6-25-2010

Topology overlays for dedicated protection Ethernet LAN services in advanced SONET/SDH networks

Srikanth Kumar Seethamraju

Srikanth Venkata Kumar Seethamraju
Candidate

Electrical and Computer Engineering
Department

This thesis is approved, and it is acceptable in quality and form for publication:

Approved by the Thesis Committee:

, Chairperson

Topology Overlays for Dedicated Protection Ethernet LAN Services in Advanced SONET/SDH Networks

BY

SRIKANTH KUMAR SEETHAMRAJU

THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science

Computer Engineering

The University of New Mexico
Albuquerque, New Mexico

May, 2010

ACKNOWLEDGEMENTS

First and foremost I offer my sincerest gratitude to my supervisor, Dr Nasir Ghani, who has supported me throughout my thesis. His patience and knowledge has given me the room to work in my own way and grow. I attribute the level of my Masters degree to his encouragement and effort and without him this thesis, too, would not have been completed or written. One simply could not wish for a better or friendlier supervisor. I also wish to thank my other Thesis Committee members, Drs. Payman Zarkesh-Ha and James Plusquellic, for their feedback and reviews. Here, each individual provided insights that guided my thinking, helping to substantially improve the finished product. Finally and most importantly, I would like to thank my parents for supporting me throughout all my university studies.

Topology Overlays for Dedicated Protection Ethernet LAN Services in Advanced SONET/SDH Networks

BY

SRIKANTH KUMAR SEETHAMRAJU

ABSTRACT OF THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Master of Science

Computer Engineering

The University of New Mexico
Albuquerque, New Mexico

May, 2010

Topology Overlays for Dedicated Protection Ethernet LAN Services in Advanced SONET/SDH Networks

By

Srikanth Kumar Seethamraju, M.S., Computer Engineering

University of New Mexico, 2010

ABSTRACT

The explosion of *information technology* (IT) services coupled with much-increased personal and scientific computing capabilities has resulted in great demand for more scalable and reliable networking services. Along these lines, carriers have spent large sums to transition their "legacy" SONET/SDH voice-based networking infrastructures to better support client-side Ethernet data interfaces, i.e., *next-generation SONET/SDH* (NGS). In particular, a key addition here has been the new *virtual concatenation* (VCAT) feature which supports inverse multiplexing to "split" larger connection requests in to a series of independently-routed "sub-connections". As these improved infrastructures have been deployed, the design of new *Ethernet over SONET/SDH* (EoS) services has become a key focus area for carriers, i.e., including point-to-point and multi-point services.

In light of the above, this thesis focuses on the study of improved *multi-point* EoS schemes in NGS networks, i.e., to provision robust "virtual LAN" capabilities over metro and wide-area domains. Indeed, as services demands grow, survivability considerations are becoming a key concern. Along these lines, the proposed solution develops novel multi-tiered

(partial) protection strategies. Specifically, graph-theoretic algorithms are first proposed to interconnect multi-point node groups using bus and *minimum spanning tree* (MST) overlays. Next, advanced multi-path routing schemes are used to provision and protect these individual overlay connections using the inverse-multiplexing capabilities of NGS. Finally, post-fault restoration features are also added to handle expanded failure conditions, e.g., multiple failures.

The performances of the proposed multi-point EoS algorithms developed in this research are gauged using advanced software-based simulation in the *OPNET Modeler*^{TM} environment. The findings indicate that both the bus and MST overlays give very good performance in terms of request blocking and carried load. However, the MST-based overlays slightly outperform the bus-based overlays as they allow more efficient topology designs. In addition, the incorporation of dynamic load state information in the selection of bus and/or MST overlays is also very beneficial as opposed to just using static hop count state. Furthermore, inverse-multiplexing is highly-effective, yielding notably higher carried loads when coupled with load-balancing sub-connection routing. Finally, results also show that post-fault restoration is also a very effective means of boosting EoS LAN throughputs for partially-protected demands, consistently matching the reliability of full-protection setups.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ATM	Asynchronous Transfer Mode
BLSR	BI-Directional Path Switched Ring
CRC	Cyclic Redundancy Checking
DES	Discrete Event Simulation
DWDM	Dense Wavelength Division Multiplexing
ESCON	Enterprise System Connection
EOS	Ethernet-Over-SONET
EPL	Ethernet Private Line
EVPL	Ethernet Virtual Private Line
FDM	Frequency Division Multiplexing
GFP	Generic Framing Protocol
GFP-F	Frame-Mapped GFP
GFP-T	Transparent Mapped GFP
IAT	Inter Arrival Time
IP	Internet Protocol
ILP	Integer Linear Programming
ISP	Internet Service Provider
ITU-T	International Telecommunications Union
LAN	Large Area Networks
LCAS	Link Capacity Adjustment Scheme
MAC	Media Access Control
MAC	Minimum Average Cost
MAH	Minimum Average Hop

MEF	Metro Ethernet Forum
MSP	Multi-Service Provisioning
MST	Minimum Spanning Tree
MTTR	Mean Time To Repair
NGS	Next Generation SONET/SDH
NSFNET	National Science Foundation Network
POS	Packet-Over SONET
RWA	Routing and Wavelength Assignment
SAN	Storage Area Networks
SDH	Synchronous Digital Hierarchy
SONET	Synchronous Optical Networks
SONET/SDH	Synchronous Optical Network/Synchronous Digital Hierarchy
STS-1	Synchronous Transport Signal Level-1
TDM	Time-Division Multiplexing
UPSR	Unidirectional Path Switched Ring
VCG	Virtual Concatenation Group
VCAT	Virtual Concatenation
WSP	Widest-Shortest Path

CHAPTER 1

INTRODUCTION

The growth of the Internet has revolutionized the world of the telecommunications in terms of data traffic and services diversity. Moreover, the explosion of information technology coupled with the increasing power of computing capabilities has resulted in sustained demand for more flexible and reliable bandwidth services. By some predictions, bandwidth usage in the Internet alone is doubling every six to twelve months [1]. In order to provision these demands, new optical network technologies have been developed most notably *dense wavelength division multiplexing* (DWDM) and *next-generation SONET/multi-service provisioning* (NGS-MSP) [2].

Optical network technologies like DWDM use *frequency division multiplexing* (FDM) techniques to carry multiple information streams over a single fiber. As such, modern DWDM systems can increase a single fiber's capacity over one hundred fold, i.e., to a throughput of terabits. For example, many transport systems now support 100-200 of wavelengths running at speeds of gigabits/sec each. Hence, DWDM allows operator to maximize their existing fiber infrastructure. Moreover, another key advantage of using DWDM is its protocol and bit rate transparency which allows it to carry different formats, e.g., packet (IP, Ethernet), circuit (SONET/SDH, voice) and even *storage area network* (SAN) protocol traffic over a common fiber base infrastructure.

Nevertheless, the deployment of DWDM technology has not been uniform across all networking domains. For example, smaller metro and edge markets have end-user clients with much smaller "sub-gigabit" bandwidth demands, e.g., for services such as fractional Gigabit

Ethernet (Layer 2), legacy private line (155 Mbps OC-3, 622 Mbps OC-12). Here, provisioning a full multi-gigabit wavelength for each user demand will inevitably yield very high costs and low bandwidth/fiber efficiencies. As a result, various technologies have been evolved with the goal of concentrating multiple end-user client services onto larger DWDM tributaries to improve efficiency [4]. A notable example is *next generation SONET/SDH* (NGS), which evolves legacy *time-division multiplexing* (TDM) SONET/SDH circuit-switching to provide much improved support for IP/Ethernet data services. Namely, NGS introduces new standards such *virtual concatenation* (VCAT), *inverse multiplexing*, *generic framing protocol* (GFP), and *link capacity adjustment scheme* (LCAS) [2-3]. Here, by combining the fine granularity of NGS and with the vast capacity of DWDM, carriers can greatly improve efficiency and economy. At the same time, they can continue to use their vast deployed "voice-centric" TDM base. Today, NGS is a very mature technology and has already been widely deployed across carrier domains [8].

1.1 MOTIVATION

As the above technologies have evolved (DWDM, NGS), the focus for many carriers has now shifted towards innovative "value-added" services. In particular, there is much interest in extending ubiquitous Ethernet data interconnection over larger distances, i.e., by mapping over DWDM and SONET/NGS transport networks [6]. These demands are being driven by the emergence of Ethernet as the premiere solution in the corporate space e.g., for applications such as voice, storage/data center extension, *Internet service provider* (ISP) interconnection, etc. Hence various industry standard organizations such as, the *International Telecommunications Union* (ITU-T) and *Metro Ethernet Forum* (MEF) are focusing and defining related services on this key area. Foremost, *Ethernet Private Line* (EPL) type offerings have been introduced to offer point-to-point connectivity and to replace existing legacy TDM private line offerings.

Meanwhile, more advanced *multipoint* Ethernet services have also been defined, e.g., such as Ethernet LAN and Ethernet Tree extension. The goal here is to extend associated Layer2 across switching capabilities across broader geographic domains, i.e., offering "virtual switch" services. Overall, as these services have emerged, various *Ethernet-Over-SONET* (EOS) algorithms have also been developed to provision them over DWDM and TDM networks, e.g., work on sub rate growing, inverse multiplexing, etc [7]. However by and large, most of these schemes have only addressed point-to-point, i.e., EPL, type offerings. Although some recent efforts have considered more advanced multipoint overlay schemes, they are premised upon simpler strategies and leave much room for improvements. This forms the core motivation for this research thesis.

1.2 Problem Statement

The thesis mainly focuses on the design of multipoint Ethernet LAN services over NGS networks. In particular, it will explore the application of new virtual concatenation and inverse multiplexing algorithms and develop novel network overlay solutions based upon efficient bus and *minimum spanning tree* (MST) topologies.

1.3 Scope

The study will evaluate the performance of the new multipoint Ethernet overlay topologies for NGS networks with virtual concatenation. In particular it will apply routing strategies, e.g., such as minimum hop count and minimum distance for different client pay loads. The performance analysis effort will be conducted using software-based simulation using the

OPNET Modeler^{TM} discrete event simulation tool. In addition, the key performance evaluation metrics will include LAN request blocking rates, carried load, and recovery rates.

1.4 Approach

In order to achieve the above objectives, the research works in partitioned into three key tasks. The first task will be to conduct a literature survey on recent advancements in NGS and carrier Ethernet technologies as well as on the network overlay schemes. Next the effort will focus on the specification and design of new overlay provisioning algorithms for multipoint carrier Ethernet services, i.e., based upon bus and MST overlays. Finally, the third task is to focus on the coding and evaluation of these schemes with and without load balancing using the *OPNET Modeler*^{M} tool. Here various network topologies will be built and the associated performances will be verified under realistic client traffic loads.

1.5 Thesis Outline

The entire thesis is categorized as follows. First, Chapter 1 presents an introduction to the thesis and also its major objectives. Chapter 2 then gives a brief review of latest advancements in NGS and Carrier Ethernet services. Next, Chapter 3 presents the bus and MST overlay algorithms along with complete pseudo code descriptions. Chapter 4 then highlights the test topologies used and the key evaluation matrices. Detailed performance evaluation results are presented in Chapter 5 in order to demonstrate the gains of using NGS technologies for multipoint Ethernet services. Finally, Chapter 6 presents conclusions of the thesis and provides the required scope and directions for the further study.

CHAPTER 2

NEXT GENERATION SONET/SDH

Over the last two decades, *Synchronous Optical Network/Synchronous Digital Hierarchy* (SONET/SDH) technology has emerged as the preferred framing solution for fiber-optic transmission. These technologies were originally designed primarily for the transport of constant bit-rate voice and private line traffic applications over a rigid multiplexing hierarchy of TDM signals, i.e., STS-3, STS-12. However in the early 2000's, the demands from data and other forms of "non-TDM traffic" increased drastically and hence transporting these "packet-based" flows over *circuit-switched* SONET/SDH networks became a big challenge. In order to overcome these limitations and to improve data efficiency, the ITU-T evolved a comprehensive new set of standards known as *next generation SONET/SDH* (NGS) or *multi-service provisioning* (MSP). As a result, modern SONET/SDH-based systems can transport constant bit rate clients (e.g., voice users) as well as variable-rate packet-oriented clients, such as IP/Ethernet users. Moreover, SONET/SDH interface speeds have steadily increased, all the way from 51 Mbps to 10 Gbps [9]. This chapter presents a more detailed overview of these NGS capabilities and also surveys their application for Ethernet service provisioning [10], [11], and [12].

2.1 Next Generation SONET/SDH

The key enhancements in the NGS framework include *generic framing procedure* protocol (GFP, ITU-T G.7041), *virtual concatenation* (VCAT), *inverse multiplexing* and the *link capacity adjustment scheme* (LCAS, ITU-T G.7042)[12].[13]. These standards are now highlighted.

Given that legacy SONET/SDH uses a standard set of payloads it is generally not possible to transport variable-length packets efficiently via these fixed formats. In order to resolve these concerns, NGS has defined new mappings to route packets over multiple timeslots and recombine them at end of the transmission i.e., *virtual concatenation.* Hence, a virtually-concatenated SONET channel is made up of N STS-1 slots that are transported as individual STS-1circuits across the network and are re-aligned and sorted to recreate the original payload at the receiver. Note that this type of payload processing only requires concatenation functionality at the path source and termination equipments. For example, Figure 2.1 shows virtual concatenation for carrying two Gigabit Ethernet streams over a 2.5 Gbps OC- 48 (21 STS-1 each) as well as one 200 Mbps Fibre Channel streams (4 STS-1) [14]. Note that when traffic from one client is sent over different routes, payload mapping at the destination node must compensate for differential delays between bifurcated streams. Along these lines, most commercially-available VCAT devices can support up to 50 ms (+/- 25ms) differential delays between individual streams via buffering, i.e., equivalent to a 10,000 km route length differential. Overall, virtual concatenation helps SONET/SDH networks carry traffic with finer granularity levels and improves link capacity/efficiency. Moreover, it also enables much better survivability provisioning [12].

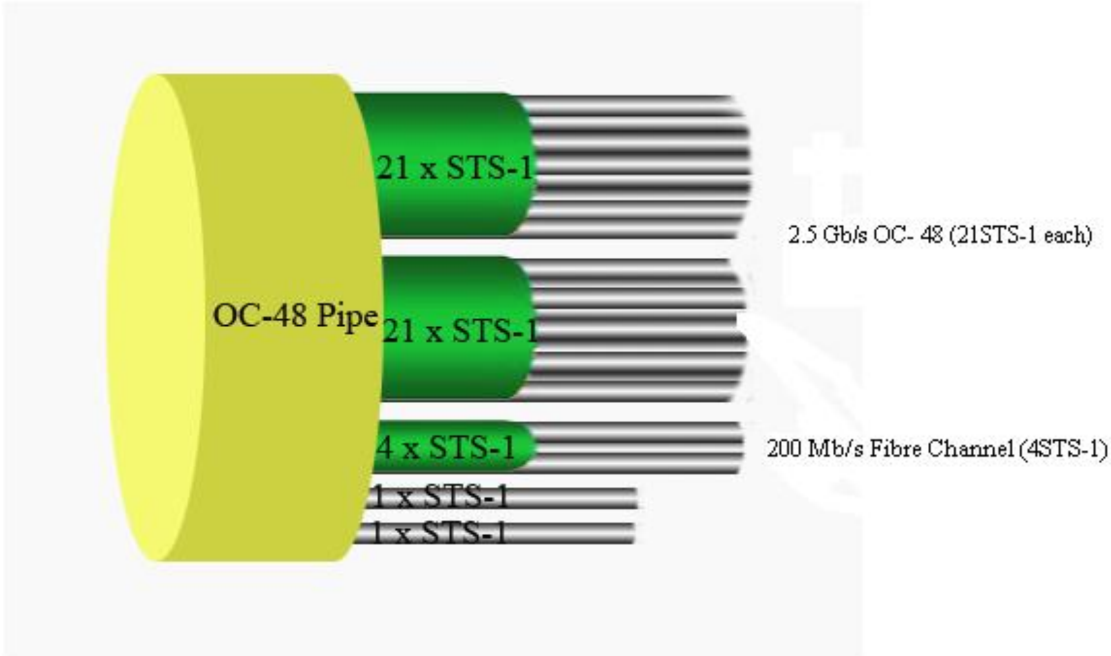

Figure 2.1 Example of using virtual concatenation to support different network services

Meanwhile, the LCAS standard further enhances virtual concatenation by allowing users to increase or decrease the capacity on virtually concatenated segments without interrupting the overall end-to-end traffic flows. In essence, LCAS allows carriers to provision "*time-varying*" fractional Ethernet services, a key advantage over legacy schemes [2], [3]. LCAS can also be applied for survivability support as well. Note however that the LCAS standard requires end-to-end signaling between the source/sink end-points in order to coordinate bandwidth adjustments.

Finally, the GFP standard defines a universal mechanism to transport a range of formats over SONET/SDH channels, e.g., Ethernet, Fibre Channel, ESCON, FICON, etc. Namely, GFP supports both point-to-point and ring applications by providing a single and flexible mechanism to map any client signal into SONET/SDH payloads. Furthermore, it also eliminates the need for byte stuffing which in turn saves bandwidth by avoiding payload-specific frame expansion i.e., as compared to earlier *packet-over SONET* (POS) schemes [16]. Now, in order cater to all mapping requirements, two mapping modes are also defined for GFP including frame-mapped GFP (GFP-F) and transparent mapped (GFP-T).

2.1.1 Frame-mapped GFP

As the name indicates, this mechanism maps the entire client into one frame and also describes that particular frame. For example, all the ETHERNET *media access control* (MAC) frames are mapped to an individual GFP frames.

2.1.2 Transparent-mapped GFP

Meanwhile, many other "low latency" client protocols use block-coded formats such as Fibre Channel, ESCON, FICON, etc. Hence the corresponding GFP transparent mode performs direct byte mapping for these client protocol blocks, thereby eliminating the need for excessive edge buffering. Specifically, in this mode, the GFP frame contains group of 8B/10B code blocks mapped into a 64B/65B block with added *cyclic redundancy checking* (CRC).

2.2 Overview on NGS Survivability Schemes

As the above NGS standards have matured, a range of related survivability schemes have emerged. These solutions improve vastly upon earlier legacy SONET/SDH strategies and are now surveyed. Foremost, legacy SONET/SDH networks have relied upon dedicated pre-fault protection schemes. Specifically, these include link level protection (e.g., 1:1, 1+1) and ring level protection e.g., 2 fiber *unidirectional path switched ring* (UPSR) and 2/4 *fiber bi-*

directional line switched ring (BLSR) [5]. Some SONET/SDH networks have also applied slower post-fault path restoration, especially in the core. Overall, these legacy schemes are best suited for protecting fixed tributaries and fibers. As such, they are very inflexible for data (i.e., Ethernet) traffic protection and primarily offer "all-or-nothing" type recovery. For example, a Gigabit Ethernet client may only need fractional service (200 Mb/s) and partial protection e.g., 100 Mb/s [16]. However, when using legacy BSLR protection, the user will have to be provisioned with two diversely-routed OC-12 (622 Mb/s) circuits, i.e., over 3 times capacity wastage. To resolve these concerns, a wide range of improved NGS-based survivability schemes have been proposed. These are now considered further.

The ability to split traffic across sub-connections has also given rise to various "*multi-path diversit*y" routing approaches. For example, [5] outlines several *protection schemes* for Ethernet over SONET (PESO), termed as PESO α, β, γ. Here, no actual protection connections are routed as the overall goal is minimize the impact of single link failures. However detailed analysis/simulation results are not presented in this study. Along the lines [17] also possess a multi-path diversity path approach to provide a degraded-service-wave provisioning. Load distribution is also applied here to minimize the maximum incremental link utilization using *integer linear programming* (ILP) techniques. The overall findings here show decent gains in blocking reduction and load balancing.

Overall since, the above multi-path diversity schemes do not actively protect sub-connections, they can be susceptible to topology limitations, i.e., limited service recovery guaranteed. As a result, other schemes have been designed to actually protect individual sub-connections including, a class of "tiered" protection schemes in [5], [13]. Specifically, here a protection threshold factor, ρ, is first defined to specify a pre-provisional level of protection and then, individual sub-connections are incrementally protected to achieve this threshold. Now, the actual sub-connection routing can be done using either a hop count or load balancing approach. Finally, post-fault restoration is also provided to recover non-protected sub-connections. Overall, results with tiered protection schemes show notable gains (lower blocking) when combining load balancing routing with increased levels of inverse multiplexing, i.e., demand is splitting. Post fault restoration is also seen to be extremely beneficial.

Meanwhile, [22] considers the application of more advanced shared protection concepts between VCG sub-connections. Namely, two shared multiplexing schemes are presented, PVIM and PREV. The latter allows backup capacity sharing between link-disjoint VCG members where as the second approach only allows sharing between link disjoint VCG members with the same source and destination. Overall, PIVM gives much higher efficiency, albeit at the cost of slow recovery and complex per-link VCG member "*conflict*" state. Hence, this scheme is amenable only to centralized implementation. Meanwhile, PREV gives faster recovery since the switchover routes are known in advance, but requires complex min-cost flow pre-calculation. Also [19], [20] develops a new effective multi-path bandwidth metric that takes into account both link bandwidth and availability constraints. Here two multi-path routing heuristics are developed to achieve desired availability levels, with both showing significant improvements versus single path provisioning strategies [18].

2.3 Multipoint EOS Services

Overall, the above schemes have only focused on point-to-point EOS connections. However, as carrier Ethernet paradigms evolve, there is a pressing need to provision more expansive multipoint schemes. In particular, consider the emergence new Ethernet LAN service definitions in the MEF [21], the goal of which is to deliver "virtual switch" connectivity over multiple dispersed geographic sites. Indeed, very few studies have considered provisioning for such multipoint EOS services. For example in [20], the authors have studied some very basic mesh and star overlay schemes for achieving such interconnectivity over SONET/SDH networks [21]. However, these schemes are very susceptible to topological limitations and have very low efficiency (high blocking). Indeed, there is much room for improvement via the design of more effective topology overlays. These concerns are now further addressed further in this thesis.

CHAPTER 3

MULTI-TIERED LAN SURVIVABILITY IN NGS NETWORKS

Extending Carrier Ethernet LAN services over advanced SONET/SDH domains requires the provision of *multi-point-to-multi-point* connectivity across dispersed metro/wide-area domains. Due to the lack of multi-casting features in SONET/SDH, here it is necessary to setup *multiple point-to-point* TDM connections, i.e., connection groups or topology overlays. Although some basic schemes have been investigated using mesh and star overlays [20], more capable variants like bus and MST overlays are now introduced.

3.1 Notation

Before detailing the solutions, the necessary notation is first introduced. Consider a physical SONET/SDH network of N nodes and M links. This network can be modeled as a graph $\mathbf{G}(\boldsymbol{V}, \boldsymbol{L})$, where $\boldsymbol{V}$ is the set of NGS nodes sites (vertices) and $\boldsymbol{L}$ is the set of SONET/SDH links (edges),i.e., $\boldsymbol{V} = \{v_1, v_2, ..., v_n\}$and $\boldsymbol{L} = \{ l_{12}, l_{13}, l_{14}....., l_{ij}\}$. Here link l_{ij} is the link between nodes i and j of C units capacity and c_{ij} is the available capacity of this link. Necessarily, if there is a link from i to j, then there is also a link from j to i since SONET/SDH links are bi-directional. Each site is also assumed to have overlying Ethernet switching capabilities.

Now Ethernet LAN request between a subset of nodes given by the vector $\boldsymbol{v}_i = \{v_{i1}, v_{i2}, ...\}$, $\boldsymbol{v}_i \subseteq \boldsymbol{V}$. This request is used to build a *connection group* comprising of a set of n_i bi-directional point-to-point TDM bypass connections, $\{\boldsymbol{s}_i, \boldsymbol{d}_i\}$, where the vector $\boldsymbol{s}_i = \{s_{i1}, s_{i2}, \ldots\} \subseteq \boldsymbol{v}_i$ and $\boldsymbol{d}_i = \{d_{i1}, d_{i2}, \ldots\} \subseteq \boldsymbol{v}_i$ represent the source/destination end-points, e.g., individual connections denoted as $s_{i1}– d_{i1}$, $s_{i2}– d_{i2}$, etc. Namely, data flowing between these connections end-points are not handled at the intermediate Ethernet switching nodes, i.e., bypass. Assuming a requested LAN throughput of x_i STS-1 units, each individual connection can also be assumed to be of size x_i STS-1 units. Now even though the exact makeup of the connection group will depend upon the overlay topology chosen, each of the constituent point-to-point connections can still be inverse-multiplexed, i.e., "split", in to multiple "sub-connections". Namely, the latter is specified by an *inverse multiplexing factor*, *K*. Hence the LAN connection group LAN request is denoted as the tuple (n_i, {$\mathbf{s}_i$, $\mathbf{d}_i$},x_i**,** *K*).

.

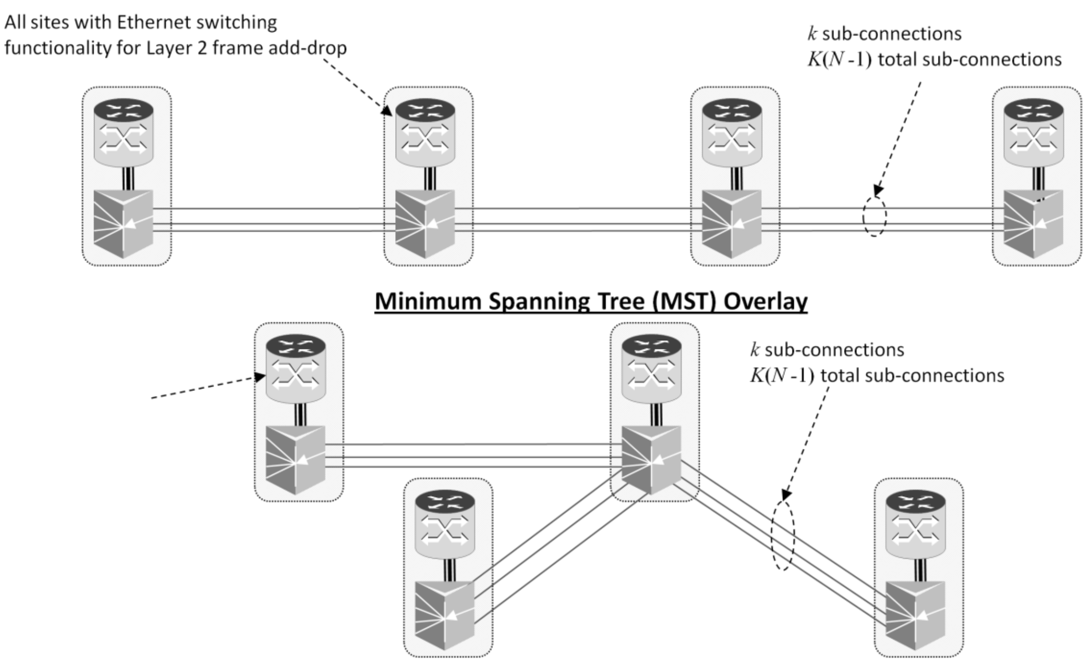

Figure 3.1: Ethernet LAN overlays: bus, MST

3.2 Topology Overlay Connection group overlay design first computes the set of LAN group connections to provisioned, i.e.,{$\mathbf{s}_i$,$\mathbf{d}_i$}. Here variants of bus and MST overlays are proposed, as shown in the Figure 3.1, expanding upon the earlier work in [20].

Given a LAN request between nodes in $\boldsymbol{v_i}$, where $|\boldsymbol{v_i}|\geq 3$

Initialize bus sequence vector $\boldsymbol{r_i}=\{\}$

Select first bus node pair (overlay link) in $\boldsymbol{r_i}=\{r_{i1}, r_{i2}\}$from v_i using min, average hop count or minimum average cost

% Generate first connection in bus overlay group

$s_{i1} = r_{i1}$

$d_{i1} = r_{i1}$

% Loop and generate rest of bus overlay

for $j=1$ to $|\boldsymbol{v_i}|-2$

{

Search for next candidate ordering node from first node in ordering

vector $\boldsymbol{r_i}$ using min. hop or min. cost, i.e., v_{ik*}, x_1 (Eqs. 5, 6)

Search for next candidate ordering node from last node in ordering

vector $\boldsymbol{r_i}$ using min. hop or min. cost, i.e., v_{im*}, x_2 (Eqs. 8, 9)

% Update bus sequence vector, generate overlay link connection

if ($x_1 \leq x_2$)

{

% Minimum hop or cost is from first node

$\boldsymbol{r_i}=\{v_{ik*}, r_{i1}, r_{i2}, \ldots, ri_{j-1}\}$

$s_{ij} = v_{ik*}$

$d_{ij} = r_{i1}$

} else

{

% Minimum hop or cost is from last node

$\boldsymbol{r_i}=\{r_{i1}, r_{i2}, \ldots, ri_{j-1}, v_{im*}\}$

$\boldsymbol{s_{ij}} = \boldsymbol{r_{ij-1}}$

$\boldsymbol{d_{ij}} = v_{im*}$

}

} *% for loop*

% Set LAN group connection count

$n_i=|v_i|-1$

Figure 3.2: Bus overlay

3.2.1. Bus Overlay

This overlay implements a linear inter-connection between the LAN sites, Figure 3.1, and requires a total of $n_i=O(|\boldsymbol{v}_i|-1)=O(|\boldsymbol{V}|)$ connections or $O(K(|\boldsymbol{v}_i|-1))=O(K|\boldsymbol{V}|)$ VCAT sub-connections. As such, it is equivalent to a specialized tree overlay with only one child per parent. Although this design has notably lower nodal in-degree requirements than a star overlay [20], i.e., 2, it requires Ethernet "add-drop" switching capabilities at all LAN group sites. Moreover, associated connection group selection is more involved here, as shown in the pseudo code listing of Figure 3.2. Specifically, the main goal here is to determine the node *sequence* that yields minimum overall bus resource utilization and/or lowers blocking. To compute this, the algorithm uses node sequence vector, $\boldsymbol{r}_i = \{r_{i1}, r_{i2}, \ldots\} \subseteq \boldsymbol{v}_i$, to iteratively build the ordering of nodes in the bus. Initially this vector is initialized to a null value to reflect the fact that all nodes in $\boldsymbol{v}_i$ are "unassigned", Figure 3.2. Subsequently, an initial node pair is "assigned" by generating the first overlay link in $\boldsymbol{r}_i$, i.e., as per a given selection strategy (detailed shortly). The algorithm then iterates through the remaining "unassigned" nodes ($\boldsymbol{v}_i$-$\boldsymbol{r}_i$) to sequentially determine their ordering (connection group overlay links). Specifically, at each iteration the first and last nodes in $\boldsymbol{r}_i$ are analyzed to "assign" the next node according to a particular strategy:

Random Selection (Bus-RS): This scheme selects consecutive bus nodes in a random manner. Namely, the next node is chosen by applying a uniform distribution over the remaining "unassigned" bus nodes.

Minimum Average Hop (Bus-MAH): This scheme chooses bus nodes in order to minimize resource consumption. Namely, the first two nodes in the bus (i.e., first overlay link) are "assigned" as the LAN group node pair with the minimum interconnecting hop count, i.e.,

$$\boldsymbol{r_i} = \{r_{i1}, r_{i2}\} = \{v_{ik*}, v_{im*}\} \qquad \text{Eq. (3.1),}$$

where $k^*, m^* = \min_{k,m}(\text{hop}(v_{ik}, v_{im}))$. Next, the scheme iterates to select the remaining bus nodes by checking the "assigned" end-points (i.e., loop over index j, $j \geq 3$, Figure 3.2). Specifically, the first and last "assigned" end-point nodes in $\boldsymbol{r}_i$, i.e., r_{i1} and r_{ij-1}, are examined to determine the next "unassigned" node with the minimum average hop count. This is done by comparing the respective minimum hop count from the first node in the current ordering vector, i.e.,

$$x_1 = \text{hop}(r_{i1}, v_{ik*}) \qquad \text{Eq. (3.2),}$$

where $k^* = \min_k(\text{hop}(r_{i1}, v_{ik}))$, $v_{ik} \notin \boldsymbol{r}_i$, versus that from the last node in the current ordering vector:

$$x_2 = \text{hop}(r_{ij-1}, v_{im*}) \qquad \text{Eq. (3.3),}$$

where $m^* = \min_m(\text{hop}(r_{ij-1}, v_{im}))$, $v_{im} \notin \boldsymbol{r}_i$. Hence the next bus node is "assigned" and appropriately inserted at the head or tail of the ordering vector, i.e., after iteration j, as follows:

$$\boldsymbol{r}_i = \begin{cases} \{v_{ik^*}, r_{i1}, r_{i2}, \ldots, r_{ij-1}\} & if \quad x_1 \leq x_2 \\ \{r_{i1}, r_{i2}, \ldots, r_{ij-1}, v_{im^*}\} & if \quad x_1 > x_2 \end{cases} \qquad \text{Eq. (3.4),}$$

Since, this approach only uses static information to choose the bus node ordering, i.e., $O(|\boldsymbol{v}_i|(|\boldsymbol{v}_i|-1))=O(|\boldsymbol{V}|^2)$ shortest path computations prior to start up with $O(|\boldsymbol{v}_i|^2)=O(|\boldsymbol{V}|^2)$ storage overheads. However at run-time $O(|\mathbf{v}_i|(|\mathbf{v}_i|-1))= O(|\mathbf{V}|^2)$ lookups are needed to select each node in the ordering, yielding a complexity of $O(|\mathbf{V}|^2)$.

Minimum Average Cost (Bus-MAC): This scheme follows the same overall flow as the Bus-MAH scheme, with the exception that hop counts are now replaced by link "costs". Namely the link cost here is dynamic and defined as inversely proportional to the available capacity on link, i.e., for the link l_{ij},

$$\text{Cost}\ (\mathbf{v}_i, \mathbf{v}_j) = \omega_{ij} = \frac{1}{c_{ij} + \varepsilon} \qquad \text{Eq. (3.5),}$$

where ε is a small quantity chosen to avoid floating-point errors.

Hence,

$$x_1 = \text{cost}(r_{i1}, v_{ik^*}) \qquad \text{Eq. (3.6),}$$

and

$$x_2 = \text{cost}(r_{ij-1}, v_{im^*}) \qquad \text{Eq. (3.7)}$$

where the cost() function is defined as $k^* = \min_k(\text{cost}(r_{i1}, v_{ik}))$, $v_{ik^*} \notin \mathbf{r}_i$ and $m^* = \min_m(\text{cost}(r_{ij-1}, v_{im}))$, $v_{im^*} \notin \mathbf{r}_i$. Since this scheme uses dynamic resource state, it also has higher compute complexity, i.e., $O(|\mathbf{v}_i|(|\mathbf{v}_i|-1))= O(|\mathbf{V}|^2)$ shortest path computations for bus node selection, yielding a total compute complexity of $O(|\mathbf{V}|^2 \cdot |\mathbf{V}| \log|\mathbf{V}|)= O(|\mathbf{V}|^3 \log|\mathbf{V}|)$.

```
Given a LAN request between nodes in v_i, where |v_i|≥3
Initialize MST nodes vector r_i={}
Select first MST node in r_i={ r_i1 }randomly from v_i
% Loop and generate rest of MST overlay
for j=1 to |v_i|-1
{
Search for next candidate MST node by using min. hop or
min. cost from existing nodes in r_i, i.e., v_ik* (Eqs. 10, 11)
% Update ring sequence vector, generate overlay link connection
        r_i={r_i1 , r_i2 , …, r_ij-1, v_ik*}
        s_ij = r_im
        d_ij  = v_ik*
}
% Set LAN group connection count
n_i=|v_i|-1
```

Figure 3.3: MST overlay based on Prim's algorithm

3.2.2 Minimum Spanning Tree (MST) Overlay

The MST overlay achieves a balance between the star and bus overlays by constructing a more generalized tree. This is achieved by adapting Prim's MST algorithm [24] for the subset of overlay LAN nodes. Namely, the initial MST node is selected randomly at first. The algorithm then loops to add nodes (links) until all LAN nodes are accounted for, similar to the Dijkstra's shortest-path search procedure.

The overall pseudo code for the MST overlay is shown in Figure 3.3. Akin to the bus overlay, a node sequence vector, $\boldsymbol{r}_i = \{r_{i1}, r_{i2}, \ldots\} \subseteq \boldsymbol{v}_i$, is used to track the nodes added to the MST overlay. Namely, the first MST node is "assigned" randomly, i.e., r_{i1}, Figure 3.3. Next, the algorithm iterates and adds new MST nodes to $\boldsymbol{r}_i$. Specifically, at each iteration all nodes in the tracking vector $\boldsymbol{r}_i$ are searched to find a new "unassigned" node from the set $\boldsymbol{v}_i$-$\boldsymbol{r}_i$ pursuant to a particular minimization strategy (akin to the star and bus overlays). Specifically, two strategies are tabled here:

Minimum Average Hop (MST-MAH): This scheme chooses MST nodes in order to minimize resource consumption. Namely, the scheme iterates (i.e., loop over index j, $j \geq 3$, Figure 3.3) to select an "unassigned" LAN node with the minimum hop count to a node in $\boldsymbol{r}_i$, i.e.,

$$r_{ij} = v_{ik^*} \text{ s.t. } \min_{m,k}(\text{hop}(r_{im}, v_{ik})), \qquad \text{Eq. (3.8)},$$

where $1 \leq m \leq j\text{-}1$, and $v_{ik} \notin \boldsymbol{r}_i$. Akin to the Bus-MAH scheme, this approach only uses static information to choose the next MST node, i.e., $O(|\boldsymbol{v}_i|(|\boldsymbol{v}_i|-1))=O(|\boldsymbol{V}|^2)$ shortest path computations needed prior to startup with $O(|\boldsymbol{v}_i|^2)=O(|\boldsymbol{V}|^2)$ storage overheads. However $O(|\boldsymbol{v}_i|(|\boldsymbol{v}_i|-1))=O(|\boldsymbol{V}|^2)$ run-time lookups are required to select all MST nodes, yielding an overall complexity of $O(|\boldsymbol{V}|^2)$.

Minimum Average Cost (MST-MAC): This scheme follows the same overall flow as MST-MAH, with the exception that hop counts are now replaced with minimum average costs as follows:

$$r_{ij} = v_{ik^*} \text{ s.t. } \min_{m,k}(\text{hop}(r_{im}, v_{ik})), \qquad \text{Eq. (3.9)},$$

where $1 \leq m \leq j\text{-}1$, $v_{ik} \notin \boldsymbol{r}_i$, and the cost() function is defined as in Eq.3.5. Since this scheme uses dynamic resource state, akin to Bus-MAC, it has higher compute complexity, i.e., $O(|\boldsymbol{v}_i|(|\boldsymbol{v}_i|-1))= O(|\boldsymbol{V}|^2)$ shortest path computations for bus node selection, yielding a total compute complexity of $O(|\boldsymbol{V}|^2 \cdot |\boldsymbol{V}| \log|\boldsymbol{V}|)= O(|\boldsymbol{V}|^3 \log|\boldsymbol{V}|)$.

3.3 Multi-Tiered LAN Group Provisioning

Carrier Ethernet LAN service users will demand flexible, multi-tiered survivability support. For example most "regular" users will suffice with partial recovery against single faults. Alternatively a subset of users may demand much more stringent 100% recovery, e.g., financial services, packet video transport, etc. To meet these requirements, the proposed framework provisions the LAN overlay connections (computed in Section 3.2) using inverse multiplexing and *tiered* protection and restoration algorithms. The overall aim here is to guarantee a minimum LAN throughput in the event of a single fault. To achieve this, a fractional protection factor, ρ ($0 \leq \rho \leq 1$), is used to specify a minimum *pre-provisioned* protection level for the LAN connection group. Namely a minimum level of ρx_i STS-1 units of dedicated protection capacity must be provisioned for all group connections. Consider the details.

3.3.1 Inverse Multiplexing Considerations

Inverse multiplexing facilitates multi-path routing of flows and within the context of a LAN overlay this concept can be applied to individual group connections. Namely, consider *i*-th LAN requesting x_i STS-1 units (mapped from Ethernet bandwidth equivalent). Here, each individual connection in the LAN group between node s_{ij} and d_{ij} will also require $x_{ij}=x_i$ STS-1 units, $0 \leq j \leq n_i$. In turn, this connection can be "resolved" into multiple "sub-connections", up to a maximum of $K \leq x_i$, as denoted by the inverse multiplexing factor. Although various policies are possible here, an "even" distribution approach is chosen in order to better distribute loads, akin to [10]. Specifically, consider integral division of x_{ij} by K yielding:

$$z = \left\lfloor \frac{x_{ij}}{K} \right\rfloor \qquad \text{Eq. (3.10),}$$

where the remainder term is given by:

$$y = x_{ij} - Kz \qquad \text{Eq. (3.11),}$$

and $0 < y < k$. For the special case of $x_{ij}=Kz$ (i.e., $r=0$), all requested sub-connections are sized at x_{ijk} STS-1 units, $1 \leq k \leq K$. However for the more general case of $y \neq 0$, the remainder term is simply distributed over the first r sub-connections. Hence the resultant generic expression for the individual capacity for the k-th requested sub-connection, x_{ijk}, in STS-1 increments is:

$$x_{ijk} = \begin{cases} z+1 & 1 \leq k \leq y \\ z & y < k \leq K \end{cases}, \quad \sum_j x_{ijk} = x_{ij} = x_i \qquad \text{Eq. (3.12)}$$

i.e., the first y connections may receive an extra STS-1 unit. Note that the above formulation assumes that the inverse multiplexing factor K is pre-specified, as will be common in most operational settings. The use of inverse multiplexing to improve LAN overlay resiliency over SONET/SDH networks is detailed next.

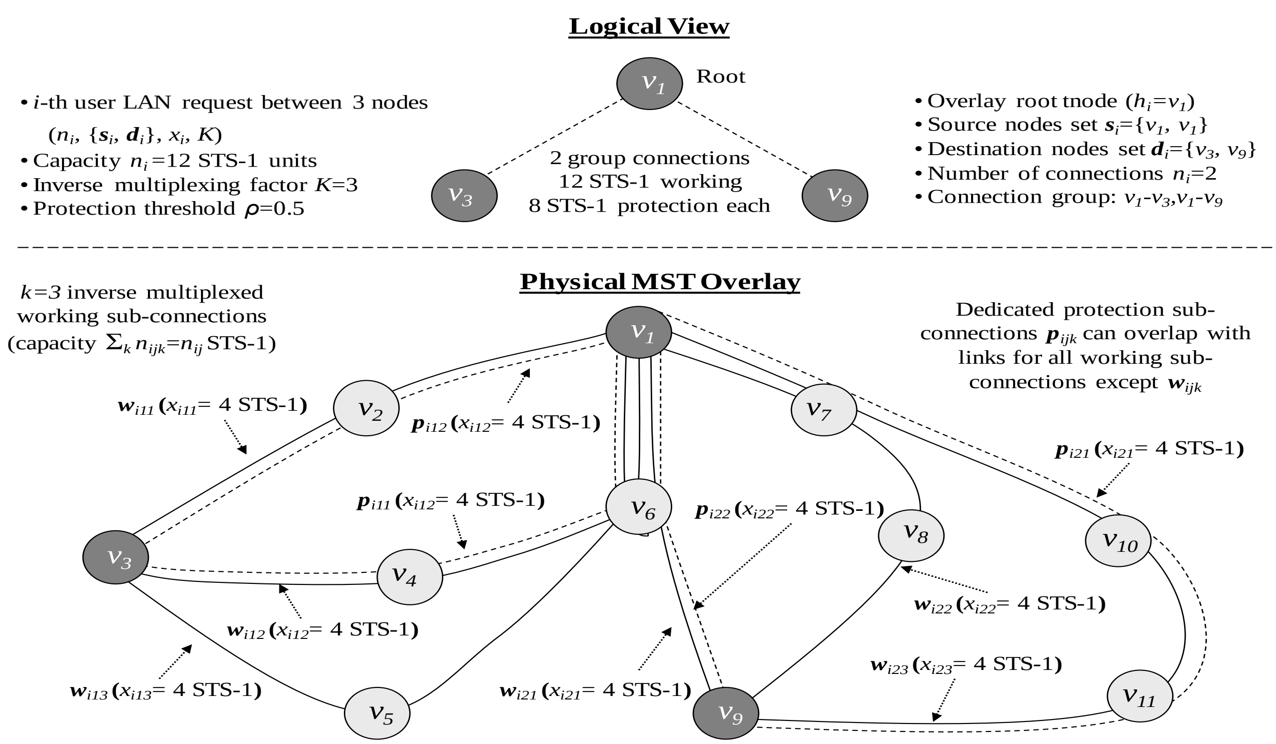

Figure 3.4: Tiered LAN scheme for sample MST overlay (one root, 2 leaves)

3.3.2 Path Computation and Protection Strategies

The LAN provisioning solution operates in two phases. First, all *working* overlay connections (sub-connections) are routed with x_i STS-1units of capacity each. Next, each of these connections is protected by provisioning a subset of its sub-connections with dedicated protection sub-connections, i.e., to achieve a minimum protection threshold of ρx_i STS-1 units. This approach ensures a "LAN-wide" throughput of at least ρx_i STS-1 units in the event of a single link failure.

This tiered protection concept is shown in Figure 3.4 for a the case of MST overlay for the *i*-th LAN request of 12 STS-1 units (approximately 600 Mb/s fractional Ethernet) between three nodes, v_1, v_3 and v_9. Here, there is one root node, v_1, connecting to two leaf nodes v_3 and v_9. (Note that similar diagrams can also be drawn for the bus overlays). The request is mapped using an inverse multiplexing factor of K=3 and a protection threshold of ρ=0.5 (50%). Assuming that the root node is selected as v_1, the LAN request is transformed into a connection group containing two bi-directional SONET/SDH connections, to the leaf nodes v_1-v_9 and v_1-v_9, respectively. In turn, each of these connections is inverse multiplexed into K=3 diversely-routed working sub-connections of 4 STS-1 units, e.g., routes $\boldsymbol{w}_{i11}$, $\boldsymbol{w}_{i12}$, and $\boldsymbol{w}_{i13}$ for LAN group connection v_1-v_3, Figure 3.4.

Now the protection threshold ρ mandates that each individual connection must have at least (0.5)·12=6 STS-1 units of protection capacity. This is achieved by using dedicated protection on a *per-sub-connection* basis, similar to [13]. Namely dedicated *link-disjoint* protection paths are computed for a minimal subset of working sub-connections until the desired threshold is achieved. Hence two protection sub-connections must be setup for each connection in Figure 3.4, i.e., 4 STS-1 units each in routes $\boldsymbol{p}_{i11}$ and $\boldsymbol{p}_{i13}$ to protect the working routes $\boldsymbol{w}_{i11}$ and

$\boldsymbol{w}_{i13}$ for the LAN connection $\boldsymbol{v}_1$-$\boldsymbol{v}_3$. Carefully note that since protection is done on a per-sub-connection basis, protection granularity is inversely proportional to the inverse multiplexing factor K. Hence it is possible for protection over-provisioning to occur for smaller values of K, as noted in [10]. For example in Figure 3.4 with K=3, a total of 8 STS-1 of protection capacity is reserved for each connection even though the threshold is 6 STS-1, i.e., 33% over-provisioning. However this inefficiency can be easily be resolved by appropriately "right-sizing" protection sub-connections. Overall, the above approach simplifies protection switchovers during link failures as it ensures equal-sized working and protection VCG members, i.e., no complex edge buffering is needed.

Now consider the actual working/protection provisioning algorithms for the LAN group connections (full pseudo code listings of which are presented in [20]). Here, the working phase first computes routes for all LAN connections/sub-connections. Namely, the algorithm makes a temporary copy of the network graph, $G'(\boldsymbol{V}, \boldsymbol{L})$, and then iterates to setup *working* routes for all n_i connections in the LAN using inverse multiplexing. Each connection is resolved into sub-connections using the above-described "even" distribution approach (Section 3.3.1), i.e., x_{ijk}, [25]. Next, a modified *successive* Dijkstra's shortest-path computation scheme is used to iteratively compute individual sub-connection route vectors, $\boldsymbol{w}_{ijk}$, for each requested sub-connection. Here if a sub-connection is successfully routed, its capacity is pruned along all route links in $G'(\boldsymbol{V}, \boldsymbol{L})$. Furthermore, the algorithm only proceeds to the next group connection if the current connection is fully routed otherwise the LAN request is dropped. This scheme places no restrictions upon link overlap between sub-connections and only *feasible* links with sufficient capacity are considered, i.e., $c_{jk} \geq x_i$.

Pending successful provisioning of all working group connections, the protection phase implements tiered (partial) protection. Here the associated algorithm uses the "left-over" capacity in $G'(\boldsymbol{V}, \boldsymbol{L})$ and implements similar steps as per the working connection routing stage. Namely, a running count of the aggregate "connection-level", *protection_capacity*, is maintained and this value is checked against the desired minimum protection threshold (ρx_i) after each successful protection sub-connection setup. If this threshold is exceeded, the LAN connection is deemed protected and the sub-connection protection paths, $\boldsymbol{p}_{ijk}$, are stored. Otherwise, the tiered LAN request is dropped. Since the maximum number of LAN group connections is $O(n_i)=O(|\boldsymbol{V}|^2)$, the working/protection provisioning algorithm is of $O(K|\boldsymbol{V}|^3 \log|\boldsymbol{V}|)$ complexity.

In addition, two different link cost "routing" metrics are used by the Dijkstra scheme when routing working/protection sub-connections, e.g., *hop count* and *cost* (as used in the overlay computation schemes, Section 3.2). Specifically, hop count routing assigns unity cost to all links and chooses the physically shortest feasible path. Conversely the minimum cost metric, Eq (3.5), distributes loads across lightly-loaded links and thereby increases multi-path routing diversity between individual sub-connections. Overall, these two strategies achieve a balance between resource minimization and multi-path diversity [13].

3.4. Post Fault Restoration

Finally, optional post-fault restoration of failed LAN connections (sub-connections) is also performed. Namely, all *non-protected* VCG members (e.g., $\boldsymbol{w}_{i13}$, Figure 3.4) traversing a failed link can be re-routed after failure notification by first pruning failed links on the network graph $G(\boldsymbol{V}, \boldsymbol{L})$. The goal here is to achieve *full* recovery for partially protected LAN requests, thereby enabling carriers to achieve improved service recovery rates for lower-priced offerings.

As such the framework is very generic and can readily incorporate regular protection (i.e., $K=1/\rho=1$) and regular restoration ($K=1/\rho=0$ with post-fault restoration).

CHAPTER 4

SIMULATION OVERVIEW

Ethernet-over-LAN performance is tested and evaluated using the *OPNET Modeler*^{TM} tool. *OPNET Modeler*^{TM} is the industry's leading discrete event simulator and is specialized for network research and development. Discrete event simulation mimics network systems by modeling each event in the system and generating user-defined sub-routines to handle them. *OPNET Modeler*^{TM} further introduces a graphical-based hierarchical strategy to organize the overall network. Namely, the hierarchy models entities from physical links to network node processes to sub-networks, see Appendix for more details.

4.1 Network Topologies

For testing purposes, two different mesh topologies are used. These networks are summarized in Table 4.1 and also presented in Figures 4.1and 4.2.

Topology	**Number of Nodes**	**Number of links**	**Nodes degrees**	**Model for connection Requests**
NSFNET	16	25	1.56	Random exponential Holding and arrival time
Regional topology	27	52	1.92	Random exponential Holding and arrival time

Table 4.1: Description of test topologies

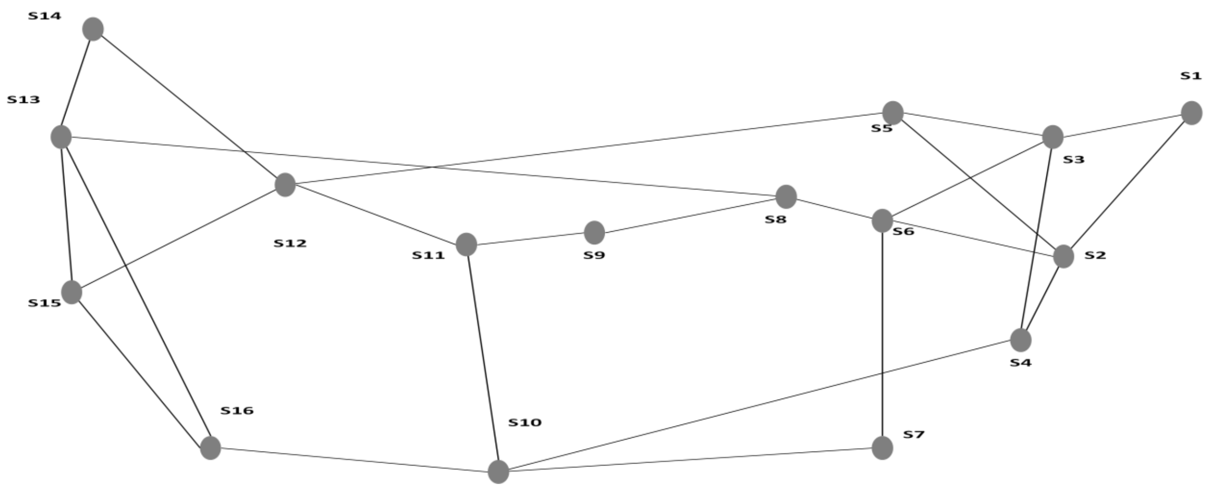

Figure 4.1: NSFNET core mesh topology (16 nodes, 25 links)

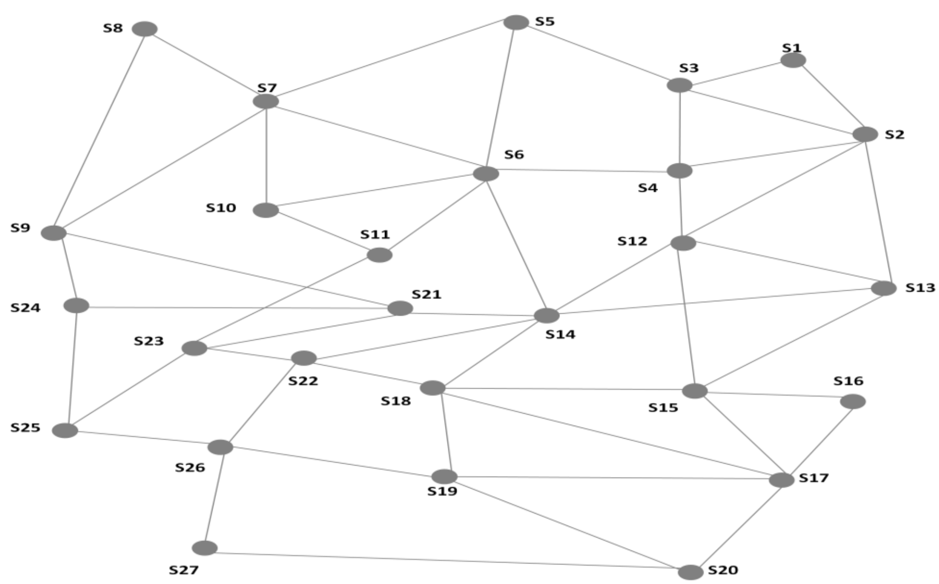

Figure 4.2: Regional network topology (27 nodes, 52 links)

Here the first network, Figure 4.1, is the ubiquitous NSFNET topology which comprises of 16 nodes and 25 links (50 bi-directional links). This network models larger backbone networks spanning across national distances. Meanwhile, the second one is a regional topology, shown in Figure 4.2 consisting of 27 nodes and 52 links (104 bi-directional) links. This network models much denser regional domains and is derived from a widely-used German network.

Overall, the respective node degrees for those two networks are 1.56 for NSFNET and 1.92 for the regional network. In general, higher node degrees will provide better connectivity for mesh provisioning (shortest-path) algorithms.

4.2 User Request Modelling

The above network topologies are tested using randomly generated user LAN requests. First, the size of the LAN connection groups is uniformly varied between 3 and 5 node sites, i.e., these nodes are serially selected from the network nodes in a uniform manner. Meanwhile, the actual LAN request sizes are varied uniformly from 200 Mbps to 1 Gbps in increments of 200 Mbps. This is expressly done to model fractional Ethernet demands. Furthermore, the average request holding time is fixed to 600 seconds (relative, scaled) and the commensurate request inter arrival rate is varied as per the loading. Here, the choice of scaled (smaller) holding time is done in order to prevent floating point overflows during simulation. As such, the associated requests inter-arrival times will also be scaled to yield a loading figure, see Section 4.3.

4.3 Performance Evaluation Metrics

The performance evaluation of the Ethernet LAN provisioning algorithms outlined in the Chapter 3 is done using various metrics. These are also now detailed further.

4.3.1 Blocking Probability

Foremost, LAN request blocking probability is used to gauge the effectiveness of the different LAN provisioning schemes. Namely, blocking probabilities is defined as the ratio of the total number of failed requests to the total number of requests made:

$$P_b = \frac{\text{Number of failed LAN requests}}{\text{Total number of LAN requests}} \qquad \text{Eq (4.1)}$$

Clearly, it is important to generate a large number of requests (i.e., on the order of 100,000 or more) in order to smooth out inconsistencies and obtain accurate values. Note that confidence intervals can also be computed here, but are omitted for simplicity's sake.

4.3.2 Network Load

Next, a loading metric is defined to measure network congestion. Now the ubiquitous Erlang load metric is commonly used in circuit switched networks to measure the network (connections) load. However, the original Erlang metric is defined for point-to-point phone cell traffic. As such, it must be modified appropriately to handle LAN requests with multiple connection groups. Hence the basic Erlang metric is first scaled by the number of connections generated by the request. In addition, the load must be further scaled by the average request size of 500 Mb/s. Hence the modified Erlang is given by:

$$\text{Modified Erlang loading (star)} = \frac{1}{3}\sum_{3}^{5}(i-1)\cdot\frac{x_{avg}\cdot T_{hold}}{IAT} \qquad \text{Eq (4.2)},$$

where x_{avg} is the average request size and T_{hold} is the average holding time (600 seconds), and IAT is the mean inter-arrival time (seconds).

CHAPTER 5

PERFORMANCE EVALUATION

The performance evaluation of the proposed multipoint Ethernet LAN provisioning scheme is now studied using simulation analysis. Namely, these tests are done using the previously-introduced network topologies and input parameters from Chapter 4 and the findings are now detailed.

5.1 NSFNET Topology

Initial results are first presented for the NSFNET topology in Figure 4.1

5.1.1 Non-Protected LAN Performance

To start out, the performance of non-protected LAN requests (i.e., $\rho = 0$) is tested for random LAN request sizes between 3-5 nodes. Specifically, the goal here is to measure the carried network load for increased inverse multiplexing factors for a given nominal request blocking rate of 2%. These types of tests are very important as they determine the true "load carrying capacity" of the network at a given "low-load" operating point. The results are shown in the Figures 5.1 and 5.2. Foremost, it is seen that the MST overlay yields slightly better results than the bus overlay, e.g., highest peak carried load for bus is about 236.06 modified Erlang whereas with MST it is 244.06. In addition, it is also seen that minimum cost-based topology selection (i.e., Bus-MAC, MST-MAC) coupled with load-balancing routing yields the highest carried loads for both overlay schemes. Moreover all "intelligent" overlay schemes outperform random selection by a good margin. Perhaps the most important finding from the above two graphs is the notable gain in carried load with increased levels of inverse multiplexing, i.e., about 20-30% higher carried load for K=5 versus K=1.

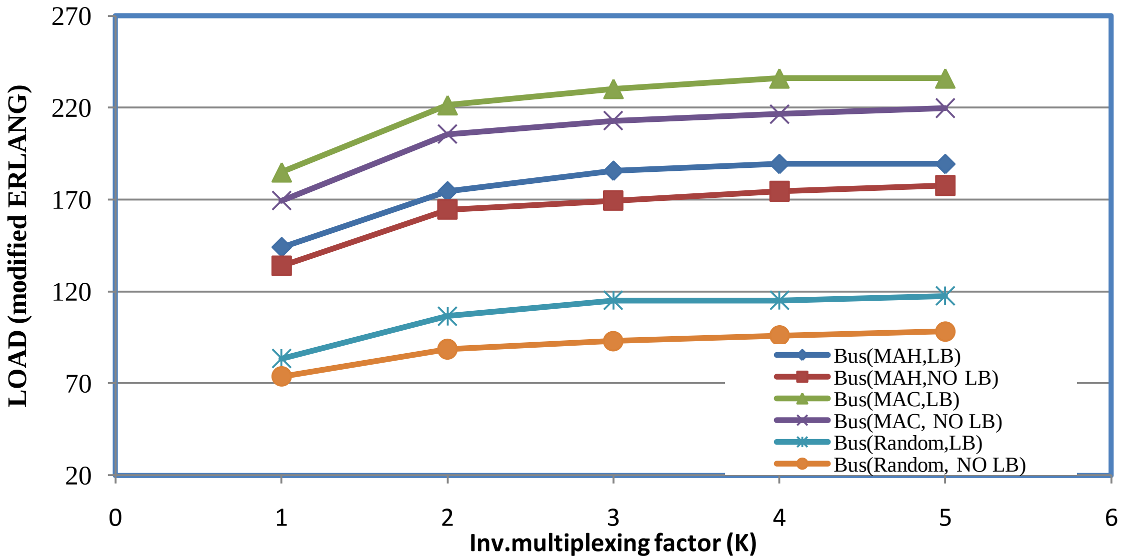

Figure 5.1: Carried load for 2% LAN request blocking, ρ = 0 for Bus overlays

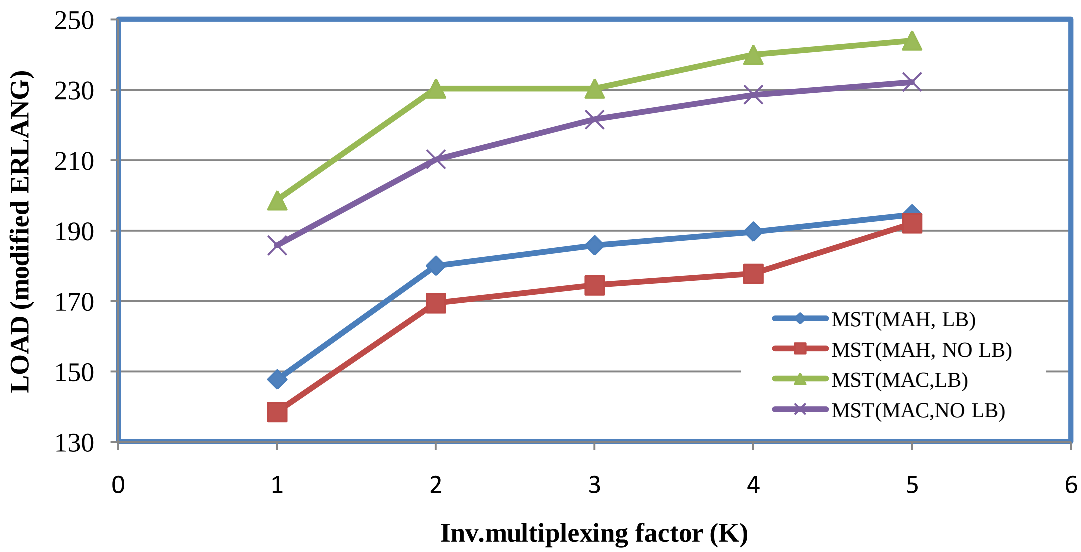

Figure 5.2: Carried load for 2% LAN request blocking, ρ=0 for MST overlays.

5.1.2 Full/Partial Protection LAN Performance

Next the performance of partial VCAT protection is studied by plotting results for varying protection thresholds (ρ=0, 0.25, 0.5) and inverse multiplexing factors (K= 1, 2, 4). Specifically sample results are shown in Figure 5.3 for bus overlays with minimum cost selection i.e., Bus-MAC, as this was shown to give better load performance in Section 5.1.1. The overall findings here clearly show that increased protection factors can significantly increase LAN request blocking rates. For example, blocking rates with full protection are over an order magnitude higher than those with partial 25% protection. Similar results are also seen with MST-based overlays (not shown). Meanwhile the individual blocking rates for different LAN sizes are also plotted for the case of K=4 and ρ=0.25 in the Figure 5.4. This plot shows that the larger 5 node LAN requests experience almost an order magnitude higher blocking than smaller 3 node LAN requests, i.e., owing to the increased number of connections in a group. Regardless, the intelligent MAC selection still yields less than 1% blocking for 5-node LAN sizes at low-mid ranges.

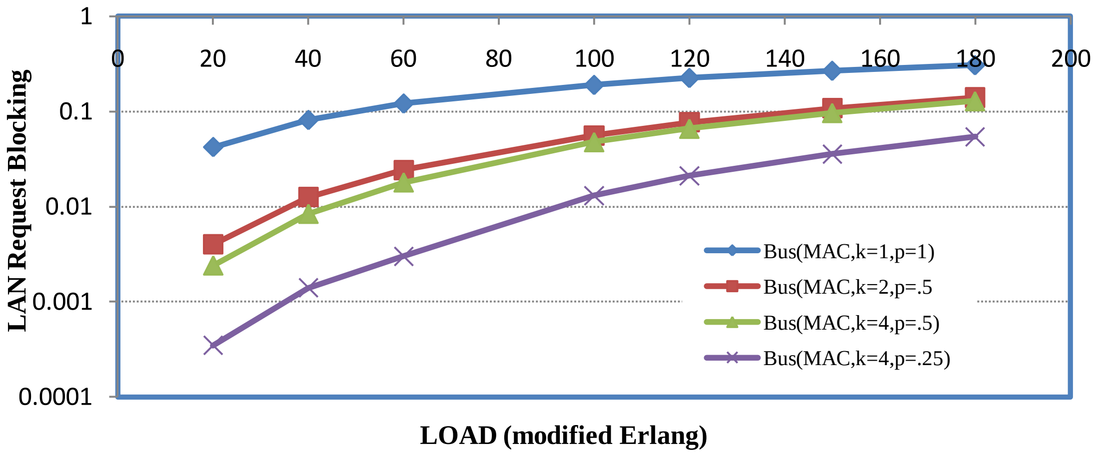

Figure5.3: LAN blocking for Bus overlays: MAC with varying *K*, *ρ*.

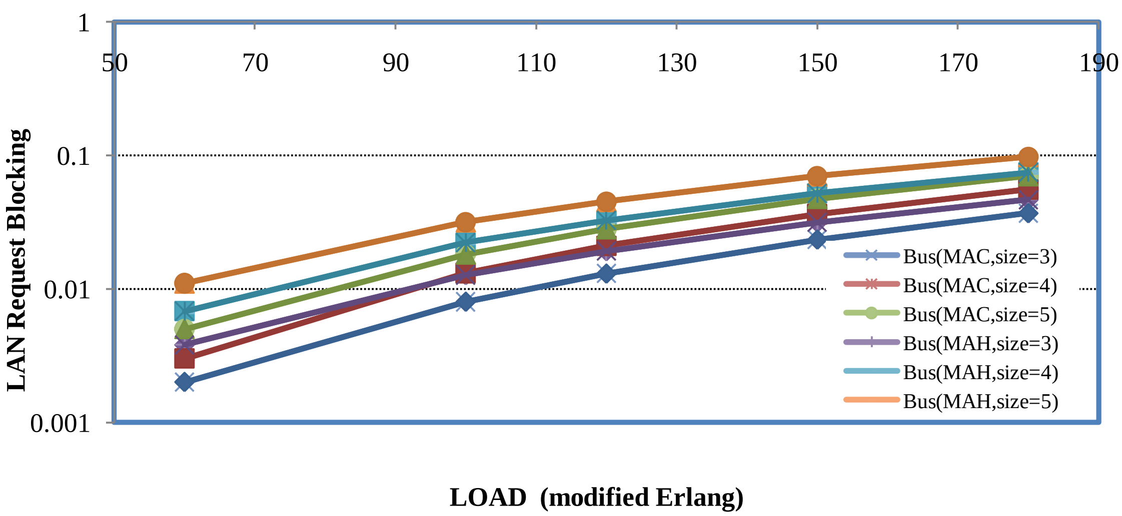

Figure 5.4: LAN blocking for Bus overlays: *K* =4, *ρ*=0.25.

5.1.3 Post-Fault Restoration Performance

Another important aspect of LAN operation is post-fault recovery and restoration. Namely, this deals with restoring the LAN connectivity back to a full capacity after a link failure, i.e., recompiling new routes for failed sub-connections. Here restoration of non-protected VCAT sub-connections is calculated by measuring *recovery rates* for higher carrier loads. Clearly this only applies for the case of partial/no-protection, i.e., ρ not equal to unity. Along these lines, LAN restoration rate is measured for increased loads up to 20% blocking. Specifically in Figures 5.5 and 5.6, the restoration rate is defined as the percentage of partially-protected LAN's which gain full throughput recovery after experiencing at least one sub-connection failure.

The results reveal some interesting findings. First of all, even though ρ=1 protection provides full recovery, the carried loads are much lower. More importantly, the notion that increased levels of over-provisioning protection result in a higher level of post fault restoration is not true here, i.e., consider an instance from results where K=4 and ρ=0 attains higher recovery than ρ=0.25 and ρ=0.5. In addition, the maximum carried loads with respect to these higher recovery rates are significantly higher, i.e., by over 100%.

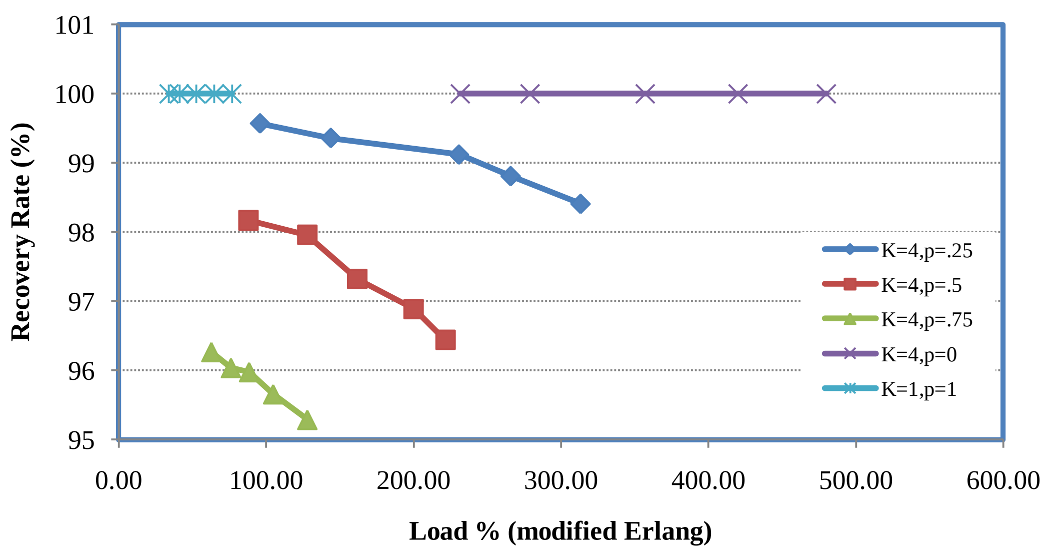

Figure 5.5: Post-fault restoration for Bus Overlay with MAC routing

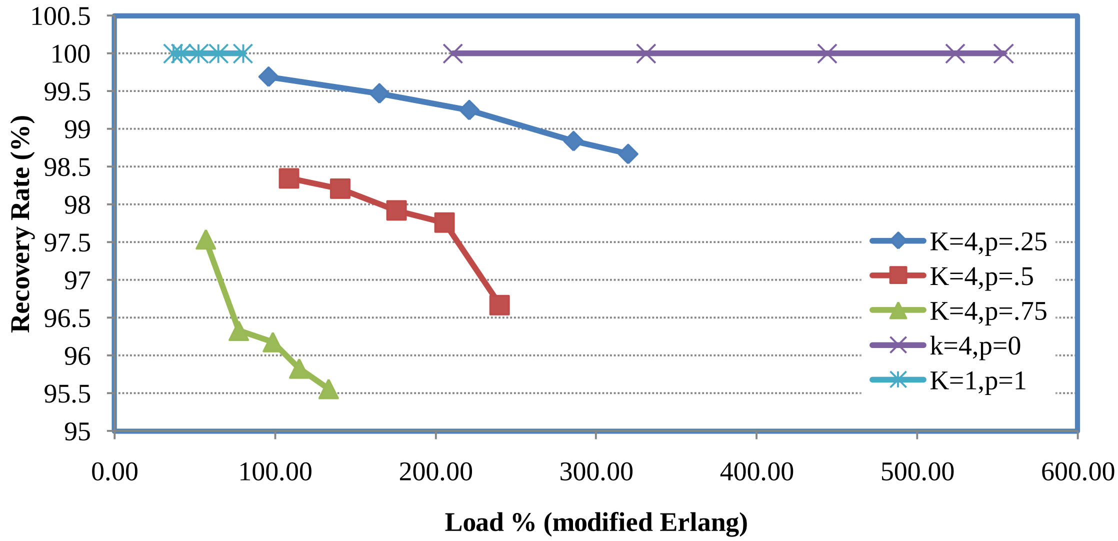

Figure 5.6: Post-fault LAN restoration for MST overlay w. load-balancing (MAC) routing

5.2 Regional Topology

Next, performance results are presented for the higher density regional German topology shown in Figure 4.2.

5.2.1 Non-Protected LAN Performance

The carried load tests of Section 5.1.1 are now repeated for the regional topology for non-protected demands ($\rho = 0$) and the plots shown in the Figures 5.7 and 5.8. From these graphs, it is clear that again the MST overlay slightly outperforms the bus overlay, i.e., peak carried load of 488 modified Erlang versus 553 modified Erlang. Also for both overlay schemes, the highest respective carrier loads are attained when minimum cost topology selection is coupled with the load balancing routing metric. Again, increased inverse multiplexing levels also give sizeable gains in carried loads, i.e., over 30% from *K*=1 to *K* =5.

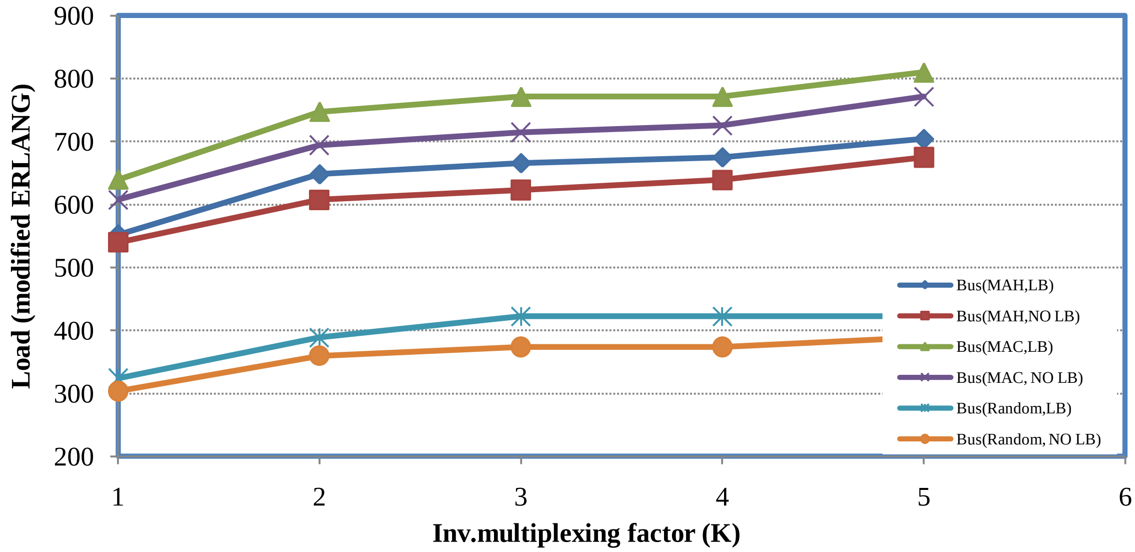

Figure 5.7. Carried load for 2% LAN request blocking, ρ=0 for Bus overlays

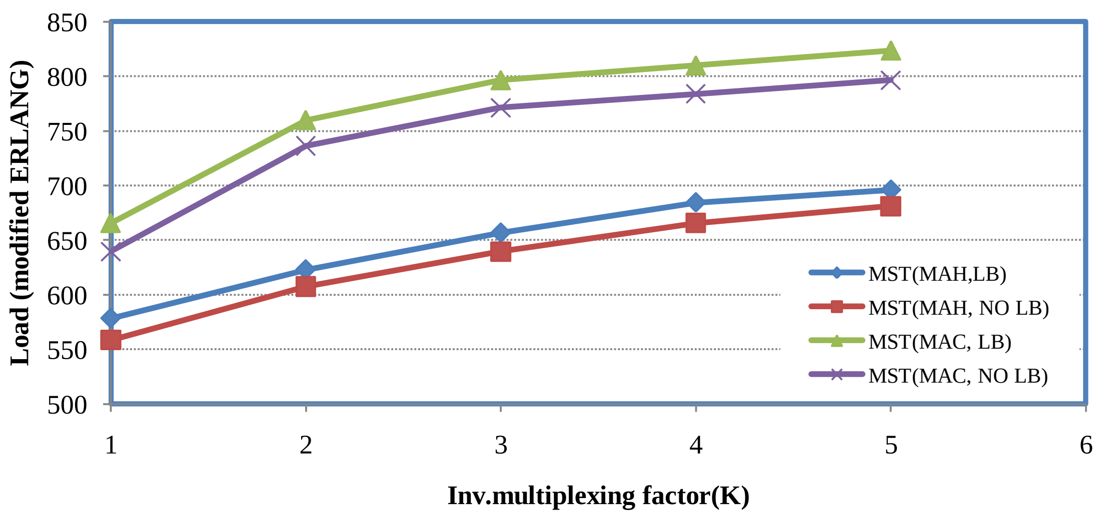

Figure 5.8 Carried load for 2% LAN request blocking, ρ=0 for MST overlays

5.2.2 Full/Partial Protection LAN Performance services

Next, the performance of partial VCAT protection services is tested for the regional topology and the results shown in Figure 5.9 and 5.10. Again it is observed that blocking performance with full protection is over an order magnitude higher than that with partial 25% protection. In addition, the individual blocking rates for different LAN sizes for K =4 and ρ=0.25 show that the larger 5 node LAN requests experience close to order magnitude higher blocking than smaller 3 node LAN requests. This discrepancy is clearly due to the added challenges in routing a larger number of working connections. However, owing to the denser connectivity of this regional network, larger 5 node LAN sizes do well, experiencing less than 1% blocking.

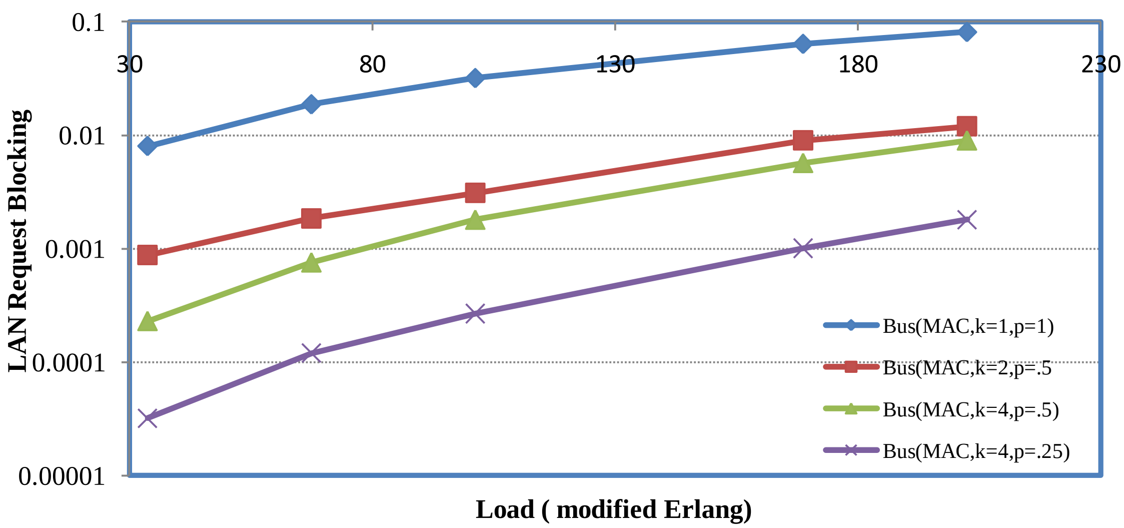

Figure 5.9 LAN blocking for Bus overlays: MAC with varying K, ρ

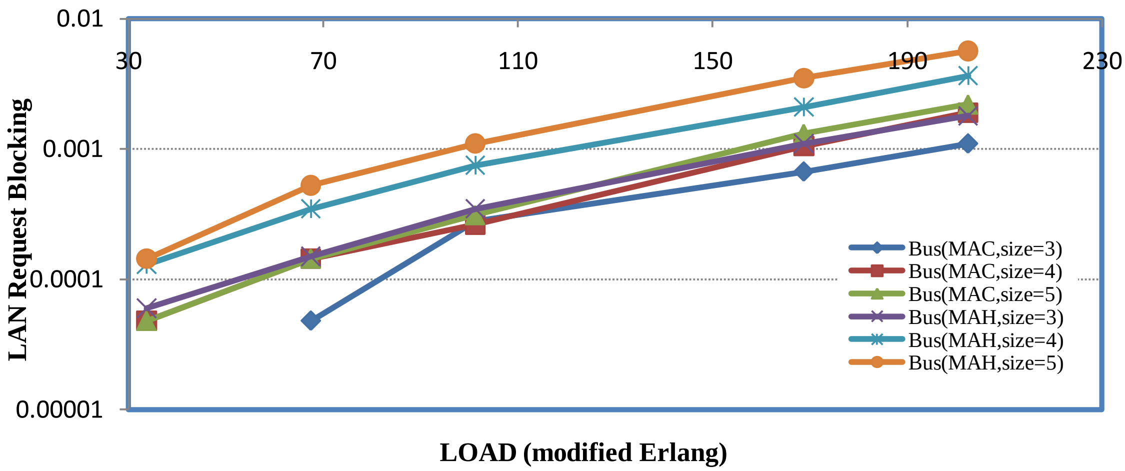

Figure 5.10 LAN blocking for Bus overlays: K=4, ρ=0.25.

5.2.3 Post-Fault Restoration Performance

Finally, post fault restorations tests are also repeated for the regional German topology. Here, the resulting graphs for both Bus and MST overlays are presented in Figures 5.11 and 5.12. These results also exhibit the same interesting features which were noted in the case of the NSFNET network i.e., Section 5.1.3. In addition, it is also observed that the regional topology yields higher recovery rates and respective carried loads than NSFNET topology for same protection factors and inverse multiplexing levels. Indeed, this is due to higher nodal interconnectivity levels of the former topology. Overall, these results show that over 97% of single link failures can be recovered fully with the aid of post-fault sub-connection restoration. Overall, these gains are very viable route for network service providers.

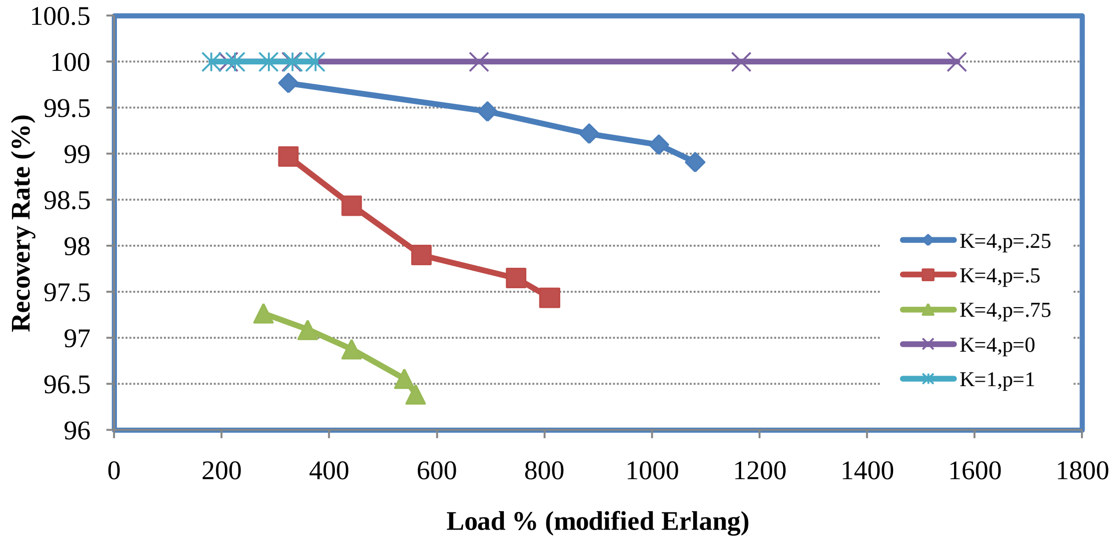

Figure 5.11: LAN restoration for Bus overlay and load-balancing (MAC) routing

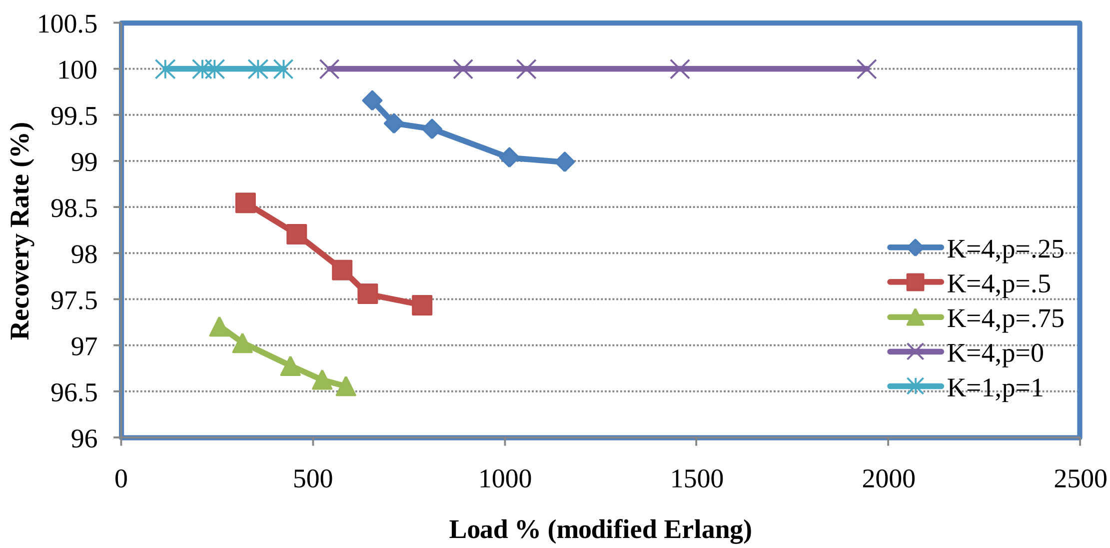

Figure 5.12: LAN restoration for MST overlay and load-balancing (MAC) routing

CHAPTER 6

CONCLUSIONS AND FUTURE RECOMMENDATIONS

Next-generation Carrier Ethernet services are seeing rapid traction in the business sphere. As Ethernet LAN interface speeds continue to scale into the multi-gigabit range, there is strong interest in extending Ethernet connectivity beyond the LAN realm and across larger geographic distances. To address these needs, the MEF has defined various new service standards for extending both point-to-point (i.e., private line) and multi-point Ethernet services over metro and wide-area networking domains.

In light of the above developments, the main focus of this thesis is to investigate the provisioning of robust, survivable multi-point Ethernet services across optical backbone networks comprising of advanced SONET/SDH technologies. The objective is to achieve *multi-tiered* capabilities to support a full spectrum of services from "best-effort" non-protected to highly stringent mission-critical offerings. Specifically, this is achieved by exploiting the inverse multiplexing capabilities of modern SONET/SDH systems, i.e., multi-path routing. In addition, novel graph-theoretic bus and MST-based connection overlay schemes are also introduced to improve LAN request provisioning efficiencies. The overall performance of the proposed schemes is then studied using the *OPNET Modeler*^{TM} discrete event simulation tool.

Conclusions

This research has yielded several key findings. Foremost, the application of intelligent connection overlay designs—particularly those using dynamic resource state—are shown to yield sizeable performance gains. For example, both the bus and MST overlays give the lowest blocking and/or highest carried load when coupled with minimum cost-based topology designs. As a result, these improved techniques will allow carriers to support larger numbers of LAN groups, increasing the revenue base.

Next, it is also seen that increased demand splitting, i.e., inverse multiplexing levels, have a very sizeable impact on network performance for both overlay types. For example, typical results at operational blocking regimes (i.e., about 2%) show about 30-50% higher carried loads when demand splitting is increased by a factor of five, e.g., K=5. Expectedly, these gains only occur when using load balancing of the sub-connections, i.e., as loads are more evenly distributed across all links. Moreover, inverse multiplexing gains are most evident with increased levels of network connectivity, i.e., node degree.

Finally, partial protection is shown to yield notably higher carried load as fewer backup resources are required, i.e., $0<\rho<1$. In particular, some interesting facts are observed for the case of post fault recovery and restoration which is another important aspect of LAN operation. In particular, unprotected LAN entities, i.e., ρ=0, are shown to achieve full (100%) post-fault recovery in almost all scenarios, owing to their relatively lower resource usages. In many cases, the carried loads for equivalent blocking are significantly higher than those with partial protection as well. Moreover, many of these findings are consistently observed over a full range of tested topologies.

Future Work Directions

This thesis builds upon earlier work on basic mesh and star/hub overlays [20] for multipoint Carrier Ethernet services by developing more advanced bus and MST-based schemes. However, all of these efforts (including this thesis) have only focused on dedicated protection of working connection groups. It is envisioned here that more advanced shared protection schemes can also be designed to allow sharing of backup resources between connections/sub-connections in LAN and outside a LAN. These techniques are expected to yield much higher levels of resource efficiency and can be investigated in the future.

REFERENCES

[1] "Technology Predictions TMT Trends 2009 - Part One", Deloitte Technology, Media & Telecommunications Industry Group article, January 2009, article available at: http://www.deloitte.com/assets/Dcom-Norway/Local%20Assets/Documents/TMT-Predictions-technology(3).pdf

[2] G. Bernstein, *et al*, *Optical Network Control: Architectures, Standards, Protocols*, Addison Wesley, Boston, 2003.

[3] E. Hernandez-Valencia, "Hybrid Transport Solutions for TDM/Data Networking Solutions," *IEEE Communications Magazine*, Vol. 40, No. 5, May 2002, pp. 104-112.

[4] A. Kasim (Editor), *Delivering Carrier Ethernet: Extending Ethernet Beyond the LAN*, McGraw Hill Publishers, November 2007.

[5] S. Acharya, *et al*, "PESO: Low Overhead Protection for Ethernet Over SONET Transport," *IEEE Infocom 2004*, Hong Kong, March 2004.

[6] N. Ghani, S. Park, A. Shami, C. Assi, K. Atthuru, B. J. Ayeleso, "Multi-Tiered Services in Next Generation SONET/SDH Networks," *IEEE GLOBECOM 2006*, Istanbul, Turkey, June 2006.

[7] L. Fang, *et al*, "The Evolution of Carrier Ethernet Services—Requirements and Deployment Case Studies," *IEEE Communications Magazine*, Vol. 46, No. 3, March 2008, pp. 69-76.

[8] N. Ghani, *et al*, "On IP-WDM Integration: A Retrospective," *IEEE Communications Magazine*, Vol. 41, No. 9, Sept. 2003, pp. 42-45.

[9] D. Cavnedish, *et al*, “New Transport Solutions for Next-Generation SONET/SDH Systems,” *IEEE Communications Magazine*, Vol. 40, No. 5, May 2002, pp. 80-87.

[10] K. Zhu, J. Zhang, B. Mukherjee, “Ethernet-over SONET (EoS) over WDM in Optical Wide-Area Networks (WANs): Benefits and Challenges,” *Photonic Network Communications*, Vol. 10, No. 1, January 2005, pp. 107-118.

[11] N. Ghani. J. Pan, X. Cheng, “Metropolitan Optical Networks,” *Optical Fiber Telecommunications IV,* Volume B (Editors I. Kamivov, T. Li), Academic Press, March 2002, pp. 329- 403.

[12] P. Bonenfant, A. Moral, “Generic Framing Procedure (GFP): The Catalyst for Efficient Data over Transport,” *IEEE Communications Magazine*, Vol. 40, No. 5, May 2002, pp. 72-79.

[13] N. Ghani, S. Park, “Multi-Tiered Service Survivability in Next-Generation SONET/SDH Networks,” *Photonic Network Communications*, Vol. 13, No. 1, January 2007, pp. 79-92.

[14] C.Ou, *et al*, “Survivable Virtual Concatenation for Data Over SONET/SDH In Optical Transport Networks”, *IEEE/ACM Transactions on Networking*, Vol. 14, No. 1, February 2006, pp. 218-231.

[15] G. Bernstein, *et al*, "VCAT/LCAS in a Clamshell," *IEEE Communications Magazine (standards report)*, Vol. 44, No. 5, May 2006, pp. 34-36.

[16] R. Roy, *et al*, “Degraded-Service-Aware Multipath Provisioning in Telecom Mesh Networks,” *IEEE/OSA OFC 2008*, San Diego, CA, February 2008.

[17] D. Zhou, S. Subramaniam, “Survivability in Optical Networks,” *IEEE Network Magazine*, Vol. 14, No. 6, November/December 2000, pp. 16-23.

[18] R. Roy, B. Mukherjee, "Degraded-Service-Aware Multipath Provisioning in Telecom Mesh Networks," *IEEE/OSA OFC 2008*, San Diego, CA, February 2008.

[19] S. Rai, *et al*, "Reliable Multipath Provisioning for High-Capacity Backbone Mesh Networks," *IEEE/ACM Transactions on Networking*, Vol. 15, No.4, August 2007, pp. 803-812.

[20] C. Xie, N. Ghani, *et al* "Multi-Point Ethernet Over Next-Generation SONET/SDH", *IEEE ICC 2009*, Dresden, Germany, June 2006.

[21] "Metro Ethernet Services-A Technical Overview," Metro Ethernet Forum, available at http://www.metroethernetforum.org .

[22] L. Fang, *et al*, "The Evolution of Carrier Ethernet Services--Requirements and Deployment Case Studies," *IEEE Communications Magazine*, Vol. 46, No. 3, March 2008, pp. 69-76.

[23] D. Andersen, *et al*, "Resilient Overlay Networks," *ACM Symposium on Operating System Principles* (SOSP),Banff, Canada, October 2001, pp. 131-185.

[24] R. C. Prim, "Shortest Connection Networks and Some Generalizations," *Bell System Technical Journal*, Vol. 36, 1957, pp. 1389–1401

[25] Z. Duan, et al, "Service Overlay Networks: SLAs, QoS, and Bandwidth Provisioning," *IEEE/ACM Transactions on Networking*, Vol. 11, No. 6, December 2003, pp. 870-883.

APPENDIX A

PERFORMANCE EVALUATION

Discrete event simulation tools model system behaviors as a chronological sequence of events, where each event occurs at an instant and marks a change in the system state. As such these offerings provide a powerful means to model and simulate complex real-time systems which may otherwise be difficult to analyze in a closed form manner. Now most network simulators are built using general purpose programming languages such as C/C++. However, a range of complete packages have also been developed to provide more modularized code blocks and model libraries, i.e., simulation languages such as *Simula*, *Siman*, and full packages such as *OPNET Modeler*^{TM}, *NS2*, *OMNET++*, *GLoMoSim*, etc. For the purpose of this research, the *OPNET Modeler*^{TM} solution is chosen as it offers the most comprehensive development environment. Some further details on this package are now presented.

A.1 Overview

OPNET Modeler^{TM} is a commercially-developed software product for simulation and modeling of network applications, protocols, and technologies. It includes a vast library of communication device, medium, and protocol process models written in a programming language called *Proto-C*. Namely, *Proto-C* is a combination of the C/C+ programming language and graphical state transition diagrams. The overall *OPNET Modeler*^{TM} design uses a series of hierarchically related editors that directly parallel the structure of actual networks. Specifically, this modeling structure includes the network/sub-network, and its associated nodes and links, and process models to control the nodes. These are now detailed further.

A.2 Project Editor

The main staging area where a network simulation is created is called the Project Editor. In a project, network models are created with the aid of models from the standard (or user-generated) library. Namely, specialized editors are provided to allow users to drag-and-drop node and link models and also define associated statistics probes to capture key run-time parameters, see Figure A.1.

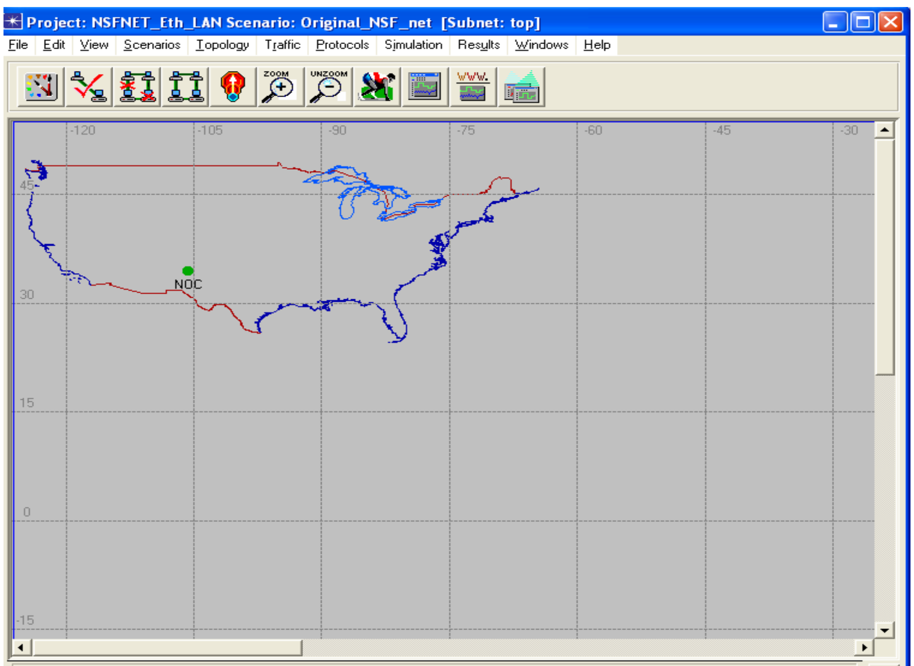

Figure A.1: Network model built in Project Editor

Now within a project, a network model may further consist of sub-networks and nodes connected by point-to-point, bus, or radio links depending on the type of network being modeled. Furthermore, sub-networks, nodes, and links can be placed within sub-networks, which can then be treated as single objects in the network model. As such, this capability is useful for separating the network diagram into manageable pieces and provides a quick way of duplicating groups of nodes and links.

A.3 Node Editor

Meanwhile, the Node Editor is used to define models for network element devices. These node models are in turn used to create node instances within networks in the Project Editor, Figure A.2. Now internally, *OPNET Modeler™* node models have a modular structure and are built by connecting process modules (detailed next) with packet streams and links. Namely, these connections between modules allow packets and status information to be exchanged between modules. Overall, each module placed in a node serves a specific purpose, such as generating packets, queuing packets, processing packets, or transmitting and receiving packets.

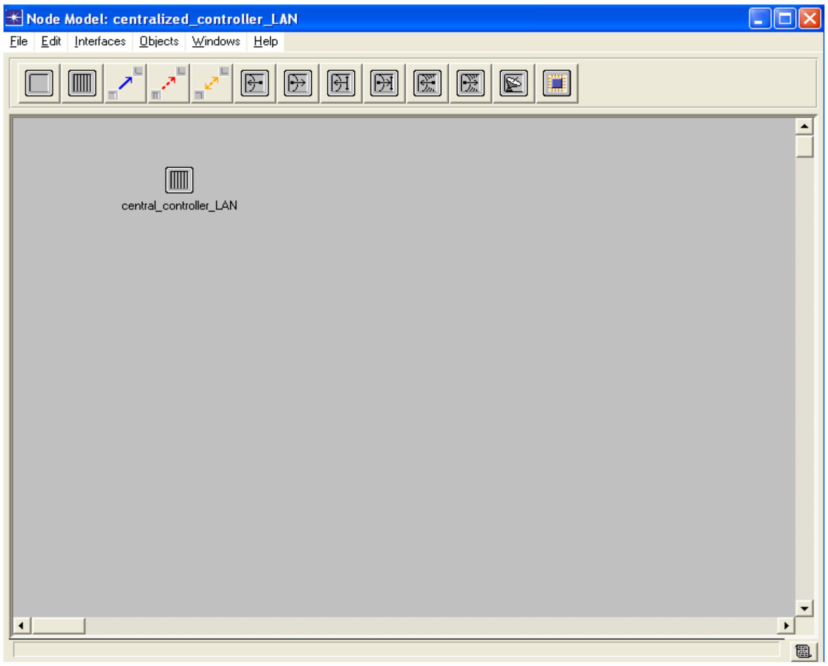

Figure A.2: Node editor

A.3 Process Model Editor

The underlying functionality of a node model is controlled by process models, built using the Process Model Editor. This editor uses a *finite state machine* (FSM) approach to describe the protocols at all levels of detail. Namely, state and transition diagrams graphically represent the process behaviors, where active states are changed in relation to incoming events. Furthermore, each state process contains C/C++ code to detail specific actions. Now many libraries can be used here for programming these module states and users can also write their own detailed code in C/C++ (as is done for this research work).

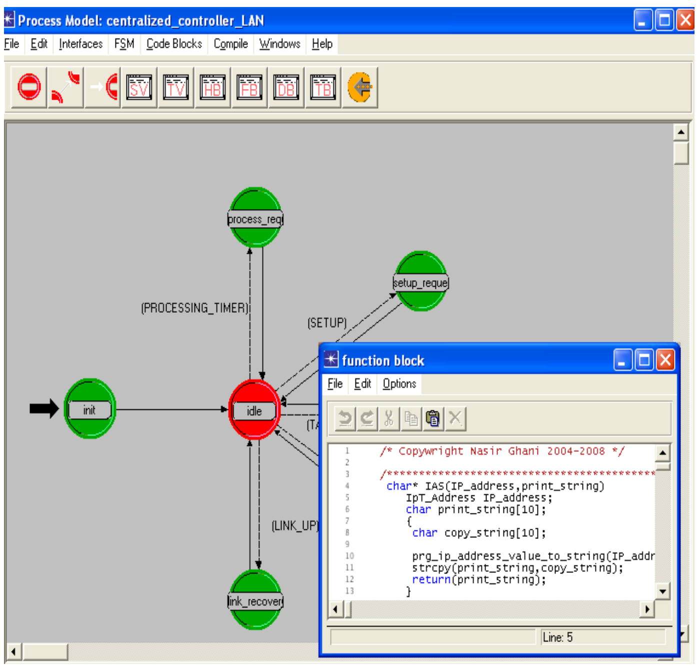

Figure A.3: Process model developed for NGS simulation

The overall process control model for the NGS multipoint provisioning controller developed for this research is shown in Figure A.3. This figure also provides a snapshot of the detailed *C/C++* function block which implements the main topology overlay computation and inverse multiplexing provisioning/survivability functions. Furthermore, states are defined to handle connection request arrivals and departures as well as link failures and recoveries.

A.4 Simulation Sequence Editor

The Simulation Sequence Editor allows users to specify additional simulation constraints, see Figure A.4. These sequences are represented by simulation icons, which contain a set of attributes that control the simulation's run-time characteristics, e.g., including initial start-up seed values, run-time durations, etc.

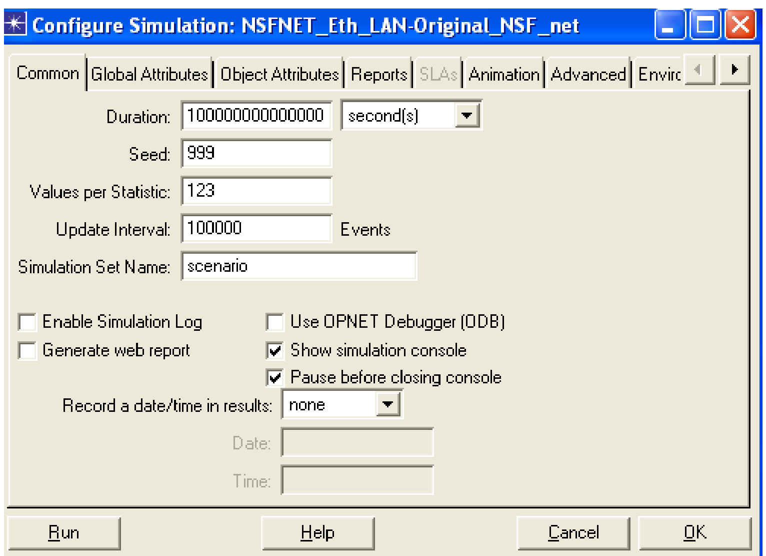

Figure A.4: The simulation sequence editor