. This can be done on two levels; on the service category level and/or on the individual connection level. Our initial studies in Ref. [13] shows that the processing order of disrupted VCs will affect; the routes selected, the connection blocking, the time used to reconnect a specific VC, and the length of the total restoration period.

Basically there are two ways for a source node to process service categories and the disrupted VCs within each category; either sequentially or in parallel. When doing it sequentially each disrupted VC is connected or rejected before the next one is processed. The local link cost database is updated after restoring each VC. When all VCs within a category are processed the VCs in the next category are handled. Whereas, in the parallel approach each source node selects new paths for all disrupted VCs in all categories, then setup messages are sent out for these VCs as quickly as possible. The link cost database is updated after restoring all VCs in all categories.

If the sequential method is adopted the order in which the VCs are processed can be done in several ways, here we consider two: (1) sorting the VCs by increasing amount of bandwidth (i.e. demand), and (2) sorting the VCs by

decreasing amount of bandwidth. The first approach is expected to give the lowest call blocking rate, the second approach is expected to maximize the demand restored. This is confirmed by our results reported in Ref. [11]. The drawback of the sequential approach is the restoration time that is nearly a linear function of the number of disrupted VCs needing restoration at a source node.

In order to reduce the setup time parts of the restoration can be parallelized. For example the setup messages for all disrupted VCs can be distributed as fast as possible in an almost simultaneous manner. The source switch then handles the corresponding replies from the destination switches in a first-come-first-serve order. A similar approach for restoration at the DCS cross connect level in circuit switched networks is discussed in Ref. [22], where it is noted that there are many practical implementation hurdles in parallel recovery schemes. Note, that under heavy loading conditions many failed setup attempts occur since the source nodes have inaccurate link cost information, extending the restoration time possibly to the extent that no gain in restoration time is achieved compared to the sequential approach.

Here, we use the sequential method since in current ATM standards signaling for parallel VC setup is not standardized. Also, the sequential method will have better call acceptance and route selection behavior since link costs updates may occur during the restoration.

5. The optimal traffic restoration problem

During fault recovery the network needs to restore many virtual circuits simultaneously. Here we formulate an optimization problem to determine whether to restore or block a disrupted VC and the restoration route for restored VCs. The problem can be modeled as a special case of a multi-commodity flow problem in a connection oriented packet switched network. A general discussion on multi-commodity flow problems can be found in Ref. [25]. Consider an arbitrary network represented by the two tuple $G = (N, L)$ where N is the ordered set of nodes and L is the ordered set of directed links. The set of service categories supported by the network is denoted by C . Let K_c denote the ordered set of disrupted VCs of category c . Each VC k of K_c defines a separate indivisible commodity with source node $s_{k_c} \in N$, destination node $d_{k_c} \in N$, and a demand of b_{k_c} units of equivalent bandwidth. For each VC k of category c , let P_{k_c} denote the ordered set of directed paths $j \in P_{k_c}$ from the source node s_{k_c} to the destination node d_{k_c} . Note that the path set considered for restoration between a source and destination node might be smaller than the set of all possible paths between the same two nodes in order to reduce the search space and to enable the network to fulfill end-to-end QoS requirements. Typically the path set is restricted by a hop count limit on the paths. Each link $\ell \in L$ has a source node $s_\ell \in N$, a destination node $\tilde{d}_\ell \in N$, and a residual

capacity R_ℓ . In general a certain portion of the bandwidth γ_ℓ of each link ℓ may be reserved for control traffic and to absorb fluctuations in traffic. Thus the usable free capacity for restoration purposes is equal to $R_\ell - \gamma_\ell$. We define two binary decision variables $y_{k,c,j}$ and $x_{k,c}$ as follows:

$$y_{k,c,j} = \begin{cases} 1 & \text{if VC } k \text{ of category } c \text{ is routed on path } j; \\ 0 & \text{otherwise.} \end{cases}$$

$$x_{k,c} = \begin{cases} 1 & \text{if VC } k \text{ of category } c \text{ is rejected;} \\ 0 & \text{otherwise.} \end{cases}$$

The path-selection variable $y_{k,c,j}$ specifies the path j on which VC k of category c is routed. Since it might be the case that there is not enough available capacity to restore all of the disrupted VCs, the rejection variable $x_{k,c}$ for VC k of category c is defined. Let the cost of routing VC k of category c on path j be $W_{k,c,j}$, let the cost of rejecting VC k of category c be $O_{k,c}$. Then the problem of selecting a route for each disrupted VC can be formulated as:

$$\min_{y_{k,c,j}, x_{k,c}} \sum_{c=1}^C \sum_{k=1}^{K_c} \sum_{j=1}^{P_{k,c}} W_{k,c,j} y_{k,c,j} + \sum_{c=1}^C \sum_{k=1}^{K_c} O_{k,c} x_{k,c} \quad (5)$$

$$\text{s.t.} \sum_{j=1}^{P_{k,c}} y_{k,c,j} + x_{k,c} = 1, \quad \forall k \in K_c, \quad \forall c \in C \quad (6)$$

$$\sum_{c=1}^C \sum_{k=1}^{K_c} \sum_{j=1}^{P_{k,c}} b_{k,c,j} \delta_{k,c,j}^\ell y_{k,c,j} \leq R_\ell - \gamma_\ell, \quad \forall \ell \in L \quad (7)$$

$$y_{k,c,j} = 0 \text{ or } 1, \quad x_{k,c} = 0 \text{ or } 1, \quad \forall c \in C, \quad \forall k \in K_c, \quad \forall j \in P_{k,c}. \quad (8)$$

In the objective function (5) the first term is the cost of choosing the j th path for VC k of category c , while the second term is the cost of rejecting VC k of category c . We define link indicator functions $\delta_{k,c,j}^\ell = 1$ if the j th path for VC k of category c uses link ℓ , $\delta_{k,c,j}^\ell = 0$ otherwise. The formulation above has a simple constraint structure since we consider paths (instead of links). There is a single constraint (6) for each VC k of each category c that states that a VC is either routed on a path $j \in P_{k,c}$ or rejected. Also, there is a constraint (7) for each link $\ell \in L$ which states that the sum of the demands of the VCs passing through the link cannot exceed the available capacity $R_\ell - \gamma_\ell$ of the link. The last set of constraints (8) ensures that only one route is chosen for a VC and that a VC is either rejected or restored.

The formulation of the optimization problem above allows us to consider various restoration scenarios by specifying appropriate path, link, and rejection costs. Here we define the path cost $W_{k,c,j}$ as the sum of link costs along the path. The general formula is given by:

$$W_{k,c,j} = \sum_{\ell=1}^L \delta_{k,c,j}^\ell M_{\ell,c} \quad (9)$$

Table 2

Range of path cost values for the four routing schemes considered

MH	MD	LDAP	LDPL
[1, 4]	[1.2, 400]	[-3.6, -0.1]	[1.1, 13]

The link cost $M_{\ell,c}$ varies according to the routing algorithm adopted for each service category c , here we utilize the link cost forms discussed in Section 3.

Note that the routing algorithm selected will implicitly impose loose priorities among the categories. The implicit nature of the priority scheme is illustrated in Table 2 which shows the range of the path cost of the four routing schemes described in Section 3 assuming normalized link capacities (to 1), maximum link utilization equal to $R_\ell - \gamma_\ell = 0.9$, and a hop count limit of four for an arbitrary network with diameter four.

From the table we see that path costs calculated by LDAP will give the lowest cost since the values are negative. Connections with path costs calculated by MH are preferred over MD and LDPL connections for the same path since MD and LDPL takes the link imposition γ_ℓ into consideration. Between MD and LDPL, LDPL will be preferred over MD for the same path. The optimization problem does not give strict prioritization among categories since the cost ranges may overlap. Clearly if the ranges does not overlap the problem reduces to a separate optimization problem for each category.

In addition to the path cost $W_{k,c,j}$, the cost $O_{k,c}$ for rejecting k th VC of category c can be specified in several ways. If the rejection cost is large for all VCs of all categories then VCs will be rejected only when there is no way to reroute all VCs under the constraints (7). On the other hand, if the rejection cost of a category is too small, then all VCs in that category will be rejected. Note that the problem formulation above is developed from the formulation developed in our paper [11] in which all VCs were assumed to belong to a single category.

6. A distributed priority restoration scheme

Given the preceding discussion we propose the following simple distributed priority scheme to be implemented at each source node: (1) strict ordering among service categories; (2) a specific routing scheme for restoration within each category; and (3) a specific rule for ordering of VCs for restoration processing within each service category. We revisit the ATM service categories to specify and justify the scheme. The scheme is summarized in Table 3. Note that we have not included the UBR category as it is intended for a private LAN environment.

- nrt-VBR: Since the nrt-VBR category is designed for applications requiring short response time, low cell loss

Table 3
Fault recovery priority scheme

Service category	Retransmission likely	Priority	Routing scheme	Ordering according to bandwidth
nrt-VBR	Yes	1	MH	Increasing
CBR	No	2	MD	Decreasing
rt-VBR	No	3	MD	Decreasing
ABR	Maybe	4	LDAP	Increasing

and retransmission of lost cells, this category is given the highest restoration priority among the service categories. Within this category the MH-scheme is used for rerouting since it has the shortest *Path Selection Time*, which will minimize the number of cells dropped. Further, in order to restore the maximum number of disrupted VCs, the VCs are restored in increasing order of bandwidth.

- **CBR:** The CBR category is designed for real-time applications that do not retransmit lost cells. It is assumed that a short service interruption is acceptable while the nrt-VBR connections are being restored giving CBR applications the second highest restoration priority among the categories. With no retransmissions the behavior is similar to steady state behavior. Consequently we adopt the MD-scheme for rerouting in this category since it is optimal under steady state network conditions. Moreover, in order to restore the maximum amount of demand the VCs are restored in a decreasing order of bandwidth. Note that, since CBR is expected to support telephony with its relatively low service outage requirement [22] it is given priority over rt-VBR.
- **rt-VBR:** This category is designed for real-time applications which do not retransmit lost cells, and the same reasoning as for CBR regarding retransmissions applies. This category is given the third highest priority among categories and MD routing is adopted for restoration along with processing the VCs in decreasing order of bandwidth.
- **ABR:** The ABR category is designed for applications that can adapt to changing network performance. Hence, the number of dropped cells should be small. Furthermore, since it has no real-time requirements it is recommended to have the lowest restoration priority among service categories. The LDAP-scheme is used for rerouting since this scheme distributes the congestion over the set of available paths and results in lower call blocking. The VCs are restored in increasing order of bandwidth to the maximize number of restored VCs.

7. Numerical examples

In this section we presents the results of numerical studies to illustrate the effectiveness of using the priority scheme, and we discuss the network performance metrics

used. The performance metrics are; the outage time, the total number of cells lost, the percentage of VCs blocked, and the percentage of disrupted demand restored. Before discussing the experiments we look at the performance metrics in more detail.

Outage Time. The outage time for a VC is the sum of the detection, notification, identification, route selection, and establishment times as discussed in Section 3. The average outage times for all reconnected VCs are reported here.

Cells Lost. The number of cells lost for a VC is estimated by calculating the number of cells lost during the outage time. It is a function of the outage time and the traffic parameters PCR, SCR and MBS for which values are available. The nrt-VBR traffic is generated according to a Markov Modulated Poisson Processes (MMPP) model for which ATM Forum specifies a general formula to calculate the number of cells sent over a time interval [18]. A deterministic process generates the CBR traffic. For CBR the number of cells lost is the cell rate times the outage time. The rt-VBR traffic is generated by a Poisson process, for which the number of cells lost is the SCR times the outage time. Note that the number of cells lost calculated above is a lower bound because a potentially large number of cells will be dropped by switches along the restoration paths during the transient congestion period occurring immediately after establishment of new connections as discussed in Section 1. The additional cells dropped also need to be retransmitted, adding a positive feedback to the source. Thus the actual number of cells lost is at least as high as the number calculated here.

Call Blocking and Demand Restored. These traditional survivability metrics are useful to characterize the network performance when the network is not able to restore all disrupted connections (due to capacity limitations).

In order to evaluate the effectiveness of the proposed distributed priority scheme, we compared it with the default distributed method where no priority scheme is being used; i.e. disrupted VCs are processed for restoration in a random order. As discussed in Section 1 a useful benchmark for comparison is to use a centralized manager to find the optimal routes and rejections. Two schemes are used for benchmark purposes; one optimizing restoration of disrupted VCs in one category at a time, and another optimizing over all categories and all disrupted VCs. We studied a sample network and calculated the metrics discussed above for the following four restoration schemes.

- (1) *Distributed Random Scheme:* use no priorities, i.e. random ordering for restoration of all disrupted connections is used. MD routing was used for *path selection* since it is optimal under steady state conditions and likely to be used if dedicated fault recovery routing is not applied.
- (2) *Proposed Distributed Priority Scheme:* the scheme of Table 3 was implemented using strict priorities among

categories with the routing and ordering of connections within each category as given in the table.

- (3) *Optimum pr. Category Benchmark Scheme*: use strict priorities among categories as given in Table 3 with connection acceptance/rejection and path selection being determined exhaustively for all disrupted VCs within a category. That is, all possible restoration paths are tried for all possible orderings/rejections of all disrupted VC's. Thus, the connection acceptance/rejection and path selection are found by solving the optimization problem (5)–(8) one category at a time. The optimization problem is first solved for disrupted VCs in the nrt-VBR category using MH routing, after which the centralized link cost database is updated. Then the problem is solved for the disrupted CBR VCs using MD routing and the link cost database is updated again, before proceeding likewise for the remaining service categories.
- (4) *Optimum over All Categories Benchmark Scheme*: use loose priorities for both categories and connections by assigning a different routing scheme to each category as specified in Table 3. The optimization problem (5)–(8) is then solved jointly for all disrupted VCs in all categories. The link cost database was updated only after processing all VCs in all categories. The new paths are found based solely on the link status at the time of failure after resources of disrupted VCs have been released.

Numerical experiments comparing the four restoration schemes described above were conducted using the network with $|N| = 10$ nodes and $|L| = 42$ directed links shown in Fig. 1. The connectivity of the topology measured by the average node degree is 4.2, which is in the range of many existing networks. A C-program model of the network was developed. The fault recovery schemes were implemented in the network model both in a distributed fashion with each node maintaining a local link cost database for schemes 1 and 2, and with a centralized manager with a centralized

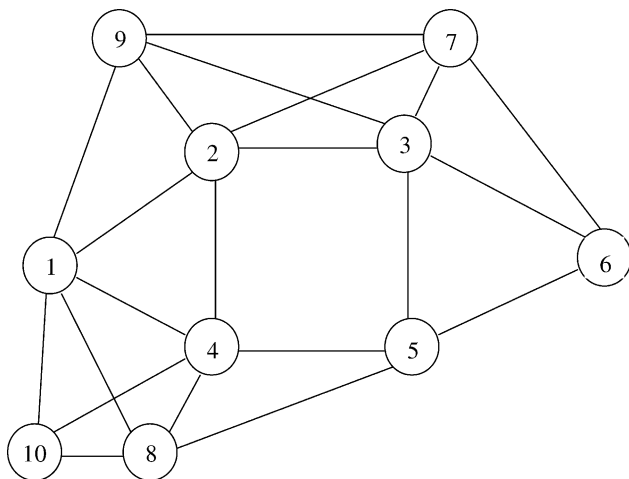


Fig. 1. Ten node network topology.

link cost database for schemes 3 and 4. For schemes 3 and 4, the restoration routes found by the centralized manager were downloaded into the source nodes in order to perform the *establishment* step reconnecting the disrupted VCs. A hop count limit of five nodes (four links) was used in the model to restrict the number of feasible paths and to speed up route selection. We let $t(H)$ denote the time it takes to reconnect a VC over a path with H switch-hops. In our calculations we use measurements from Ref. [20] where $t(2) = 395$, $t(3) = 535$, $t(4) = 619$, and $t(5) = 669$, all in ms. In the calculations a link capacity equal to OC-3 rate was used. A maximum steady-state link utilization threshold of 0.9 was used for all links and VCs were rejected if no route could be found without exceeding utilization the threshold. A sample set of results with nrt-VBR, CBR and rt-VBR traffic are reported here for two load scenarios: LITE load with average link utilization of 0.3 before the failure, and HEAVY load with average link utilization of 0.7 before the failure. For each load case a total of 10 experiments were conducted, in each individual experiment enough VCs were setup to create the desired average link utilization (≈ 600 and 1600 VCs on average for the LITE and HEAVY load, respectively). Each VC was set up by first selecting a source–destination pair randomly, then, a random service category was selected before a demand was picked. The demand was also picked randomly from the following ranges: nrt-VBR: 9.6–155 Kbit/s, CBR: a rate was picked from a discrete set of standard ISDN and B-ISDN rates (16, 64, 144, 384, 1488, 1536, 1544, 1920, 3152, 44 736, 274 176 all in kbps) rt-VBR: 9.6–3100 Kbit/s. The ranges were selected based on the rates of typical applications intended for these categories. A MMPP model was used for generating the nrt-VBR traffic, a Poisson model for creating the rt-VBR traffic and a deterministic model for generating CBR traffic.

After the VC's were set up, link 2–4 was failed. On average the link failure caused approximately 6% of the VC's to be disrupted. The time to detect the failure at node 4 and notify the source nodes was set to a constant of 1 s. The Route Selection time was set to a constant of 250 μ s per VC as in Ref. [20]. Note that a constant route selection time is not realistic for schemes 3 and 4 since we solve the optimization problem to find new paths. For each of the four schemes studied we calculated: the amount of demand restored, the number of VCs blocked, the outage time, and the number of cells lost. The results were then grouped by service category and averaged over the 10 experiments. The numerical results are summarized in Tables 4 and 5 for the LITE and HEAVY network load cases, respectively.

The LITE load case results in Table 4 show that the priority scheme (column 2) results in a shorter outage time and smaller number of lost cells for the nrt-VBR VCs when compared with the random case (column 1). Specifically, the average outage time per VC is reduced by approximately 50% from 0.54 to 0.26 s. At the same time the average number of lost cells per VC is reduced by 85 from 771 to

Table 4
Results of numerical examples for the LITE load case

Category	Metric	Random	Prioritized	Optimum pr. cat	Optimum all cat
nrt-VBR	Outage time	0.54	0.26	0.26	0.26
CBR	Outage time	0.47	0.46	0.46	0.46
rt-VBR	Outage time	0.53	0.74	0.73	0.76
nrt-VBR	Cells lost	771	117	117	117
CBR	Cells lost	3279	2614	2604	2617
rt-VBR	Cells lost	1796	2387	2360	2359
nrt-VBR	%Demand rest	100.0	100.0	100.0	100.0
CBR	%Demand rest	100.0	100.0	100.0	100.0
rt-VBR	%Demand rest	100.0	100.0	100.0	100.0
nrt-VBR	%VCs blocked	0.0	0.0	0.0	0.0
CBR	%VCs blocked	0.0	0.0	0.0	0.0
rt-VBR	%VCs blocked	0.0	0.0	0.0	0.0

117. Since these cells need retransmission the reduction will significantly improve network performance after the failure in terms of the transient congestion. Also, since the number of cells lost calculated here is a lower bound the effect of giving priority to nrt-VBR traffic would be even more pronounced in a real network. The CBR connections experience a slightly better performance when using the priority scheme: the outage time is nearly identical, and the number of cells lost is reduced by 20% from 3279 to 2614. These are normally not retransmitted. The performance gain for nrt-VBR and CBR traffic is at the cost of increasing the average outage time per VC for rt-VBR VCs by 40% from 0.53 to 0.74, and increasing the number of cells lost by 33% from 1796 to 2387. These cells are normally not retransmitted. Note that for the LITE load scenario sufficient spare capacity exists in the network such that no VCs are blocked. In comparing the *Prioritized* column with the *Optimum pr. cat* and *Optimum all cat* columns we see that the numbers reported are nearly identical. The main reason for this is, at light loads the solution of the optimization problem differs little from the paths selected by the proposed distributed priority scheme of Table 2.

For the lightly loaded network scenario all disrupted VCs were restored, while for a heavy loaded network scenario some VCs were restored and some were blocked. Thus the

analysis of the results given in Table 5 for the heavy load scenario is more complex. Both the outage time and the number of lost cells must be considered in light of the amount of demand restored and the number of VCs blocked. First, let us look at the nrt-VBR results in column 1 and 2 of Table 5 for random and prioritized restoration, respectively. The average outage time per VC is reduced by approximately 30% from 6.5 to 3.4 s. The average number of cells lost per VC is reduced by 15% from 6306 to 3805 cells. These cells need retransmission, and in a heavily loaded network this may cause the switch buffers along the new paths to overflow, causing a large number of additional cells in need of being retransmitted. As in the LITE load situation the number of cells lost calculated here is a lower bound so the effect of giving priority to nrt-VBR traffic would be even more pronounced in a real network. Note that the gain in performance for nrt-VBR performance is even better than indicated by the outage time and cells lost parameters. This is found by looking at the number of VCs blocked and amount of demand restored. By using the priority scheme the percentage of demand restored is increased by 9.2% from 24.3 to 33.5. The percentage of VCs blocked is reduced by 10.2% from 67.9 to 57.7.

For CBR traffic the performance results are approximately the same for the Random and Prioritized situations.

Table 5
Results of numerical examples for the HEAVY load case

Category	Metric	Random	Prioritized	Optimum pr. cat	Optimum all cat
nrt-VBR	Outage time	6.5	3.4	3.9	4.5
CBR	Outage time	6.1	7.4	7.8	7.6
rt-VBR	Outage time	6.8	11.0	13.5	11.7
nrt-VBR	Cells lost	6,306	3,805	5,358	6,384
CBR	Cells lost	16,061	22,110	22,235	18,456
rt-VBR	Cells lost	19,550	30,203	29,016	26,092
nrt-VBR	%Demand rest	24.3	33.5	42.6	41.7
CBR	%Demand rest	8.0	10.6	11.7	7.5
rt-VBR	%Demand rest	14.3	10.8	8.9	13.5
nrt-VBR	%VCs blocked	67.9	57.7	50.2	50.7
CBR	%VCs blocked	86.0	83.1	81.2	84.8
rt-VBR	%VCs blocked	84.4	89.1	89.2	82.6

The rt-VBR connections being restored third suffer by a reduced performance for all four parameters for the priority scheme in comparison with the random scheme.

Considering at the results from the optimum cases for HEAVY load the first observation is that the results for the two situations are nearly identical. This is explained by two factors; first, the cost range of MH and MD overlap, but only for a small part of the range of MD. Consequently the results for solving the categories separately versus jointly should be nearly equal. Also this is in part due to the network loading, while the number of generated VCs is equal for each service category (33.3% of each), the demand varies for each category, causing a small percentage of the demand to be for nrt-VBR VCs (0.4%), a large amount is for CBR traffic (83.3%), and a small amount for rt-VBR traffic (6.3%). Thus the capacity allocation is mostly controlled by the CBR connections, causing the routing in the two situations to be nearly the same. Now we compare the results of the two centralized optimization benchmark schemes 3 and 4 with the results from the proposed distributed priority scheme 2. We see that a centralized optimization based scheme results in significantly higher utilization of link capacity. For nrt-VBR connections approximately 9% more demand is restored, and 7% fewer VCs are blocked. We see that this comes at a cost of 35% increase of outage time and 65% increase of cells lost. For CBR and rt-VBR the performance is approximately the same for the distributed priority situation and the centralized situations.

Additional numerical results, illustrating similar behavior (including ABR traffic) is given in Refs. [13,21]. From our numerical results one can see that a priority scheme for traffic restoration after failures improves network performance. Furthermore it was seen that under heavy load conditions when VC call blocking occurs during restoration, there is a trade-off between the call blocking and the amount of demand restored which in turn results in a trade-off between outage time and cells lost. Note that in a real network each source node might have hundreds or thousands of disrupted VC connections to restore. Thus the number of lost cells will be more significant indicating that congestion will be an important issue. For these situations it generally may not be possible to restore all nrt-VBR connections and still satisfy the recommended restoration time constraints on the CBR and rt-VBR connections. Therefore it might be better to use a scheduling scheme to give a time slot to each category. For example, using the principle of weighted fair queueing common in processor sharing algorithms, the time spent for each category can be set proportional to the number of disrupted VCs in the category. This is currently under study.

8. Summary

In this paper we presented a new distributed scheme for prioritized restoration of virtual circuits in QoS based wide

area networks. Specifically the scheme gives highest priority to applications requiring short response time, low packet loss, and retransmission of lost packets. Second and third priority is given to real-time applications that do not retransmit lost cells, and among these, applications requesting constant bit rate service are given priority over applications requesting statistical guarantees on sustainable bit rate and maximum burst size. Other applications that can adapt to feedback from the network are given lowest priority.

As a benchmark for comparison with our distributed priority scheme we formulated the restoration problem as an integer programming optimization problem. The problem was first solved separately for each service category assuming fixed priorities among categories, then the problem was solved for all connections in all categories with relative priorities among categories. The relative priorities were imposed by assigning a suitable routing scheme together with a weight factor for path costs for each scheme and a rejection cost for each connection and each category.

The effect of using the prioritized restoration scheme was explored for two workload situations under centralized and distributed control. When comparing the proposed priority scheme and the default random scheme under distributed control, we find that the priority scheme outperforms the random scheme for all metrics for both lite and heavy workloads.

When comparing the proposed priority scheme and the centralized optimization based scheme for lite load situations we find that the schemes have practically the same performance since for lite load since the spare resources in the network are plentiful. In heavy load situations the optimization based scheme performs better than the distributed scheme regarding both amount of demand restored and number of connections restored. However, this comes at the cost of extended outage time and higher loss ratio for high priority connections. Thus there is a trade-off between the metrics. We see that if we could improve the distributed scheme regarding demand restored and number of connections restored we will increase connection outage time and loss ratio.

References

- [1] T.H. Wu, *Fiber Network Service Survivability*, Artech House, Boston, MA, 1992.
- [2] Andrea Fumagalli, I. Cernetti, M. Tacca, F. Masetti, R. Jagannathan, S. Alagar, *Survivable networks based on optimal routing and WDM self-healing rings*, In *Proceedings of INFOCOM'99* 2 (1999) 726–733.
- [3] *Special Issue, Integrity of Public Telecommunications Networks*, *IEEE Journal of Selected Areas in Communications*, Editors Tsong-Ho Wu, J.C. McDonald, T.P. Flanagan, Ken-Ichi Sato, 12(1), January 1994.
- [4] R. Doverspike, B. Wilson, *Comparison of capacity efficiency of DCS network restoration routing techniques*, *Journal of Network and System Management* 2 (2) (1994) 95–123.

- [5] W. Grover, J. Doucette, M. Clouqueur, D. Leung, D. Stamatelakis, New options and insights for survivable transport networks, *IEEE Communications Magazine* 40 (1) (2002) 34–41.
- [6] H. Fuji, N. Yoshikai, Restoration message transfer mechanism and restoration characteristics of double search self-healing ATM network, *IEEE Journal of Selected Areas in Communications* 12 (1) (1994) 149–158.
- [7] K. Murakami, H. Kim, Virtual path routing for survivable ATM networks, *IEEE/ACM Transactions on Networking* 4 (1) (1996) 22–39.
- [8] K.R. Krishnan, R. Doverspike, C. Pack, Improved survivability with multi-layer dynamic routing, *IEEE Communications Magazine* 33 (7) (1995) 62–68.
- [9] K. Kyandoghere, Survivability performance analysis of rerouting strategies in an ATM/VP DCS survivable mesh networks, *ACM Computer Communication Review* 28 (5) (1998).
- [10] A. Banerjee, Fault recovery for guaranteed performance communications connections, *IEEE/ACM Transactions on Networking* 7 (5) (1999) 653–668.
- [11] W.P. Wang, D. Tipper, B. Jaeger, D. Medhi, Fault Recovery Routing in Wide Area Packet Networks, *Proceedings of 15th International Teletraffic Congress—ITC 15*, Washington, DC, 22–27 June 1997, Elsevier, 1997, pp. 1077–1086.
- [12] D. Hsing, B. Cheng, G. Goncu, L. Kant, A restoration methodology based on pre-planned source routing in ATM networks, *Proceedings of the IEEE International Conference on Communications, IEEE ICC'97*, Montreal, Canada, 8–12 June 1 (1997) 277–282.
- [13] B. Jæger, D. Tipper, On fault recovery priority in ATM networks, *Proceedings of IEEE International Conference on Communications, Proceedings of the IEEE International Conference on Communications, ICC'98*, Atlanta, GA, June 7–11 3 (1998) 1410–1414.
- [14] A. Alles, ATM Internetworking, *InterOp + Network Engineering Conference*, Las Vegas, NV, March 1995, <http://www-europe.cisco.com/warp/public/614/12.html>, accessed June 25, 2002.
- [15] American National Standards Institute (ANSI) Committee T1, Network Survivability Performance, Technical Report No. 24, November 1993.
- [16] ITU-T Recommendation I.371 (06/97), Traffic Control and Congestion Control in B-ISDN, Geneva, June 1997.
- [17] The ATM Forum Technical Committee, Traffic Management Specification Version 4.0, ATM Forum Document af-tm-0056.000, April 1996.
- [18] The ATM Forum Technical Committee, Private Network—Network Interface Specification Version 1.0 (PNNI 1.0), ATM Forum Document af-pnni-0055.000, March 1996.
- [19] The ATM Forum Technical Committee, BISDN Inter Carrier Interface (B-ICI) Specification Version 2.0 (Integrated), ATM Forum Document af-bici-0013.003, December 1995.
- [20] D. Niehaus, A. Battou, A. McFarland, B. Decina, H. Dardy, V. Sirkay, B. Edwards, Performance benchmarking of signaling in ATM networks, *IEEE Communications Magazine* 35 (8) (1997) 134–143.
- [21] B. Jæger, *Traffic Restoration in Survivable Wide Area Communication Networks*, PhD Dissertation, Department of Informatics, University of Bergen, Bergen, Norway, September 2000.
- [22] J. Sosnosky, Service applications for SONET DCS distributed restoration, *IEEE Journal of Selected Areas in Communications* 12 (1) (1994) 59–68.
- [23] D. Tipper, J. Hammond, S. Sharma, A. Khetan, K. Balakrishnan, S. Menon, An analysis of the congestion effects of link failures in wide area networks, *IEEE Journal of Selected Areas in Communications* 12 (1) (1994) 179–192.
- [24] D. Bertsekas, R. Gallager, *Data Networks*, Second ed., Prentice Hall, 1992.
- [25] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, *Network Flows*, Prentice Hall, 1993.
- [26] J. Kobza, *The significance of transients following failures and repairs in packet-switched networks*, PhD Thesis, Virginia Tech, February 1993.
- [27] D. Logothetis, K. Trivedi, The effect of detection and restoration times on error recovery in communications networks, *Journal of Network and Systems Management* 5 (2) (1997) 173–195.
- [28] M. Rumsewicz, D. Smith, A comparison of SS7 congestion control options during mass call-in situations, *IEEE/ACM Transactions on Networking* 3 (1) (1995) 1–9.



Bjørn Jæger is an Associate Professor of Informatics in the Department of Informatics at Molde University College, Norway. Prior to joining Molde University College in 1993, he was a system designer and group leader at ABB Nera, Department of Telecommunication Management Networks, Bergen, Norway. He received the MS degree in nuclear physics and the PhD degree in informatics from the University of Bergen, Norway. His current research interests are fault tolerant distributed infrastructures, traffic restoration procedures for survivable networks, network management and broadband infrastructure development. Professor Jæger's research has been supported by grants from Norwegian Research Council and Department of Informatics, University of Bergen.



David Tipper is an Associate Professor of Telecommunications in the Department of Information Science and Telecommunications at the University of Pittsburgh, USA. Prior to joining Pitt in 1994, he was an Associate Professor of Electrical and Computer Engineering at Clemson University in Clemson, SC. He is a graduate of the University of Arizona and Virginia Tech. His current research interests are network design and traffic restoration procedures for survivable networks, network control (i.e. routing, flow control, etc.), performance analysis, wireless and wired network design. Professor Tipper's research has been supported by grants from various government and corporate sources such as NSF, DARPA, NIST, IBM and AT&T. He is the Technical Program Chair of DRCN 2003 and on the editorial board of the *Journal of Network and Systems Management*.