A load-balancing spare capacity reallocation approach in service-rich SONET metro mesh networks

Lu Shen
University of Nebraska - Lincoln, lshen@cse.unl.edu

Xi Yang
University of Nebraska - Lincoln, xyang@cse.unl.edu

Byrav Ramamurthy
University of Nebraska - Lincoln, bramamurthy2@unl.edu

Follow this and additional works at: http://digitalcommons.unl.edu/cseconfwork

Part of the Computer Sciences Commons

http://digitalcommons.unl.edu/cseconfwork/79

This Article is brought to you for free and open access by the Computer Science and Engineering, Department of at DigitalCommons@University of Nebraska - Lincoln. It has been accepted for inclusion in CSE Conference and Workshop Papers by an authorized administrator of DigitalCommons@University of Nebraska - Lincoln.
A load-balancing spare capacity reallocation approach in service-rich SONET
metro mesh networks†

Lu Shen, Xi Yang and Byrav Ramamurthy
Department of Computer Science and Engineering
University of Nebraska – Lincoln
Lincoln NE 68588-0115
Email: {lshen, xyang, byrav}@cse.unl.edu

Abstract

The next-generation SONET metro network is evolving into a service-rich infrastructure. At the edge of such a network, multi-service provisioning platforms (MSPPs) provide efficient data mapping enabled by Generic Framing Procedure (GFP) and Virtual Concatenation (VC). The core of the network tends to be a meshed architecture equipped with Multi-Service Switches (MSSs). In the context of these emerging technologies, we propose a load-balancing spare capacity reallocation approach to improve network utilization in the next-generation SONET metro networks. Using our approach, carriers can postpone network upgrades, resulting in increased revenue with reduced capital expenditures (CAPEX). For the first time, we consider the spare capacity reallocation problem from a capacity upgrade and network planning perspective. Our approach can operate in the context of shared-path protection (with backup multiplexing) because it reallocates spare capacity without disrupting working services. Unlike previous spare capacity reallocation approaches which aim at minimizing total spare capacity, our load-balancing approach minimizes the network load vector (NLV), which is a novel metric that reflects the network load distribution. Because NLV takes into consideration both uniform and non-uniform link capacity distribution, our approach can benefit both uniform and non-uniform networks. We develop a greedy load-balancing spare capacity reallocation (GLB-SCR) heuristic algorithm to implement this approach. Our experimental results show that GLB-SCR outperforms a previously proposed algorithm (SSR) in terms of established connection capacity and total network capacity in both uniform and non-uniform networks.

1. Introduction

The next-generation SONET metro network will serve as a service-rich infrastructure that responds quickly and efficiently to emerging data services [1][2][3]. In such a network, multi-service provisioning platforms (MSPPs) are deployed at the edge where efficient data mapping is enabled by Generic Framing Procedure (GFP) and Virtual Concatenation (VC) technologies [1][2][3][4]. Virtual concatenation, which is standardized and widely accepted in industry [2][3][5][6], is an inverse-multiplexing technology that groups multiple base-rate SONET circuits (e.g., VT-1.5 or STS-1) and eliminates the continuous constraints on conventional SONET multiplexing hierarchy. Equipped with multi-service switches (MSSs) [1], the metro core network is evolving from ring to mesh topologies that provide enhanced routing flexibility and resource utilization. However, due to high traffic uncertainty in metro networks, leveraging new technologies to achieve efficient network resource utilization remains a challenge.

Provisioning in real-world networks is usually implemented without the knowledge of how network resources will be used in the future. As connections are added and deleted from the network, the network resource utilization becomes sub-optimal [7][8][9]. This problem becomes more severe in the next-generation metro networks, where traffic is data-dominant and tends to be more unpredictable. Bandwidth reallocation, a method of reconfiguring the existing connections, is an attractive solution to fill in the gap between current and optimal of network resource utilization. [8][9].

As the metro network becomes a bottleneck in the entire network, carriers feel pressed to provide more network capacity. When bandwidth demands exceed network capacity, carriers need to upgrade their network by deploying extra equipment such as fibers, wavelengths, transceivers and switching fabrics. Due to uncertainty of future traffic volume and equipment costs, carriers need a flexible, incremental capacity upgrade solution to avoid over-investment on equipment [10][11]. Taking capacity planning into account, we propose a load-balancing spare capacity reallocation approach to postpone network

† This work was supported in part by NSF grants (ANI-0074121 and EPS-0091900).
upgrades, resulting in substantial cost savings. Because our approach provides a flexible and efficient capacity upgrade solution to handle traffic uncertainty, carriers may incorporate it into their network planning tools to maintain robustness against unpredictability, which is an important requirement for network planning tools in next-generation multi-service networks [11]. To the best of our knowledge, our work, for the first time, studies the bandwidth reallocation problem from a capacity planning and network upgrade perspective. Our proposed spare capacity reallocation approach only changes protection paths, and hence does not interfere with the ongoing services on the working paths and is therefore risk-free. Our approach is suitable for the networks with shared-path protection, which has proven to be cost-efficient in the next generation metro mesh networks [12][13].

Unlike previous spare capacity reallocation approaches whose objective is minimizing the total spare capacity, our load-balancing approach reconfigures the existing protection paths with the objective of minimizing the network load vector (NLV). NLV is a new metric defined in this paper to reflect the network load distribution. By minimizing NLV, the network load distribution is changed to accommodate more future connections before a network upgrade is needed. Network upgrade is usually triggered based on carriers’ policies, e.g., when the number of congested links exceeds a threshold or when a provisioning upgrade is needed. Network upgrade is usually triggered due to budget crunch, resulting in non-uniform distribution of link capacities. (This kind of upgrade is called a non-uniform upgrade in this paper.) The conventional spare capacity reallocation approaches aim at minimizing spare capacity, but cannot deal with the non-uniform distribution of link capacity. In contrast, the definition of NLV takes into consideration the non-uniform distribution of link capacity. Therefore our load-balancing approach can benefit both uniform and non-uniform networks.

Previous studies on the spare capacity reallocation problem include [8][13][15]. The work in [13] proposed a distributed spare capacity optimization approach in WDM networks. The authors in [15] presented a matrix-based spare capacity allocation approach called successive survivable routing (SSR). Both the approaches in [13] and [15] re-provision their protection paths iteratively to minimize total spare capacity. Each iteration in their approaches uses the dynamic protection path selection approach to reroute protection path. State-of-the-art dynamic protection path selection schemes include partial information restoration (PIR) [14], full information restoration (FIR) [18], and distributed partial information management (DPIM) [16][17]. All of these previous approaches focus on minimizing spare capacity. However, this objective does not directly relate to cost savings because minimizing spare capacity does not necessarily maximize the capability of accommodating future connections. Thus, service providers may still need to pay for early network upgrades. Our previous study in [19] proposed a load-balancing shared-protection-path reconfiguration approach for wavelength-routed networks, but did not take into consideration the non-uniform link load distribution.

In this study, we develop a greedy load-balancing spare capacity reallocation (GLB-SCR) heuristic algorithm to minimize NLV. We conduct substantial experiments to simulate the service-rich environment in next-generation SONET mesh networks and compare GLB-SCR with SSR, a conventional approach introduced in [15]. The experimental results show that GLB-SCR can establish more connections under the same amount of total network capacity in uniform networks. In non-uniform networks, GLB-SCR obtains constant delays in network upgrades, resulting in significant cost savings on network capacity investments.

The remainder of this paper is organized as follows. Section 2 presents the network model and gives the problem formulation. Section 3 describes our approach to solve this problem. Section 4 presents the experimental design and numerical results. Section 5 concludes this paper.

2. Network model and problem formulation

2.1 Network model

In this paper, we consider next-generation SONET mesh networks. In such a network, nodes are interconnected by a set of fibers, each having a certain amount of link capacity. Different fibers are allowed to have different link capacities. Each network node is capable of cross-connecting circuits at a base rate (e.g., STS-1), which is called switching granularity. Capacity is represented by an integer with the unit of switching granularity. For example, in a network with an STS-1 switching granularity, link capacity 48 represents a capacity of 48 STS-1, i.e., OC-48. Each link in the network is associated with a working capacity, used for working services, and a spare capacity, used for protection. The sum of the working capacity and the spare capacity of a link is the used capacity of this link. The free capacity on a link can be obtained by subtracting the used capacity from its link capacity. To calculate spare capacity, we maintain two arrays, called FailOther and FailSelf, for each link. FailOther records the amount of
spare capacity needed on the link to restore working services upon failure of other links. FailSelf records the amount of spare capacity needed on other links to restore working services upon failure of this link. These two arrays were originally introduced in [18] for protection path selection. The definition of each member in these arrays will be given in Section 3.1.

A user service on the network is represented by a connection. A connection comprises a source, a destination, and a bandwidth requirement. We assume that every node has virtual concatenation functionality. In other words, the source node is capable of dividing a connection into multiple circuits at the rate of switching granularity, and the destination node is capable of combining these circuits to reconstruct the original connection. These circuits are called virtual concatenation (VC) group members. They can be routed, cross-connected, and protected independently in the network. Each VC group member has shared-path protection against single-link failures. In particular, the spare capacity is assigned such that, under a single-link failure, all the disrupted working circuits that traverse the failed link can be restored. The VC group members of a connection that traverse through the same link capacity (i.e., STS-1-21v). These 21 VC group members are further divided into two sub-connections that traverse two working paths, W1 with 15 STS-1 circuits and W2 with 6 STS-1 circuits, each having a protection path (P1 and P2 respectively). They are combined at node D to reconstruct the original GbE connection. Fig. 1 also shows the spare capacity information. Note that link (C, D) only needs to reserve a spare capacity of 15 STS-1, although all three protection paths traverse through it. This is because the three working paths will not break down simultaneously under any single-link failure.

2.2 Problem formulation

Our problem is formulated as follows: Given a network topology, the capacity information of each link, and a set of existing connections, determine how to reallocate spare capacity with the objective of minimizing the network load vector (NLV).

Network load vector (NLV) is a novel network performance metric introduced in this paper. Let \( G(V,E) \) denote the network topology, where \( V \) is the set of nodes and \( E \) is the set of links. Let \( C_{ab} \) denote the link capacity of the link \((a,b)\), \( C = \max \{C_{ab}\} \) denote the maximum link capacity among all the links, and \( W_{ab} \) and \( S_{ab} \) denote the working capacity and spare capacity on the link \((a,b)\) respectively. Note that \( C_{ab}, W_{ab}, \) and \( S_{ab} \) are integers with the unit of switching granularity. We define the NLV of network \( G \) as a vector: \( \mathbf{L}_G = [l_0, \ldots, l_i, \ldots, l_{C}] \), where \( i \) is an integer, and \( l_i = |\{(a,b) : [C \cdot (W_{ab} + S_{ab})/C_{ab}] = i\}| \). In other words, \( l_i \) denotes the number of links of the same used capacity that satisfies the equation \([C \cdot (W_{ab} + S_{ab})/C_{ab}] = i\), where \((W_{ab} + S_{ab})\) is the used capacity on the link \((a,b)\), \([C \cdot (W_{ab} + S_{ab})/C_{ab}]\) is the normalized used capacity that is scaled to \( C \) levels according to the percentage of the used capacity over the link capacity (i.e., \((W_{ab} + S_{ab})/C_{ab}\)) on the local link. In this paper, link load refers to this normalized capacity. The index \((0 \leq i \leq C)\) of a NLV represents a specific value of the link load. The value of a member in NLV is the number of links whose load equals its index. The definition of NLV applies to both uniform and non-uniform networks. Note that in the uniform networks, \( l_i \) becomes the number of links with used capacity \((W_{ab} + S_{ab})\) because \( C = C_{ab}, \forall (a,b) \in E \).

We define the relationship between two NLVs, \( \mathbf{L}_{G_1} \) and \( \mathbf{L}_{G_2} \), as follows:

- \( \mathbf{L}_{G_1} = \mathbf{L}_{G_2} \) if \( \mathbf{L}_{G_1}[i] = \mathbf{L}_{G_2}[i], \forall i : 0 \leq i \leq C \).
3. Load-balancing spare capacity reallocation approach

We propose a greedy heuristic algorithm to achieve our load-balancing objective (i.e., minimizing NLV) through spare capacity reallocation. The heuristic operates on sub-connections because all the VC group members (circuits) belonging to a sub-connection traverse the same pair of working and protection paths. This is also to satisfy the constraint on the number of diverse paths (i.e., sub-connections) of each connection, which is usually imposed by the implementation of virtual concatenation [5]. By treating each sub-connection as one operational unit, the number of sub-connections (diverse paths) remains the same after reallocation. In this section, we first define the notations to be used. We then introduce the spare capacity allocation and release procedures for operating one sub-connection, which are the basic operations in our heuristic. In the remaining two subsections, we present the design of our heuristic algorithm and discuss some operational considerations for our approach.

3.1 Notations

- \( G(V,E) \) denotes the network topology with a set of nodes \( V \) and a set of unidirectional links \( E \).
- \( N = \vert V \vert \) denotes the number of nodes in the network.
- \( C_{ij} \) is an integer that denotes the link capacity of the link \((i,j) \in E\).
- \( C = \max_{(i,j) \in E} \{ C_{ij} \} \) is an integer that denotes the maximum link capacity among all the link in the network.
- \( W_{ij} \) is an integer that denotes the working capacity on the link \((i,j) \in E\).
- \( S_{ij} \) is an integer that denotes the spare capacity reserved on the link \((i,j) \in E\).
- \( F_{ij} \) is an integer that denotes the free capacity on the link \((i,j) \in E\), where \( F_{ij} = C_{ij} - W_{ij} - S_{ij} \).
- \( \text{FailSelf}_{ij}(x,y) \) is an integer that denotes the spare capacity needed on the link \((x,y)\) in order to protect the working services upon failure of the link \((i,j)\), where \((i,j) \in E\) and \((x,y) \in E\). In other words, \(\text{FailSelf}_{ij}(x,y) = \max_{(x,y) \in E} \{ \text{FailOther}_{ij}(x,y) \} \) is the sum of the capacity of all the sub-connections whose working paths traverse the link \((i,j)\) and protection paths traverse the link \((x,y)\).
- \( \text{FailOther}_{ij}(x,y) \) is an integer that denotes the spare capacity needed on the link \((i,j)\) in order to protect the working services upon failure of the link \((x,y)\), where \((i,j) \in E\) and \((x,y) \in E\). In other words, \(\text{FailOther}_{ij}(x,y) = \max_{(x,y) \in E} \{ \text{FailOther}_{ij}(x,y) \} \) is the sum of the capacity of all the sub-connections whose working paths traverse the link \((x,y)\) and protection paths traverse the link \((i,j)\). Note that \( S_{ij} = \max_{(x,y) \in E} \{ \text{FailOther}_{ij}(x,y) \} \) in order to protect the working services against any single-link failure.

3.2 Spare capacity allocation and release procedures

Spare capacity allocation (SC-Allocation) and release (SC-Release) procedures are the basic operations performed on a sub-connection in our heuristic. We use \( (s,d,b,W,P) \) to present a sub-connection, where \( s \) denotes its source...
node, d denotes its destination node, b denotes its capacity (or the number of VC group members), W denotes its working path, and P denotes its protection path. The spare capacity allocation and release procedures update the link information along the working and protection paths. The information includes spare capacity, free capacity and the members in the arrays FailSelf and FailOther.

\[
\text{SC-Allocation}(G, (s, d, b, W, P))
\]

1) Update the working path
   for each link \((i, j) \in W\), do
      for each link \((x, y) \in P\), do
         \(\text{FailSelf}_{ij}(x, y) \leftarrow \text{FailSelf}_{ij}(x, y) + b\);
   2) Update the protection path
      for each link \((i, j) \in P\), do
         for each link \((x, y) \in W\), do
            \(\text{FailOther}_{ij}(x, y) \leftarrow \text{FailOther}_{ij}(x, y) + b\);
            \(S_{ij} \leftarrow \max \{\text{FailOther}_{ij}(x, y)\};\)

(a) Spare capacity allocation procedure

\[
\text{SC-Release}(G, (s, d, b, W, P))
\]

1) Assign bandwidth on the working path
   for each link \((i, j) \in W\), do
      for each link \((x, y) \in P\), do
         \(\text{FailSelf}_{ij}(x, y) \leftarrow \text{FailSelf}_{ij}(x, y) - b\);
   2) Assign bandwidth on the protection path
      for each link \((i, j) \in P\), do
         for each link \((x, y) \in W\), do
            \(\text{FailOther}_{ij}(x, y) \leftarrow \text{FailOther}_{ij}(x, y) - b\);
            \(S_{ij} \leftarrow \max \{\text{FailOther}_{ij}(x, y)\};\)

(a) Spare capacity release procedure

Fig. 3 SC allocation and release procedures.

Fig. 3 (a) and (b) present the SC-Allocation and SC-Release procedures respectively. The input for both procedures is the network topology along with the link capacity information and a sub-connection. In the SC-Allocation procedure, \(\text{FailSelf}_{ij}(x, y)\) is incremented by the value of \(b\) if the link \((i, j)\) is on the working path and the link \((x, y)\) is on the protection path of this sub-connection. \(\text{FailOther}_{ij}(x, y)\) is incremented by the value of \(b\) if \((i, j)\) is on the protection path and \((x, y)\) is on the working path. Then, the spare capacity of the link \((i, j)\) along the protection path is set to the maximum value of \(\text{FailOther}_{ij}(x, y)\) among all the links \((x, y) \in E\). The SC-Release procedure has steps similar to the SC-Allocation procedure, but it decreases the corresponding members in the arrays by the value of \(b\).

3.3 Greedy load-balancing spare capacity reallocation (GLB-SCR) heuristic

We develop a greedy load-balancing spare capacity reallocation (GLB-SCR) heuristic, which iteratively reallocates all the sub-connections one by one until the NLV of the network cannot be reduced. Fig. 4 presents the procedure of the GLB-SCR heuristic, in which a spare capacity reallocation (SCR) procedure is called iteratively.

Greedy Load-Balancing Spare Capacity Reallocation (GLB-SCR)

1) \(\text{ONLV} \leftarrow \text{NLV of the network} \ G\)
2) \(\text{max}\_load \leftarrow \max \{(W_{ij} + S_{ij}) \cdot C_i / C_j\}) \)  /* find the maximum link load on the protection path */
3) \(\text{SC-Release}(G, s, d, b, W, P), */ in Fig. 3 (b) */
   /\* construct a weighted graph (shown in Fig. 6) */
4) \(G_w \leftarrow \text{least-NLV-weight(max_load, (s, d, b, W, P))}\)
5) \(P_N \leftarrow \text{the shortest path on the weighted graph } G_w\)
6) \(\text{SC-Allocation}(G, (s, d, b, W, P_N))\); /* in Fig. 3 (a) */
7) \(\text{NLV}_N \leftarrow \text{the NLV of the network } G\)
8) if \(\text{NLV}_N < \text{NLV}_O\), then
   a. update the sub-connection by changing its protection path \(P\) to \(P_N\);
   b. return success;
9) if \(\text{NLV}_N \geq \text{NLV}_O\), then
   a. \(\text{SC-Release}(G, (s, d, b, W, P_N))\);
   b. \(\text{SC-Allocation}(G, (s, d, b, W, P))\);
   c. return failure;

Fig. 5 Spare capacity reallocation (SCR) procedure.

SCR is the key procedure in our approach. It tries to reallocate the protection path of an existing sub-connection to reduce the NLV of the network. Its input is the network.
topology with link capacity usage information, and an existing sub-connection. If it successfully reduces NLV, it returns success. Otherwise it returns failure. In the SCR procedure, the spare capacity on the existing protection path of the sub-connection is released by the SC-Release procedure (shown in Fig. 3). Then, without changing its working path, a new protection path (\( P_N \)) is selected by calculating the shortest path on a weighted graph. In the weighted graph, which has the same topology as the input graph \( G \), each link is assigned a weight according to the least-NLV weight function (shown in Fig. 6). Then, the spare capacity of the sub-connection is re-allocated based on its original working path and the new protection path. If the NLV of the network is reduced, the sub-connection is changed by replacing the new protection path with its original protection path and a success is returned. If the NLV is not reduced, the sub-connection is not changed, the protection path of this sub-connection is restored to its original one and a failure is returned. The time complexity of SCR is \( O(N^2) \), where \( N \) is the number of nodes in the network.

Given the network topology, a sub-connection \((s,d,b,W,P)\) and the maximum link load (i.e., \( \text{max}_\text{load} \) calculated in the step 2 of SCR) among all the links on the original protection path of this sub-connection, the least-NLV weight function returns a weighted graph which is used for protection path selection in SCR. In the weighted graph, a weight, denoted by \( \text{weight}(i,j) \), is assigned to each link \((i,j)\) to facilitate selecting a new protection path that tends to reduce NLV. Before giving the definition of the least-NLV weight function, we further define the notation: \( T(i,j) = \max (\{\text{FailSelf}_{xy}(i,j)\} \) is the spare capacity needed on the link \((i,j)\) to protect the working path \( W \). Note that the weighted graph is computed after the spare capacity on the existing protection path is released (i.e., after the step 3 in SCR). Thus, \( T(i,j) \) is the spare capacity needed on the link \((i,j)\) to protect the working path of this sub-connection if the new protection path traverses through the link \((i,j)\). When \( T(i,j) > S_{ij} \), an amount of \((T(i,j) - S_{ij})\) extra spare capacity needs to be reserved on the link \((i,j)\).

The least-NLV weight function is defined in Fig. 6, where \( \text{weight}(i,j) \) is computed according to the following principles: (1) if the link \((i,j)\) is on the working path, its weight is set to infinity to satisfy the link-diversity constraint on the working and protection path pairs; (2) if the link \((i,j)\) is not on the working path, and its spare capacity needs to be increased, but the free capacity is not sufficient to satisfy the incremental spare capacity, its weight is set to infinity because of lack of the spare capacity; (3) if the link is not on the working path, and its spare capacity needs to be increased, and the free capacity is sufficient for the incremental spare capacity, and its future link load is greater or equal to \( \text{max}_\text{load} \), and the link \((i,j)\) is not on the original protection path, its weight is set to infinity. This weight is set to detour the new protection path from the links with maximum link load on the original protection paths, resulting in a reduced NLV; (4-5) for all other links, their weight is set to the power of the number of the nodes in the graph \( N \) to its future link load.

The link weight, which is the power of \( N \) to the link load, leads to a minimum NLV when selecting the shortest path. This is because the maximum number of links on the shortest path is \( N-1 \), resulting in the inequality \( N^1 > (N)^{N-1} \). Although such a weight is exponential to \( N \), the time complexity of computing integer exponentiation is polynomial under the square-and-multiply method. However, this kind of weight is a large number that may overflow any conventional data type in computer languages. In our implementation, we use the GNU Multiple Precision (GMP) arithmetic library to manipulate these large numbers [20].

### 3.4 Operational considerations

In Section 3.3, we have presented a load-balancing spare capacity reallocation heuristic. In this sub-section, we discuss how to implement our spare capacity reallocation approach in real-world networks based on this heuristic. We focus on a centralized implementation, but our approach can be both centralized and distributed. A centralized network operator maintains the network
4. Illustrative Numerical Results

We conduct experiments in a mesh network, which has 24 nodes and 86 unidirectional links as shown in Fig. 7. In the experiments, we compare the GLB-SCR approach with the successive survivable routing (SSR) approach proposed in [15]. SSR also reallocates spare capacity iteratively, but aims at minimizing the total spare capacity of the network. At the beginning of our experiments, each link in the network has an initial capacity (IC). During the experiments, we provision connection requests one-by-one until blocking occurs. Then, the spare capacity reallocation procedure is triggered to operate on all existing connections. After the reallocation, we continue to provision the connection request that was previously blocked. If it is still blocked, we upgrade the link capacity by the amount of upgrade capacity (UC). We consider both cases of uniform and non-uniform networks. In uniform networks, every link is upgraded by the amount of UC capacity. In non-uniform networks, only the links with maximum link load are upgraded by the amount of UC.

We assume the network has an STS-1 switching granularity. To simulate traffic in the real-world metro networks, we provide various connection requests with bandwidth requirements ranging from 45Mbps to 1 Gbps. Table 1 lists different connection requests with their bandwidth requirements, service type, capacity values with the unit of switching granularity (i.e., STS-1), and arrival distribution. The service types and their arrival distributions are close to those in practical networks [1][2][6]. In the experiments, we use the working path selection approach proposed in [5] and use the protection path selection scheme (called FIR) introduced in [18]. We set the number 16 as the limit on the number of sub-connections (diverse-paths) for each connection (our experiments show that the maximum number of sub-connections does not exceed 6 under our provisioning method). We use two metrics for comparison: established connection capacity and total network capacity. Established connection capacity is the sum of the capacities of all the established connections. The total network capacity is sum of the link capacities of all the links in the network.

![Fig. 7 The 24-node and 86-link network.](image)

Table 1 Service types and their distribution

<table>
<thead>
<tr>
<th>Bandwidth Requirement</th>
<th>Service Type</th>
<th># of STS-1 Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>45Mbps</td>
<td>Aggregated lower streams, e.g., DS-1</td>
<td>1 100 (41%)</td>
</tr>
<tr>
<td>100 Mbps</td>
<td>Fast Ethernet</td>
<td>2 50 (20%)</td>
</tr>
<tr>
<td>150 Mbps</td>
<td>STS-3c</td>
<td>3 50 (20%)</td>
</tr>
<tr>
<td>160 Mbps</td>
<td>ESCON</td>
<td>4 20 (8%)</td>
</tr>
<tr>
<td>400 Mbps</td>
<td>Gigabit Ethernet (at a lower average rate)</td>
<td>8 10 (4%)</td>
</tr>
<tr>
<td>600 Mbps</td>
<td>ATM or Fast Internet Router Connection</td>
<td>12 10 (4%)</td>
</tr>
<tr>
<td>850 Mbps</td>
<td>Fiber Channel or FICON</td>
<td>18 2 (0.8%)</td>
</tr>
<tr>
<td>1 Gbps</td>
<td>Gigabit Ethernet (at its peak rate)</td>
<td>21 1 (0.4%)</td>
</tr>
</tbody>
</table>

In this section, we first present the results obtained for the cases of a uniform network. Then, we compare these two approaches for the cases of a non-uniform network. At the end of this section, we compare the network utilization of uniform and non-uniform upgrade.

4.1 Results for uniform networks

In this sub-section, we present experimental results for the cases of a uniform network. We conduct two sets of experiments, one with an initial capacity of 48 (i.e., line rate of OC-48) and an upgrade capacity (UC) of 48, and the other with an initial capacity of 192 (i.e., line rate of OC-192) and an upgrade capacity of 192. Table 2 and Table 3 show the results of the two cases respectively. Both tables...
list the link capacity of the network, the established connection capacity (of all existing connections) before the link capacity is upgraded to the next higher level, the improvement of GLB-SCR over SSR in terms of established connection capacity under the same amount of link capacity, and the number of triggered reallocation events. The results show that the network with GLB-SCR can establish more connections in terms of the established connection capacity than the one with SSR. On average, the network with GLB-SCR can accommodate 2.4% and 1% more connections (in terms of the established connection capacity) in the two cases respectively. As shown in the tables, the network with GLB-SCR invokes a few more reallocation events to obtain this improvement, which translates into increased revenue for carriers.

### 4.2 Results for non-uniform networks

In Section 4.1, we have presented the experimental results for uniform networks. In non-uniform networks, the improvement of GLB-SCR over SSR becomes more significant because the design of GLB-SCR takes into consideration the non-uniform link capacity distribution. In this subsection, we present the experimental results for non-uniform networks. We conduct four sets of experiments: (a) IC=48 and UC=48; (b) IC=48 and UC=96; (c) IC=192 and UC=192; (d) IC=192 and UC=384. During the experiments, the links with maximum link load are upgraded in the network upgrade. Fig. 8, Fig. 9, Fig. 10, and Fig. 11 plot the network upgrade trace and reallocation trace in these cases respectively. The X-axis is the established connection capacity of all the existing connections and the Y-axis is the total network capacity (i.e., the sum of capacity of all the links). The upgrade trace line shows the footprint of the network upgrade as connections are provisioned. The points of the reallocation trace indicate when a reallocation event is triggered. Note that the Y-axis values of each reallocation trace (the symbols × and + in these figures) have no meaning.

In all cases, the network with GLB-SCR needs less total network capacity to provision the same amount of connections most of the time. During a few short periods, the total network capacity required by the network with GLB-SCR may temporarily reach the same or a slightly higher level, compared to the network with SSR. However, the network with GLB-SCR shows constant savings in total network capacity over the long run. The tables listed following the figures (i.e., Table 4, Table 5, Table 6, and Table 7) show the improvement of GLB-SCR over SSR in terms of total network capacity with the same amount of established connection capacity. The data in each table are obtained by sampling the cross-points of the upgrade trace and five equal-distance vertical lines (as shown in the corresponding figures). The tables show that the average improvement of GLB-SCR over SSR in terms of total network capacity is 2.9%, 7.9%, 5.7%, and 9% for each case respectively. There are two observations: (1) the higher the initial capacity, the more improvement is obtained; and (2) the greater the ratio of upgrade capacity over initial capacity, the more the improvement. This is because more free capacity creates more space for the GLB-SCR approach to balance the network load distribution. Improvement (or reduction) in total network capacity translates into savings in equipment investment. The reallocation trace in these figures shows that this improvement is achieved by triggering slightly more reallocations.

Fig. 8, Fig. 9, Fig. 10, and Fig. 11 also show that each upgrade in the network with GLB-SCR needs fewer capacity increments (due to the design of NLV). This property is especially important to carriers who have strict budget limits and want to achieve short-term cost savings. On the other hand, the results also show that carriers using the GLB-SCR approach can benefit from tremendous savings in capacity investments over a long-term period.

The merit of the GLB-SCR approach comes from the design of NLV and its objective of minimizing NLV. Fig. 12 shows the link load distribution before and after a GLB-SCR reallocation in the case with IC=48 and UC=48. It gives a direct vision for the value of each member in a NLV and explains how the link load distribution changes after a reallocation. The number of congested links is reduced after the reallocation by moving some of the load to the links with lower load. For example, the number of links that have more than 90% capacity used (i.e., have loads greater than 43 in this example) is reduced after the reallocation.
Table 4 The improvement of GLB-SCR over SSR for the non-uniform network when IC=48 and UC=48.

<table>
<thead>
<tr>
<th>Established Connection Capacity (STS-1)</th>
<th>Total Network Capacity (STS-1)</th>
<th>Improvement</th>
<th>GLB-SCR</th>
<th>Average = 2.9%</th>
</tr>
</thead>
<tbody>
<tr>
<td>538</td>
<td>4416</td>
<td>4320</td>
<td>2.2%</td>
<td></td>
</tr>
<tr>
<td>676</td>
<td>4848</td>
<td>4704</td>
<td>3.0%</td>
<td></td>
</tr>
<tr>
<td>814</td>
<td>5856</td>
<td>5760</td>
<td>1.6%</td>
<td></td>
</tr>
<tr>
<td>952</td>
<td>6384</td>
<td>6240</td>
<td>2.3%</td>
<td></td>
</tr>
<tr>
<td>1090</td>
<td>6960</td>
<td>6576</td>
<td>5.5%</td>
<td></td>
</tr>
</tbody>
</table>

Table 5 The improvement of GLB-SCR over SSR for the non-uniform network when IC=48 and UC=96.

<table>
<thead>
<tr>
<th>Established Connection Capacity (STS-1)</th>
<th>Total Network Capacity (STS-1)</th>
<th>Improvement</th>
<th>GLB-SCR</th>
<th>Average = 7.9%</th>
</tr>
</thead>
<tbody>
<tr>
<td>605</td>
<td>5568</td>
<td>4992</td>
<td>10.3%</td>
<td></td>
</tr>
<tr>
<td>810</td>
<td>7584</td>
<td>6816</td>
<td>10.1%</td>
<td></td>
</tr>
<tr>
<td>1015</td>
<td>9120</td>
<td>7776</td>
<td>14.7%</td>
<td></td>
</tr>
<tr>
<td>1220</td>
<td>9984</td>
<td>9792</td>
<td>1.9%</td>
<td></td>
</tr>
<tr>
<td>1425</td>
<td>11040</td>
<td>10752</td>
<td>2.6%</td>
<td></td>
</tr>
</tbody>
</table>

Table 6 The improvement of GLB-SCR over SSR for the non-uniform network when IC=192 and UC=192.

<table>
<thead>
<tr>
<th>Established Connection Capacity (STS-1)</th>
<th>Total Network Capacity (STS-1)</th>
<th>Improvement</th>
<th>GLB-SCR</th>
<th>Average = 5.7</th>
</tr>
</thead>
<tbody>
<tr>
<td>2500</td>
<td>19584</td>
<td>18240</td>
<td>6.9%</td>
<td></td>
</tr>
<tr>
<td>3500</td>
<td>23808</td>
<td>22636</td>
<td>4.8%</td>
<td></td>
</tr>
<tr>
<td>4500</td>
<td>30336</td>
<td>28032</td>
<td>7.6%</td>
<td></td>
</tr>
<tr>
<td>5500</td>
<td>33204</td>
<td>30720</td>
<td>7.0%</td>
<td></td>
</tr>
<tr>
<td>6500</td>
<td>35712</td>
<td>34944</td>
<td>2.2%</td>
<td></td>
</tr>
</tbody>
</table>

Table 7 The improvement of GLB-SCR over SSR for the non-uniform network when IC=192 and UC=384.

<table>
<thead>
<tr>
<th>Established Connection Capacity (STS-1)</th>
<th>Total Network Capacity (STS-1)</th>
<th>Improvement</th>
<th>GLB-SCR</th>
<th>Average = 9.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>2100</td>
<td>17280</td>
<td>16512</td>
<td>4.4%</td>
<td></td>
</tr>
<tr>
<td>2700</td>
<td>22656</td>
<td>20736</td>
<td>8.5%</td>
<td></td>
</tr>
<tr>
<td>3300</td>
<td>26496</td>
<td>24192</td>
<td>8.7%</td>
<td></td>
</tr>
<tr>
<td>3900</td>
<td>33792</td>
<td>30336</td>
<td>10.2%</td>
<td></td>
</tr>
<tr>
<td>4500</td>
<td>41088</td>
<td>35712</td>
<td>13.1%</td>
<td></td>
</tr>
</tbody>
</table>

4.3 Comparison of uniform upgrade and non-uniform upgrade

In Section 4.1 and 4.2, we have shown that the GLB-SCR approach outperforms the SSR approach in terms of established connection capacity for uniform networks and in terms of total network capacity for non-uniform networks. From a network planning perspective, uniform networks are formed by uniform upgrade, which increments the same amount of capacity for every link in the network. An alternative is to upgrade a portion of the links in the network, which is called a non-uniform upgrade. This section compares the uniform upgrade and non-uniform upgrade schemes under the GLB-SCR approach. We only show the results of the case with IC=48 and UC=48. Other cases have similar results. Table 8 gives the total network capacity needed for the uniform upgrade and for the non-uniform upgrade respectively. The data are sampled at the time when a uniform upgrade occurs (or the same amount of connection capacity is established for the case of non-uniform upgrade). It shows that the non-uniform upgrade scheme saves, on average, 29% total network capacity compared to the uniform upgrade scheme. However, this saving is obtained at the expense of more
network upgrades, resulting in increased operations expenditure (OPEX) for more frequent equipment installation. Carriers can choose their own scheme based on estimates of CAPEX and OPEX.

Table 8 Comparison of the uniform upgrade and non-uniform upgrade for the case when IC=48 and UC=48

<table>
<thead>
<tr>
<th>Established connection capacity (STS-1)</th>
<th>Total Network Capacity (STS-1)</th>
<th>Improvement of uniform upgrade over non-uniform upgrade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform Upgrade</td>
<td>Non-uniform Upgrade</td>
<td>Average = 29%</td>
</tr>
<tr>
<td>1038</td>
<td>8256</td>
<td>6576</td>
</tr>
<tr>
<td>1508</td>
<td>12384</td>
<td>8976</td>
</tr>
<tr>
<td>2158</td>
<td>16512</td>
<td>11904</td>
</tr>
<tr>
<td>2480</td>
<td>20640</td>
<td>13104</td>
</tr>
<tr>
<td>3308</td>
<td>24768</td>
<td>16416</td>
</tr>
</tbody>
</table>

5. Conclusions

In this paper, we proposed a load-balancing spare capacity reallocation approach in next-generation SONET metro networks. We defined a novel network load vector (NLV), which reflects the network load distribution and takes into consideration the non-uniform link capacity distribution. Our proposed approach aims at minimizing this NLV. For the first time, we proposed a spare capacity reallocation approach from a capacity planning and network upgrade perspective. We developed a greedy heuristic (GLB-SCR) to implement our approach and conducted experiments to compare GLB-SCR with a previous approach (SSR). Our experiments show that GLB-SCR outperforms SSR in terms of established connection capacity and total network capacity in both uniform and non-uniform networks. This improvement translates into increased revenue for carriers. Because of the way the NLV is designed, the advantage of GLB-SCR over SSR is more significant in non-uniform networks, which are more likely to be the case in real-world metro networks. In the experiments, we further showed that the non-uniform upgrade can bring significant savings on capacity investment at the expense of more network upgrades.

This paper focuses on the next-generation SONET metro networks. However, our approach can be applied to any circuit-switching network. Although our approach only operates on protection paths, it can be slightly modified to reallocate both working and protection paths when some connections allow their working paths to be rerouted under their service level agreements (SLAs).

6. References