

Simulation and modeling of the restoration performance of path based restoration schemes in planar mesh networks

Manish Bhardwaj and Leon McCaughan

*Electrical and Computer Engineering Department, University of Wisconsin-Madison, 1415 Engineering Drive,
Madison, Wisconsin 53706
mbhardwaj@wisc.edu*

Anatoli Olkhovets and Steven K. Korotky

*Bell Laboratories, Lucent Technologies, 791 Holmdel-Keyport Road, Holmdel, New Jersey 07733
skk@lucent.com*

In this paper we formulate an analytic framework for the restoration performance of path based restoration schemes in planar mesh networks. We analyze different switch architectures and signaling schemes and model their total restoration interval. We also evaluate the network global expectation value of the time to restore a demand as a function of network parameters. We analyze a wide range of nominally capacity-optimal planar mesh networks and find our analytic model to be in good agreement with numerical simulation data.

OCIS codes: 060.2330, 060.4250.

1. Introduction

The consideration of mesh network topologies and architectures as candidate optical data transport strategies has stimulated work to quantify the requirements and potential benefits of adopting these architectures and also to formulate algorithms for provisioning capacity and routing demands. As optical networks are required to be survivable against link and node failures, the analysis of the restoration capacity requirements and temporal performance of restoration protocols is critical to this understanding and various simulation studies have been performed on the restoration capacity and time of planar mesh networks, [1-8]. Analytic models of the restoration capacity requirements show that path based restoration schemes, such as path restoration disjoint (PRd), are more capacity efficient than link based schemes, such as link restoration (LR), an observation confirmed by simulation studies, [5-10]. However it is qualitatively understood that shared path restoration is more complicated to implement and incurs higher operational expenditure in the form of larger amount of signaling between cross-connects (XCs) and processing of restoration requests, [11]. To better understand the operational versus capital expenditure tradeoffs for shared mesh restoration we investigate the time to restore demands for shared path disjoint restoration. We model both the *total* time to restore (TTTR) all demands for all single link failure events and the *mean* time to restore (MTTR) a demand for shared PRd. In developing the present analytic model we take into account the network topology, the demand profile, the XC architecture employed, and the signaling scheme used to convey requests to the switches along the restoration path. We test the accuracy and predictive capabilities of the analytic model by comparing it to simulation data from a large set of capacity optimal nominally planar networks of varying size, average degree and demand profile.

In shared mesh restoration, spare capacity is connected on the restoration path of a demand *after* a link failure has occurred. The restoration interval depends on some, and in the worst case scenario, all, of the failed demands [9, 7, 12]. Although determination of the total restoration interval based on the worst case scenario may be accurate for link based restoration, we find that such a scenario does not reflect the restoration interval for path based restoration. Previous simulation analyses have shown that the TTTR is approximately linear with the total number of demands in a network [2, 7]. Other simulation studies also point to the dependence of restoration interval on the number of demands [1, 3, 4]. We have recently described a model for the MTTR based on such a linear relationship [13].

The remainder of the paper is organized as follows. In Section 2 we describe the network model and define network variables. In Sections 3 and 4 we derive the TTTR and introduce our model for the MTTR for various XC architectures and signaling schemes. In Section 5 we describe our simulation methodology and present simulation results. We conclude the paper in Section 6.

2. Network model

In this section we define the key variables utilized in our analysis, as well as enumerate the network characteristics. The average value of a set $\{q\}$ of the variable q will be represented by $\langle q \rangle$ and its variance will be represented by $\sigma^2(q)$. The covariance of two sets $\{p\}$ and $\{q\}$ having the same number of elements is represented by $\sigma^2(p, q)$. The network is represented as a directed graph, $G(N, L)$, where $\{n\}$ is the set of nodes and $\{l\}$ is the set of links and N and L are the numbers of nodes and directed links in the network, respectively. The degree of a node, δ , is the number of duplex physical links attached to the node which is half the number of directed links attached to the node. The global average of degrees of nodes is then

$$\langle \delta \rangle = \frac{L}{N} \quad (1)$$

Throughout this paper all demands are considered as one-way (unidirectional) and the route of a demand when the network has no failures is called its primary path. The total number of demands in the network is denoted by D . We consider that each demand occupies a single unit of capacity, e.g. a channel or a wavelength, and there can be multiple demands between a node pair. The working capacity on a link W_0 is therefore the number of primary paths (demands) traversing the link and the global expectation of the working capacity on a link may be expressed as [14]

$$\langle W_0 \rangle = \frac{D \langle h \rangle}{L} = \frac{\langle d \rangle \langle h \rangle}{\langle \delta \rangle} \quad (2)$$

where $\langle d \rangle$ is the average number of demands that originate at a node, and $\langle h \rangle$ is the demand-weighted average of the numbers of hops in the primary paths of the demands. The W_0 demands on a link are comprised of terminating demands, W^t , i.e. demands that originate/terminate at one of the nodes attached to the link, and through demands, W^{th} . The demands on a link are counted relative to one of the nodes attached to the link. If d_i are the number of demands that originate or terminate at a node n_i of degree δ_i , then on average there are d_i/δ_i terminating demands on each directed link connected to n_i and $\langle W^t \rangle$ and $\langle W^{th} \rangle$ can be approximated by Eqs. 3 and 4, respectively.

$$\langle W^t \rangle \cong \frac{\langle d \rangle}{\langle \delta \rangle} \cong \frac{\langle W_0 \rangle}{\langle h \rangle} \quad (3)$$

$$\langle W^{th} \rangle = \langle W_0 \rangle - \langle W^t \rangle \cong \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle} \right) \quad (4)$$

In the event that a link along the primary path fails, the demand may be rerouted around the failed link (LR) or on a path disjoint route (PRd). In either case the route used for the recovery of the demand when the primary path has failed will be referred to as the backup path and the demand-weighted average length of the backup path is denoted by $\langle h_r \rangle$. Restoration capacity must therefore be allocated on surviving links to reroute failed demands and this capacity is shared among backup paths. The restoration capacity can be expressed relative to the working capacity in the network by the measure *extra capacity*. The global expectation of the extra capacity is the fractional increase in total network capacity to ensure survivability against any single link failure over the *minimum* network capacity necessary to support the working demands alone. The minimum network capacity required to support working demand is the working capacity in the network when the primary paths are routed via minimum hop routing and therefore $\langle h \rangle$ represents the average number of hops of the primary paths when demands are routed via minimum hop routing. The total capacity on a link is the capacity required to route the primary and backup paths traversing the link and its mean can be expressed as

$$\langle W \rangle = \langle W_0 \rangle (1 + \langle \kappa \rangle) \quad (5)$$

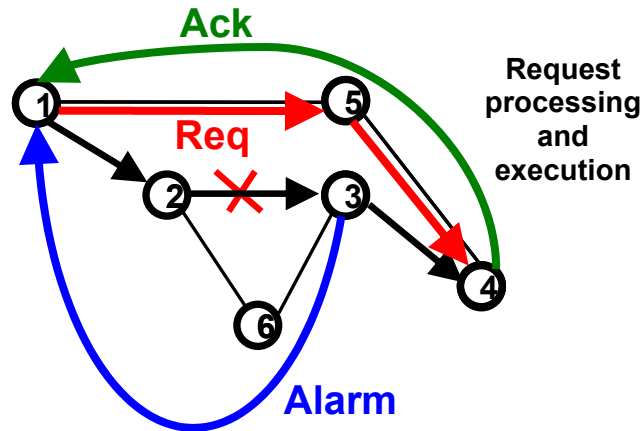


Fig.1. Example restoration process

where $\langle \kappa \rangle$ is the average extra capacity and is described analytically for LR and PRd in [10]. Although PRd is more capacity efficient than LR, it is qualitatively understood to have higher restoration latency and we investigate the restoration time performance of PRd quantitatively in the next Section.

3. Analytic analysis of TTTR

A relevant parameter for assessing network restoration operational performance is the total time to restore demands after a link failure event. Various simulation studies have suggested that the time to restore all failed demands totaled over all single link failure events (TTTR) is approximately linear with the number of demands, D , for a particular network and demand profile. Additionally, the analysis of TTTR must take into account the XC architecture and the restoration signaling scheme employed, and we describe first the architectures and signaling schemes that we consider, followed by the development of the analytic model of the TTTR.

3.A. Restoration signaling

We begin the discussion of the TTTR with a brief description of the two signaling protocols that we analyze and introduce the various parameters relevant to the analytic model. We consider first the two-way source based routing (SBR) signaling scheme and summarize the sequence of events that take place to restore demands after a single link failure event. When a link failure occurs, the nodes adjacent to the failed link detect the loss of signal after an interval T_D and begin notification of the failure to all source nodes whose demands have failed. The nodes detecting the failure send an alarm message that propagates hop by hop upstream on the primary paths to the source nodes. Upon notification of the failure of demands, source nodes initiate the reconfiguration stage of the restoration operation by requesting assignment of capacity on the backup path by transmitting a request (Req) message, which propagates hop by hop along the backup path. The link transit time (one hop flight time) is denoted by T_L . Each XC may receive multiple requests for capacity, and processing of requests is assumed to be performed serially while the XC reconfiguration operations may be performed serially or in parallel depending upon the XC architecture. Thus a XC can be modeled as two buffers in series – a processing buffer and a reconfiguration buffer, with service delays denoted by T_P and T_{SC} respectively. Finally when the reconfiguration stage is complete the destination node sends an acknowledgement message to the source node. This completes the description of SBR. An example restoration scenario is shown in Fig. 1 where the link between nodes 2 and 3 has failed. Source node 1, when notified of the failure of its demand to node 4, sends a request on the backup route 1-5-4 and receives an acknowledgement from node 4 after XCs at nodes 1, 5 and 4 have been reconfigured.

More sophisticated one-way messaging schemes that have been recently suggested, such as enhanced source based routing (eSBR), combine features of optical mesh networks with parallel control architectures that incorporate out of band signaling and one-way resource reservation protocols, [12]. The

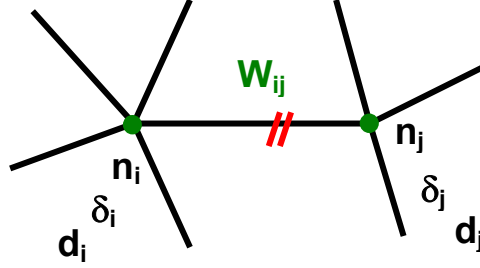


Fig.2 Single link failure event.

advantage of out of band signaling is that control messages can be forwarded from one XC to the next without waiting for the requested reconfiguration operations to complete. As the Req message does not wait at a XC after being processed, but immediately moves to the next XC in the backup path, the request for reconfiguration for a path will in general be present in more than one reconfiguration buffer in the backup path. As a limiting case, we may consider that the link transit interval is 0. The Req will therefore reach all the XCs in the backup path at effectively the same time (assuming the message processing interval is also small). Thus the reconfiguration operation is considered complete when the XC with the longest reconfiguration buffer length has serviced the request, i.e. performed the reconfiguration operation. We notice if the link transit interval is greater than zero that the XC reconfiguration operation is being performed in parallel with the transmission of the Req message. Therefore, if the link transit interval is much smaller than the time required to reconfigure XCs on the backup path, we expect the reconfiguration interval for eSBR to not include a flight time component. The link transit interval may be comparable to T_{SC} for long haul networks, but a link is modeled as a delay line while a XC is modeled as a series of buffers and therefore the time required to reconfigure a XC will be many multiples of T_{SC} for serial XC architectures. For parallel XC architectures, the flight time may be comparable to the time required to configure XCs on the backup path for long haul networks and the flight time will be included in the reconfiguration interval.

Additionally, the acknowledgement message (Ack) is not present in eSBR. A source node starts transmitting on the backup path after waiting for a fixed guard time after sending a request message (Req) along the backup path. Ideally this guard time should be equal to the time required to configure the XCs on the backup path, referred to as the reconfiguration interval; therefore the accurate prediction of the reconfiguration interval is crucial to the proper functioning of these protocols. Underestimation of the guard time can result in data arriving at a XC that has not been reconfigured, and the data will be lost. This will become especially important for future all-optical networks, in which the source nodes cannot practically buffer the data. Conversely, overestimation of the guard time will render the protocol inefficient.

3.B. Analytic model

In the discussion above we outlined the XC architectures-serial and parallel, and signaling schemes- SBR and eSBR, which when combined constitute the four path based restoration architectures for which we derive the total time to restore. We start the discussion with the TTTR for eSBR and SBR signaling schemes for a network with serial XC architecture. The time interval required to restore a demand is dominated by the time required to reconfigure the backup path, the reconfiguration interval, for that demand. Therefore the time to restore all demands after a link failure is the maximum reconfiguration interval of all demands affected by the failure and the TTTR is the sum of all such maximum reconfiguration intervals after all single link failures. For eSBR estimating the TTTR therefore involves estimating the length of the longest reconfiguration buffer after a link failure, i.e. estimating the length of the reconfiguration buffer at the XC that receives the maximum number of Req messages. We begin by estimating the length of the longest reconfiguration buffer after a link failure and then estimating the TTTR for eSBR for serial XC architectures. We then add the contribution of the flight time of Req messages to estimate the TTTR for SBR. Finally we model the TTTR for eSBR and SBR signaling schemes for networks employing parallel XC architectures.

To estimate the length of the longest reconfiguration buffer we focus on the nodes adjacent to the link failure. The reasoning for this is that all terminating demands on the link have failed and the Req

messages for these demands immediately occupy the reconfiguration buffers at the nodes adjacent to the link failure. In addition to the terminating demands these nodes also reconfigure a fraction of the through demands, which will increase the buffer lengths further. Since the nodes are adjacent to the failure, the failure notification interval is small, and in the eSBR signaling scheme there is no Ack message. The TTTR can therefore be expressed entirely by the time required to service all Req messages in the reconfiguration buffers in these nodes. We refer to Fig. 2 where the link, l_{ij} , between nodes n_i and n_j has failed. The reconfiguration queue for node n_i , b_i , is d_i/δ_i and for n_j is $b_j = d_j/\delta_j$. Thus the maximum number of reconfiguration operations that a node must perform based on terminating demand considerations when link l_{ij} fails is

$$O_{ij} = \max\left(\frac{d_i}{\delta_i}, \frac{d_j}{\delta_j}\right) \quad (6).$$

The TTTR for terminating demands summed over all single link failures can therefore be expressed as

$$\text{TTTR} = \sum_{(i,j) \in E} O_{ij} T_{SC} \quad (7)$$

where E is the set of directed edges in the network graph. We now consider the contribution of through demands. The backup path of a through demand involves $(\langle h_r \rangle + 1)$ nodes on average from among a possible N choices. Thus the average number of Req messages for restoration of through demands that a node will receive is $\langle W_{th} \rangle (\langle h_r \rangle + 1)/N$. Including the interval required for reconfiguration operations of these through demands in the total time to restore gives us the final expression for TTTR as

$$\text{TTTR} = \sum_{(i,j) \in E} O_{ij} T_{SC} + \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle}\right) \frac{(\langle h_r \rangle + 1)}{N} L T_{SC} \quad (8)$$

where the second term in Eq. 8 includes the multiplication factor of L because the TTTR is tallied for all single link failures.

For a regular topology and uniform demand profile we observe that Eq. 6 can be expressed as

$$O_{ij} = \frac{d}{\delta} \quad \forall (i, j) \in E \quad (9).$$

Substituting Eq. 9 in Eq. 8, the TTTR for networks with regular topologies and uniform demand profile may be expressed as

$$\text{TTTR} = \frac{d}{\delta} L T_{SC} + \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle}\right) \frac{(\langle h_r \rangle + 1)}{N} L T_{SC} \quad (10).$$

Noting that $L/\delta = N$, the number of nodes, and that $Nd = D$, the total number of demands, the expression for the total time to restore for a regular network with uniform demand profile may be expressed as

$$\text{TTTR} = D T_{SC} + \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle}\right) \frac{(\langle h_r \rangle + 1)}{N} L T_{SC} \quad (11).$$

The above result, Eq. 11, shows that the total time to restore is approximately linear in the total number of demands D in the network, as the average working capacity on a link, $\langle W_0 \rangle$, is linear with D (see Eq. 2). Latencies of SBR will include a flight time component in addition to the reconfiguration buffer delay as discussed in Section 3.A, and the global expectation of the flight time along the backup path can be

approximated as $\langle h_r \rangle T_L / 2$ i.e. the global expectation of the point of failure is midway between the source and destination nodes of a failed demand. Additionally the acknowledgement notification interval can be approximated as $\langle h_r \rangle (T_P + T_L)$ which includes the global expectation of the processing interval of the acknowledgement at XCs on the backup path and the flight time of the acknowledgement message. Thus the TTTR for two-way signaling employing the serial XC architecture can be expressed as

$$\text{TTTR} = \sum_{(i,j) \in E} O_{ij} T_{SC} + \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle} \right) \frac{(\langle h_r \rangle + 1)}{N} L T_{SC} + \frac{3\langle h_r \rangle}{2} L T_L + \langle h_r \rangle L T_P \quad (12),$$

for general mesh networks. For uniform demand profile and regular topology, the TTTR for SBR signaling scheme can be found by applying Eq. 9 to Eq. 12 in analogy to the evolution of Eq. 11 from Eq. 8.

For parallel XC architectures, each reconfiguration operation is performed in parallel and therefore the reconfiguration queue is 1 for every Req message at each XC. Consequently the time to restore all failed demands after a link failure is determined by the maximum time it takes for a Req message of a failed demand to arrive at the destination node, i.e. the maximum length of the backup path among all failed demands. When link l_{ij} fails we denote the set of failed demands as D_{ij} which is a subset of the set of demands D in the network. Thus the TTTR for eSBR employing parallel XCs can be expressed as

$$\text{TTTR} = \sum_{(i,j) \in E} \max_{k \in D_{ij}} \left(T_D + \frac{h_k}{2} (T_P + T_L) + h_{rk} (T_P + T_L) + T_{SC} \right) \quad (13),$$

where h_k and h_{rk} are the lengths of the primary and backup paths respectively, of a demand $k \in D_{ij}$. The first term inside the parenthesis on the right hand side is the time to detect a link failure, the second term is the average notification interval and the processing interval of alarm messages. The third term is the flight time and processing interval of the Req message on the backup path and the last term is the XC reconfiguration interval at the destination node. Distinct from the serial XC scenario, the TTTR for eSBR for parallel XC architecture includes a contribution from the flight time interval of the Req message; however, for SBR we must also include the acknowledgement notification interval. Therefore the TTTR for parallel XC architecture can be expressed as

$$\text{TTTR} = \sum_{(i,j) \in E} \max_{k \in D_{ij}} \left(T_D + \frac{h_k}{2} (T_P + T_L) + 2h_{rk} (T_P + T_L) + (h_{rk} + 1) T_{SC} \right) \quad (14).$$

Here the third term inside the parenthesis on the right hand side has a factor of 2 due to the addition of flight and processing interval of the acknowledgement message. The last term in the parenthesis on the right hand side of Eq. 14 has a factor of $(h_{rk} + 1)$ representing the latency due to the Req message waiting at intermediate nodes on the backup path before being forwarded to the next XC due to in-band signaling.

The TTTR estimate for networks with parallel XC architecture expressed by Eqs. 13 and 14 can only be calculated from knowledge of the routing information in order to find the maximum length of a backup path of a demand affected by a link failure or through a suitable approximation of the average length of the maximum backup path lengths. The TTTR estimate for serial XCs for a general network does not require routing information and can be calculated quickly from knowledge of the demand profile and network topology (see Eqs. 8 and 12). An approximate closed form expression for the TTTR of serial XC architectures can be arrived at by exploring the relationship between TTTR and restoration capacity for regular networks with uniform demand profiles. The average restoration capacity on a link is represented as $\langle R \rangle \equiv \langle \kappa \rangle \langle W_0 \rangle$, and therefore the total restoration capacity in the network is $\langle R \rangle L = \langle \kappa \rangle \langle W_0 \rangle L$. In [10], we identified two estimates for the average extra capacity for a general mesh network. The indivisible bound represents the situation where only one link attached to a node participates in restoration of failed demands. In the divisible bound all surviving links (or links disjoint from the primary path) directed in the same manner as the failed link, which are $\delta - 1$ in number, attached to a node participate in the restoration process. The divisible bound is a lower estimate of restoration capacity due to participation of more links in restoration and the average restoration capacity on a link in a regular network with uniform demand profile in the divisible bound is expressed as

$$\langle R \rangle \equiv \langle \kappa \rangle \langle W_0 \rangle = \frac{D}{L(\delta-1)} + \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle} \right) \left(\frac{\langle h_r \rangle}{L/2 - \langle h \rangle} \right) \quad (15).$$

Comparing Eqs. 11 and 15, we see that for eSBR signaling in a network with serial XCs, the TTTR can be expressed approximately in terms of extra capacity as

$$\text{TTTR} \approx (\langle \delta \rangle - 1) \langle \kappa \rangle \langle W_0 \rangle L T_{SC} \quad (16).$$

The average length of the longest reconfiguration buffer after a link failure is denoted by $\langle B \rangle$. Recalling that the TTTR for eSBR for serial XCs is the sum of the times required to service all demands in the longest reconfiguration buffers after a link failure, we can express TTTR as

$$\text{TTTR} \equiv \langle B \rangle L T_{SC} \quad (17).$$

Comparing Eqs. 16 and 17, we see that in the divisible bound, the average length of the longest reconfiguration buffer after a link failure can be expressed as

$$\langle B \rangle \approx (\langle \delta \rangle - 1) \langle \kappa \rangle \langle W_0 \rangle = (\langle \delta \rangle - 1) \langle R \rangle \quad (18).$$

The average restoration capacity of a link is derived by focusing on the links incident on a node adjacent to the link failure and directed in the same manner as the failed link. Therefore Eq. 18 indicates that the average length of the longest reconfiguration buffer is approximately equal to the average restoration capacity used for restoration at a node adjacent to the link failure if all $\delta-1$ surviving directed links participate in the restoration process i.e. the divisible bound. At the other extreme only one link incident on the node adjacent to the link failure participates in restoration. This situation represents the indivisible bound and for full redundancy two links attached to the node adjacent to link failure and directed in the same manner must be assigned restoration capacity, denoted by $\langle R' \rangle$. However only one link will actually participate in restoration of failed demands. There are $2N$ number of links assigned restoration capacity in the network. Since the average restoration capacity on a link is represented by $\langle R \rangle$, in the indivisible bound the total restoration capacity in the network can be represented by $\langle R \rangle L = 2 \langle R' \rangle N$. Therefore the restoration capacity on links incident on the node adjacent to the failure and participating in restoration is $\langle R' \rangle = \langle R \rangle L / 2N = \langle R \rangle \delta / 2$. Thus in the indivisible bound, the average length of the longest reconfiguration queue can be approximated as

$$\langle B \rangle \approx \frac{\langle \delta \rangle \langle \kappa \rangle \langle W_0 \rangle}{2} \quad (19).$$

Utilizing the analytic formalism for extra capacity in the divisible or indivisible bound for general mesh networks, [10], Eqs. 18 and 19 when substituted into Eq. 17 represent an approximate closed form expression for TTTR for general mesh networks. Alternatively if the extra capacity of a network is known, the TTTR can be bounded by Eqs. 17, 18, and 19.

The analysis of the TTTR has been performed by viewing it as the sum of the maximum restoration intervals of demands affected by a link failure, however it gives little insight into the *average* restoration interval of a demand after link failure. This is because the distribution of the restoration intervals of demands is unknown, making analytic derivations of the MTTR difficult. We take a semi-empirical modeling approach to this problem below by systematically studying the dependencies of the MTTR on network variables.

4. Modeling of MTTR

The mean time to restore a demand after a link failure is defined as the global expectation of the restoration interval of a demand. As the expectation is taken over demands, the MTTR is independent of the *number*

of demands in the network and becomes a characteristic parameter of the network topology similar to the extra capacity, by which different topologies may be compared. We first develop the relationship between the MTTR and network variables by quantifying the various contributions to the restoration interval. We then introduce a construction wherein a demand is modeled as multiple flows, which results in a simple closed form relationship between MTTR and network variables

4.A. Analytic model

We refer again to the restoration process described in Sec. 3.A for SBR and eSBR. When a link failure occurs, the nodes adjacent to the failure detect the failure after an interval T_D , and send Alarm signals upstream along the primary paths to notify nodes whose demands have been disrupted. An Alarm message is processed and then forwarded at each XC on the backup path. In our network model, a failure can occur at any hop of a primary path with equal probability and therefore the global expectation of the hop distance between a node adjacent to the failure and a source node is half the primary path length. Therefore the average failure detection and notification interval, $\langle T_{DN} \rangle$, may be expressed as

$$\langle T_{DN} \rangle = T_D + \left(\frac{\langle h \rangle}{2} \right) (T_P + T_L) \quad (20)$$

Similarly the time interval of the acknowledgment stage, $\langle T_{ACK} \rangle$, is just $\langle h_i \rangle (T_P + T_L)$. The calculation of these intervals is based on the routing information and can be estimated by approximating the average lengths of the primary and backup paths. The time interval of the reconfiguration stage of the restoration process, denoted as $\langle T_{RC} \rangle$, however depends not only on the average path lengths but also the average length of the reconfiguration buffer at XCs on the backup paths.

The timelines of the reconfiguration stage for SBR for serial and parallel XC architectures are shown in Figs 3(a) and (b), respectively. The timelines in Fig. 3 represent a simplified scenario wherein the buffer length at each XC on the backup path is the same. More rigorously, we consider the case when link l_{ij} fails and the set of failed demands is denoted by D_{ij} , which is a subset of the set of demands D in the network. We denote by b_{kn} the reconfiguration buffer length at cross-connect XC_n on the backup path of length h_{rk} after the arrival of a reconfiguration request message for demand $k \in D_{ij}$. From Fig. 3(a) it can be seen that the reconfiguration interval, T_{RC}^k , of demand k after link l_{ij} fails for SBR with serial XC architecture is

$$T_{RC}^k = \sum_{n=0}^{h_{rk}} (T_P + b_{kn} T_{SC}) + h_{rk} T_L \quad (21)$$

Expressing the average reconfiguration buffer length along the backup path of demand k , $\langle b_k \rangle$, as

$$\langle b_k \rangle = \frac{1}{h_{rk} + 1} \sum_{n=0}^{h_{rk}} b_{kn} \quad (22)$$

Eq. 21 may be rewritten in terms of average reconfiguration buffer length, $\langle b_k \rangle$, as

$$T_{RC}^k = (h_{rk} + 1) (T_P + \langle b_k \rangle T_{SC}) + h_{rk} T_L \quad (23).$$

The sum of the reconfiguration intervals of all demands $k \in D_{ij}$ for all link failures is therefore expressed as

$$\sum_{(i,j) \in E} \sum_{k \in D_{ij}} T_{RC}^k = L \langle W_0 \rangle \langle T_{RC} \rangle = \sum_{(i,j) \in E} \sum_{k \in D_{ij}} (h_{rk} + 1) (T_P + \langle b_k \rangle T_{SC}) + h_{rk} T_L \quad (24),$$

because the global expectation of the number of demands that fail when a link fails is the average working

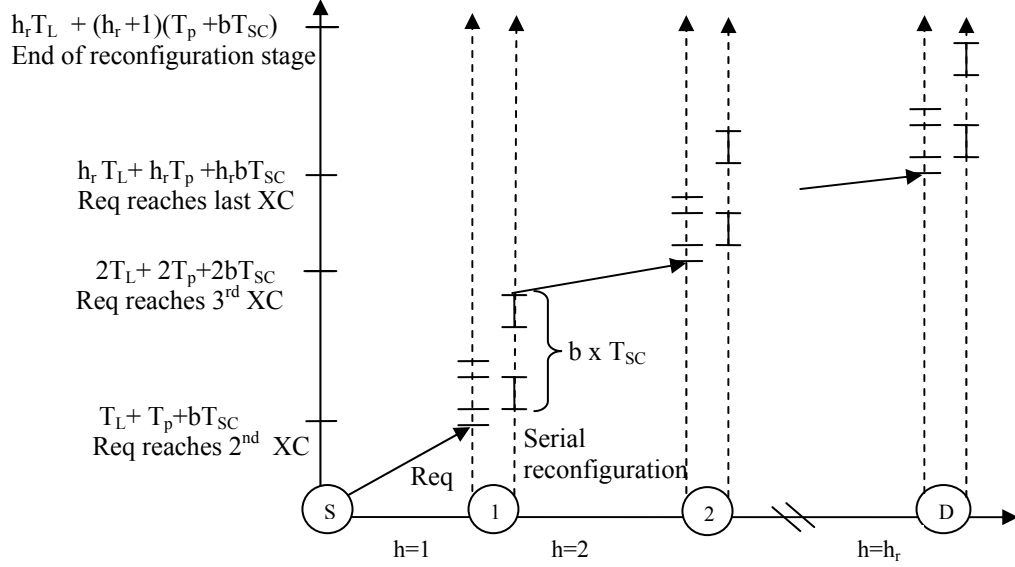


Fig.3 a. Source based routing (SBR) reconfiguration stage timeline representation for serial XCs.

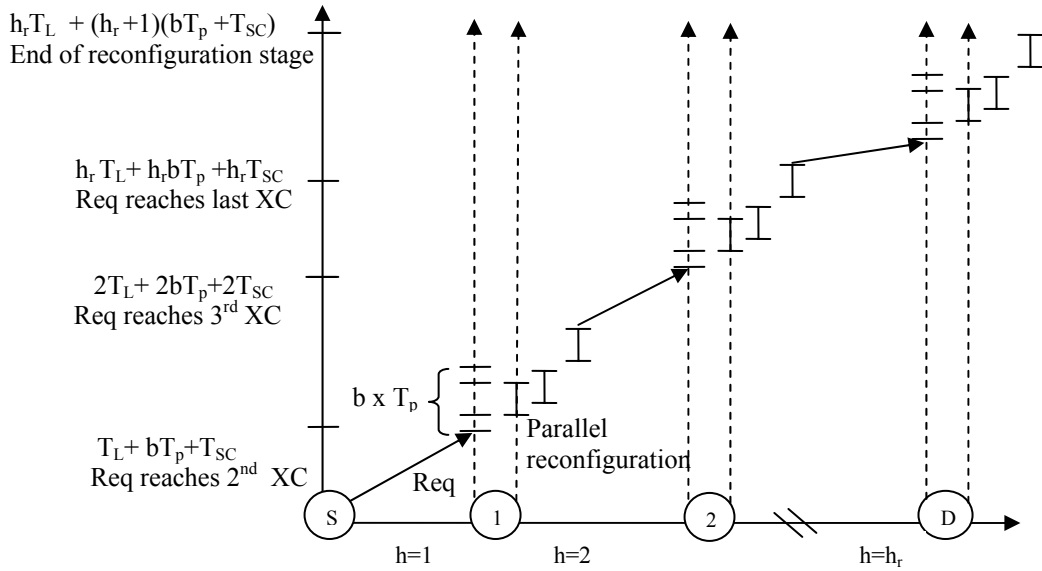


Fig.3 b. SBR reconfiguration stage timeline representation for parallel XCs.

capacity on a link, $\langle W_0 \rangle$. The summation on the right hand side can be related to the demand-weighted expectation of backup path lengths by noting that

$$\sum_{(i,j) \in E} \sum_{k \in D_{ij}} h_{rk} = \sum_{k \in D} h_k h_{rk} = \frac{D}{D} \sum_{k \in D} h_k h_{rk} = D \langle h, h_r \rangle \quad (25).$$

The joint expectation of two variables may be expressed as

$$\langle pq \rangle = \langle p \rangle \langle q \rangle + \sigma^2(p, q) \quad (26).$$

Using the relationship that $D\langle h \rangle = L\langle W_0 \rangle$ from Eq. 2, and Eq. 26, the summation of backup path lengths in Eq. 25 can be expressed as

$$\sum_{(i,j) \in E} \sum_{k \in D_{ij}} h_{rk} = L\langle W_0 \rangle \langle h_r \rangle \left(1 + \frac{\sigma^2(h, h_r)}{\langle h \rangle \langle h_r \rangle} \right) \quad (27).$$

The second term in the parenthesis on the right of Eq. 27 represents a perturbation in the demand-weighted global expectation of the backup hop lengths due to the correlation of the hop lengths of the primary and backup paths. We find that for typical networks this perturbation represents a small change in the average backup path length (only 1 out of the 19 networks we simulated showed a perturbation greater than 10% in the global expectation value). If the correlation between the primary and backup hop lengths in Eq. 27 is small, the demand weighted global expectation of the backup hop lengths can be expressed as

$$\langle h_r \rangle \approx \frac{1}{L\langle W_0 \rangle} \sum_{(i,j) \in E} \sum_{k \in D_{ij}} h_{rk} \quad (28).$$

Expressing the global expectation of the reconfiguration buffer, $\langle b \rangle$, as

$$\langle b \rangle = \frac{\sum_{(i,j) \in E} \sum_{k \in D_{ij}} \sum_{n=0}^{h_{rk}} b_{kn}}{L\langle W_0 \rangle (\langle h_r \rangle + 1)} \quad (29),$$

from Eq. 24 the global expectation of the reconfiguration interval, $\langle T_{RC} \rangle$, can be expressed as

$$\langle T_{RC} \rangle = (\langle h_r \rangle + 1) \langle T_P \rangle + \langle b \rangle T_{SC} + \langle h_r \rangle T_L \quad (30).$$

The processing interval, T_P (10's of μs), is typically much smaller than the configuration interval, T_{SC} (10's of ms), and therefore the average processing buffer length is small (~ 1). Thus the average time interval of the reconfiguration interval for parallel XC architectures may be expressed as a special case of Eq. 30 where the average reconfiguration buffer length is $\langle b \rangle = 1$. The timelines for eSBR are similar with the exception that the Req message is forwarded from one XC to the next immediately after the request processing stage. Typically the reconfiguration interval for a request at a XC is longer than the flight time of the Req message along the backup path. As the link transit and reconfiguration operations occur in parallel in eSBR, the average reconfiguration interval for eSBR for serial XC architectures does not include a link transit interval component and therefore may be expressed as

$$\langle T_{RC} \rangle = (\langle h_r \rangle + 1) \langle T_P \rangle + \langle b \rangle T_{SC} \quad (31).$$

From the description of the eSBR protocol, we observe that a particular Req message will be present in many, if not all, of the XC buffers on its backup path. The reconfiguration interval for this request is the length of the longest of the reconfiguration buffers at the XCs on the backup path after the arrival of the Req message multiplied by T_{SC} . The average number of failed demands when a link failure occurs is $\langle W_0 \rangle$, and each demand requires $\langle h_r \rangle + 1$ reconfiguration operations, which are performed by a set of nodes that participate in the restoration of demands after a link failure. Thus the average number of reconfiguration operations performed by a node, O_n , may be expressed as

$$O_n = \frac{\langle W_0 \rangle (\langle h_r \rangle + 1)}{N_R} = \frac{D\langle h \rangle (\langle h_r \rangle + 1)}{N_R L} \quad (32)$$

where N_R is the average number of nodes that participate in the restoration process. For a unit uniform demand profile, $D = N(N-1)$ in Eq. 32. To estimate the average number of nodes participating in the restoration process, consider a demand that has failed during a link failure event. The network can be viewed as a tree with the source node of the failed demand as the root. There are δ outgoing branches of the tree attached to the root, two of which are used to route the working and backup paths of a demand, thus the nodes that are part of the tree accessible via these 2 branches are likely candidates for routing. Therefore the average number of nodes participating in the restoration process is approximately $2N/\delta$. Substituting for D and N_R in Eq. 32, we can express the average number of reconfiguration operations as

$$O_n = \frac{N(N-1)}{(2N/\langle\delta\rangle)L} \langle h \rangle (\langle h_r \rangle + 1) \approx \frac{1}{2} \langle h \rangle (\langle h_r \rangle + 1) \quad (33).$$

Neglecting the message processing interval, the form of Eq. 31 expresses the average length of the longest reconfiguration buffer as $(\langle h_r \rangle + 1) \langle b \rangle$, and comparing with Eq. 33, we find that the average length of the reconfiguration buffer is $\langle b \rangle \sim \langle h \rangle / 2$. It is, however, common methodology to consider a link failure event as the failure of both directed links that form a bidirectional link between neighboring nodes. In such a scenario, the average reconfiguration buffer length is $\langle b \rangle \sim \langle h \rangle$.

In the discussion above we have estimated the average number of nodes participating in the restoration process by considering the network as a tree rooted at a source node. For all the nodes in the network to participate in the restoration process of a demand, the demand must be restored via all $\langle \delta \rangle - 1$ surviving branches of the tree. This construction entails each demand or demand set between node pairs being split into $\langle \delta \rangle - 1$ flows. This situation will arise if the demands between a node pair can be routed independently of each other. It can also occur implicitly if the routing is modeled by a linear programming (LP) tool, which may route each demand via multiple flows by treating the capacity requirement of each demand as a real variable. In the latter case the flows construction provides an approximation for the integer demand routing scenario which is a much harder problem to solve. This flows construction ensures that all nodes are candidates for restoration routing, i.e, $N_R \sim N$. However the interaction of the flows with each other affects the average length of the reconfiguration buffers and we incorporate the effect of flows on the reconfiguration interval below.

4.B. Demand flows construction

Although not all nodes in a network may be utilized to reroute the backup paths after a link failure, and estimating the number of participant nodes is difficult, we analyze the situation by visualizing the rerouting of demands via physically diverse ‘flows’. Node pairs in a network may exchange multiple pairs of demands between themselves, which may be subdivided into different network flows. Thus a node with a failed demand set may subdivide this demand set into $\delta - 1$ flows for restoration (excluding the link on the failed path due to disjointness of the primary and backup paths) each requiring $1/(\delta - 1)$ of the reconfiguration resources of the demand set and therefore contributing $1/(\delta - 1)$ to each buffer length. The large number of flows generated now makes the participation of all nodes in the network plausible. Substituting $N_R \sim N$, and $D = N(N-1)$ in Eq. 32, yields the average number of reconfiguration operations performed by a node upon a link failure $O_n \sim \langle h \rangle (\langle h_r \rangle + 1) / \langle \delta \rangle$. When compared to Eq. 31 as before, this indicates that the average buffer length, $\langle b \rangle \sim \langle h \rangle / \langle \delta \rangle$. However, although this prescription of flows will not affect the total length of buffers, its effect on the global expectation of the buffer length as defined by Eq. 29 is to introduce a factor of $\langle \delta \rangle / 2$ in the calculation of the average buffer length. To illustrate this, consider a limiting scenario wherein one Req message arrives at a XC. Thus the total and average buffer length, $\langle b \rangle = 1$. However if this XC is part of a network of mean degree $\langle \delta \rangle = 3$, and we utilize the flows prescription then each demand is split into two flows and the XC will have two Req messages in its buffer (possibly from flows of different demands). However each Req message contributes $1/(\langle \delta \rangle - 1) = 0.5$ to the buffer length. The total buffer length is still 1, however from Eq. 29 the average buffer length is now 1.5 due to the fact that the Req message from the first flow will require 0.5 time units (i.e. T_{SC}) to be serviced and the second one will require 1. We see therefore that the average buffer length has now increased by a factor of $\langle \delta \rangle / 2 = 1.5$. Therefore the average buffer length is now $\langle b \rangle \sim \langle h \rangle / \langle \delta \rangle \times \langle \delta \rangle / 2 = \langle h \rangle / 2$, the same result as Eq. 33, indicating the equivalence of the flows construction and the tree representation of the network. The relationship between the average buffer length and the average hops is intuitive in that the average numbers

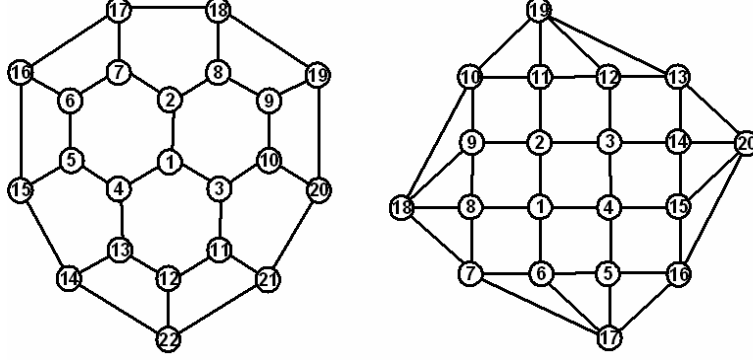


Fig. 4. Example uniform planar networks of constant degree 3 and 4.

of hops captures not only the structure of the demand profile but also the interaction of the demand profile and the physical topology.

The flows construction differs from the tree representation in that it also introduces interactions of flows of the same demand with each other. The Req messages of flows of the same demand are present in the reconfiguration buffer of the source and destination nodes of the demand and possibly other nodes as well because their paths are only required to be link disjoint. For SBR the end of reconfiguration operations of all these flows at the destination node is equal to the reconfiguration interval. If one of these flows is chosen to represent the demand then the reconfiguration buffer length experienced by the Req message of the last flow of this demand to arrive at the destination node has increased by $\delta-2$ (one flow to represent the demand and $\delta-2$ extra flows) if all the Req messages for the $\delta-2$ flows arrive before the XC has begun reconfiguration operations for any of the flows. Incorporating all the effects of interactions of flows on the average buffer length, $\langle b \rangle$, and substituting into Eq. 30 we find that the average reconfiguration interval for the scenario when the link failure is modeled as the failure of both directed links for SBR with serial XC architecture can be expressed as

$$\langle T_{RC} \rangle = \langle h_r \rangle + 1 \left(T_p + \langle h \rangle \left(1 + \frac{\langle \delta \rangle - 2}{\langle h \rangle (\langle h_r \rangle + 1)} \right) T_{SC} \right) + \langle h_r \rangle T_L \quad (34)$$

and from Eq. 31, the average reconfiguration interval for eSBR with serial XC architecture can be expressed as

$$\langle T_{RC} \rangle = \langle h_r \rangle + 1 \left(T_p + \langle h \rangle \left(1 + \frac{\langle \delta \rangle - 2}{\langle h \rangle (\langle h_r \rangle + 1)} \right) T_{SC} \right) \quad (35).$$

5. Simulation analysis and results

To test the analytic model presented above, we emulated the restoration process for a large number of planar mesh networks of varying size and nodal degree and measured restoration performance parameters on a software platform. First we pre-provisioned working and restoration capacity in the networks employing shared path disjoint restoration scheme using a LP tool, SPIDER [15]. The tool sought to minimize total capacity (working + spare) with the cost function of all links equal to one and no hop limits on the primary and backup paths. We simulated multiple demands between node pairs with the LP tool routing each demand independently. The routing information was then used by a network and network element emulation program to simulate network link failures and perform restoration routing using SBR or eSBR restoration signaling between nodes. Each node is connected to a XC which could be parallel or serial architecture. An emulation experiment involves choosing the network parameters, i.e. signaling scheme, XC architecture, link length, and the measured network performance parameter i.e. TTTR or $\langle T_{RC} \rangle$. Each node in the network is emulated as a separate thread and messages are passed between threads using

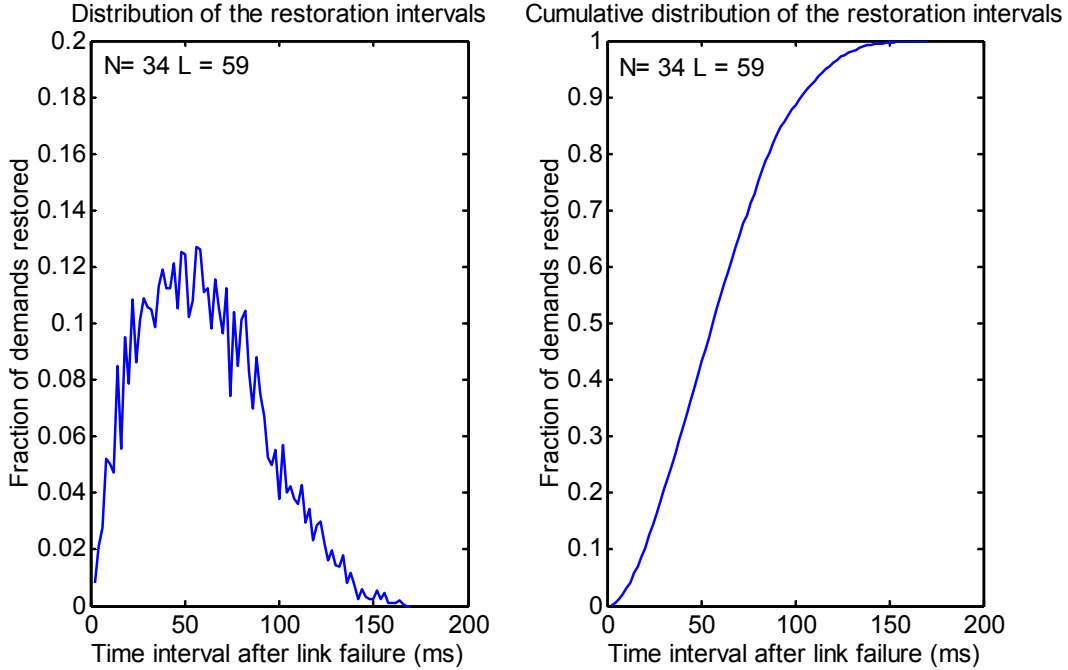


Fig.5 Distribution of the restoration intervals of demands of a network with $N = 34$, $L = 59$ (Network 3).

Windows PostThreadMessage function. The computer's system clock is used to time stamp messages and to emulate time lapse for the network. Emulated intervals, such as processing interval, link transit interval etc. are at least two orders of magnitude larger than the computer platform's processing time, and consequently any time lapse unrelated to the network emulation is not reflected in the restoration intervals recorded.

The emulated network set consists of 11 regular planar graphs of nodal degree 3 and 4 with uniform demand profiles, 6 non-regular graphs with uniform demand profiles and 2 non-regular graphs with non-uniform demand profiles. Two of the networks from among the 11 regular networks we simulated are shown in Fig. 4. These allow us to analyze the dependence of TTTR and $\langle T_{RC} \rangle$ on the size and nodal degree of the network. One of the regular networks has 34 nodes and is of degree 3. To further study the variation of TTTR and $\langle T_{RC} \rangle$ with average nodal degree we ran emulations of 3 irregular networks, labeled Network 1, 2 and 3, with 34 nodes and average nodal degrees 2.7, 2.8 and 3.5 respectively and uniform demand profiles. Network 4 consists of 36 nodes of average degree 2.6 and Network 5 has 27 nodes of average degree 5.1. To illustrate the results of the emulation experiments, the distribution of the restoration intervals of the demands of Network 3 after all single link failures is shown in Fig. 5.

In addition to simulating the restoration performance of networks with uniform demand profiles, we also considered a non-uniform demand profile imposed on the non-regular topology of 32 nodes and 51 links described in [8], which we denote here as Network 6. The demand profile we used was based on a gravity attraction model where the measure of attraction is the degree of a node. The ratio of the maximum to minimum number of demands per node pair was 2.6 for the demands between the 496 node pairs in this network scenario. Networks 7 and 8 consist of a metropolitan area model with 15 nodes and 28 links and a uniform demand profile and a sparse demand profile (over 1/3 of the entries are 0) between the 105 node pairs [16] respectively. For Network 8 the minimum number of demands between node pairs is 1 while the maximum is 22 over the entire range of node pairs that generate demands. We note that recently DeMaesschalck, et al, have modeled the anticipated traffic demand for intelligent optical networks and their predicted distribution of the traffic demand among points-of-presence (PoPs) for 2006 exhibits a similarly large variation with approximately 99.8% of the inter-terminal demands falling within a ratio of 21:1 [17].

We begin by considering the eSBR signaling protocol for networks with serial XC architectures and short link lengths ($\approx 100\text{Km}$). The results of our TTTR network emulations and model analysis are

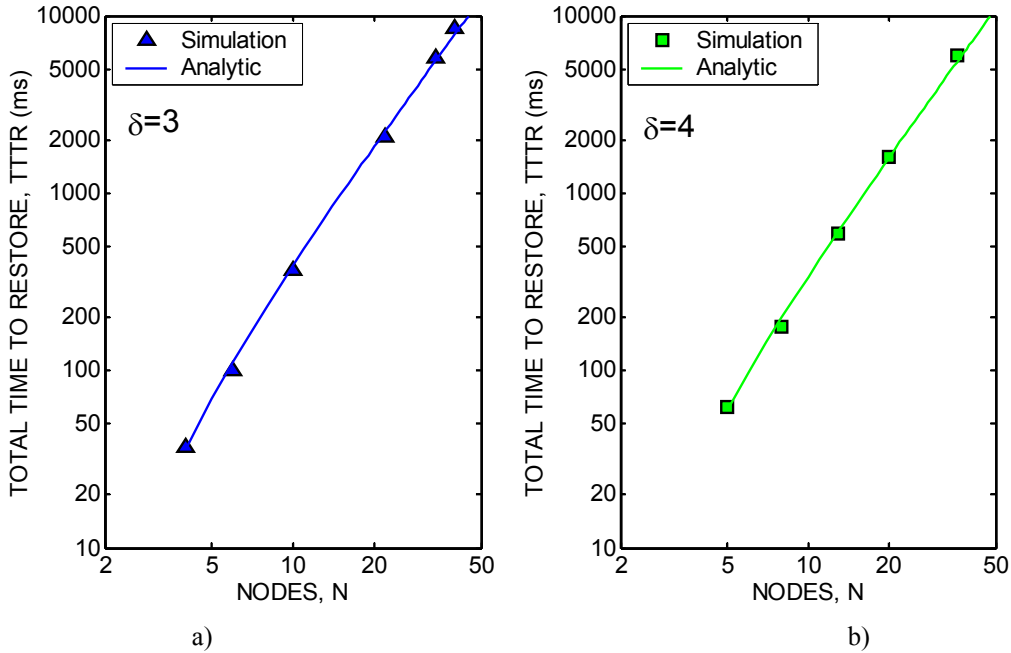


Fig.6 eSBR TTTR for regular topologies of degree a) 3 and b) 4, link length = 100 Km and uniform demand profiles.

Table 1. Network parameters of mixed degree networks

Network	N	L	TTTR (Simulation)	TTTR (Eq. 8)
1	34	46	6761	6049
2	34	48	6514	5817
3	34	59	5708	5421
4	36	47	8361	7073
5	27	69	2970	2744
6	32	51	6610	5487
7	15	28	1178	1004
8	15	28	2016	1963

shown on a log-log plot of the TTTR versus the number of nodes in Fig. 6 a) and b) for regular networks of degree 3 and 4 and in Table 1 for networks of mixed degree. The TTTR is calculated using Eq. 8 where T_{SC} and T_P are chosen to be 3ms and 10 μ s respectively. The average hop length, $\langle h \rangle$, of the working paths may be approximated as in [14]; however we use empirical values in the analytic model calculations as they can be computed quickly. While the average length of the backup paths, $\langle h_r \rangle$, can also be obtained from the LP routing tool; the detailed optimization may take between a few minutes to several hours for large networks. To reduce computation time when using the analytic model, $\langle h_r \rangle$ has been approximated semi empirically as being equal to twice $\langle h \rangle$, [10], an observation also made for non-uniform nominally planar mesh networks with average graph degree greater than approximately 3 and employing dynamic restoration heuristics to achieve optimal capacity assignment [18]. A more accurate approximation of $\langle h_r \rangle$ can be arrived at by observing that the hop lengths over all demands in a planar graph scales approximately as the square root of the size of the graph. We also note that for the smallest size regular network of a given degree, eg. $N = 4$ for $\delta = 3$ and $N = 5$ for $\delta = 4$, $h = 1$ and $h_r = 2$ for every demand. We consider therefore that the expectation value of the backup path hop length may be expressed semi-empirically by the relation.

$$\langle h_r \rangle \cong 2 \left(\frac{N-1}{\langle \delta \rangle} \right)^{\frac{1}{2}} \quad (36)$$

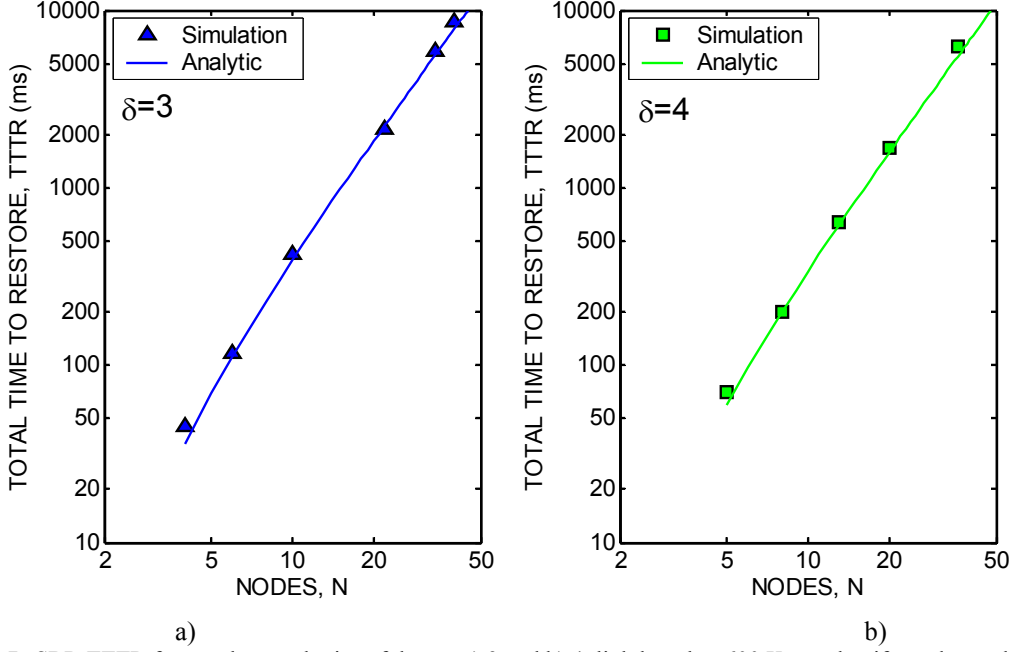


Fig.7 eSBR TTTR for regular topologies of degree a) 3 and b) 4, link length = 600 Km and uniform demand profiles.

Table 2. Network parameters of mixed degree networks

Network	N	L	TTTR (Simulation)	TTTR (Eq. 8)
1	34	46	6884	6049
2	34	48	6659	5817
3	34	59	6059	5421
4	36	47	8465	7073
5	27	69	3214	2744
6	32	51	6846	5487
7	15	28	1232	1004
8	15	28	2334	1963

which has a RMS error of 4.3% for the 19 networks in our network test set. Thus the mean number hops in both the primary and backup paths are proportional to the square root of the number of nodes in nominally planar networks having a mean degree greater than or equal to approximately 3. Note, however, that as the graph degree falls below 3 the dependence of the mean path length on the number of nodes becomes increasingly linear, and in the limit of $\delta = 2$, i.e. ring network, the average primary path and backup path lengths are approximated by $N/2$ for a unidirectional ring, and by $N/4$ and $3N/4$, respectively, for a bidirectional ring. The RMS error between the analytic approximation and network emulations for TTTR for the 19 data points is 9%. Next we increase the link lengths to 600 Km and the results of these simulations are shown in Fig. 7 and Table 2 along with the analytic values for comparison. The RMS error between the simulation data and the analytic model is 13%. The small increase in RMS error despite a 6 fold increase in link lengths indicates that the link propagation delay has a negligible effect on the TTTR for eSBR signaling schemes and serial XC architectures.

The average reconfiguration interval, $\langle T_{RC} \rangle$, for eSBR with serial architecture is calculated using Eq. 35 and the results of the simulation experiment for networks with link lengths of 100Km are shown on a log-log plot of the $\langle T_{RC} \rangle$ versus the number of nodes in Fig. 8 a) and b) for regular networks of degree 3 and 4 and in Table 3 for networks of mixed degree. The RMS error for the 19 data points between the simulations and the analytic approximation is 8%. The results with link lengths of 600Km are shown in Fig. 9 and Table 4. The RMS error for the 19 data points between the simulations and the analytic

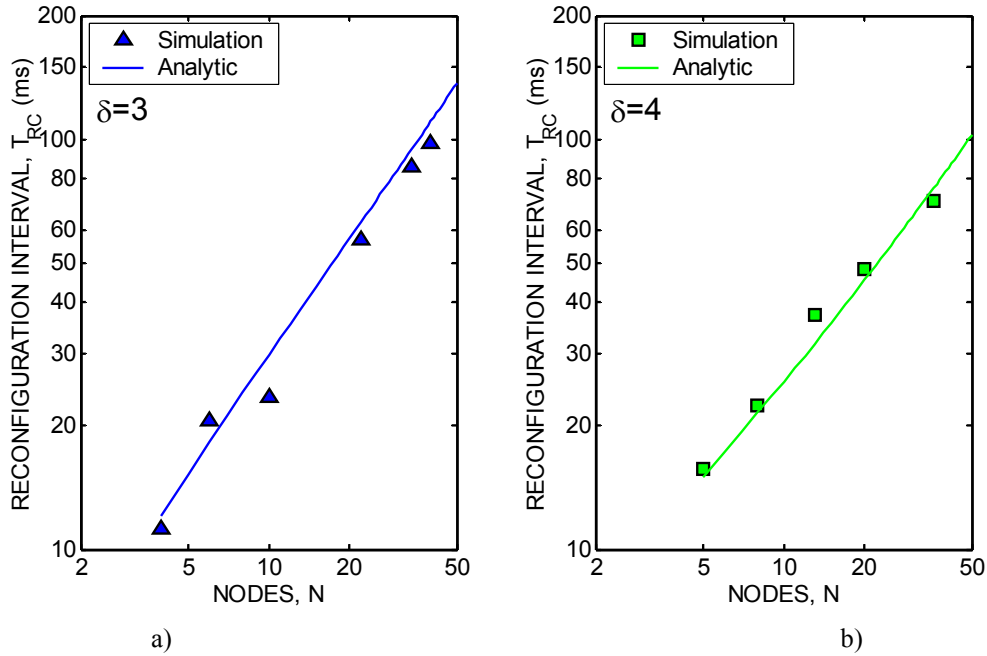


Fig.8 eSBR reconfiguration interval for regular topologies of degree a) 3 and b) 4, link length = 100 Km and uniform demand profiles.

Table 3. Network parameters of mixed degree networks

Network	N	L	$\langle T_{RC} \rangle$ (Simulation)	$\langle T_{RC} \rangle$ (Eq. 35)
1	34	46	97	97
2	34	48	92	94
3	34	59	78	80
4	36	47	110	107
5	27	69	52	52
6	32	51	84	82
7	15	28	40	37
8	15	28	27	30

approximation is 11%. We alert the reader that based on RMS error; the analytic model for $\langle T_{RC} \rangle$ is more accurate than that for TTTR even though Figs. 6 and 7 would appear to suggest otherwise. This is because the Y axis logarithmic time scales for TTTR are two orders of magnitude greater than for $\langle T_{RC} \rangle$ and consequently the visual separation between the analytic model and the simulation data in the graphs of TTTR is reduced relative to the separation in the graphs of $\langle T_{RC} \rangle$.

The analytic results of the TTTR and $\langle T_{RC} \rangle$ analysis for SBR signaling in serial XC architectures are also in good agreement with simulation data; however we do not show these in this presentation for conciseness. The TTTR for parallel XC architecture networks is expressed by Eqs. 13 and 14 but these are not in closed form and require routing data in order to be evaluated. Utilizing the routing data, the results of Eqs. 13 and 14 are in good agreement with simulation data for the TTTR for networks with parallel XC architectures which confirms the correct implementation of the signaling schemes. In order to arrive at a compact expression for the TTTR for parallel XC architecture networks we must approximate the longest backup path length after a link failure and this is a topic of future research. The $\langle T_{RC} \rangle$ for parallel XC architecture networks can be evaluated using Eqs. 30 and 31 for SBR and eSBR respectively where the average buffer length, $\langle b \rangle = 1$. The difference between the analytic model and the simulation therefore reflects the uncertainty in estimating the average backup path hop length, $\langle h_r \rangle$. The RMS error between the analytic model and the simulation data is $\sim 4\%$ when using experimental values of $\langle h_r \rangle$.

The reader will note that the range of possible topologies and demand profiles is very large and it

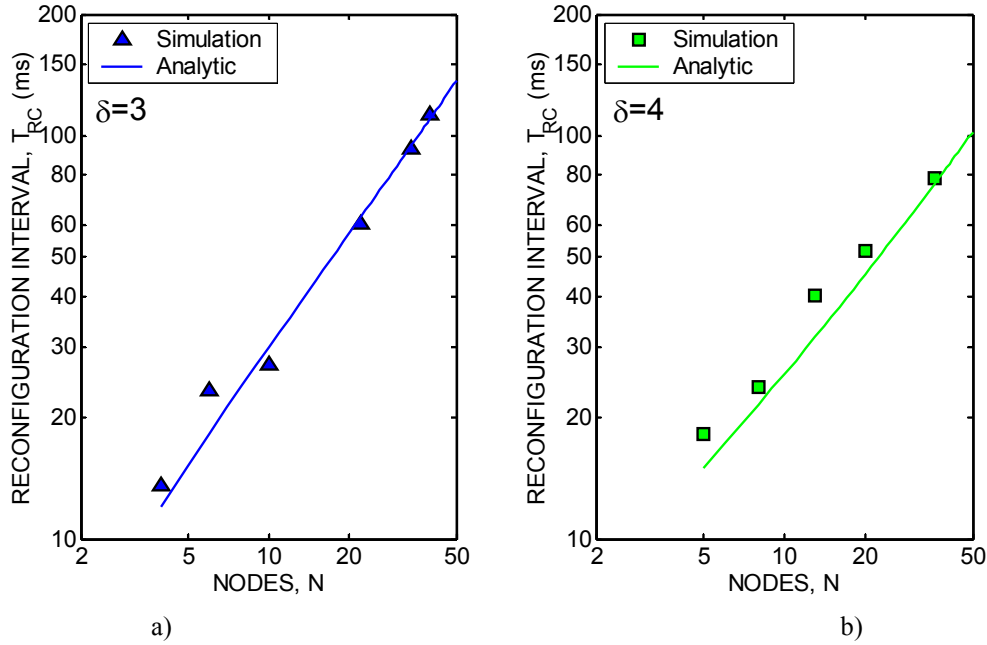


Fig.9 eSBR reconfiguration interval for regular topologies of degree a) 3 and b) 4, link length = 600 Km and uniform demand profiles.

Table 4. Network parameters of mixed degree networks

Network	N	L	$\langle T_{RC} \rangle$ (Simulation)	$\langle T_{RC} \rangle$ (Eq. 35)
1	34	46	106	97
2	34	48	99	94
3	34	59	84	80
4	36	47	121	107
5	27	69	57	52
6	32	51	90	82
7	15	28	43	37
8	15	28	30	30

is always possible that the analytic model may not accurately describe particular scenarios. We have included in our simulation set networks of practical size and typical values of network parameters, and an RMS error of $\sim 10\%$ over the 19 network scenarios indicates the usefulness of the model for quick estimation of network restoration performance. We should caution however that the analytic model assumes that the nodes in a network, in particular the node adjacent to the link failure, are never idle. These idle times are small relative to the entire restoration interval for networks of practical interest, however may become relevant for networks with large link lengths when a node having finished all reconfiguration operations in its queue is waiting for a Req message to arrive which is delayed due to long flight times. We believe this contributes to the increased RMS error between the analytic model and the simulation data for networks with link lengths of 600Km as opposed to link lengths of 100Km. Additionally we have neglected the contribution of the correlation between the primary and backup path lengths as we have found it to be small. However network scenarios with significant correlation values can exist which would adversely impact the accuracy of the approximation of the backup path length and consequently the analytic model of the restoration time. Moreover future XC technologies may reduce the reconfiguration interval, T_{SC} , further and the flight time rather than the reconfiguration interval may dominate the restoration interval. This might increase idle times and reduce the accuracy of the model. While we have limited our treatment to serial and parallel XC architectures, batch processing XC architectures are increasingly gaining prominence. We believe that the model can be modified to include batch processing and this is a topic for future research.

Table 5. Key expressions of the analytic restoration time model.

XC architecture	Signaling scheme	Relationship	Equation
Serial	SBR	$TTTR = \sum_{(i,j) \in E} O_{ij} T_{SC} + \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle} \right) \frac{(\langle h_r \rangle + 1)}{N} L T_{SC} + \frac{3\langle h_r \rangle}{2} L T_L + \langle h_r \rangle L T_P$	(12)
Serial	eSBR	$TTTR = \sum_{(i,j) \in E} O_{ij} T_{SC} + \langle W_0 \rangle \left(1 - \frac{1}{\langle h \rangle} \right) \frac{(\langle h_r \rangle + 1)}{N} L T_{SC}$	(8)
Parallel	SBR	$TTTR = \sum_{(i,j) \in E} \max_{k \in D_{ij}} \left(T_D + \frac{h_k}{2} (T_P + T_L) + 2h_{rk} (T_P + T_L) + (h_{rk} + 1) T_{SC} \right)$	(14)
Parallel	eSBR	$TTTR = \sum_{(i,j) \in E} \max_{k \in D_{ij}} \left(T_D + \frac{h_k}{2} (T_P + T_L) + h_{rk} (T_P + T_L) + T_{SC} \right)$	(13)
Serial	SBR	$\langle T_{RC} \rangle = (\langle h_r \rangle + 1) \left(T_P + \langle h \rangle \left(1 + \frac{\langle \delta \rangle - 2}{\langle h \rangle (\langle h_r \rangle + 1)} \right) T_{SC} \right) + \langle h_r \rangle T_L$	(34)
Serial	eSBR	$\langle T_{RC} \rangle = (\langle h_r \rangle + 1) \left(T_P + \langle h \rangle \left(1 + \frac{\langle \delta \rangle - 2}{\langle h \rangle (\langle h_r \rangle + 1)} \right) T_{SC} \right)$	(35)

6. Conclusion

In this paper we formulate an analytic model for the total restoration time of path based restoration schemes in general mesh networks and relate it to restoration capacity hardware requirements. We also present an analytic model of the average restoration time for a demand and find that it is closely related to network parameters and in particular the average primary and backup path lengths. For the convenience of the reader key results of the model are summarized in Table 5. We have simulated a large number of capacity optimal networks with different demand profiles, link lengths, XC architectures, and signaling schemes and found our analytic model to be in close agreement with simulation data. This work combines the hardware requirements and operational performance of path based restoration strategies in a consistent analytical framework allowing quantification of the tradeoffs involved.

References

1. R. K. Sinha, F. Yu, and R. Doverspike, "Performance study of mesh restoration," Proc. Conf. Optical Fiber Commun., Anaheim, CA (2005) paper OTuK2.
2. M. Goyal, G. Li, and J. Yates, "Shared mesh restoration: A simulation study," Proc. Conf. Optical Fiber Commun., Anaheim, CA, 2002, paper ThO2.
3. A. A. Akyamac, J. F. Labourdette, K. Hua, Z. Bogdanowicz, S. Chaudhuri, and S. French, "Optical mesh network modeling: simulation and analysis of restoration performance," Proc. National Fiber Optic Engineers Conf., Dallas, TX, 2002, 1408-1418.
4. S. Koo, and S. Subramaniam, "Trade-offs between speed, capacity and restorability in optical mesh network restoration," Proc. Conf. Optical Fiber Commun., Anaheim, CA, 2002.
5. S. Baroni, P. Bayvel, R. J. Gibbens, and S. K. Korotky, "Analysis and design of resilient multifiber wavelength-routed optical transport networks," J. Lightwave Technol., vol. 17, no. 5, pp. 743-758, 1999.
6. B.V. Caenegem, N. Wauters, and P. Demeester, "Spare capacity assignment for different restoration strategies in mesh survivable networks," in Proc. Int. Conf. Commun., vol. 1, pp. 288-292, 1997.
7. S. Ramamurthy, L. Sahasrabudde, and B. Mukherjee, "Survivable WDM mesh networks," J. Lightwave Technol., vol. 21, no. 4, pp. 870-883, 2003.
8. J. Doucette, and W.D. Grover, "Comparison of mesh protection and restoration schemes and the dependency on graph connectivity," Proc. Design Reliable Commun. Networks, Budapest, pp. 121-128, 2001.
9. J.F. Labourdette, E. Bouillet, R. Ramamurthy, and A.A. Akyamac, "Fast approximate dimensioning and performance analysis of mesh optical networks," in Proc. Design Reliable Commun. Networks, Banff, Alberta, Canada, pp. 428-439, 2003.

10. M. Bhardwaj, L. McCaughan, S. K. Korotky, and I. Saniee, "Analytical description of shared restoration capacity for mesh networks," *Journal of Optical Networking*, **4**, 130-141 (2005).
11. T. Y. Chow, F. Chudak, and A. M. Ffrench, "Fast optical layer mesh protection using pre-cross-connected trails," in *IEEE/ACM Trans. Networking*, vol. 12, no. 3, pp. 539-548, 2004.
12. C. Assi, Y. Ye, S. Dixit, and M. Ali, "Control and management protocols for survivable optical mesh networks," *J. Lightwave Technol.*, **21**, 2638-2651 (2003).
13. M. Bhardwaj, L. McCaughan, A. Olkhovets, and S. K. Korotky, "Simulation and modeling of the mean time to restore for optical mesh networks," Proc. Conf. Optical Fiber Commun., Anaheim, CA (2005) paper OTuK3.
14. S. K. Korotky, "Network Global Expectation Model: A statistical formalism for quickly quantifying network needs and costs," *J. Lightwave Technol.*, **22**, 703-722 (2004).
15. R.D. Davis, K. Kumaran, G. Liu, and I. Saniee, "SPIDER: A simple and flexible tool for design and provisioning of protected lightpaths in optical networks," *Bell Labs Tech. J.*, **6**, 82-97 (2001).
16. "The role of digital crossconnect systems in transport network survivability," Bellcore, Special Rep. SR-NWT-002514, Issue 1, January, 1993.
17. S. De Maesschalck, L. Nederlof, M. Vaughn, and R. E. Wagner, "Traffic studies for fast optical switching in an intelligent optical network," *Photonic Network Commun.*, vol. 3, pp. 285-307, 2004.
18. C. Ou, J. Zhang, H. Zang, L.H. Sahasrabudhe, and B. Mukherjee, "New and improved approaches for shared-path protection in WDM mesh networks," *J. Lightwave Technol.*, vol. 22, pp 1223-1232, 2004.