AVAILABILITY-BASED RESILIENT VIRTUAL OPTICAL NETWORK MAPPING

by

Jian Kong

APPROVED BY SUPERVISORY COMMITTEE:

Jason Jue, Chair

Andrea Fumagalli

András Faragó

Jorge A. Cobb

Copyright © 2018

Jian Kong

All rights reserved

To my parents and my family

AVAILABILITY-BASED RESILIENT VIRTUAL OPTICAL NETWORK MAPPING

by

JIAN KONG, BS, MS

DISSERTATION

Presented to the Faculty of

The University of Texas at Dallas

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY IN

TELECOMMUNICATIONS ENGINEERING

THE UNIVERSITY OF TEXAS AT DALLAS

December 2018

ACKNOWLEDGMENTS

My sincere gratitude goes to my advisor, Dr. Jason Jue, who not only guided me throughout my research and study but also guided my attitude towards my career and life. His comprehensive knowledge, excellent perception, and great attention to detail help me overcome many research challenges. I am also highly motived by his great passion for research. Without his constant encouragement and financial support, this dissertation would not have been possible.

I would like to express my sincere gratitude to my committee members, Dr. Andrea Fumagalli, Dr. Andras Farago and Dr. Jorge Cobb for their guidance and insightful comments.

I am also grateful to my collaborators, Dr. Inwoong Kim, Dr. Xi Wang, Dr. Qiong Zhang, Dr. Hakki C. Cankaya, Dr. Weisheng Xie, and Mr. Tadashi Ikeuchi for their patience and insightful comments. I would like to express my gratitude to Dr. Xi Wang, Dr. Nannan Wang and Dr. Paparao Palacharla for their collaboration and valuable guidance during my summer internship at Fujitsu Labs of America.

I would like to thank all the members in the UTD Advanced Networks Research Lab (ANRL) for their help, cooperation and encouragement. Many thanks to Dr. Sanjin Hong, Dr. Fahim Khandaker, Dr. Nannan Wang, Dr. Ruijie Zhu, Dr. Xingbin Yin, Riti Gour, Ashkan Yousefpour, Genya Ishigaki and Qizhe Guo.

I especially thank my parents for their unconditional love, patience, and encouragement. I also thank my wife, Huijie, for her continuous understanding and encouragement during my study. Without their support, I would not have been able to complete this work.

November 2018

AVAILABILITY-BASED RESILIENT VIRTUAL OPTICAL NETWORK MAPPING

Jian Kong, PhD
The University of Texas at Dallas, 2018

Supervising Professor: Jason Jue, Chair

With the growth of bandwidth-demanding and dynamic network services such as cloud computing, big data, and machine learning applications, a key challenge for network operators is how to efficiently deploy the dynamic network infrastructures to support these various services and applications. Network virtualization provides a feasible way to share computing resources and network resources among virtual network requests. Availability is a key factor in the service level agreements (SLAs) when a network service provider provides the computing and network resources to the customers. A virtual optical network (VON) request may consist of many services which depend on each other, and the failures of single or multiple network elements may lead to the inability of the VON to support these services. Therefore, it is important to evaluate a VON based on its availability, where a VON is defined to be available only when all of its virtual nodes and virtual links are available at the same time.

We first study the availability-guaranteed VON mapping problem over the physical optical network where the network elements suffer independent failures. The objective is to minimize the total link cost of a VON mapping while guaranteeing the VONs availability requirement. Based on the proposed availability analysis model, we propose an ADAPTIVE algorithm using a new cost metric to provide selective dedicated protection. We also provide shared backup path protection for VONs by finding the maximum availability VON mapping based on a group node-weighted Steiner tree solution. We then address high-availability VON

mapping over the physical optical network where its physical links may suffer correlated failures. We propose the Group Steiner tree problem under correlated cost (GSTP-CC) to determine the maximum availability of the VON mapping without any protection, and a dedicated protection mechanism is provided to meet the availability of each VON, if needed. Finally, we focus on availability-guaranteed service function chain (SFC) mapping in inter-datacenter networks. We propose different coordinated protection mechanisms that adopt both backup path protection in the network and VNF replicas at nodes to guarantee a SFCs availability while minimizing the total computing and link resources.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Virtual optical network

With the growth of bandwidth-demanding, dynamic network services such as cloud/fog computing, big data, and machine learning applications, a key challenge for network operators is how to efficiently deploy the dynamic network infrastructures to support these various services and applications. Network virtualization provides a feasible way to share computing resources and network resources among virtual network requests, and also introduces greater flexibility and efficiency in the allocation of computing resources and link resources. The virtual network mapping or virtual network embedding (VNE) is to allocate the physical computing resources and network resources for the virtual network request, where each virtual node is mapped to a physical computing node, and each virtual link is mapped to a path which consists of a set of physical links in the physical network [61, 79, 12].

As the amount of data stored and processed by applications and services continues to increase, optical networks will play an important role in next-generation virtual network infrastructures. Optical network virtualization can be realized by the partitioning or aggregation of the physical optical network elements, including the sliceable bandwidth-variable optical transponders, optical cross-connects (OXCs), sliceable regenerators, reconfigurable optical add-drop multiplexers (ROADMs), optical fiber links, wavelengths, and spectrum slices [41, 39].

A key issue of the virtual optical network (VON) is to allocate the computing resources and network resources over the physical optical network. A typical virtual optical network request is specified by a virtual topology that includes a set of virtual nodes interconnected by virtual links. A virtual link may have certain bandwidth requirements while a virtual node may have specific computing requirements. The virtual optical network mapping is to map each virtual edge to a path with the appropriate bandwidth, and map each virtual node to one or more physical candidate nodes that satisfy the resource requirement in the substrate network [57, 26].

Due to the fact that the interruption of high-speed optical components, such as fiber, OXC, can affect a large amount of data and services, survivability has grown into an issue with great importance in traditional wavelength division multiplexing networks (WDMs) and emerging elastic optical networks (EONs) [53, 64, 27]. A single failure or disaster failure of the physical optical network may result in the inability of a VON to satisfy requests. Therefore, survivable virtual optical network mapping focuses on providing a different protection mechanism to avoid or reduce the impact of physical network failures [38, 74]. Protection mechanisms typically provide backup resources along paths, segment or physical links, which are disjoint for the working paths, segments or links. In addition, physical optical network components may be affected by various failures, such as fiber cuts and natural disasters. Failures can be categorized as independent failures, which only affect a single network component independently of other failures, or correlated failures, which affect a set of network components simultaneously. Therefore, different kinds of failures should be considered in survivable VON mapping [56, 31].

1.2 Network function virtualization

Traditionally, network functions are deployed as physical appliances that are tightly coupled with specific hardware, such as the firewalls, routers, and switches in networks. However, physically deployed network functions hinder the rapid deployment of new services and result in operational challenges. Network function virtualization (NFV) provides a more agile network by implementing the virtual network functions (VNFs) in software that is decoupled from the underlying hardware [33, 37], and which therefore, significantly decreases the OPEX and CAPEX. The VNFs can be implemented in software that can run on standardized high

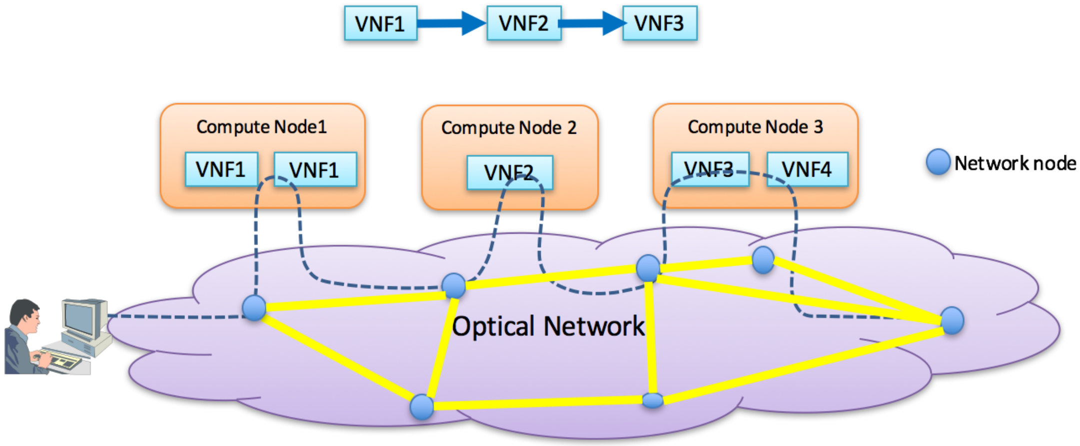


Figure 1.1. Service function chain mapping problem

performance servers or clouds. VNFs also provide a possible solution to share the physical hardware resources to improve the resource utilization. Based on the service provider's demand, end-to-end network services may consist of an ordered set of VNFs, and such ordered functions connected by logical links are referred to a service function chain (SFC) [32, 6].

Similar to the virtual network, a SFC can be described as one network topology with additional network functionality and mapping constraints. Typically, there may be dependence requirement among different VNFs in one SFC, and mutiple VNFs can be mapped to the same physical computing node. The SFC mapping problem is to find out the placement of the VNFs, allocate the VNFs to physical nodes with centain computing resources, and determine the path (a set of physical links) to connect different VNFs.

1.3 High-availability VON and NFV

Availability is a key factor in the service level agreements (SLAs) when a network service provider provides the computing and network resources to its customers. Availability is defined as the fraction of the time that the system is providing service to its users. Most of the existing work related to optical networks focuses on the end-to-end path availability and provides various types of the protection mechanisms to guarantee availability [81]. However, a VON request may consist of many services which depend on each other, and the failure of a single or multiple network elements may lead to the inability of the whole VON to support these services. For example, a VON may be deployed for a big data service with different virtual nodes handling different data processing tasks. A task in one virtual node may require the output of another task running on a different virtual node. A service function chain (SFC) consists of many dependable virtualized network functions (VNFs), such as load-balancers, DPIs and so on. Those functions are deployed in different data-centers which are connected by fiber links. If one part of the physical optical network element (e.g., OXC, fiber links, transponders) fails, the entire SFC will not be available. Therefore, it is important to evaluate a VON based on its availability, where a VON is defined to be available only when all of its virtual nodes and virtual links are available at the same time.

The availability of a network component is affected by various failures, such as fiber cuts and natural disasters. Failures can be categorized as independent failures, which only affect a single network component independently of other failures, or correlated failures, which affect a set of network components simultaneously. Most existing works related to availability are based on the assumption that the failures of network components are independent. However, correlated failures are very common and may significantly affect the availability of network components. For example, many network elements and computing nodes in one datacenter may be malfunction at the same time because of a power issue in the datacenter. Also, some fiber links may be affected by natural disasters, such as hurricanes, and floods. If such correlated failures are ignored, it may not be possible to find a VON mapping that satisfies the VON's availability requirement, or the mapping may not be optimal in terms of cost.

Availability is an important factor in measuring the quality of service in a network and can also be applied as a metric to measure the service quality in one SFC. In general, VNFs are deployed on generalized hardware that may lack the robustness of standard telecommunications equipment. Physical hardware failures may also affect several VNFs in one SFC. Furthermore, the malfunction of the software-based VNFs and the loss of logical connections in one SFC may also result in the unavailability of the service [58]. Therefore, software-based VNFs and flexible SFC mapping bring new challenges for network operators to provide availability-guaranteed network services. The availability-guaranteed SFC mapping problem is to determine the mapping of the VNFs in one SFC to guarantee the SFC's availability. The availability of both the network service and the SFC can be improved by adopting different protection mechanisms. Typically, network-layer protection (e.g., backup path) is provided to improve the availability of traditional network services. For software-based VNFs (one VNF may be deployed on one virtual machine), several replicas of each VNF can be provided to guarantee availability [34, 19, 76].

1.4 Background and related work

1.4.1 Research in reliable VON mapping

Besides many existing works to address the VON/VNE mapping problem [7, 21, 62, 26, 82], availability-aware routing over wavelength-division multiplexing (WDM) and elastic optical networks (EONs) has also been extensively investigated. In [72], the authors guarantee the availability of the application and services in a WDM mesh networks from the perspective of time, and a time aware availability guarantee (TAAG) scheme is proposed by adjusting the routing paths of each connection based on the different availability requirements in different time spans. In [67], the authors consider the case in which the connections for differentiated services may have different availability requirements, and therefore they may be provisioned with different protection (if needed) based on the network state and their availability requirements. In [51], the authors propose a shared-protection mechanism by monitoring the actual downtime experienced by one connection and the network state changes. and then estimate an updated availability target for each connection according to the predictable network-state changes. In [10], the authors develop a flexible availability-aware differentiated protection approach to meet different service availability requirements by changing the path protection scheme in EONs. The work in [40] focuses on the availability-aware survivable virtual network embedding (A-SVNE) problem in optical interdatacenter networks, and the authors aim to guarantee the availability requirement of each virtual link and each virtual node in a virtual network. In [81], the authors propose an availability analysis model, and then provide differentiated protection, including unprotected, dedicated protected, and shared protected, to meet customers' availability requirements.

Many existing work focus on the survivable virtual network embedding (VNE) problem to improve the service availability while considering single network component failures. The work in [35] guarantees an overall network availability by formulating the VNE problem with explicit link availability requirements. The authors in [36] focus on the trade-off between cost and availability when realizing virtual networks, and provide a comprehensive evaluation based on different network topologies and the influence of different parameters.

Correlated failures of links can often be modeled using the concept of a shared risk link groups (SRLG). The authors in [15] consider shared risk resource groups (SRRGs) and provide a polynomial time graph transformation techniques to tolerate multiple failures. In [23], the authors use a p-SRLG framework to handle the correlated and probabilistic failures for multi-domain optical networks. The paper [2] provides a unified model with an arbitrary probability density function for the network vulnerability, and then they identify vulnerable points by using computational geometric tools. Based on spatial point processes for representing stress centers on the geographical plane, the paper [60] proposes a stochastic model for correlated link failures in networks of arbitrary topologies due to multiple external stresses. The authors of [29] [28] propose a probabilistic model to find the simultaneous failure probability of a pair of paths under circular geographic failures with uniform and non-uniform distributions of epicenter location. Based on the topology aggregation information from each domain, they propose an inter-domain minimum overlapping area routing algorithm to improve the survivability of the service.

Some works also focus on disaster/regional failures in the optical network and provide different protection mechanisms to address the survivability of VON mapping. In [55], the authors aim to reduce the expected VN disconnections and capacity loss in the disaster-resilient and post-disaster-survivable VN mapping problem, and they build a probabilistic model and provide an adpative solution to minimize VN disconnections caused by any post-disaster single-physical-link failure. In [52], the authors focus on reducing the overall damage defined as the numbers of disconnected virtual networks, failed virtual links, and failed physical links, and they introduce the travalling repair problem to restore the optical network with minimum damage. In [78], the authors focus on the regional failure-resilient VI mapping problem, and provide a failure-dependent protection mechanism to share the protection resources among different regional failure scenarios. In [22], considering the large-scale regional failures, the authors propose a region-disjoint mapping (RDM) algorithm by mapping the primary and backup virtual networks into nonoverlapping geographical areas to improve the resilience.

1.4.2 Research in reliable NFV

Most of the existing works focus on providing several replicas for each VNF to improve the availability of the NFV/SFC. The authors in [5] proposed the VNF placement problem in datacenters to meet the latency constraints while minimizing the overall network load. The authors in [1] formulated the VNF placement and routing (VNF-PR) optimization problem considering the compression/decompression constraints and the forwarding latency regimes, and they designed a math-heuristic approach for large instances of the problem.

In [13], the VNF placement problem is separated into the facility location problem and the generalized assignment problem, and rounding-based heuristics are proposed to reduce the OPEX compared with greedy-based solutions. The authors in [50] investigated an integrated design for network function instance allocation and end-to-end demand realization sharing the same physical substrate network, and they proposed a game theory based algorithm by capturing the competition on physical resources between network function instance allocation and routing. The authors in [80] formulate the function placement and chaining problem of NFV as an integer optimization problem which can be mapped to an exponential number of min-cost flow problems. Then they address it by relaxing it into a fractional linear problem. By jointly considering the network topology, VNF instance number and deployment, and flow scheduling, the authors in [30] formulate the VNF deployment and flow scheduling problem as a MILP, and propose a relaxation-based algorithm to address it.

The authors in [19] proposed a reliability measurement model to estimate the reliability of the SFC mapping, and tried to back up the unreliable VNF in each SFC to guarantee the availability of each SFC request. In [17], the authors proposed a cost-aware importance measure (CIM) to select the qualified backups for each VNF, and to map the backups to the physical nodes with high reliability. The authors in [76] focused on the joint topology design and mapping (JTDM) problem, and evaluate service availability with no protection, dedicated protection, and shared protection. The authors in [9] proposed a probabilistic approach to find the best assignment of VMs to servers while guaranteeing protection by replication. In [59], the authors proposed a reliability-aware and delay-constrained routing optimization framework to address the optimal network service VNF placement and traffic routing problem in NFV-enabled data center networks. The authors in [42] investigated the tradeoff between end-to-end reliability and computational load per server via a joint design of VNF chain composition and forward graph embedding. The authors in [34] investigated different VNF backup deployment strategies in different data center topologies, and demonstrate the data-center topologies that offer the best cost-per-throughput relation for a given resilience/availability for VNF service chains.

1.5 Dissertation organization

The rest of this dissertation is organized as follows: In Chapter 2, we analyze the availability of virtual optical networks mapped over a physical substrate and propose VON mapping algorithms with selective dedicated protection to minimize cost while meeting availability requirements. In Chapter 3, we consider virtual optical network (VON) mapping with the objective of minimizing total network link cost while guaranteeing VON availability, where VON availability is supported by providing shared backup path protection for selected VON links. We transform the problem to a group node-weighted Steiner tree problem and propose an efficient auxiliary-graph-based availability (AA) algorithm to find a VON mapping with high availability. Based on the availability evaluation, we propose a heuristic algorithm to map the VON. In Chapter 4, we address the high-availability VON mapping considering the correlated failures in optical network. We aim to find out the VON mapping with the maximum availability without any protection, and then provide a dedicated path protection mechanism to guarantee the availability of each VON if needed. We propose the Group Steiner tree problem under correlated cost (GSTP-CC) to find out the maximum availability of the VON mapping without any protection. We then address the VON mapping under correlated failures by providing dedicated protection based on the GSTP-CC algorithms. In Chapter 5, we propose a coordinated protection mechanism that adopts both backup path protection in the network and VNF replicas at nodes to guarantee a SFCs availability. The proposed mechanism determines the number of replicas required for each VNF in the SFC, and allocates the replicas to physical nodes on the working and backup paths while maintaining ordered dependency among VNFs. we also propose an efficient segment protection mechanism to guarantee each SFCs availability requirement. In Chapter 6, considering the availability of the datacenter network elements, we propose a coordinated protection mechanism that adopts both backup path protection and SFC replicas distributed among datacenters to support high availability while reducing total cost.

CHAPTER 2

AVAILABILITY-GUARANTEED VIRTUAL OPTICAL NETWORK MAPPING WITH SELECTIVE PATH PROTECTION[1]

In this chapter, we study the availability-guaranteed VON mapping problem with the objective to reduce the total link cost as well as reducing the VON blocking rate. We propose a mapping boolean matrix to analyze the availability of a VON mapping. A heuristic algorithm using a new cost metric is proposed to provide partial dedicated protection while meeting the VON availability requirement[44].

2.1 Introduction

Network virtualization provides an effective way to share physical network resources among virtual network requests and introduces greater flexibility and efficiency in allocating resources. The application of virtualization concepts to optical networks will enable service providers to efficiently provision virtual optical network (VON) slices in support of dynamic high-bandwidth applications and services.

An important factor for a service provider when provisioning a VON is the ability to provide a high degree of availability for the VON customer in support of the service level agreement (SLA)[81]. Availability is defined as a fraction of time the system is providing service to its users. In general, network availability can be improved by adopting high quality hardware components; however, a more feasible strategy is the adoption of protection mechanisms for end-to-end paths. To guarantee a certain level of availability for a VON, it may not be necessary to provide full dedicated protection for each virtual link in the VON[73].

[1]©IEEE. Reprinted, with permission, from Jian Kong, Sangjin Hong, Jason P. Jue, Inwoong Kim, Xi Wang, Qiong Zhang, Hakki C. Cankaya, Weisheng Xie, and Tadashi Ikeuchi. "Availability-guaranteed virtual optical network mapping with selective path protection." In *Optical Fiber Communications Conference and Exhibition (OFC)*, 2016.

Rather, it may still be possible to meet availability requirements by appropriately mapping the virtual links to the physical substrate layer and by carefully selecting the virtual links for which protection should be provided. In this chapter, we develop a model for evaluating the availability of a VON mapped over a physical optical network, and we address the problem of finding a survivable mapping and determining an appropriate protection strategy in order to guarantee high availability for a VON while minimizing the total link cost.

2.2 Problem statement

We define a VON to be available only when all of its virtual nodes and virtual links are available at the same time. We analyze the availability of a VON mapped to a physical optical network under the following assumptions[81]: 1) network components used by the VON are either available or unavailable, 2) different network components fail independently and the repair process of each network component is also independent, 3) compared to the repair time of failures, the switching time from working network components to backup components is negligible, 4) we only consider the effect of physical layer failures on VON availability.

The availability-based partial dedicated protection VON mapping problem is described as follows. We are given 1) a physical optical network, $G_p(N_s, L_s)$, where N_s is the set of physical nodes, L_s is the set of physical fiber links, and the availability of each physical network component is also given; 2) a virtual optical network request with topology $G_v(N_v, L_v)$, where N_v is the set of virtual nodes, L_v is the set of virtual links, and each virtual link has a bandwidth requirement; and 3) the availability requirement of the VON. It may not be necessary to provide dedicated path protection for each virtual link to meet the VON's availability requirement. Therefore, we need to first determine the virtual links that should be protected and then map the virtual nodes and virtual links to the physical optical network. The objective is to find a mapping of VON request to physical optical network with partial dedicated protection to minimize the total link cost and reduce the VON blocking rate while meeting the VON's availability requirement.

2.3 VON availability analysis and calculation

Here we propose a boolean matrix model to analysis and calculate VON availability. As defined above, a VON is available only when all of its virtual nodes and virtual links are available at the same time. Since most network failures are caused by fiber cable cuts [81], we focus on link failures and assume that other network components are always available. However, the following availability analysis and calculation steps can be easiliy extended to other physical network components failures. Dedicated protection can increase the availability of VON. For example, as shown in Fig. 2.1, a VON has one virtual link which is mapped to physical link. Without any protection, its availability is $\mathbf{A}_{von} = a_1 \cdot a_2 = 0.81$. With dedicated protection, the VON is available when either working path or backup path is available. Therefore, the availability of the VON with dedicated protection is $\mathbf{A}_{von} = a_1 \cdot a_2 + (1 - a_1 \cdot a_2) \cdot a_3 \cdot a_4 = 0.9544$. The availability of the backup path is not considered when links for the working path are available, which can help to reduce the calculation steps of VON availability. We use the following example to show the calculation details. A VON request, as shown in Fig. 2.2, with $m_v = 3$ virtual nodes and $n_v = 8$ virtual links, is mapped to a physical optical network which has $m_p = 10$ physical nodes and $n_p = 8$ fiber links. Some virtual links (w_1 and w_2) are provided with dedicated protection to guarantee the service reliability.

Step 1: We create a boolean matrix $\mathbf{M_0}(n_p \times 2n_v)$ where row i represents fiber link l_i, columns $0 \sim (n_v - 1)$ represent the working path mapping of each virtual link, while columns $n_v \sim (2n_v - 1)$ represent the backup path mapping of virtual links. i) If fiber link $link_i$ is used by working path w_j, $\mathbf{M_o}[link_i, w_j]$ is set to 1, otherwise 0. ii) If fiber link $link_i$ is used by backup path b_j, $\mathbf{M_o}[link_i, b_j]$ is set to 1, otherwise 0. iii) If there is no backup path for working path w_i, it can be seen that its backup path is the same as the working path, therefore, let $\mathbf{M_o}[link_i, b_j] = \mathbf{M_o}[link_i, w_j]$.

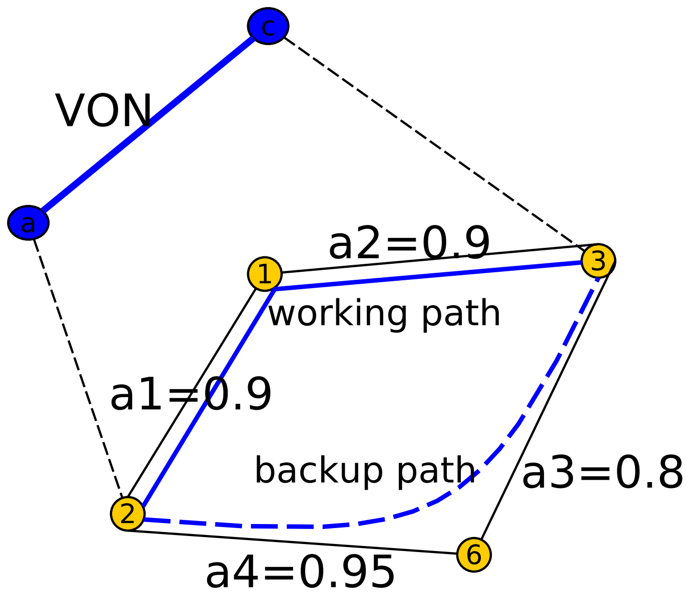


Figure 2.1. Availability example

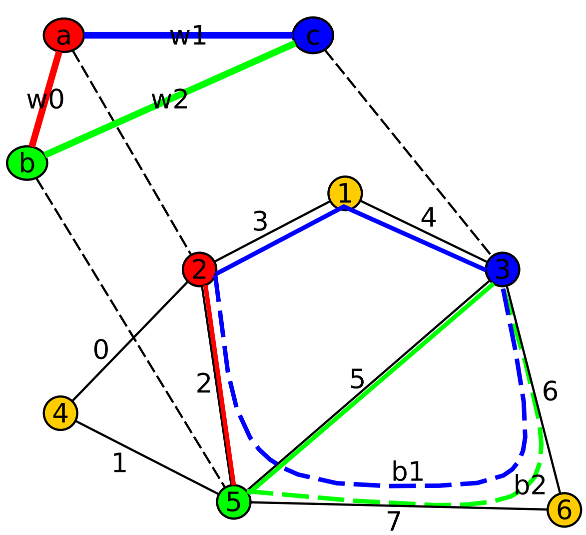


Figure 2.2. VON mapping with selective protection example

$$
\mathbf{M_0} = \begin{array}{c} \\ link_0 \\ link_1 \\ link_2 \\ link_3 \\ link_4 \\ link_5 \\ link_6 \\ link_7 \end{array} \begin{array}{c} \begin{array}{cccccc} w_0 & w_1 & w_2 & b_0 & b_1 & b_2 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array}
$$

$$
\mathbf{M} = \begin{array}{c} \\ L_0 \\ L_1 \\ L_2 \\ L_3 \end{array} \begin{array}{c} \begin{array}{cccccc} w_0 & w_1 & w_2 & b_0 & b_1 & b_2 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array} = \begin{bmatrix} \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{pmatrix} \mathbf{W}_0 & \mathbf{B}_0 \\ \mathbf{W}_1 & \mathbf{B}_1 \\ \mathbf{W}_2 & \mathbf{B}_2 \\ \mathbf{W}_3 & \mathbf{B}_3 \end{pmatrix}
$$

$$
\mathbf{M} : m \times 2n_v \; \mathbf{C} : m \times n_v
$$

$$
\mathbf{D} : m \times n_v \; \mathbf{W}_i : 1 \times n_v \; \mathbf{W}_i : 1 \times n_v
$$

Step 2: We simplify the mapping boolean matrix in the following two ways: i) Remove the rows of fiber links not used by virtual links ($\| link_i \| = 0$), such as $link_0$ and $link_1$, from mapping boolean matrix $\mathbf{M_o}$. ii) Merge the fiber links used by the same virtual links. We define L_i as the collection of fibers link used by the same virtual links. For example, $link_6$ and $link_7$ are used by backup path b_1 and backup path b_2, thus we define L_3 as the collection of $link_6$ and $link_7$. We finally obtain a simplified mapping boolean matrix $\mathbf{M}(m \times 2n_v)$, where L_0 is the collection of $link_2$; L_1 is the collection of $link_3$ and $link_4$; L_2 is the collection of $link_5$; and so on.

Step 3: We define WL as the set of L_i that are used by working paths of virtual links in the VON. The availability of the VON is:

$$\mathbf{A}_{VON} = \sum_{\forall \Delta, \Delta \subseteq \Omega} \mathbf{K}_{\Delta} \mathbf{P}_{\Delta}, \tag{2.1}$$

where Ω is a set of WLs and Δ is the subset of Ω, and $\mathbf{P}_{\Delta}$ is the probability that all WLs in Δ fail at the same time. In the above example, L_0, L_1, L_2 are in WL and $\Omega = \{\{\phi\}, \{L_0\}, \{L_1\}, \cdots, \{L_0, L_1, L_2\}\}$. We take $\Delta = \{L_i, L_j\}$ as an example and calculate the availability in the following steps:

$$\mathbf{H} = \mathbf{W_i} \vee \mathbf{W_j}$$

$$\mathbf{L} = \mathbf{B_i} \vee \mathbf{B_j}$$

$$\mathbf{I} = [\mathbf{1} \ \cdots \ \mathbf{1}]$$

$$\mathbf{F} = \mathbf{C} \odot \mathbf{I^T} \vee \mathbf{D} \odot \mathbf{H^T}$$

Then we obtain:

$$\mathbf{P}_{\Delta} = (1 - \mathbf{A}_{L_i})(1 - \mathbf{A}_{L_j}) \prod_{k \neq i,j, L_k \in \Delta} (\mathbf{A}_{L_k})^{f_k},$$

$$\mathbf{K}_{\Delta} = \begin{cases} 0 & \text{if} \quad \mathbf{H} \odot \mathbf{L^T} = true \\ 1 & \text{if} \quad \mathbf{H} \odot \mathbf{L^T} = false \end{cases}$$

where

$$f_k = \begin{cases} 1 & \text{if} \quad \mathbf{F}[k] = true \\ 0 & \text{if} \quad \mathbf{F}[k] = false \end{cases} \quad (k = 0, 1 \cdots m-1)$$

To be specific, if there is no link failure($\Delta = \phi$), $\mathbf{H} = \mathbf{L} = [\mathbf{0} \ \cdots \mathbf{0}]$.

2.4 VON mapping heuristics

We first propose two simple heuristic algorithms to solve the problem of VON mapping. In the REDUCE algorithm, we initially find protection paths for all virtual links. A greedy algorithm is used for virtual node mapping, and Suurballe's algorithm is used for virtual link mapping. If the VON mapping exceeds the availability requirement, we remove the backup path of the working path with the highest availability. The algorithm stops when the availability requirement is no longer met when removing a backup path.

In the ADD algorithm, the VON request is initially mapped without any protection. If the availability requirement is not met, protection paths are added one at a time to the working path with the lowest availability until the availability requirement is reached. Dijkstra's algorithm is used for working paths without protection, and Suurballe's algorithm is used for working paths with backup paths. If the availability requirement cannot be met when all virtual links are protected, the VON request is blocked.

The above two algorithms use the physical link cost to calcuate the shortest working path and backup path for each virtual link. In order to minimize the total link cost of the VON mapping and reduce the blocking caused by the availability requirement, we introduce a third ADAPTIVE algorithm that utilizes a new adpative cost metric for VON mapping: $\mathbf{metric}_{ij} = \mathbf{cost}_{ij} - \gamma \log \mathbf{a}_{ij}$, where $\mathbf{cost}_{ij}$ is the physical link cost between node i and node j, and $\mathbf{a}_{ij}$ is the availability of the link between node i and node j. γ is adaptive to the difference between the current availability ($\mathbf{A}_{von}$) and availability requirement ($\mathbf{A}_r$), where $\gamma = k(\mathbf{A}_r - \mathbf{A}_{von})$. The new cost metric takes the physical link cost and current availability of the VON into consideration to achieve a balance between the total link cost and the blocking rate of VON mappings. Initially, there is no protection for any virtual links of the VON, resulting in a greater difference between $\mathbf{A}_r$ and $\mathbf{A}_{von}$. Therefore, γ is larger and the algorithm tries to map to virtual links with higher availability. As the number of protected virtual links increases, $\mathbf{A}_{von}$ approaches $\mathbf{A}_r$. Therefore, the physical link cost plays a more important role in finding the shoretst path. Thus, the ADAPTIVE algorithm tries to decrease the total link cost by reducing the number of backup paths needed.

2.5 Numerical results

We use the 24-node U.S. mesh network [14] to evaluate the performance of our proposed algorithms. For each physical fiber link, we consider availability ranging from 0.99 to 0.999999. We assume that there are sufficient spectrum slots on each fiber link. For each VON request, the number of virtual nodes is between 3 and 5. We randomly generate 500 VON requests to evaluate the performance.

Table 2.1 illustrates the blocking rate of VON mapping requests. The physical link availability ranges from 0.99 to 0.999999, and the VON requirement ranges from 0.99 to 0.999999. From the table, the physical optical network can gurantee 6 nines availability for a VON with dedicated protection when all of its physical links have 0.9999 availability. The ADAPTIVE algorithm performs better than the REDUCE and ADD algorithms because the ADAPTIVE algorithm tries to find paths with high availability using the new cost metric if there is a big gap between its current availability and its requirement.

Fig. 2.3 shows the total link cost of VON mappings. The availability of physical links range between 0.9999 and 0.999999 with uniform distribution, and the VON availability requirement ranges from 0.999 to 0.999999. Initially, low availability requirements can be supported without any protection. As the availability requirement increases, the VON mapping requires more and more dedicated protection to meet the availability requirement. The ADAPTIVE algorithm results in lower cost compared to the other two algorithms because it can choose the working and backup paths with higher availability; therefore, the ADAPTIVE algorithm can use fewer dedicated protection paths to meet the availability requirement. Fig. 2.4 shows the total link cost as the number of VON requests increases for the case in which physical links have 0.9999 availability and the VON availability requirement is 0.9999, and Fig. 2.5 shows the ratio between the number of virtual links with protection and total number of virtual links.

Table 2.1. VON blocking rate

Physical Link Avail. \ Required VON Avail.	0.99	0.999	0.9999	0.99999	0.999999
0.99	0	0.38	0.99	1.0	1.0
0.999	0	0	0	0.602	0.994
0.9999	0	0	0	0	0
0.99999	0	0	0	0	0
0.999999	0	0	0	0	0

(a). REDUCE algorithm

Physical Link Avail. \ Required VON Avail.	0.99	0.999	0.9999	0.99999	0.999999
0.99	0	0.384	0.99	1.0	1.0
0.999	0	0	0	0.604	0.994
0.9999	0	0	0	0	0
0.99999	0	0	0	0	0
0.999999	0	0	0	0	0

(b). ADD algorithm

Physical Link Avail. \ Required VON Avail.	0.99	0.999	0.9999	0.99999	0.999999
0.99	0	0.192	0.988	1.0	1.0
0.999	0	0	0	0.486	0.994
0.9999	0	0	0	0	0
0.99999	0	0	0	0	0
0.999999	0	0	0	0	0

(c). ADAPTIVE algorithm

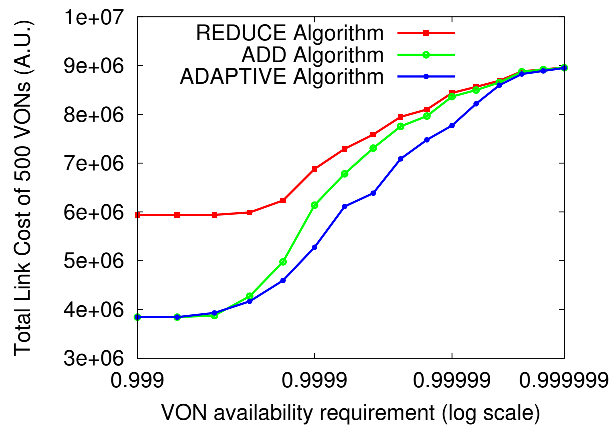


Figure 2.3. Cost vs. availability

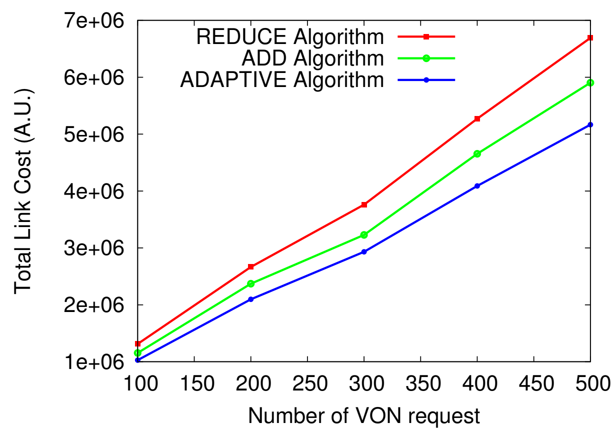


Figure 2.4. Cost vs. number of VONs

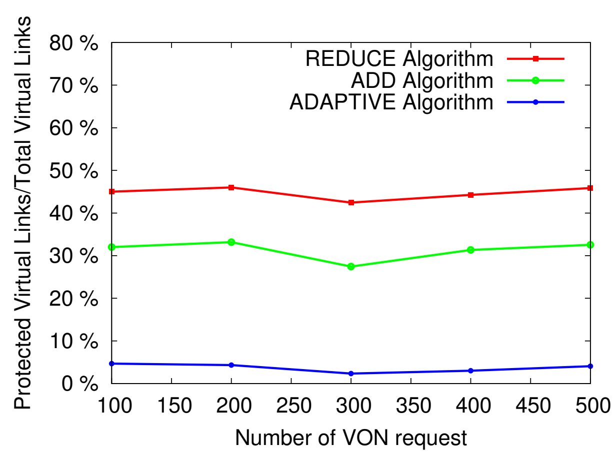


Figure 2.5. Protected virtual links

2.6 Conclusion

In this chapter, we define the availability of a VON mapping and propose a mapping boolean matrix to analyze the availability of a VON mapping. We propose an ADAPTIVE algorithm using a new cost metric to provide partial dedicated protection while meeting the VON availability requirement. Simulation results demonstrate that the proposed ADAPTIVE algorithm can significantly reduce the total link cost while reducing the VON blocking rate.

CHAPTER 3

AVAILABILITY-GUARANTEED VIRTUAL OPTICAL NETWORK MAPPING WITH SHARED BACKUP PATH PROTECTION[1]

In this chapter, we consider virtual optical network (VON) mapping with the objective of minimizing total network link cost while guaranteeing VON availability, where VON availability is supported by providing shared backup path protection for selected VON links. We develop a matrix-based approach for calculating the availability of a VON mapping with shared backup path protection. In order to efficiently evaluate the maximum availability of a VON mapping, we transform the problem to a group node-weighted Steiner tree problem and propose an efficient auxiliary-graph-based availability (AA) algorithm to find a VON mapping with high availability. Based on the availability evaluation, we propose a heuristic algorithm to map the VON [46].

3.1 Introduction

With the deployment of virtual optical networks, the issue of how to efficiently and accurately evaluate the QoS of the VON is becoming more important. A VON request may consist of many services which may depend on each other. The failure of a single network element (physical node, link, or storage) may result in an inability of the VON to support one or more of its intended services. For example, a VON may be deployed for a big data application with different virtual nodes handling different data processing tasks. A task in one virtual node may require the output of another task running on a different virtual node. If a physical link on the path between these two nodes fails, or if one of the two physical nodes

[1]©IEEE. Reprinted, with permission, from Jian Kong, Jason P. Jue, Inwoong Kim, Xi Wang, Qiong Zhang, Hakki C. Cankaya, Weisheng Xie, and Tadashi Ikeuchi. "Availability-guaranteed virtual optical network mapping with shared backup path protection." In *IEEE Global Communications Conference (GLOBECOM)*, 2016.

fails, the entire application must stop. In this chapter, we also evaluate a VON based on its availability, where a VON is defined to be available only when all of its virtual nodes and virtual links are available at the same time.

In this chapter, we study the VON mapping problem with the objective of minimizing the total link cost of a VON mapping while guaranteeing the VON's availability requirement. To reduce the link cost, there is no need to provide protection for the VON if its availability can be easily reached. However, if its availability can not be guaranteed without any protection, a shared backup path protection mechanism is applied to selected VON links to improve availability. In this paper, shared backup path protection among different VONs is applied to further reduce the total link cost.

An efficient way to determine whether there is a need to provide protection for a VON is to find the VON mapping with the maximum availability. In this paper, we transform the problem of finding the VON mapping with the maximum availability to a Steiner tree problem (STP)[24] by constructing an auxiliary graph in which the edge weights are functions of the availability of network elements. We then propose a heuristic algorithm that finds a VON mapping with near-optimal availability.

The rest of this chapter is organized as follows. In Section 3.2, the availability-guaranteed VON mapping problem is defined, and we present a simple mapping example to show the availability calculation. In Section 3.3, we extend a boolean mapping matrix to calculate the availability of a VON mapping with shared backup path protection. In Section 3.4, an auxiliary graph based heuristic algorithm is proposed to approximate the maximum availability of a VON mapping. Based on the heuristic algorithm, we propose an effective heuristic algorithm in Section to map the VONs. In Section 3.6, numerical results are presented, and we conclude this chapter in Section 3.7.

3.2 Problem statement

In this problem, we are given a physical optical network $G_p(N_s, L_s, S_s, A_s)$, where N_s is the set of physical nodes, L_s is the set of physical fiber links, S_s is vector specifying the capacity of each physical link in spectrum slots, and A_s is a vector specifying the availability of each physical node and each physical link. We make the following assumptions regarding availability in the physical optical network[81]: 1) physical nodes and links are either available or unavailable, 2) physical nodes or links fail independently, and the repair process is also independent, 3) a reverting model, which means that, when failures of the working path are repaired, the traffic will change from backup path back to the working path, 4) a resource-locked model, which means that, if there are many failed connections, connections can be recovered in the exact order of their failure sequence, i.e., earliest failure recovered first, 5) and an active-recovery network, which means that, when the working path and its backup path suffer failures, if the backup path is repaired, the working path can switch to the backup path.

We are also given a virtual optical network request with topology $G_v(N_v, L_v, A_v)$, where N_v is the set of virtual nodes, L_v is the set of virtual links and A_v is the availability requirement of the VON. Each virtual node has a set of physical candidate nodes and each virtual link has a bandwidth requirement.

The problem is to map the virtual nodes to the candidate physical nodes, find the routing paths for each virtual link, and allocate spectrum slots along the paths to meet the bandwidth demand of each VON link. Our goal is to find a mapping for each VON while guaranteeing the availability requirement of each VON. If the availability can be easily met, there is no need to provide protection for the VON. Otherwise, we provide shared backup path protection among different VONs. By using shared backup path protection, the total link cost can be reduced while meeting the VON availability requirement. The solution to the problem is subject to the following constraints.

1. Spectrum continuity constraint: the spectrum allocated for a VON link must be continuous across the entire physical path on which the VON link is mapped.

2. Node mapping constraint: one virtual node should be mapped to only one candidate physical node, and one physical node cannot be mapped to multiple virtual nodes in one VON.

3.3 Availability calculation of VON mapping

After a VON is mapped to the physical optical network with some virtual links sharing the backup path resources, we need to calculate the VON's availability. We assume that there are m_p physical network elements (physical nodes and links) and there are n_v virtual links for one VON request. For example, as shown in Fig. 3.1, there are 3 VON requests $VON1$, $VON2$ and $VON3$, where w_{11} in $VON1$ and w_{21} in $VON2$ share the physical link 6 as the backup path, and w_{12} in $VON1$ and w_{31} in $VON3$ share the physical link 8 as the backup path. The availability of $VON1$ is calculated in the following steps.

Step 1: We first obtain the boolean matrix mapping $\mathbf{C}(m_p \times n_v)$ for working paths where row i represents network element i, and each column represents the working path mapping of each virtual link. We also obtain $\mathbf{D}(m_p \times n_v)$ for backup paths where each column represents the backup path mapping of each virtual link. In a boolean matrix, 1 represents true while 0 represents false. i) If network element i is used by the working path of virtual link j, $\mathbf{C}[i][j] = 1$, otherwise 0. ii) If network element i is used by the backup path of virtual link j, $\mathbf{D}[i][j] = 1$, otherwise 0. iii) If there is no backup path for the virtual link j, it can be seen that its backup path is the same as the working path, therefore, let $\mathbf{D}[i][j] = \mathbf{C}[i][j]$. For example, as shown in Fig. 3.1, the three virtual links (w_{11}, w_{12}, w_{13}) of $VON1$ are mapped to the physical optical network with dedicated path protection for w_{11}, w_{12} and no protection for w_{13}. We obtain the mapping matrices $\mathbf{C}$ and $\mathbf{D}$ as shown in Fig. 3.2.

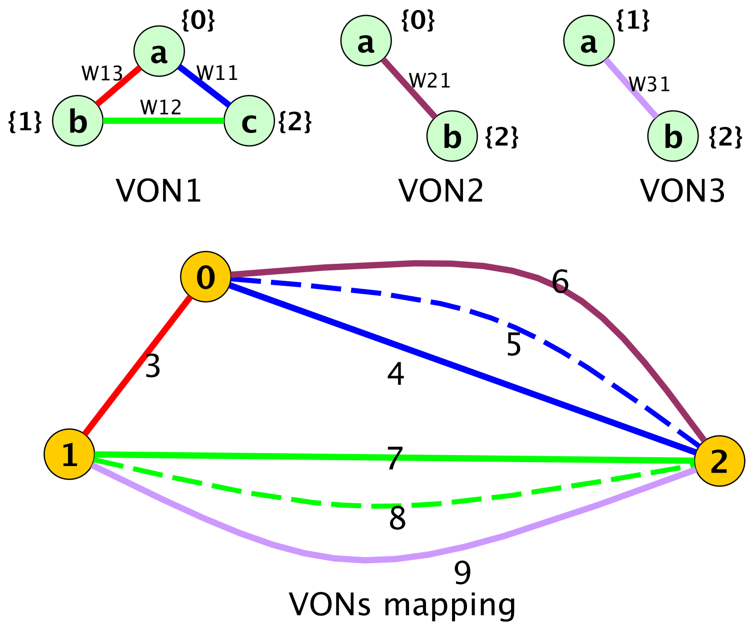


Figure 3.1. VON mapping example

Step 2: We create a boolean mapping matrix $\mathbf{Q_i}$ for each virtual link i. In the above example, the virtual link w_{11} of $VON1$ shares the network elements 1, 6, and 3 with the virtual link w_{21} of $VON2$. The first column of $\mathbf{Q_1}$ represents the working path mapping of w_{11} and the second column represents the working path mapping of w_{21}. We can also obtain $\mathbf{Q_2}$ and $\mathbf{Q_3}$ for the virtual links w_{12} and w_{13}, as shown in Fig. 3.2. Note that, $\mathbf{Q_3}$ is a single column vector because there is no backup path shared for the virtual link w_{13}.

Step 3: We create a boolean matrix $\mathbf{E}$ to represent the network elements used by working paths in the current VON and working paths in other VONs that share the backup paths with the current VON, and define Ω as the set of such network elements. The availability of the VON is:

$$\mathbf{A}_{VON} = \sum_{\forall\Delta,\Delta\subseteq\Omega} \mathbf{K}_{\Delta}\mathbf{P}_{\Delta}O_{\Delta} \tag{3.1}$$

where Δ is the subset of Ω, $\mathbf{K}_{\Delta}$ shows whether the VON is available when all network elements in Δ fail at the same time, $\mathbf{P}_{\Delta}$ is the probability that all network elements in Δ

Table 3.1. Matrices used in availability calculation

Nota.	Description
m_p	number of physical network elements(physical nodes and links)
n_v	number of virtual links in current VON
w_{ij}	the jth virtual link in VON i
$\mathbf{C}$	$m_p \times n_v$, mapping matrix of current VON's working paths
$\mathbf{D}$	$m_p \times n_v$, mapping matrix of current VON's backup paths
$\mathbf{E}$	$m_p \times 1$, network elements used by working paths
$\mathbf{Q_i}$	mapping matrix that the virtual links share the backup path with the ith virtual link in current VON)
$\mathbf{H}$	$1 \times n_v$, virtual links whose working paths are affected by failures
$\mathbf{L}$	$1 \times n_v$, virtual links whose backup paths are affected by failures
$\mathbf{F}$	$m_p \times 1$, network elements to guarantee VON available

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{M_0} \\ \mathbf{M_1} \\ \mathbf{M_2} \\ \mathbf{M_3} \\ \mathbf{M_4} \\ \mathbf{M_5} \\ \mathbf{M_6} \\ \mathbf{M_7} \\ \mathbf{M_8} \\ \mathbf{M_9} \end{bmatrix}, \mathbf{D} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{B_0} \\ \mathbf{B_1} \\ \mathbf{B_2} \\ \mathbf{B_3} \\ \mathbf{B_4} \\ \mathbf{B_5} \\ \mathbf{B_6} \\ \mathbf{B_7} \\ \mathbf{B_8} \\ \mathbf{B_9} \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

$$\mathbf{Q_1} = \begin{bmatrix} 1 & 1 \\ 0 & 0 \\ 1 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{S_{10}} \\ \mathbf{S_{11}} \\ \mathbf{S_{12}} \\ \mathbf{S_{13}} \\ \mathbf{S_{14}} \\ \mathbf{S_{15}} \\ \mathbf{S_{16}} \\ \mathbf{S_{17}} \\ \mathbf{S_{18}} \\ \mathbf{S_{19}} \end{bmatrix}, \mathbf{Q_2} = \begin{bmatrix} 0 & 0 \\ 1 & 1 \\ 1 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{S_{20}} \\ \mathbf{S_{21}} \\ \mathbf{S_{22}} \\ \mathbf{S_{23}} \\ \mathbf{S_{24}} \\ \mathbf{S_{25}} \\ \mathbf{S_{26}} \\ \mathbf{S_{27}} \\ \mathbf{S_{28}} \\ \mathbf{S_{29}} \end{bmatrix}, \mathbf{Q_3} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{S_{30}} \\ \mathbf{S_{31}} \\ \mathbf{S_{32}} \\ \mathbf{S_{33}} \\ \mathbf{S_{34}} \\ \mathbf{S_{35}} \\ \mathbf{S_{36}} \\ \mathbf{S_{37}} \\ \mathbf{S_{38}} \\ \mathbf{S_{39}} \end{bmatrix}$$

Figure 3.2. The VON mapping boolean matrix

fail at the same time, and O_Δ is the probability that the current VON occupies the backup path when many VONs are affected by the failures. We take $\Delta = \{i, j\}$ as an example and calculate the availability in the following steps:

$$\mathbf{H} = \mathbf{M_i} \vee \mathbf{M_j};\ \ \mathbf{L} = \mathbf{B_i} \vee \mathbf{B_j};\ \ \mathbf{F} = \mathbf{E} \vee (\mathbf{D} \odot \mathbf{H^T}).$$

If there is no link failure ($\Delta = \emptyset$), $\mathbf{H} = \mathbf{L} = [\mathbf{0} \cdots \mathbf{0}]$.

Then we obtain:

$$\mathbf{P}_\Delta = \prod_{k=i,j} (1 - a_k) \prod_{k \neq i,j}^{k \in \Omega} (a_k)^{f_k} \tag{3.2}$$

where,

$$f_k = \begin{cases} 1 & \text{if} \quad \mathbf{F}[k][0] \ is\ true \\ 0 & \text{if} \quad \mathbf{F}[k][0] \ is\ false \end{cases} \quad (k = 0, 1 \cdots m_p - 1) \tag{3.3}$$

and,

$$\mathbf{K}_\Delta = \begin{cases} 0 & \text{if} \quad \mathbf{H} \odot \mathbf{L^T} \ is\ true \\ 1 & \text{if} \quad \mathbf{H} \odot \mathbf{L^T} \ is\ false. \end{cases} \tag{3.4}$$

For a reverting system with active-recovery, the working paths that share the backup path have the same probability of obtaining the backup path when there is a failure [81]. Then, we obtain:

$$\mathbf{O}_k = \begin{cases} \mathbf{S}_{ki} \vee \mathbf{S}_{kj} & \text{if} \quad \Delta^{'} \neq \emptyset, \\ \emptyset & \text{if} \quad \Delta^{'} = \emptyset \end{cases} \tag{3.5}$$

where $\Delta^{'}$ is the set of network elements used by this VON in Δ. And,

$$O_\Delta = \prod_{k=1}^{n_v} \frac{1}{\|\mathbf{O}_k\|}. \tag{3.6}$$

where $\|\mathbf{O}_k\|$ is the number of *true* elements in $\mathbf{O}_k$. To be specific, if there are no *true* elements in $\mathbf{O}_k$, $\|\mathbf{O}_k\| = 1$. To reduce the computation, we can also simplify the boolean mapping matrices as proposed in Chapter 2.

For example, in the calculation of the $VON1$ availability, $\Omega = \{0, 1, 2, 3, 4, 6, 7, 9\}$. If physical links 4 and 6 fail, $\Delta = \{4, 6\}$ and we obtain,

$$\mathbf{H} = \mathbf{M_4} \vee \mathbf{M_6} = [1\ 0\ 0] \vee [0\ 0\ 0] = [1\ 0\ 0]$$

$$\mathbf{L} = \mathbf{B_4} \vee \mathbf{B_6} = [0\ 0\ 0] \vee [0\ 0\ 0] = [0\ 0\ 0]$$

$$\mathbf{D} \odot \mathbf{H^T} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \odot [1\ 0\ 0]^{\mathbf{T}} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{H} \odot \mathbf{L^T} = [1\ 0\ 0] \odot [0\ 0\ 0] = 0$$

Therefore, we obtain

$$\mathbf{K}_\Delta = 1$$

$$\mathbf{F} = \mathbf{E} \vee (\mathbf{D} \odot \mathbf{H^T}) = [1\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 0\ 1]^{\mathbf{T}} \vee [1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0]^{\mathbf{T}} = [1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 1]^{\mathbf{T}}$$

We can also obtain:

$$\mathbf{P}_\Delta = (1 - a_4)(1 - a_6)a_0 a_1 a_2 a_3 a_5 a_7 a_9$$

Here, $\Delta' = \{4\} \neq \emptyset$ and we obtain:

$$\mathbf{O}_1 = \mathbf{S}_{14} \vee \mathbf{S}_{16} = [1\ 0] \vee [0\ 1] = [1\ 1],\ \|\mathbf{O}_1\| = \tfrac{1}{2}$$

$$\mathbf{O}_2 = \mathbf{S}_{24} \vee \mathbf{S}_{26} = [0\ 0] \vee [0\ 0] = [0\ 0],\ \|\mathbf{O}_2\| = 1$$

$$\mathbf{O}_3 = \mathbf{S}_{34} \vee \mathbf{S}_{36} = [0] \vee [0] = [0],\ \|\mathbf{O}_3\| = 1$$

So we can obtain $O_\Delta = \frac{1}{2}$.

Therefore, the probability that $VON1$ is available when the network elements 4 and 6 fail at the same time is:

$$\mathbf{K}_\Delta \mathbf{P}_\Delta O_\Delta = \frac{1}{2}(1 - a_4)(1 - a_6)a_0 a_1 a_2 a_3 a_5 a_7 a_9$$

3.4 Maximum availability for the VON mapping without protection

In this section, we try to find the maximum availability of a VON mapping without protection. We do not consider the spectrum allocation and assume that there is sufficient spectrum for each physical link. We first transform the maximum availability problem to the group node-weighted Steiner tree problem (STP). Then we propose an Auxiliary-graph-based Availability (AA) algorithm to evaluate the maximum availability of the VON mapping. Finally, an example is given to show the details of the AA algorithm.

3.4.1 Problem transformation

In this paper, we solve the VON mapping problem with the objective of minimizing the total link cost while guaranteeing each VON availability requirement. To reduce the link cost, there is no need to provide protection for the VON if its availability can be easily reached. Different VON mapping solutions result in different VON availability. Therefore, we need to find the VON mapping solution with the maximum availability, and then determine whether protection is needed. We transform the problem of finding the maximum availability VON mapping into a group node-weighted Steiner tree problem.

The group node-weighted Steiner tree problem [16, 43] is, given a graph $\mathbf{G}$ with edge weights and node weights, and a collection $g_1, g_2, \ldots, g_k$ of node sets called *groups*, to find a minimum-weight connected subgraph of $\mathbf{G}$ that contains at least one node from each group.

We transform our problem into the group node-weighted steiner tree problem by constructing an availability-based graph $\mathbf{G_a}$ in which the node weights and link weights are determined by

$$weight_i = -\log a_i, \tag{3.7}$$

where a_i is the availability of the physical network element. Therefore, we can get the maximum availability by finding the minimum total cost using the *weight*. Each group is the set of each virtual node's candidate nodes.

Theorem 1. *Given a VON request without any protection and given several candidate nodes for each virtual node, the VON's maximum availability mapping is the solution to the group node-weighted Steiner tree problem in the graph $\mathbf{G_a}$, where the groups $g_1, g_2, \ldots, g_k$ are the sets of each virtual node's candidate nodes.*

Proof. We prove Theorem 1 by contradiction. Assume that the total link cost of the group node-weighted Steiner tree is C, and the availability of the mapping is $A = e^{-C}$. If the maximum availability mapping is not its group node-weighted Steiner tree in the graph $\mathbf{G_a}$, there must be a mapping with availability $\bar{A} > A$, therefore, the total link cost of this solution $\bar{C} < C$. This implies that there is one connected subgraph with less cost than the current group Each group is the set of node-weighted Steiner tree, which contradicts the definition of a group node-weighted Steiner tree. □

3.4.2 Auxiliary-graph-based availability (AA) algorithm

After transforming the maximum availability VON mapping problem to the group node-weighted Steiner tree problem, we propose a heuristic algorithm. The main idea is to introduce auxiliary nodes for each virtual node in the VON and then to add the physical nodes one by one to find the minimum spanning tree. We also consider the node constraint that one physical node cannot be mapped to different virtual nodes in the VON at the same time, and that each virtual node is mapped to one physical node. We introduce the node's factor μ_k for the physical node k to be the sum of the cost of the minimum-cost paths from the node k to each of the auxiliary nodes.

The details of the proposed Auxiliary-graph-based Availability (AA) Algorithm are as follows:

1. Construct the auxiliary graph $\mathbf{G_{aux}}$: start with $\mathbf{G_a}$, and add an auxiliary node for each virtual node in the VON. Connect each auxiliary node to their candidate nodes. The

weight of each auxiliary node and the links from the auxiliary node to its candidate nodes are set to 0.

2. Sort all physical nodes in ascending order according to each node's factor μ_k, and store the physical nodes in a list Γ.

3. Add all auxiliary nodes to set Θ, and construct a graph $\mathbf{G_{mst}}$ with all the nodes in Θ. Initially, there are no links between the nodes in $\mathbf{G_{mst}}$. Set the minimum total weight $C_m = \infty$.

4. Select the first node n_0 from Γ and calculate its shortest paths to all nodes in Θ. Check the nodes along the shortest path. If there is one node $n_m \in \Theta$, add one link between n_0 and n_m in $\mathbf{G_{mst}}$. Then go to **Step 5**. If there is a candidate node n_c along the shortest path and the next node along the shortest path is an auxilary node, move n_c from Γ to Θ and add a link between n_0 and n_c. Then remove all the links between n_c and all other auxiliary nodes in $\mathbf{G_{aux}}$ to follow the VON node mapping constraints.

5. Change the link weight of all auxiliary links in $\mathbf{G_{mst}}$ to a very large value (e.g. the total weight of all links), and calculate the minimum spanning tree (MST) and obtain the minimum weight C_v. To be noted, the weight of the auxiliary links in current MST is not added for the calculation of C_v. If $C_v < C_m$, let $C_m = C_v$ and remove n_0 from Θ. Then repeat the **Step 4**. If not, roll back $\mathbf{G_{mst}}$, Γ, and Θ to that before adding n_0 and move n_0 to the end of Γ. Then repeat **Step 4**. If all nodes in Γ have been tried, and a total weight less than C_m cannot be found, stop here.

3.4.3 An illustrative example

We give an example illustrating the calculation details in Fig. 3.3. As shown in Fig. 3.3(a), A VON request with three virtual nodes and three virtual links are mapped to a physical network with six nodes. First we add the auxiliary nodes 6, 7, and 8 and their corresponding auxiliary links to the graph, as shown in Fig. 3.3(b). Then we calculate the factor μ for each physical node. For example, $\mu_0 = 8, \mu_1 = 30, \mu_2 = 16, \mu_3 = 16, \mu_4 = 15, \mu_5 = 23$. Then we obtain $\Gamma = \{0, 4, 2, 3, 5, 1\}$. Initially, $\Theta = \{6, 7, 8\}$. We then select Node 0 and calculate the shortest path from Node 0 to Node 6. The shortest path spans $link_{0,6}$. Because Node 0 is one candidate of virtual node 0, and Node 6 is an auxiliary node, we delete the link between Node 0 and Node 7 to follow the node mapping constraints. The shortest path from Node 0 to Node 8 spans $link_{0,2}$ and $link_{2,8}$. Node 2 is also one candidate of virtual node 2, therefore Node 2 is moved from Γ to Θ. Node 3 is added in the same way. We obtain the current $\mathbf{G_{mst}}$ and calculate the MST by Prims algorithm[**?**], as shown in Fig. 3.3(d). Because $C_v = 16 < C_m = \infty$, we keep this solution. Then we move Node 4 from Γ (current $\Gamma = \{4, 5, 1\}$) to Θ and calculate the shortest path to each node in Θ as shown in Fig. 3.3(e). We obtain the current $\mathbf{G_{mst}}$ and find the MST shown in Fig. 3.3(f). Because $C_v = 15 < C_m = 16$, we keep this solution and upadate $C_m = 15$. Next, we add Node 5 to Θ (current $\Gamma = \{5, 1\}$) and add the shortest path to $\mathbf{G_{mst}}$ in the same way. Note that the shortest path from Node 5 to Node 6 spans $link_{5,4}, link_{4,0}$, and $link_{0,6}$. Since Node 4 is already in Θ, only the link between Node 5 and Node 4 is added, as shown in Fig. 3.3(g). We obtain the $\mathbf{G_{mst}}$ shown in Fig. 3.3(h) and the MST shown by the blue lines. Since $C_v = 20 > C_m$, Node 5 is moved to the end of Γ, and we then try Node 1. We can easily check that the weight of adding Node 1 is also greater than C_m. Since the nodes in Γ have been tried, the algorithm stops. The final solution consists of the blue lines in Fig. 3.3(f), and virtual node 0 is mapped to physical node 0, virtual node 1 is mapped to physical node 3 and virtual node 2 is mapped to physical node 2.

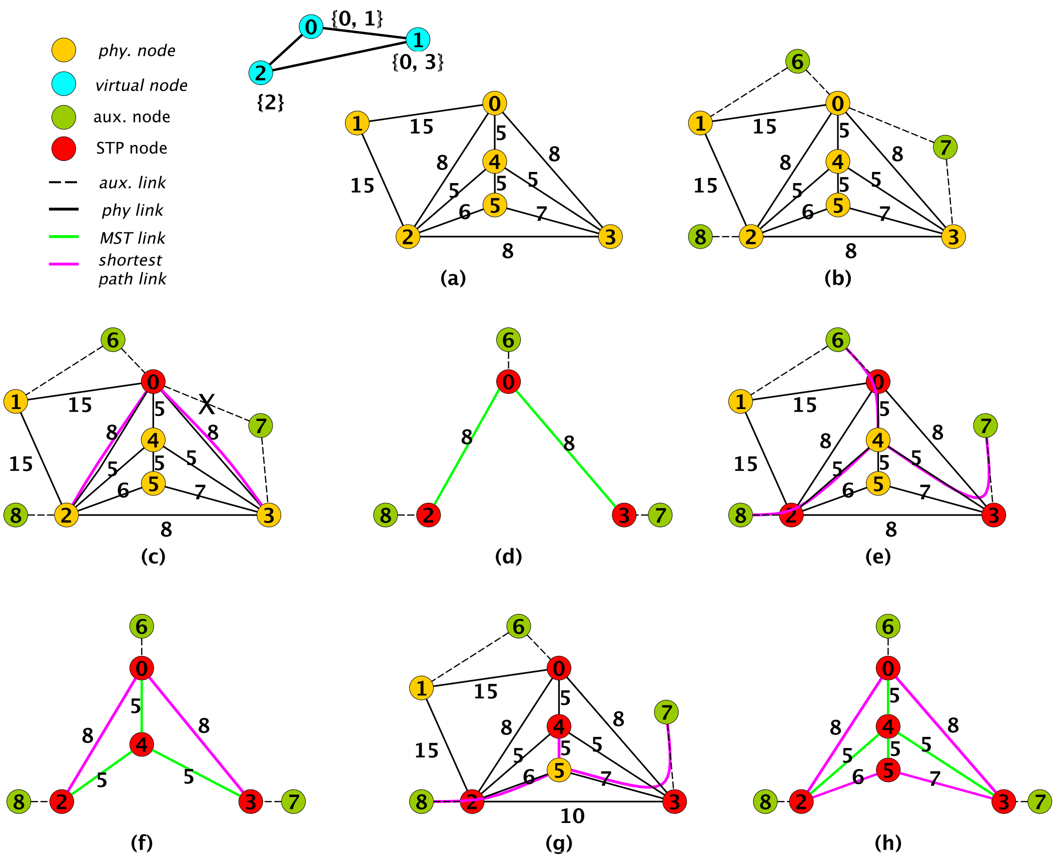


Figure 3.3. The AA algorithm illustrative example

3.5 Effective heuristic algorithm for VON mapping

In order to reduce the total link cost while guaranteeing the availability requirement of the VON, we propose a heuristic algorithm called the Mix algorithm for mapping VONs. In short, we first evaluate the maximum availability using the AA algorithm. If the maximum availability can meet the VON's availability requirement, there is no need to provide any protection. Otherwise, all virtual links are protected to increase availability, and shared backup path protection among VONs is adopted to reduce the total link cost. Each spectrum has a sharing limit [18] to avoid too many VONs being mapped to the same spectrum. For virtual node mapping, we map the virtual nodes to the physical nodes with the maximum availability using the AA algorithm. Then we map the virtual links and allocate spectrum based on each VON's bandwidth demand.

The details of the algorithm are as follows.

1. Find the maximum availability (A_{max}) and $\mathbf{G_{mst}}$ of the VON mapping using the AA algorithm.

2. Map each virtual node to its candidate node in $\mathbf{G_{mst}}$.

3. If $A_{max} > A_{req}$, go to **Step** 4 to find the VON mapping without any protection. Otherwise, go to **Step** 5 to provide protection for the VON.

4. Provide no protection for the VON. Map each virtual link to the shortest path found in $\mathbf{G_{mst}}$ and allocate spectrum by First-Fit policy. Then find K shortest paths for each virtual link based on $\mathbf{G}$. For each virtual link, try the K shortest paths one by one until the spectrum allocation succeeds and the VON's availability is met. Otherwise, go to **Step** 5 to provide shared backup path protection.

5. Provide shared backup path protection for VONs. First, find a link-disjoint path in $\mathbf{G_{mst}}$ for each virtual link as its backup path. Allocate spectrum for the working path by First-Fit policy, and allocate spectrum for the backup path with the maximum sharing of the backup path. Then check whether the VON's availability is met and the availability of the VONs that share the backup path are also met. If yes, stop here. Otherwise, re-allocate the spectrum with less sharing of the backup path until all VONs' availabilities are met. If no solution can be found, block this VON request.

To compare with the Mix algorithm, we use a greedy mapping algorithm. We select the physical node pair from the physical candidate nodes with the shortest path (with the availability function as the weight), and then map the virtual nodes to the physical nodes and map the virtual link to the shortest path as the working path. Then we find another shortest path between the one of the already-mapped virtual nodes and one of the unmapped virtual nodes' candidate nodes, and map the virtual nodes and virtual links one by one until the VON is mapped. After VON mapping, we calculate the current VON's availability. If the availability is met, replace the working path of each virtual link by the path with the shortest physical link cost and allocate the spectrum by First-Fit policy until the availability is not met. If the availability is not met, keep the nodes and the working paths mapped and find a link-disjoint path for each virtual link, and then provide shared backup path protection in the same way as the Mix algorithm. Finally, if the availability is not met or if the spectrum allocation fails, block this VON request.

3.6 Numerical results

In this section, we evaluate the performance of the proposed Auxiliary-graph-based Availability (AA) algorithm and the Mix algorithm in the 24-node U.S. mesh network. For each physical fiber link, we consider availability ranging from 0.999 to 0.999999 and for each physical node, we consider an availability of 0.99999. Each physical link has a spectrum capacity of 4 THz with a spacing of 12.5 GHz per spectrum slot, and thus there are 320 slots on each fiber link [69, 71]. For each VON request, the number of virtual nodes is between 3 and 5 and virtual links bandwidth range from 2 to 5 spectrum slots. Each VON's availability requirement is randomly generated based on its optimal availability.

We evaluate the performance of the proposed AA algorithm to find the maximum availability. We assume that there are sufficient spectrum slots and spectrum allocation is not considered in the availability calculation. We compare the performance of the AA algorithm with the optimal availability and the greedy mapping algorithm. For the optimal availability, we construct a topology graph by transforming each network element's availability using Eq. 3.7, and then enumerate all the possible subgraphs of the topology graph that can map the virtual nodes and virtual links of the VON. For each subgraph, we find the minimum spanning tree using Prim's algorithm [11] and calculate the total cost. Finally, we calculate the optimal availability based on the subgraph with minimum total cost. For the greedy mapping algorithm mentioned in Section 3.5, the physical node pair with the shortest path is chosen for node mapping, and the virtual links are mapped to the shortest paths. After all virtual nodes and virtual links are mapped, we calculate the current availability for comparison. We evaluate the availability of 600 randomly generated VONs and calculate the geometric mean of the availability. As shown in Table 5.1, our proposed AA algorithm achieves an availability that is very close to the optimal availability and performs better than the greedy mapping algorithm. In the AA algorithm, we introduce the node's factor to evaluate each physical node, and nodes with higher factor contribute towards higher availability VON mappings.

We compare the total link cost of VON mappings and the blocking rate between the Mix algorithm and the greedy mapping algorithm. The link cost is the product of the physical link distance and the spectrum slots used by the VON. We obtain the cumulative link cost from 20 VONs to 100 VONs and each data point is the mean value over six simulation runs in which there are 100 VON requests each. When the sharing limit is 1, dedicated protection for virtual links in a VON is provided. We compare the total links cost of the VONs that are mapped successfully by both algorithms. As shown in Fig. 3.4, we find that the Mix algorithm performs better in reducing the total link cost compared to the greedy mapping algorithm. Because the Mix algorithm maps the nodes and links with the highest availability, it contributes to reducing the total link cost by increasing the availability of VON mappings without protection. Therefore, there is no need to provide protection to meet the availability requirement of some VONs. Furthermore, the total link cost decreases with an increase of the sharing limit because the spectrum can be shared by more VONs for backup paths. We also compare the Mix algorithm and the greedy algorithm in terms of the blocking rate

Table 3.2. Availability comparision

	Optimal	AA Algorithm	Greedy Algorithm
Availability	0.999247	0.999231	0.998457

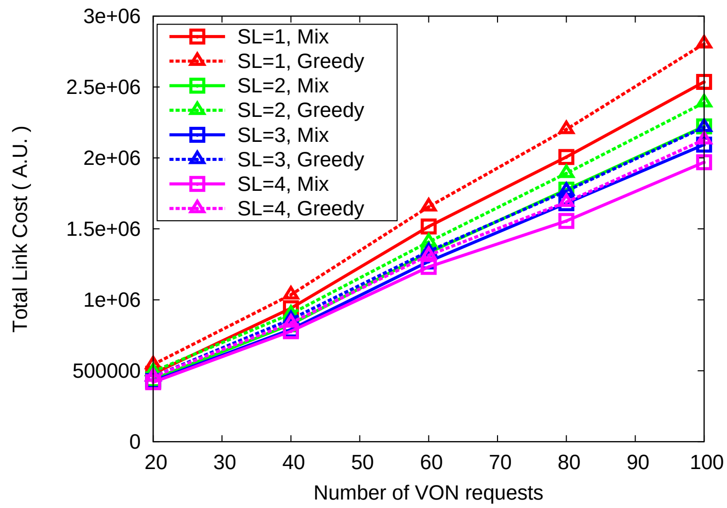


Figure 3.4. Total Link Cost

which is caused primarily by failure to satisfy the required availability. We obtain the mean blocking rate over six simulation runs in which there are 100 VON requests each, and we vary the sharing limit from 1 to 4. As shown in Fig. 3.5, the Mix algorithm can reduce the blocking rate significantly. Because the Mix algorithm is based on the AA algorithm, which finds the VON mapping with the maximum availability, it can provide higher availability than the greedy algorithm and thus reduce the blocking rate caused by availability.

3.7 Conclusion

In this chapter, we consider the problem of virtual optical network mapping with availability requirements, and we propose an efficient heuristic algorithm to find the maximum availability for the VON mapping problem. We provide shared backup path protection for VONs

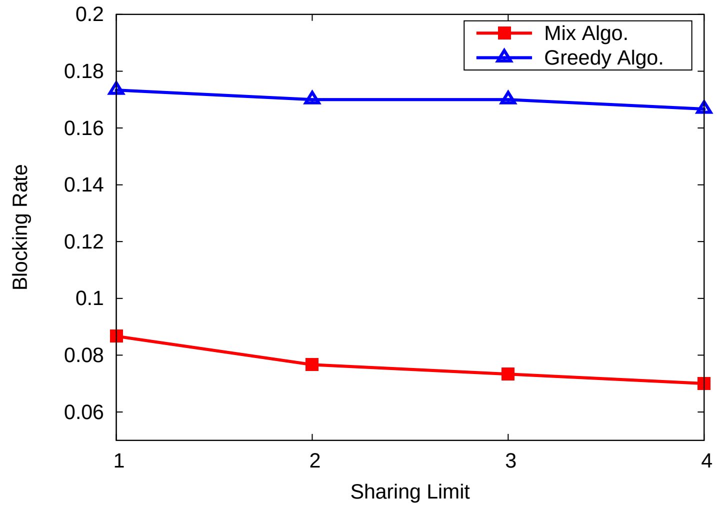


Figure 3.5. Blocking Rate

to reduce the total link cost while guaranteeing each VON's availability requirement. To find the maximum availability VON mapping without protection, we transform the problem to a group node-weighted Steiner tree problem and propose an efficient heuristic algorithm to find the maximum availability. Numerical results show that our proposed AA algorithm performs close to the optimal availability, and our Mix algorithm performs well in terms of reducing total link cost and blocking rate.

CHAPTER 4

CORRELATED-FAILURE-AWARE RESILIENT VIRTUAL OPTICAL NETWORK MAPPING[1]

In this chapter, we address the high-availability VON mapping problem over physical optical networks in which physical links may suffer correlated failures. We first aim to find a VON mapping that provides the maximum availability without any protection, and then we provide a dedicated path protection mechanism to guarantee the availability of each VON if needed. We propose the Group Steiner tree problem under correlated cost (GSTP-CC), which can be used to determine the maximum availability of a VON mapping when no protection resources are provisioned. For a special case of this problem in which there are only nodal correlated failures, called the Group Steiner tree problem under nodal correlated failure (GSTP-NCF), we propose an approximate solution based on hierarchically well-separated trees. For the general GSTP-CC problem, we propose two heuristic algorithms to find the maximum availability of the VON mapping under correlated failures. We then address the VON mapping under correlated failures by providing dedicated protection based on the GSTP-CC algorithms [45, 49].

4.1 Introduction

When virtual optical networks (VONs) are deployed over physical optical network infrastructures, correlated failures among physical optical network components may significantly affect the SLA of VON services. A major challenge is to find a high-availability VON mapping that is resilient to correlated failures in the physical optical network. In this chapter, we address the high-availability VON mapping problem over physical optical networks in

[1]©IEEE. Reprinted, with permission, from Jian Kong, Nannan Wang, Jason P. Jue, Inwoong Kim, Xi Wang, Qiong Zhang, Hakki C. Cankaya, Weisheng Xie, and Tadashi Ikeuchi. "Correlated-failure-aware VON mapping." In *Optical Fiber Communications Conference and Exhibition (OFC)*, 2017.

which physical links may suffer correlated failures. We first aim to find a VON mapping that provides the maximum availability without any protection, and then we provide a selective dedicated path protection mechanism to guarantee the availability of each VON if needed. We start by introducing a Markov model to calculate the availability of a VON mapping under correlated failures. We then propose the Group Steiner tree problem under correlated cost (GSTP-CC), which can be used to determine the maximum availability of a VON mapping when no protection resources are provisioned. For a special case of this problem in which there are only nodal correlated failures, called the Group Steiner tree problem under nodal correlated failure (GSTP-NCF), we propose an approximate solution based on hierarchically well-separated trees with approximation ratio $O(\log N \log n \log\log n \log k)$. For the general GSTP-CC problem, we propose two heuristic algorithms, the Contract-based GSTP-CC algorithm and the Replace-based GSTP-CC algorithm, to find the maximum availability of the VON mapping under correlated failures. We then address the VON mapping under correlated failures by providing dedicated protection based on the GSTP-CC algorithms. Simulation results demonstrate the efficiency of the proposed algorithms.

4.2 Problem statement

4.2.1 Impact of correlated failures

Correlated failures among physical network components can affect the availability when those components are used at the same time. If we ignore such correlated failures when finding the availability-guaranteed VON mapping, the VON mapping solution may be not optimal.

For example, as shown in the Fig. 4.1, a physical optical network has 5 fiber links with each link having a cost of 100. Assume that the $link_{1,2}$ and $link_{2,4}$ suffer not only from independent failures but also a correlated failures while all other links suffer independent failures. We assume that $link_{1,2}$ and $link_{2,4}$ are 100% correlated, and they are either both

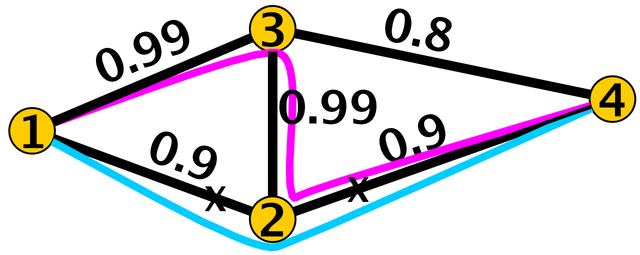


Figure 4.1. Availability of the VON mapping under correlated failure

available or both unavailable at any given time. Therefore, the combined availability of those two links is 0.9, which is also the availability of each individual link. There is a VON request with only one virtual link whose endpoints are mapped to nodes 1 and 4, and the VON's availability requirement is 0.85. The objective is to find the VON link mapping with minimum total link cost while guaranteeing its availability requirement. If we don't consider the correlated failure between the links, the best VON mapping is to map the virtual link to the path ($link_{1,3}$, $link_{3,2}$, $link_{2,4}$) with total link cost 300 and the availability is $0.99 * 0.99 * 0.9 = 0.882$. The path ($link_{1,2}$, $link_{2,3}$) is not selected because those two links are assumed to be independent and the availability is $0.9 * 0.9 = 0.81 < 0.85$. However, if we consider the correlated failures between $link_{1,2}$ and $link_{2,4}$, the optimal solution for the VON mapping is to map the virtual link to the path ($link_{1,2}$, $link_{2,4}$) with total link cost 200 and availability 0.9.

Therefore, the correlation between the physical network elements can affect the availability of each network element, thus, we need to take the correlation into consideration when we find the VON mapping solution.

4.2.2 VON mapping under correlated failures

In this chapter, we are given a physical optical network $G_s(N_s, L_s, \lambda_{ind}, \lambda_{corr}, \mu_s)$, where N_s is the set of physical nodes, L_s is the set of physical fiber links, λ_{ind} is the set of each link's independent failure rates, where $\lambda_i \in \lambda_{ind}$ is the rate at which link i fails, λ_{corr} is the set of the links' correlated failure rates, where $\lambda_{ij} \in \lambda_{corr}$ is the rate at which links i and j fail at the same time due to a correlated failure, and μ_s is the set of repair rates. We assume that each physical link has a sufficient number of spectrum slots to accommodate the VONs.

We are also given a virtual optical network request with topology $G_v(N_v, L_v, A_v)$, where N_v is the set of virtual nodes, L_v is the set of virtual links, and A_v is the availability requirement of the VON. We are also given a set of candidate physical nodes for each virtual node. Our work differs from most existing works on availability in that the VON is defined to be available only when all its virtual nodes and virtual links are available at the same time.

The problem in this paper is to find the mapping for each virtual node and the paths for each virtual link while satisfying the VON's availability requirement when correlated link failures are considered in the physical optical network. Dedicated protection resources may also be provisioned to meet the availability requirement. In the case that availability cannot be guaranteed even when protection resources are deployed, we define a penalty cost as a function of the difference between the availability requirement and the actual availability provided by the VON mapping. We categorize the different levels of availability in Table 4.1. Given a VON's availability requirement A_v and its mapping availability A_m, the penalty cost for the VON mapping is defined as:

$$Pen. = \begin{cases} Level(A_v) - Level(A_m) & \text{if} \quad A_m < A_v \\ 0 & \text{if} \quad A_m \geq A_v \end{cases} \tag{4.1}$$

For example, if the availability provided by a VON mapping is $A_m = 0.92$, but the availability requirement is requirement $A_v = 0.99$, the penalty cost is calculated as $Level(0.99) - Level(0.92) = 4 - 1 = 3$. To be noted, the granularity of $Level$ can be changed according to the specific SLA between the service provider and its customers. Our primary goal is to find VON mappings which have high availability in order to minimize the penalty cost, while a secondary objective is to minimize the total link cost used for the VON mapping.

Table 4.1. Availability level

Level	Availability	*Level*	Availability
0	[0, 0.9)	5	[0.993, 0.996)
1	[0.9, 0.93)	6	[0.996, 0.999)
2	[0.93, 0.96)	7	[0.999, 0.9993)
3	[0.96, 0.99)	8	[0.9993, 0.9996)
4	[0.99, 0.993)	9	$\cdots$

In this paper, we address VON mapping under the assumptions that the wavelength allocated for a VON link must be continuous across the entire physical path on which the VON link is mapped (wavelength continuity constraint), and the assumption that one virtual node should be mapped to only one candidate physical node, and one physical node cannot be mapped to multiple virtual nodes in one VON (one-to-one node mapping constraint).

4.2.3 VON mapping availability analysis under correlated failures

Here we propose a Markov model based approach to analyze the VON availability when the VON is mapped to a physical optical network with correlated failures. We focus on links failures and assume that other network components are always available. However, the following availability analysis and calculation steps can be easily extended to other physical network component failures.

For example, as shown in Fig. 4.2, a VON with one virtual link is mapped with (L_0, L_1) as the working path and (L_2) as the backup path. We assume that L_0 and L_1 are correlated to each other. We are also given the failure rate and repair rate for each physical links in the optical network. For example, the independent failure rates of L_0, L_1 are λ_0, λ_1, their correlated failure rate is $\lambda_{0,1}$ and the repair rates are μ_0 and μ_1. We define Ω as the set of all physical links used by the VON mapping, Φ as the set of all virtual links in the VON, θ as a set of correlated links used by the VON mapping, and Θ as the set of all correlated link sets used by the VON mapping. Therefore, we obtain $\Omega = \{L_0, L_1, L_2\}$, and

$\Theta = \{\{L_0, L_1\}, \{L_2\}\}$, We can obtain the VON availability from Eq. (4.2) - Eq. (4.4).

$$\mathbf{Avail}_{VON} = \sum_{\forall \Delta, \Delta \subseteq \Omega} \mathbf{K}_\Delta \mathbf{P}_\Delta, \tag{4.2}$$

$$\mathbf{K}_\Delta = \bigwedge_{\forall i, i \subseteq \Phi} (\mathbf{A}_{i,wp,\Delta} \bigvee \mathbf{A}_{i,bp,\Delta}) \tag{4.3}$$

$$\mathbf{P}_\Delta = \prod_{\forall \theta, \theta \subseteq \Theta} \pi_{\theta,\Delta} \tag{4.4}$$

$\mathbf{K}_\Delta = 1$ if the VON is still available when all network elements in Δ fail at the same time, otherwise, $K_\Delta = 0$. $\mathbf{P}_\Delta$ is the probability that all network elements in Δ fail at the same time. $\mathbf{A}_{i,wp,\Delta}$ (or $\mathbf{A}_{i,bp,\Delta}$) is 1 if the working (or backup) path of virtual link i is available when the physical links in Δ fail, otherwise it is 0. A virtual link i is available when either its working path or backup path is available ($A_{i,wp,\Delta} \bigvee A_{i,bp,\Delta}$), and a VON is available when all of its virtual links are available (Eq. 4.4).

To calculate $\mathbf{P}_\Delta$, we can construct a Markov model $\mathbf{M}_\theta$ for each correlated link set θ based on the correlated links' failure rates and repair rates. $\pi_{\theta,\Delta}$ is the probability of the state that links in $(\theta \bigcap \Delta)$ fail while other links in θ are available in $\mathbf{M}_\theta$. The Markov model illustrating the failure states for L_0 and L_1 is shown in Fig. 4.3, where π_{L_0,L_1} is the probability that L_0 and L_1 is available at the same time, $\pi_{\overline{L_0},L_1}$ is the probability that L_0 fails while L_1 is not failed, $\pi_{L_0,\overline{L_1}}$ is the probability that L_0 is not failed while L_1 fails, and $\pi_{\overline{L_0},\overline{L_1}}$ is the probability that L_0 and L_1 fail at the same time. Consider the failure set $\Delta = \{L_0, L_2\}$. We obtain $\mathbf{P}_\Delta = \pi_{\overline{L_2}}\pi_{\overline{L_0},L_1}$. Since the working path and backup path fail at the same time, $\mathbf{K}_\Delta = 0$ in this case. We can enumerate all the cases and finally calculate the availability for the VON mapping. Note that the probability that many fiber links fail simultaneously is fairly low; therefore, we can place a limit on the number of links that fail at the same time in order to reduce the calculation complexity.

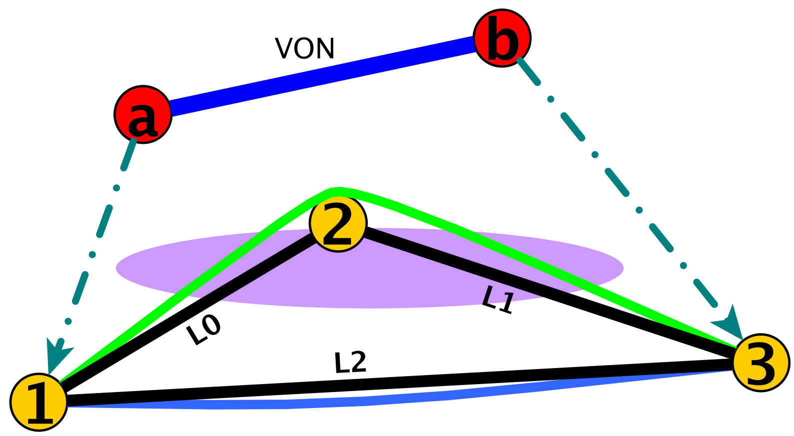


Figure 4.2. Availability calculation example

4.3 Maximum-availability VON mapping under correlated failures without protection

In some cases, it may be possible to satisfy the availability requirement for a VON without providing any additional protection. Therefore, to minimize the total penalty cost and the total physical link cost, we first attempt to find the maximum-availability VON mapping under correlated failures without any protection, and we check to see whether or not the availability requirement is met. We assume that there are sufficient wavelength resources for each physical link, and do not consider the wavelength allocation in this step. In the following subsections, we present variations of the Steiner tree problem and then show how these problems are equivalent to the problem of finding a maximum-availability VON mapping under correlated failures.

4.3.1 Problem transformation

Given a physical optical network with a correlated links set L_s with several correlated links, we can calculate the availability that one or many links in L_s are used based on the above

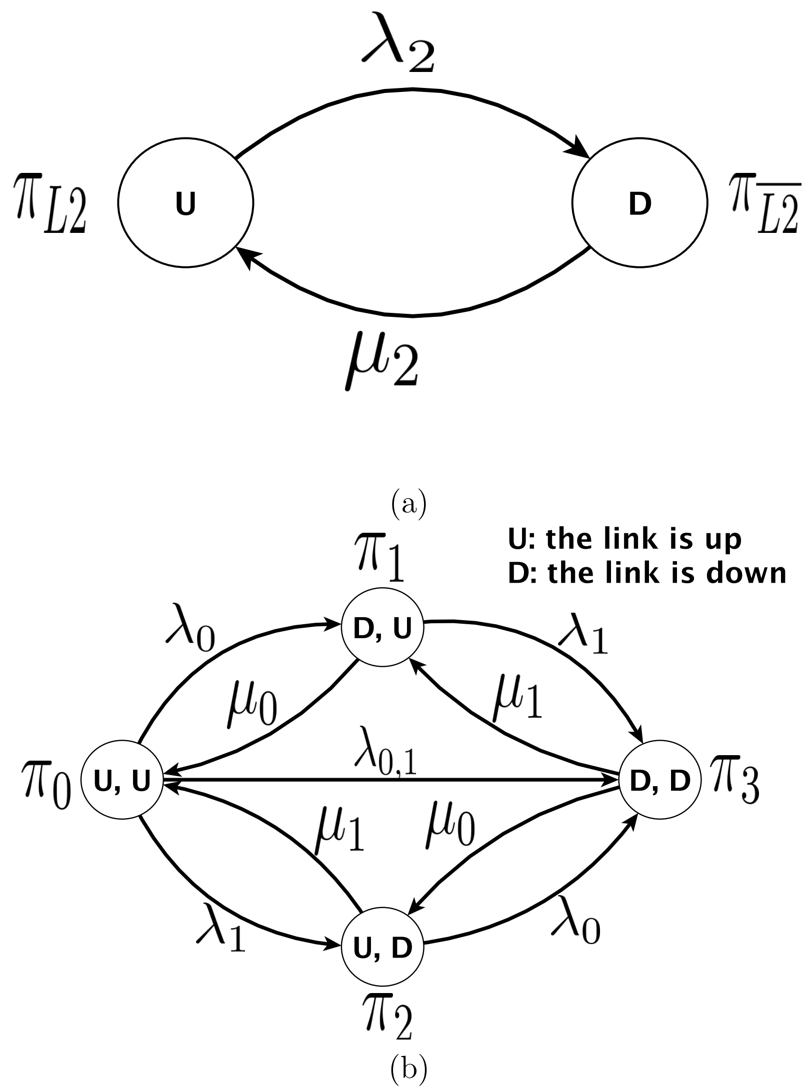


Figure 4.3. Markov model for correlated links L_0 and L_1

Markov analysis model. As shown in Fig. 4.4, the availability of L_0 is $a_0 = \pi_{L_0,L_1} + \pi_{L_0,\overline{L_1}}$, the availability of L_1 is $a_1 = \pi_{L_0,L_1} + \pi_{\overline{L_0},L_1}$, and the availability of L_2 is $a_2 = \pi_{L_2}$. Note that the availability that L_0 and L_1 are used at the same time is $a_{0,1} = \pi_{L_0,L_1}$.

We use the notation a_Δ to represent the availability when the links in Δ ($\Delta \in \Omega$ and $\Delta \neq \emptyset$) are used at the same time. Then we construct an availability-based graph $G_a(N_s, w)$

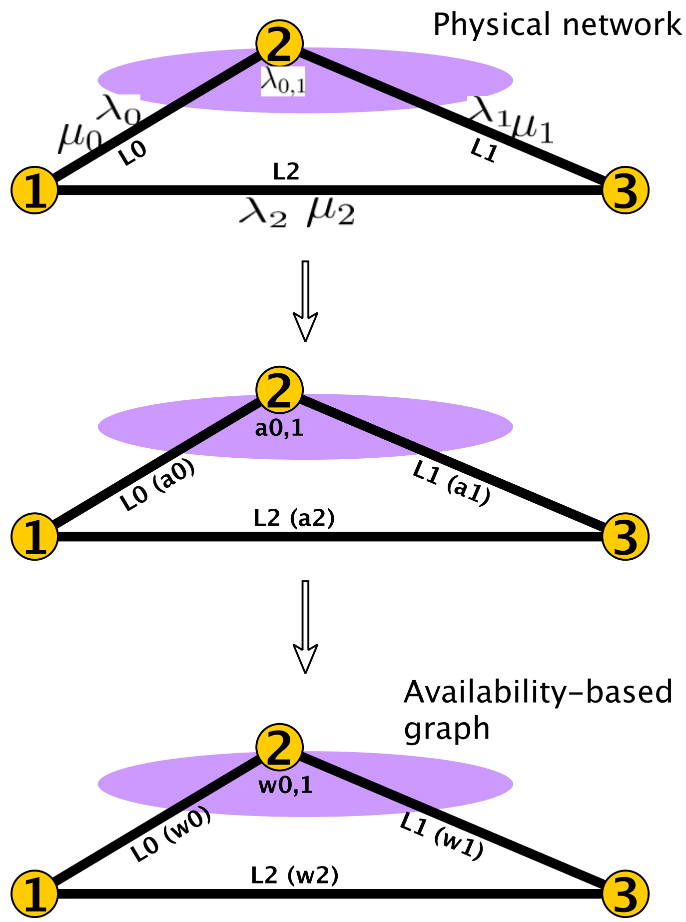


Figure 4.4. Availability-based graph

in which the link weights are determined by

$$w_\Delta = -\log a_\Delta \quad (\Delta \in \Omega, \Delta \neq \emptyset) \tag{4.5}$$

4.3.2 STP-CC and GSTP-CC

We first introduce the Steiner tree problem under correlated cost (STP-CC). The STP-CC problem differs from existing work on Steiner trees in that it considers correlated costs among different edges in the graph. Each correlated cost is represented by a set of edges and a corresponding cost for the set, such that if all of the edges in the set are included in the final solution, the set cost is incurred rather than the sum of individual edge costs. The STP-CC is described as follows:

Definition 1. *STP-CC: Given a graph $G = (V, E)$ with both independent and correlated edge weights, and a collection of terminal nodes of $t_1, t_2, \ldots, t_k \in V$, find a minimum-weight connected subgraph of G that connects each terminal node.*

Considering that one terminal node may have several candidates, we also introduce the group Steiner tree problem under correlated costs (GSTP-CC) as follows:

Definition 2. *GSTP-CC: Given a graph $G = (V, E)$ with both independent and correlated edge weights and node weights, and a collection $g_1, g_2, \ldots, g_k \subset V$ of node sets called groups, find a minimum-weight connected subgraph of G that contains at least one node from each group.*

Therefore, based on the above defination, we have the following conclusion:

Theorem 2. *Given a VON request without any protection and given several candidate nodes for each virtual node, the VON's maximum availability mapping is the solution to the GSTP-CC in the graph $G_a(N_s, w)$, where the groups $g_1, g_2, \ldots, g_k$ are the sets of each virtual node's candidate nodes.*

Proof. We prove Theorem 1 by contradiction. Assume that the total link cost of the group Steiner tree under correlated cost is C, and the availability of the mapping is $A = e^{-C}$. If the maximum availability mapping is not its group Steiner tree under correlated cost in the graph G_a, there must be a mapping with availability $\bar{A} > A$, therefore, the total link cost of this solution $\bar{C} < C$. This implies that there is one connected subgraph with less cost than the current group, which contradicts the definition of the GSTP-CC. □

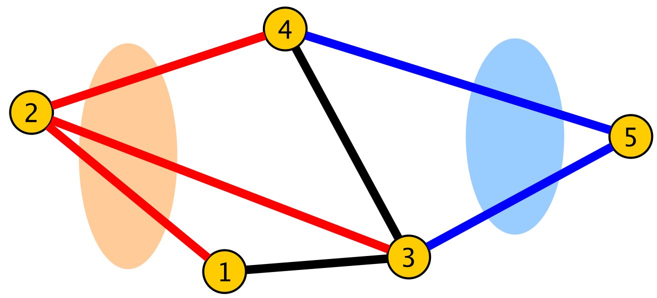


Figure 4.5. Nodal correlated failures in network

4.3.3 A special case: GSTP-NCF

A special case of GSTP-CC is the case in which only links incident to the same node are correlated. We refer to this problem as the group Steiner tree problem under nodal correlated failures (GSTP-NCF). We define Ψ as a set of links that share a correlated failure. Note that under the condition of nodal correlated failures, the links in Ψ must all be incident to a common node. As shown in Fig. 4.5, Ψ_1 contains correlated links $link_{2,1}$, $link_{2,3}$, and $link_{2,4}$ which are all incident to $node_2$, while Ψ_2 contains correlated links $link_{5,3}$ and $link_{5,4}$, which are both incident to $node_5$. The assumption of nodal correlated failures is reasonable, since links that share a node usually have higher correlation than links that do not share a common node. We assume that there is no overlapping links among different Ψ.

4.4 Approximation algorithm for GSTP-NCF in VON mapping

In this section, we focus on GSTP-NCF, and propose an approximation algorithm that guarantees availability within an approximation ratio. A possible solution for the general GSTP problem (not considering correlated cost) is to approximate the graph metric by a distribution over trees [4], and it has been proved that any metric space can be probabilistically-approximated by hierarchically well-separated trees (HST) with a polylogarithmic distortion.

We extend the work in [3] to solve GSTP-NCF in VON mapping by guaranteeing that (N_s, w) in the auxiliary graph $G_a(N_s, w)$ of the physical network graph is a metric space.

Based on the nodal correlated failures assumption in the physical optical network [75], we can prove that (N_s, w) in $G_a(N_s, w)$ is a metric space when the availability of the physical links is high enough.

Theorem 3. *(N_s, w) is a metric space when*

$$\pi_0 > \frac{\sqrt{5}+1}{2} \approx 0.618 \tag{4.6}$$

where π_0 is the availability when any two of the nodal correlated links are available at the same time.

Proof. See the Appendix. □

We observe that typical network components have availability higher than 0.9, and therefore, the GSTP-NCF in VON mapping can easily satisfy the above conclusion.

We refer to [4] and have the following theorem:

Theorem 4. *Every metric space M over a weighted connected graph V with n vertices can be α-probabilistically-approximated by a set of hierarchically well-separated trees (HSTs), where $\alpha = O(\log n \log\log n)$ in polynomial time.*

Finally, we address GSTP-NCF based on Naveen's approach [25] of the general GSTP, and we have the following result:

Theorem 5. *Given weighted connected graph G with correlated cost, for any $\epsilon > 0$, we can find a group Steiner tree whose cost is $O(\log N \log n \log\log n \log k)$ times the cost of the optimal tree with probability 1-ϵ in polynomial-time, where N is the maximum size of a group, k is the number of groups, and n is the number of vertices.*

4.5 Heuristic algorithms for GSTP-CC in VON mapping

For the more general GSTP-CC, We propose the following two heuristic algorithms, the Contract-based GSTP-CC algorithm and the Replace-based GSTP-CC algorithm. The details of the proposed algorithms are as follows.

4.5.1 The Contract-based GSTP-CC algorithm for VON Mapping

The proposed Contract-based GSTP-CC algorithm is a variant of the algorithm proposed in [43] which didn't consider correlated cost among different edges. The main idea of the Contract-based GSTP-CC algorithm is to iteratively merge the trees into large trees until there is only one tree that connects all the nodes while considering the correlated cost during each merge step.

We are given the topology graph $G = (V, E)$, and a set of trees, T, each tree $t \in T$ is not connected to the others, and each node $v \in V$ belongs to only one tree. We define a density factor for a node $v \in V$ in each iterative step as following:

$$F_v = \min \frac{\sum_{\forall t \in S} D_{v,t}}{N_S}, \tag{4.7}$$

where S is a subset of T, N_S is the number of the trees in S, and $D_{v,t}$ is the distance between node v to the tree t. Different from the case that there is correlated cost in G, we calculate $D_{v,t}$ by choosing the minimium distance among the K shortest paths between node v and the tree t without considering the correlated cost.

Based on the density factor above, we propose the Contract-based GSTP-CC algorithm as follows. We first construct an auxiliary graph G_{aux} by adding an auxiliary node for each virtual node, and connecting the auxiliary node with its candidate nodes. After determining the node and trees with the minimum density factor, we then connect the node and those trees with the shortest paths until all the nodes in G are connected. The details of the Contract-based GSTP-CC are described in Algorithm 4.5.1.

4.5.2 The Replace-based GSTP-CC algorithm for VON Mapping

The above Contract-based GSTP-CC algorithm tries to merge the trees into large trees step by step, and the solution in the previous steps is fixed, and cannot be changed. However, when we consider the correlated cost among different eges, the previous solution may be

Algorithm 1 The Contract-based GSTP-CC algorithm

Input:

The topology graph G with node set V
A set of groups P, with each group containing several candidate nodes

Output: Mapping solution

Initialization :
The tree set $T := \{\}$
Construct an auxiliary graph G_{aux} by:
for $\forall group_i \in P$ **do**
    Add one auxiliary node v_a in G
    for $\forall v \in group_i$ **do**
        Connect v_a and v
        $cost_{v_a,v} \leftarrow \infty$
    end for
    Create a new tree t_i with node v_a
    Add t_i to T
end for
Main steps:
while T has more than one tree **do**
    $min := 0$, $F_{min} = \infty$, $S_{min} := \{\}$
    for $\forall i \in V$ **do**
        for each subset S of T **do**
            Calculate $F_i = (\sum_{\forall s \in S} D_{i,s})/N_S$
            if $F_i < F_{min}$ **then**
                $min \leftarrow i$, $F_{min} \leftarrow F_i$, $S_{min} \leftarrow S$
            end if
        end for
    end for
    for $\forall s \in S_{min}$ **do**
        Connect node v_{min} to tree S with the shortest path
    end for
end while

updated to get a better result. Therefore, we propose another approach, the Replace-based heuristic algorithm, to address the general GSTP-CC problem for VON mapping. This approach differs from the Contract-based algorithm, which merges trees, in that it iteratively improves the result when a new node is added to the tree.

In this approach, we introduce a new metric, node factor η_k, for physical node k, which is the sum of the cost of the minimum-cost paths from node k to each of the auxiliary nodes. Note that correlated cost among different edges in G is not considered when calculating η_k. Similar to the Contract-based GSTP-CC algorithm, we first build an auxiliary graph G_{aux} by adding an auxiliary node for each virtual node, and then we connect the auxiliary node to its candidate nodes. We then build a tree to connect all the auxiliary nodes, and refine the result if the minimum cost is less than current result. The details of the proposed algorithm are described in Algorithm 4.5.2.

4.6 Heuristic algorithms for VON Mapping with protection under correlated failures

We aim to minimize the penalty cost by providing dedicated path protection for the VON request if the VON mapping from previously-introduced algorithms does not meet the specified VON availability requirement. When the dedicated path protection is provided in the VON mapping, correlated failures can affect the availability of the VON mapping in several different ways. As show in Fig. 4.6, the availability of the VON mapping can be affected by correlated failures among physical links that are on the same working (or backup) path ($l_{1,2}, l_{2,3}$ of type 1), or correlated failures among physical links where some links are on the working path and other links are on the backup path ($l_{3,4}, l_{4,5}$ of type 2), or a combination of those two cases ($l_{2,3}, l_{3,4}, l_{4,5}$ of type 3). Therefore, we need take different types of correlated failures into consideration when addressing the VON mapping problem with protection over physical optical network under correlated failures.

Algorithm 2 The Replace-based GSTP-CC algorithm

Input:

The topology graph G with node set V

A set of groups P, and each group has several candidates

Output: Mapping solution

Initialisation :

The minimum total weight $C_m := \infty$

Auxiliary nodes set $\Theta := \{\}$

for $\forall k \in G$ **do**

  Calculate η_k for node k

end for

Sort each node based on η_k in ascending order, and store in list Γ

Construct an auxiliary graph G_{aux} by connecting η_k with each candidate node

Construct another graph G_{mst} with all the nodes in Θ by connecting each nodes with the shortest cost paths

Main steps :

while $\Theta \neq \emptyset$ **do**

  $\tau :=$ the first node in Γ

  for $\forall \theta \in \Theta$ **do**

    Calculate K shortest paths $\zeta_{\tau,\theta,k}$ from τ to θ

  end for

  for one shortest paths combination between τ and Θ **do**

    for $\forall \theta \in \Theta$ **do**

      for each node θ_m along $\zeta_{\tau,\theta,k}$ **do**

        if $\theta_m \in \Theta$ **then**

          connect τ and θ_m in G_{mst}

        end if

      end for

    end for

    Find the minmum spanning tree of G_{mst} with weight C_v

    if $C_v < C_m$ **then**

      $C_m = C_v$

      Remove τ from Θ

    else

      Roll back G_{mst}

      Move τ to the end of Γ

    end if

  end for

end while

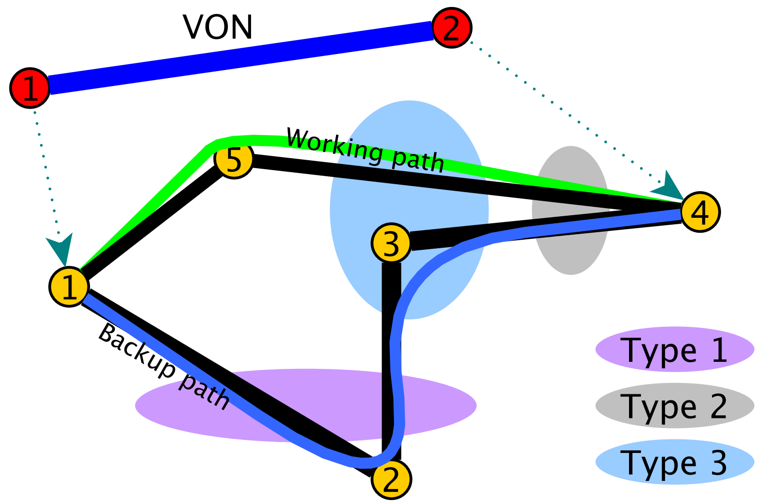


Figure 4.6. Different types of correlated failures in physical optical network

Based on the contract-based GSTP-CC algorithm and the replace-based GSTP-CC algorithm, we propose two heuristic algorithms to find a VON mapping with the objective of minimizing the penalty cost and the total link cost. The main framework of the proposed heuristics is to first find the VON mapping with the maximum availability without any protection by addressing the GSTP-CC problem, and then to provide dedicated path protection for the VON request if its availability requirement cannot be met.

4.6.1 The Contract-based VON mapping algorithm with protection

In this algorithm, we firstly apply the Contract-based GSTP-CC algorithm to find out the maximum availability when there is no protection for VON mapping, and if the availability cannot meet its requirement, the dedicated protection for each virtual links is provided to improve its availability. The correlated cost should be considered when finding out the backup path of the VON virtual links. To be noted that, the wavelength continuity and consistence constraints are also considered when allocating the wavelength for a working path and backup path. The details of the Contract-based VON mapping algorithm is described in Algorithm 4.6.1.

Algorithm 3 The Contract-based VON mapping algorithm

Input:

The physical optical network G_p with the physical links set PL and each physical link's failure rate and repair rate

The VON request V with virtual links set VL, virtual nodes set VN, and availability requirement A_{req}

Output: VON mapping solution

Main steps:

Create an availability graph G_a by:

for $\forall l_i \in PL$ **do**

  Calculate its availability A_{l_i} based on the Markov model

  Set the metric of each edge i as $logA_{l_i}$.

end for

Apply the Contract-based GSTP-CC algorithm based on G_a to find out the VON mapping without protection

for $\forall vl_i \in VL$ **do**

  for K shortest paths between nodes of vl_i **do**

    if Allocate the wavelength by the First-fit (FF) algorithm successfully **then**

      Break;

    end if

  end for

  if vl_i cannot be mapped successfully **then**

    Block this VON request, and return

  end if

end for

Calculate current availability of the VON mapping A_{act} and the total link cost L_t

if $A_{act} > A_{req}$ **then**

  for $\forall vl \in VL$ **do**

    Find the shortest cost path pl_s of vl based on G_p

    if $L_c < L_t$ and $A_{act} > A_{req}$ when pl_s is mapped to vl **then**

      Map vl to pl_s

    end if

  end for

else

  for $\forall vl \in VL$ **do**

    Find the shortest cost path pl_s as the working path of vl based on G_a

    Find K-shortest paths BL disjoint with pl_s

    Choose the path from BL with highest availability of VON mapping as the backup path.

  end for

end if

return The penalty cost and total physical link cost of the VON mapping

4.6.2 The Replaced-based VON mapping algorithm with protection

Similar to the Contract-based VON mapping algorithm, the Replaced-based VON mapping algorithm deploys the Replaced-based GSTP-CC algorithm to find the node mapping and working path of the VON mapping. We first deploy the Replaced-based GSTP-CC algorithm to find out the maximium availability when there is no protection for VON mapping, and dedicated path protection for each virtual links are provided to improve its availability if the requirement cannot be guaranteed. The correlated coast among different physical links are also considered. The details of the proposed algorithm are described in Algorithm 4.6.2:

4.6.3 Correlated-failure-unaware VON mapping algorithm

For comparison, we refer to the Mix VON mapping algorithm [46] as the correlated-failure-unaware VON mapping algorithm and we don't consider the correlated failures in the physical optical network. In this approach, we find out the maximium availabiltiy of the VON mapping by addressing the Group Steiner Tree problem (GSTP), and determine the virtual node mapping based on the GSTP solution. Each virtual link is mapped to the physical links by GSTP solution. If current availability of the VON mapping cannot meet the requirement, a dedicated path protection for each virtual link is provided.

4.7 Numerical results

In this section, we evaluate the performance of the proposed heuristic algorithms in the 24-node U.S. mesh network [68]. For each fiber link, we consider an availability range from 0.999 to 0.99999. Different correlated failures can affect the correlated link's availability and finally affect the VON mapping. We assume that one fiber link can be independent or correlated with one other link or correlated with two other links. We randomly generate the correlated links in the optical network. We consider four different levels of correlated failure

Algorithm 4 The Replace-based VON mapping algorithm

Input:

The physical optical network G_p with the physical links set PL and each physical link's failure rate and repair rate
The VON request V with virtual links set VL, virtual nodes set VN, and availability requirement A_{req}

Output: VON mapping solution

Main steps:
Create an availability graph G_a by:
for $\forall l_i \in PL$ **do**
  Calculate its availability A_{l_i} based on the Markov model
  Set the metric of each edge i as $logA_{l_i}$.
end for
Apply the Replace-based GSTP-CC algorithm based on G_a to find out the VON mapping without protection
for $\forall vl_i \in VL$ **do**
  for K shortest paths between nodes of vl_i **do**
    if Allocate the wavelength by the First-fit (FF) algorithm successfully **then**
      Break;
    end if
  end for
  if vl_i cannot be mapped successfully **then**
    Block this VON request, and return
  end if
end for
Calculate current availability of the VON mapping A_{act} and the total link cost L_t
if $A_{act} > A_{req}$ **then**
  for $\forall vl \in VL$ **do**
    Find the shortest cost path pl_s of vl based on G_p
    if $L_c < L_t$ and $A_{act} > A_{req}$ when pl_s is mapped to vl **then**
      Map vl to pl_s
    end if
  end for
else
  for $\forall vl \in VL$ **do**
    Find the shortest cost path pl_s as the working path of vl based on G_a
    Find K-shortest paths BL disjoint with pl_s
    Choose the path from BL with highest availability of VON mapping as the backup path.
  end for
end if
return The penalty cost and total physical link cost of the VON mapping

cases shown in Table II and Table III. λ is the failure rate for one physical link with the unit $failures\ per\ million\ hours(fpmh)$ [20], and we assume that the failure repair rate η for each physical link is 100000.0 $fpmh$. For each VON request, the number of virtual nodes is between 3 and 5, and the bandwidth of virtual links ranges from 2 to 5 wavelengths. We evaluate the performance considering three different levels of VON availability requirement: 0.999, 0.9999 and 0.99999.

We first evaluate the performance of the Contract-based, the Replace-based, and the correlated-failure-unaware GSTP-CC algorithms under correlated failures. For comparison, we also determine the highest possible attainable availability of the GSTP-CC problem by enumerating all possible VON mapping solutions, which is denoted as Optimal Algorithm. We assume that there are sufficient wavelength resources and the wavelength allocation is not considered in the availability calculation. We evaluate the availability of 200 randomly generated VONs and calculate the geometric mean of the availability. As shown in Table 4.4, the proposed Replaced-based GSTP-cc algorithm performs the best among the heuristic algorithms for different correlated failure cases. Compared to the Contract-based GSTP-CC algorithm that merge the trees into large trees until there is only one tree which connect all the nodes, we iteratively improve the result when incorporating a new node in the Replace-based GSTP-CC algorithm, which contributes to a higher availability. We also note that the proposed Contract-based and Replace-based GSTP-CC algorithms perform better than the Correlated-unware algorithms which shows the impact of correlated failures on the VON mapping solution.

We then evaluate the total penalty cost based on the penalty *Level* defined in the Table 4.1 for the proposed VON mapping algorithms. The availability requirement of each VON request is 0.999, 9.9999, or 0.99999. As show in Figs. 4.7 - 4.8, the Replace-based VON mapping algorithm performs the best in reducing the penalty cost. The Replace-based VON mapping algorithm not only deploys the Replace-based GSTP-CC algorithm to find a VON

Table 4.2. Correlated failure rate for two links (fpmh)

	λ_1	λ_2	λ_{12}
case 1	20.0	20.0	1.0
case 2	1.0	1.0	20.0

Table 4.3. Correlated failure rate for three links (fpmh)

	λ_1	λ_2	λ_3	λ_{12}	λ_{13}	λ_{23}	λ_{123}
case 1	30.0	30.0	30.0	1.0	1.0	1.0	1.0
case 2	1.00	1.00	1.00	1.00	1.00	1.00	30.0

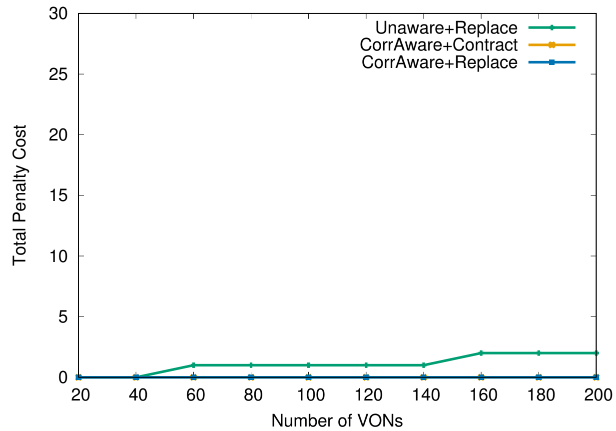
mapping without any protection, which contribute to a higher availability, but also considers the different types of correlated failures for the VON mapping. We also find that correlated failures can affect the VON mapping availability when comparing the different failure level cases. We also compare the total link cost used by the proposed algorithms. As shown in Figs. 4.9 - 4.10, the proposed Replace-based algorithm result in less link cost than other algorithms because it can provide higher availability, resulting in lower need to provide dedicated protection for some VON request.

4.8 Conclusion

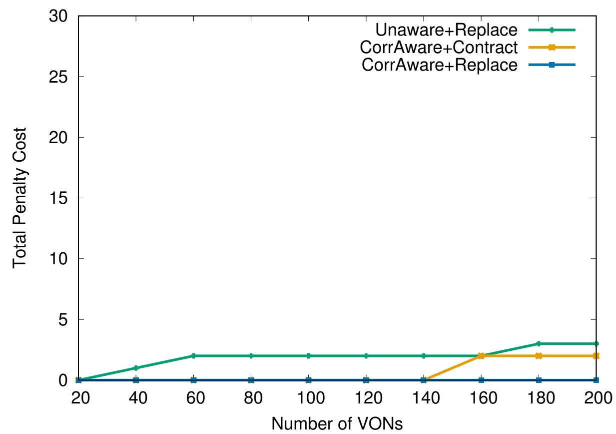
In this chapter, we address the problem of high-availability VON mapping over a physical optical network under correlated failures. We introduce the Group Steiner tree problem under correlated cost (GSTP-CC), and propose the Contract-based and the Replace-based heuristic GSTP-CC algorithms to address the problem. For a special case, GSTP-NCF, we propose an approximation algorithm based on hierarchically well-separated trees with an approximation ratio $O(\log N \log n \log\log n \log k)$. We solve the VON mapping problem under correlated failures by providing dedicated path protection based on the proposed GSTP-CC algorithms. Simulation results demonstrate the efficiency of the proposed algorithms.

Table 4.4. GSTP-CC availability comparison

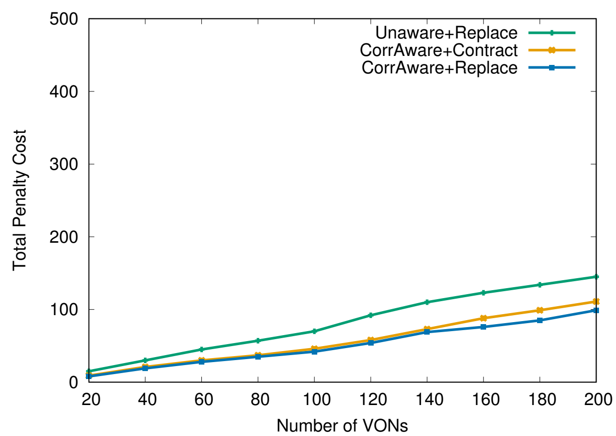
	Corr-Unaware Algo.	Corr-Aware-Contract Algo.	Corr-Aware-Replace Algo.	Optimal Algorithm
Case 1	0.9991108	0.9991508	0.9991718	0.9991822
Case 2	0.9991880	0.9992450	0.9992587	0.9992625



(a) $A_{req} = 0.999$

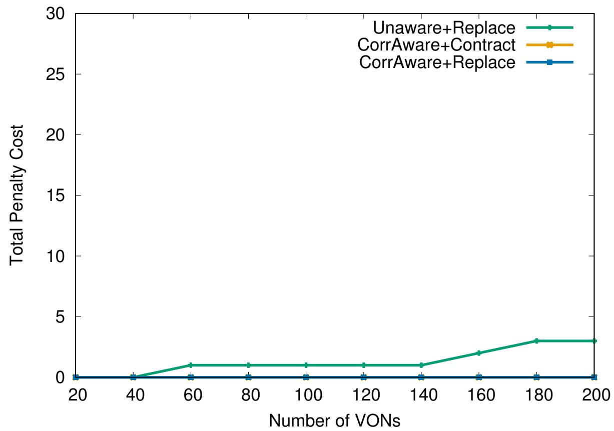


(b) $A_{req} = 0.9999$

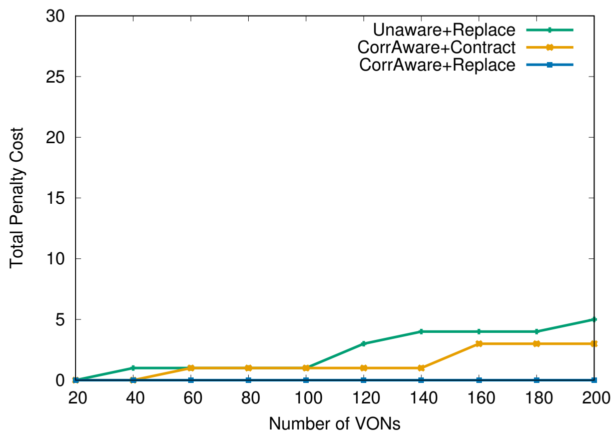


(c) $A_{req} = 0.99999$

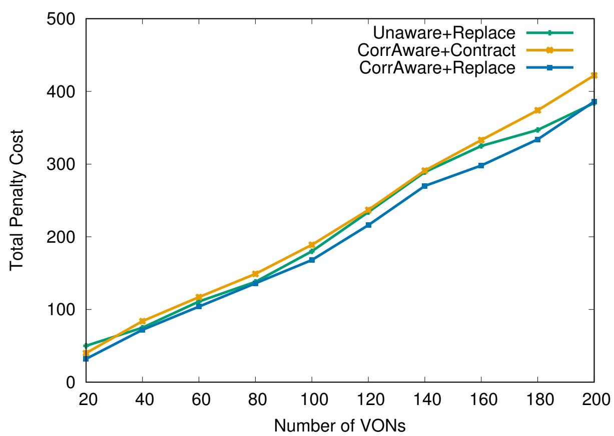
Figure 4.7. Penalty Cost Comparison for Case 1



(a) $A_{req} = 0.999$

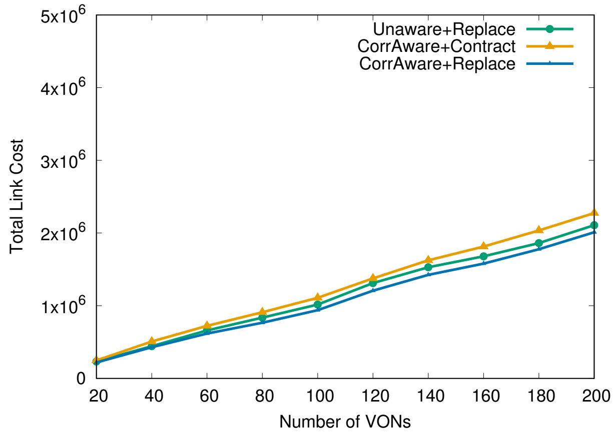


(b) $A_{req} = 0.9999$

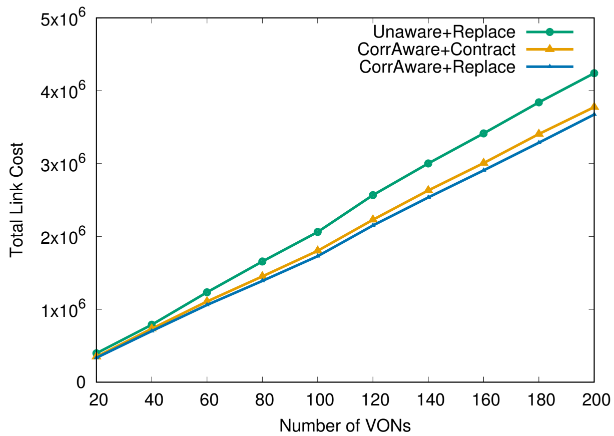


(a) $A_{req} = 0.99999$

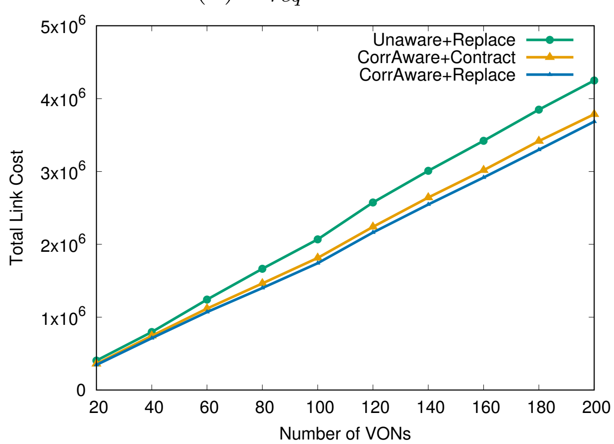
Figure 4.8. Penalty Cost Comparison for Case 2



(a) $A_{req} = 0.999$

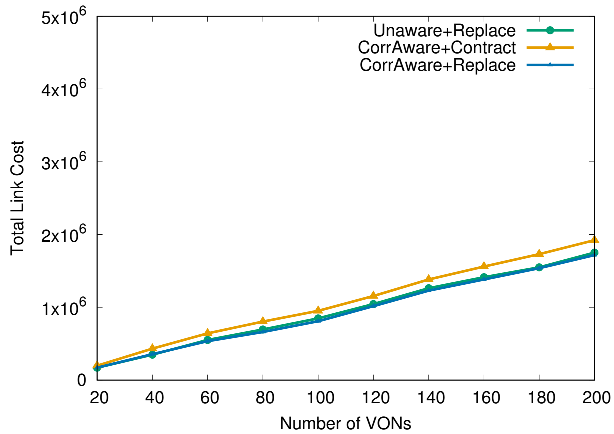


(b) $A_{req} = 0.9999$

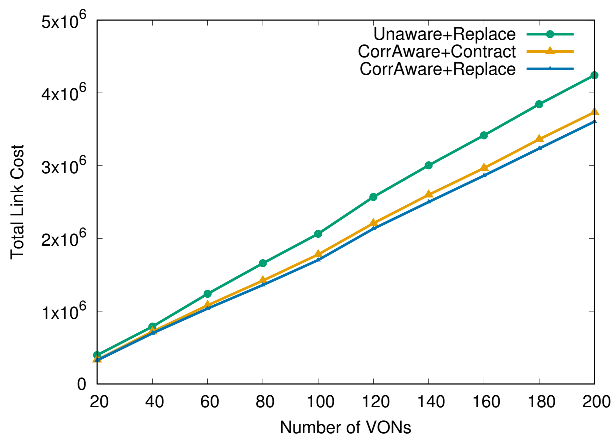


(c) $A_{req} = 0.99999$

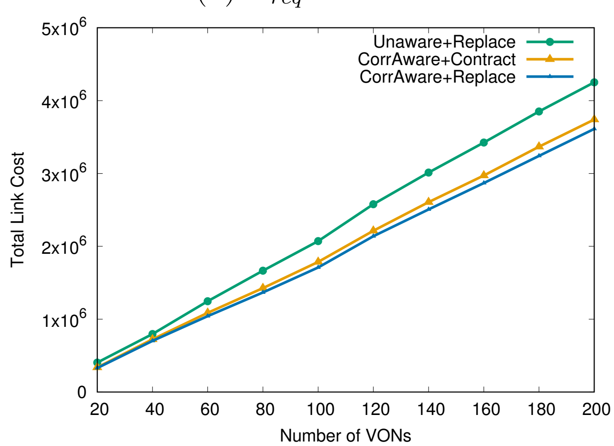
Figure 4.9. Total Link Cost for Case 1



(a) $A_{req} = 0.999$



(b) $A_{req} = 0.9999$



(c) $A_{req} = 0.99999$

Figure 4.10. Total Link Cost for Case 2

CHAPTER 5

HIGH-AVAILABILITY-GUARANTEED SERVICE FUNCTION CHAIN MAPPING WITH COLLABORATIVE NETWORK PROTECTION AND VNF REPLICATION[1]

Network function virtualization (NFV) provides an efficient and flexible way to deploy network services in the form of service function chains (SFCs) by adopting generalized equipment. However, software-based virtualized network functions (VNFs) bring new challenge for network operators in providing service availability guarantees. Traditionally, network-level protection mechanisms are considered separately from function-level VNF backup mechanisms. However, a SFC's availability cannot be guaranteed if only network-level protection mechanisms or only function-level VNF backup mechanisms are considered. In this chapter, we propose a coordinated protection mechanism that adopts both backup path protection in the network and VNF replicas at nodes to guarantee a SFC's availability. The proposed mechanism determines the number of replicas required for each VNF in the SFC, and allocates the replicas to physical nodes on the working and backup paths while maintaining ordered dependency among VNFs. To further improve the SFC request acceptance and reduce the total computing and link resource consumption, we also propose an efficient segment protection mechanism to guarantee each SFC's availability requirement [47].

5.1 Introduction

With the increasing growth of dynamic services based on technologies such as cloud/fog computing and big data, a key challenge for network providers is how to flexibly and efficiently deploy new network functions or change existing network services [77]. Traditionally,

[1]©IEEE. Reprinted, with permission, from Jian Kong, Inwoong Kim, Xi Wang, Qiong Zhang, Hakki C. Cankaya, Weisheng Xie, Tadashi Ikeuchi, and Jason P. Jue. "Guaranteed-Availability Network Function Virtualization with Network Protection and VNF Replication." In *IEEE Global Communications Conference (GLOBECOM)*, 2017.

network functions are deployed as physical appliances that are tightly coupled with specific hardware. Physically deployed network functions hinder the rapid deployment of new services and result in operational challenges. Network function virtualization (NFV) provides a more agile network by implementing the virtual network functions (VNFs) in software that is decoupled from the underlying hardware, and which therefore, significantly decreases the OPEX and CAPEX. Based on the service provider's demand, end-to-end network services may consist of an ordered set of VNFs, and such ordered functions connected by logical links are referred to a service function chain (SFC)[32].

Availability, which is defined as the fraction of time that a system is providing service to its users, is an important factor in measuring the quality of service in a network and can also be applied as a metric to measure the service quality in one SFC. In general, VNFs are deployed on generalized hardware that may lack the robustness of standard telecommunications equipment. Physical hardware failures may also affect several VNFs in one SFC. Furthermore, the malfunction of the software-based VNFs and the loss of logical connections in one SFC may also result in the unavailability of the service[58]. Therefore, software-based VNFs and flexible SFC mapping bring new challenges for network operators to provide availability-guaranteed network services. The availability-guaranteed SFC mapping problem is to determine the mapping of the VNFs in one SFC to guarantee the SFC's availability.

The availability of both the network service and the SFC can be improved by adopting different protection mechanisms. Typically, network-layer protection (e.g., backup path) is provided to improve the availability of traditional network services. For software-based VNFs (one VNF may be deployed on one virtual machine), several replicas of each VNF can be provided to guarantee availability[34, 19, 76]. However, existing works provide replicas for VNFs while ignoring the availability of physical network components. For example, as shown in Fig. 5.1, consider a SFC with two VNFs mapped to the physical network, and with an availability requirement 0.999. If there is no network-layer protection, the maximum

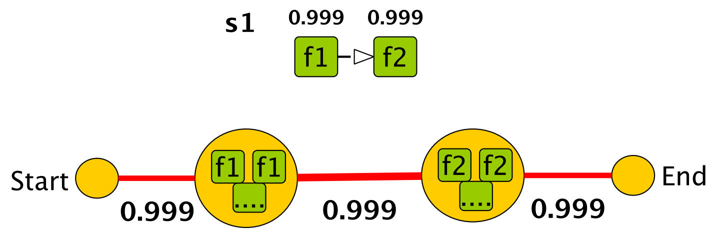


Figure 5.1. Availability of SFC example

availability of the SFC mapping will be less than the availability of the physical path (around 0.997). Furthermore, if no backup replicas are provided for VNFs, the maximum availability of the SFC mapping will be less than 0.998. Therefore, only providing backup paths in the network layer or only allocating replicas for each VNF in the application layer cannot guarantee the SFC's availability requirement. In contrast to existing work, we address the availability-guaranteed SFC mapping problem by considering both path protection in the network-layer and replicas for each VNF in the application layer in a coordinated manner.

In this chapter, we analyze SFC availability, considering both network link availability and VNF availability, and we develop an approach for guaranteeing the availability requirement of an SFC utilizing both network-layer protection and application-layer VNF replicas. By analyzing how to distribute VNF replicas between the working path and backup path in order to maximize the SFC's availability, we determine the number of replicas for each VNF in the SFC, and we allocate the replicas to physical nodes while maintaining ordered dependency among VNFs. To further improve the SFC request acceptance and to reduce the total computing and link resource consumption, we extend our previous results by proposing an efficient segment protection mechanism to guarantee each SFC's availability requirement. Furthermore, we provide a comprehensive performance evaluation of the proposed algorithms considering different levels of physical network availability and different levels of VNF availability.

The rest of this chapter is organized as follows. In Section 5.2, we present the SFC mapping problem and show the availability calculation when both network-layer protection and application-layer VNF replicas are implemented. In Section 5.3, we illustrate the heuristic algorithm for the availability-guaranteed SFC mapping problem, which determines the working path, the backup path, the number of replicas for each VNF, and how to distribute the replicas to the working path and the backup path. We also present a replica allocation algorithm that allocates replicas to the physical nodes in a way that balances the number of replicas among the nodes. In Section 5.4, numerical results are presented, and we conclude the chapter in Section 5.5.

5.2 Problem statement

In this chapter, we aim to find an availability-guaranteed SFC mapping, and we investigate protection mechanisms for increasing SFC availability.

5.2.1 Availability of the SFC mapping

The availability of an SFC can be affected by many factors, such as physical network component failures or the malfunction of the virtual machines for one VNF. We categorize these factors into two different layers: the physical network layer and the application layer. SFC availability is affected by both the physical network availability and the VNF availability in the SFC.

Network-layer protection

One possible approach to improve an SFC's availability is to provide a backup path in the physical network layer. For example, as shown in Fig. 5.2(a), an SFC is protected with a dedicated path, and has only one replica for each VNF on the working path and one replica

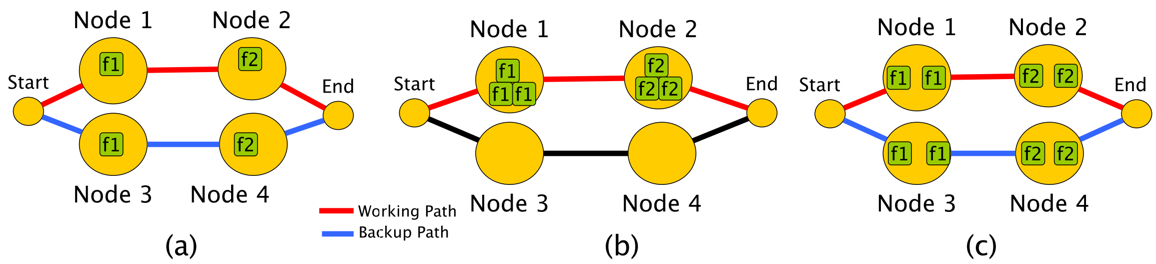


Figure 5.2. Different layer protection mechanisms

for each VNF on the backup path. The SFC's availability can be represented as:

$$A_w = \prod \alpha_{w_i} \prod \alpha_{f_i} \tag{5.1}$$

$$A_b = \prod \alpha_{b_i} \prod \alpha_{f_i} \tag{5.2}$$

$$A_{sfc} = A_w + A_b - A_w A_b < \prod \alpha_{f_i}, \tag{5.3}$$

where α_{w_i} is the availability of physical network component i in the working path , α_{b_i} is the availability of physical network component i on the backup path, and α_{f_i} is the availability of VNF f_i.

Application-layer protection

Another way to improve the SFC's availability is to provide several replicas for each VNF. As shown in Fig. 5.2(b), the availability of the SFC with k_i replicas for one VNF f_i can be represented as:

$$A_{sfc} = \prod \alpha_{p_i} \prod (1 - (1 - \alpha_{f_i})^{k_i}) < \prod \alpha_{p_i}, \tag{5.4}$$

where α_{p_i} is the availability of the physical network component i. We note that the SFC's availability with only application-layer protection is typically limited by the physical path's availability.

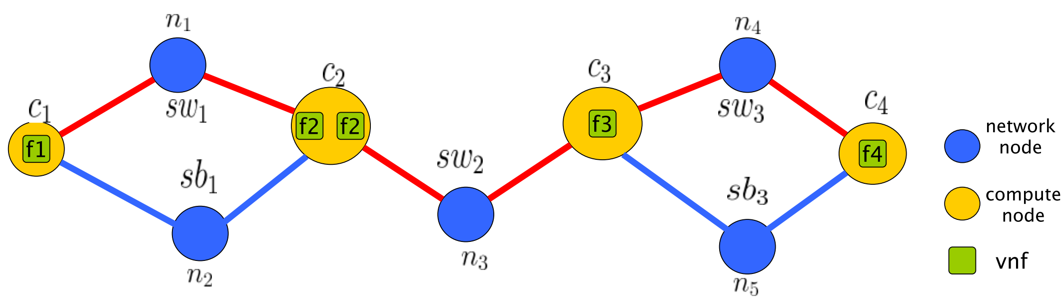


Figure 5.3. Segment protection with VNF replicas mechanism

Coordinated protection

To eliminate the limitations in the above two protection mechanisms, we adopt a coordinated protection mechanism by providing both physical network-layer path protection as well as application-layer protection, as shown in Fig. 5.2(c). The SFC availability is calculated as:

$$A_w = \prod \alpha_{w_i} \prod (1 - (1 - \alpha_{f_i})^{kw_i}) \tag{5.5}$$

$$A_b = \prod \alpha_{b_i} \prod (1 - (1 - \alpha_{f_i})^{kb_i}) \tag{5.6}$$

$$A_{sfc} = A_w + A_b - A_w A_b, \tag{5.7}$$

where kw_i is the number of replicas for VNF f_i on the working path, and kb_i is the number of replicas for VNF f_i on the backup path.

Segment protection mechanism

Segment protection is another effective protection mechanism for guaranteeing service [66]. As shown in Fig. 5.3, the computing nodes are connected by the physical network which consists of physical links and network nodes (e.g., routers, switches, transponders). We consider one segment as the path between two computing nodes. To improve the resource utilization, there is no need to provide segment protection for each segment along the SFC's working path. As shown in Fig. 5.3, the SFC's working path has three working segment paths sw_1, sw_2, and sw_3. The working segment sw_1 consists of the physical links $link_{c_1,n_1}$ and $link_{n_1,c_2}$. It's backup segment sb_1 consists of physical links $link_{c_1,n_2}$ and $link_{n_2,c_2}$.

Given a SFC mapping with segment protection, its availability is calculated as:

$$A_w = \prod \alpha_{s_i} \prod (1 - (1 - \alpha_{f_i})^{k_i}) \tag{5.8}$$

$$\alpha_{s_i} = \begin{cases} \prod a_{sw_i} & \text{w/o} \quad SP \\ 1 - (1 - \prod a_{sw_i})(1 - \prod a_{sb_i}) & \text{with} \quad SP, \end{cases} \tag{5.9}$$

where k_i is the number of replicas for VNF f_i, and α_{s_i} is the availability of each segment path, which depends on the availability of the working segment path a_{sw_i} and the availability of the backup segment path a_{sb_i} if needed.

5.2.2 Problem definition

In this chapter, we focus on the availability-guaranteed SFC mapping problem with coordinated protection. We are given a physical inter-datacenter network $G(V_p, E_p)$, where V_p is a set of the physical nodes, and CN_v is the number of computing resources units at physical node $v \in V_p$. E_p is the set of the physical links connecting the physical nodes, α_l is the availability of physical link l, and CL_l is the link cost of physical link $l \in E_p$.

We are also given a SFC request $S(F_s, B, SA)$, where F_s is a set of the VNFs in the SFC, CR_i is the number of computing resource units required by VNF f_i ($f_i \in F_s$), and β_i is the availability of the VNF f_i. B is the traffic bandwidth, and SA is availability requirement of the SFC.

In this problem, we aim to find the working path and backup path (if needed) for the SFC when dedicated path protection is provided, or the working segments and backup segments when segment protection is provided. We also find the number of replicas for each VNF, and the placement of replicas along the working and backup path (or segments). The objective is to minimize SFC blocking and to minimize the use of computing resources while guaranteeing the SFC's availability requirement. In this chapter, we make the following assumptions:

- We consider a generalized inter-datacenter network where VNFs are deployed to physical computing nodes (datacenters).
- For one specific VNF, its replicas have the same availability and consume the same amount of computing resources.

5.3 Heuristic algorithm

In this section, we propose heuristic algorithms that guarantee the SFC's availability by providing a coordinated protection mechanism. The algorithms first find the physical layer working path (as well as a backup path in dedicated path protection, or backup segments in segment protection if needed), then determine the number of replicas for each VNF in the SFC, and finally deploy the VNF replicas to the physical nodes.

5.3.1 Is there a need to provide physical-layer protection?

Physical layer protection is not needed if the SFC availability requirement can be easily met using a single path. We propose two different approaches to find the physical layer path(s) as follows:

Link-cost-based path algorithm

We assume that each VNF has a maximum of L replicas. We first find the shortest path by deploying Dijkstra's algorithm in G_l, where G_l is the physical network topology. If the path can meet the availability requirement with L replicas for each VNF along the path, then there is no need to provide physical layer protection. Otherwise, we find two link-disjoint paths with the least total cost as the working path and the backup path by deploying Suurballe's algorithm[65].

Availability-based path algorithm

To maximize the physical layer availability, we propose an $availability - based$ algorithm by constructing an auxiliary graph G_a, in which the edge weights are determined by the following transformation:

$$c_i = -\log \alpha_i, \tag{5.10}$$

where α_i is the availability of physical link l_i. Given the maximum number L of replicas for each VNF, we first find the minimum-weight path in G_a and assign this path as the working path. We then check whether the SFC's availability can meet its requirement if each VNF has L replicas along this path. If the SFC's availability can meet its requirement, no physical layer protection is needed. Otherwise, we find two link-disjoint paths on G_a by applying Suurballe's algorithm. The authors in [63] proved that, given the total length of two disjoint paths, a greater difference in length between the two paths yields a higher reliability for the two paths. We follow their conclusion to determine the working path and the backup path.

In this chapter, we consider that the nodes with computing resources are connected by fiber links with a specific number of wavelengths. In case that there is not sufficient wavelengths along the working/backup path, we remove the links without sufficient network resources from graph G_l or G_a, and apply the above path algorithms to find another possible solution. The SFC request will be blocked if no solution is found.

5.3.2 How to distribute the replicas between the working and backup paths?

If path layer protection is needed, after finding the working and the backup path, we need to determine how many replicas are needed for each VNF, and how to distribute the replicas among the working and backup path. We provide three different algorithms for determining the number of replicas on each path.

Vulnerable-vnf-first replicas algorithm

This algorithm attempts to provide replicas for the VNFs that have the lowest availability. Initially, there is one replica for each VNF on the working path and one replica for each VNF on the backup path. We first determine the VNF with the lowest availability by using the following equation:

$$a_{vnf_i} = 1 - (1 - a_{f_i})^{n_i} \tag{5.11}$$

where a_{f_i} is availability of one replica of VNF f_i, and n_i is the number of replicas of VNF f_i on the working or backup path. After determining the most vulnerable VNF and the path it belongs to, we provide one additional replica for that VNF along that path, and then check whether the SFC's availability is reached. If not, we repeat the process of finding a VNF with the lowest availability and provide an additional replica for that VNF.

Even if the SFC's availability can be improved by providing more replicas for each VNF, it may not be reasonable to provide an unlimited number of replicas due to the physical node's computing resource limitation and the cost to the service provider. Based on the availability calculation (Eq. 5.7), the SFC's availability is significantly affected by both the number of replicas for each VNF as well as the distribution of VNFs between the working path and backup path. For example, in Fig. 5.4, there is a SFC with one VNF f_1 mapped to both the working path and the backup path. The availability of the working path is α_w while the availability of the backup path is α_b. There are six replicas for f_1. We can allocate one to five replicas to the working path while the other replicas are allocated to the backup path. The SFC will have different availability under different replica distribution solutions. Therefore, we propose the following replica distribution algorithm:

Path-balance replicas algorithm

Given a total of K replicas, we try to determine the distribution of replicas to the working path and the backup path such that availability is maximum. If availability cannot meet the

SFC's requirement, we increase K until the SFC's availability can be reached. We propose the following lemma:

Lemma 1: Given a SFC with one VNF, f_1 mapped to the working path and the backup path, the availability α_f of VNF f_1, the availability α_1 of the working path, and the availability α_2 of the backup path. If there are total of K ($K > 1$) replicas for f_1, the SFC's maximum availability solution is to distribute k_w replicas in the working path and $k_b = K - k_w$ replicas in the backup path, where

$$k_w = \begin{cases} \lfloor k_1 \rfloor & \text{if} \quad 0 < \lfloor k_1 \rfloor < K \\ 1 & \text{if} \quad \lfloor k_1 \rfloor \leqq 1 \\ K-1 & \text{if} \quad \lfloor k_1 \rfloor \geqq K, \end{cases} \tag{5.12}$$

and

$$k_1 = \frac{K}{2} + \frac{1}{2}(\log_{1-\alpha_f}^{\frac{1}{\alpha_1}-1} - \log_{1-\alpha_f}^{\frac{1}{\alpha_2}-1}). \tag{5.13}$$

Proof: We assume that the unavailability of f_1 is β where $\beta = (1-\alpha_f)$, and $k_2 = K - k_1$ is the number of replicas deployed to the backup path. Therefore, the availability of the SFC is:

$$\begin{aligned} A_c &= \alpha_1(1-\beta^{k_1}) + \alpha_2(1-\beta^{k_2}) - \alpha_1(1-\beta^{k_1})\alpha_2(1-\beta^{k_2}) \\ &= \alpha_1 + \alpha_2 - \alpha_1\beta^{k_1} - \alpha_2\beta^{k_2} \\ &\quad -\alpha_1\alpha_2 + \alpha_1\alpha_2(\beta^{k_1} + \beta^{k_2}) - \alpha_1\alpha_2\beta^{k_1+k_2} \\ &= \alpha_1 + \alpha_2 - \alpha_1\alpha_2(1+\beta^K) - \alpha_1\beta^{k_1} \\ &\quad -\alpha_2\beta^{k_2} + \alpha_1\alpha_2(\beta^{k_1} + \beta^{k_2}) \\ &= \alpha_1 + \alpha_2 - \alpha_1\alpha_2(1+\beta^K) - \alpha_1\alpha_2\Delta, \end{aligned} \tag{5.14}$$

where

$$\begin{aligned} \Delta &= \tfrac{\beta^{k_1}}{\alpha_2} + \tfrac{\beta^{k_2}}{\alpha_1} - \beta^{k_1} - \beta^{k_2} \\ &= (\tfrac{1}{\alpha_2} - 1)\beta^{k_1} + (\tfrac{1}{\alpha_1} - 1)\beta^{k_2} \\ &= (\tfrac{1}{\alpha_2} - 1)\beta^{k_1} + \frac{(\frac{1}{\alpha_1}-1)\beta^K}{\beta^{k_1}} \\ &\geqq 2\sqrt{(\tfrac{1}{\alpha_2} - 1)(\tfrac{1}{\alpha_1} - 1)\beta^K}. \end{aligned} \tag{5.15}$$

Therefore, the function A_c reaches the peak point (maximum availability) when the following holds:

$$(\frac{1}{\alpha_2} - 1)\beta^{k_1} = \frac{(\frac{1}{\alpha_1} - 1)\beta^K}{\beta^{k_1}}. \tag{5.16}$$

We obtain:

$$k_1 = \frac{K}{2} + \frac{1}{2}(\log_\beta^{\frac{1}{\alpha_1}-1} - \log_\beta^{\frac{1}{\alpha_2}-1}). \tag{5.17}$$

The conclusion holds. □

Lemma 2: Given the same conditions as in **Lemma 1**, we assume that there are at least two replicas ($k_w > 1$ and $k_b > 1$) for one VNF f_1 on the working path and at least two replicas on the backup path. If we can reduce two replicas of the VNF f_1, the SFC's maximum availability solution is to remove one replica each from both the working path and the backup path.

Proof: As shown in Eq. 5.17, the SFC's availability is maximized when:

$$k_w - k_b = \log_\beta^{\frac{1}{\alpha_1}-1} - \log_\beta^{\frac{1}{\alpha_2}-1}, \tag{5.18}$$

which means that the difference between the number of replicas on the working path and the number of replicas on the backup path is a fixed number. Therefore, if the total number of replicas of VNF f_1 is reduced by two, we can remove one replica from the working path and another one from the backup path to keep the SFC's availability maximized. □

Vulnerable-vnf-path-balance replicas algorithm

Different from the *vulnerable-vnf-first replicas algorithm*, this algorithm determines the VNF with the lowest availability by considering the replicas on the working path and the backup path together. Initially, each VNF has one replica on the working path and one on the backup path. We determine the VNF f_i with the lowest availability based on Eq. (5.11), where n_i is the total number of replicas of VNF f_i on the working path and backup path combined. We increase the number of replicas of f_i, distribute these replicas among the working path and the backup path based on **Lemma 1**, and repeat to find the replicas for all VNFs until the SFC's availability can meet its requirement.

5.3.3 Seg-vnf algorithm

Segment protection (SP) is an efficient protection mechanism to improve the availability of the service while reducing the resource consumption. To further improve the availability of the SFC and reduce the total computing resources and links resources consumption, we propose a coordinated segment protection mechanism in the network layer as well as VNF replicas protection in the application layer. The details are as follows:

1. Determine the working path based on the *availability-based* route selection, and separate the working path into several segments, where each segment is the physical path between two adjacent computing nodes along the working path.

2. Determine the segment s_i with the lowest availability among the segments not protected, and also determine the VNF f_i which has the lowest availability among all the VNFs. If $a_{s_i} < a_{f_i}$, provide segment protection for segment s_i, otherwise, provide one more replica for VNF f_i.

3. Repeat the above step until the availability of the SFC request can meet its requirement. Block the SFC request if the availability of the SFC request cannot meet its requirement when all the segments are protected and the total number of replicas reach the maximum threshold.

5.3.4 Replicas balance deployment algorithm

After determining the working path (and backup path if needed) and the number of replicas for each VNF on each path, we deploy the replicas to the physical nodes in a way that balances the number of replicas among the nodes[8, 54].

Lemma 3: Assume there are unlimited computing resources on each physical node. We are given an SFC consisting of a set of VNFs without dependency among the VNFs, and the SFC is mapped to two link-disjoint paths as its working chain and backup chain. For VNF f_i, there are n_{w_i} replicas in the working chain and n_{b_i} replicas in the backup chain. The availability of the SFC will not be affected by how the n_{w_i} replicas are deployed along the physical nodes on the working chain or how the n_{b_i} replicas are deployed along the physical nodes of the backup chain. In other words, we can deploy the replicas of the VNFs on the working path (or backup path) to any physical nodes along the working path (or backup path) without changing the SFC's availability.

Proof: As shown in Eq. 5.7, we see that the availability of the SFC is determined by the physical path availability and each VNF's availability. For example, there is a SFC with one VNF f_1, as shown in Fig. 5.4. There are four replicas for f_1 on the working path and two replicas on the backup path. Given the two link-disjoint paths as the mapping of the working path and the backup path, we can deploy the four replicas on the working path to $Node_1$ or $Node_2$ along the working path without changing the SFC's availability. $\square$

Furthermore, if some VNFs are dependent on other VNFs in the SFC, we can also conclude that the replicas of the working chain can be mapped to the physical nodes in the working path without changing its availability as long as the dependencies among the VNFs are maintained.

Therefore, after the number of replicas on the working path (and the number of replicas on the backup path) is determined, we can deploy the replicas for the working and backup chains separately without changing the SFC's availability. We provide an example to show the details of the replica deployment along the working path. As shown in Fig. 5.5, there is a SFC S with four VNFs, f_1, f_2, f_3, and f_4. The number of replicas for the four VNFs are 3, 2, 2, 3, and the replica computing resource unit requirements are 2, 2, 1, and 1. There may also be dependencies among different VNFs. In this example, f_3 is dependent on f_1

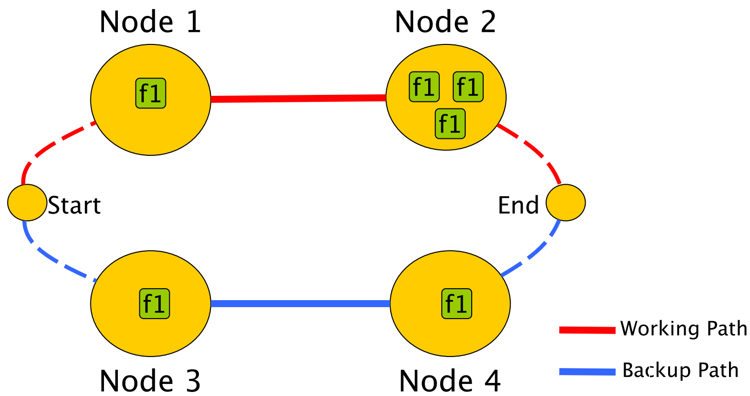


Figure 5.4. Replicas distribution among the working and backup paths

($f_1 \rightarrow f_3$), and f_4 is dependent on f_2 ($f_2 \rightarrow f_4$). S is mapped to a physical network with two nodes, and the available number of computing resource units at $Node_1$ and $Node_2$ is 10 and 8, respectively.

As shown in Fig. 5.6(a), we first separate the SFC into several sub-chains according to the dependencies s_1 ($f_1 \rightarrow f_3$) and s_2 ($f_2 \rightarrow f_4$). Then we construct a set Θ by choosing the first VNF from each sub-chain (f_1 from s_1 and f_2 from s_2). Then we deploy the replicas of the VNF with the maximum total resource consumption to $Node_1$ and then update Θ and each node's available capacity, as shown in Fig. 5.6(b). We can deploy all the VNFs to the nodes in the same way, as shown in Fig. 5.6(c),(d),(e). Note that, if the available capacity of one node cannot support all the replicas of one VNF, we deploy as many replicas as possible in the current node, and then allocate the remaining replicas to the following node.

5.3.5 SFC mapping heuristic algorithms

Based on the above analysis and details, we propose the following algorithms to address the availability-based SFC mapping problem. The *Avail-bal*, *Avail-vul* and *Vvpb* algorithms utilize *availability-based* route selection, while *LC-bal* and *LC-vul* utilize *link-cost-based* route selection. For determining the number of replicas and the distribution of replicas between the working and backup paths, Algorithms *Avail-bal* and *LC-bal* utilize the

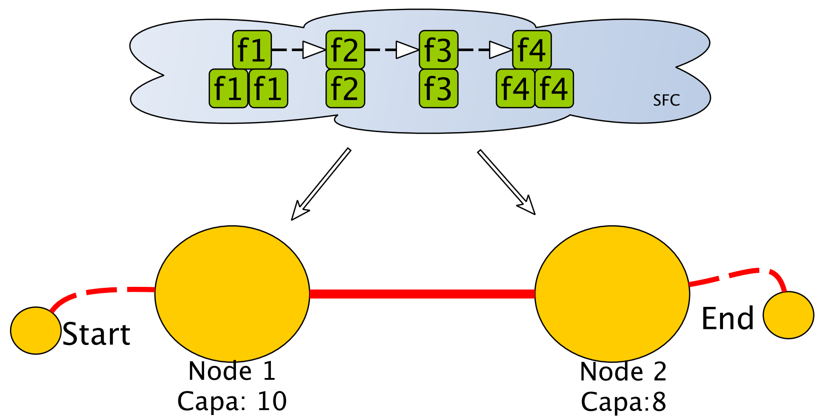


Figure 5.5. VNFs with dependency

path-balance replicas algorithm, while *Avail-vul* and *LC-vul* algorithms utilize the *vulnerable-vnf-first replicas* algorithm. The *Vvpb* algorithm utilizes the *vulnerable-vnf-path-balance* algorithm. The *Seg-vnf* algorithm utilizes *availability-based* route selection to determine the working and backup segment paths. All of the proposed heuristics use the same balance deployment algorithm for determining at which node along a path replicas should be deployed. A summary of the proposed heuristics is shown in Table I. To show the efficiency of the proposed heuristic algorithms, we also find out the optimal solution for comparison. In *Ded-opt* algorithm, we enumerate all possible mapping solutions without any protection and with dedicated protection. Then we choose the maximum SFC request acceptance rate, the minimum total computing resource, and the minimum total link cost as the optimal value for comparison. Similarly, we enumerate all possible mapping solutions without any protection and with segment protection in *Seg-opt* algorithm.

5.4 Performance evaluation

In this section, we evaluate the performance of the proposed heuristic algorithms in terms of the SFC request acceptance rate, total computing resources, and total physical link resources

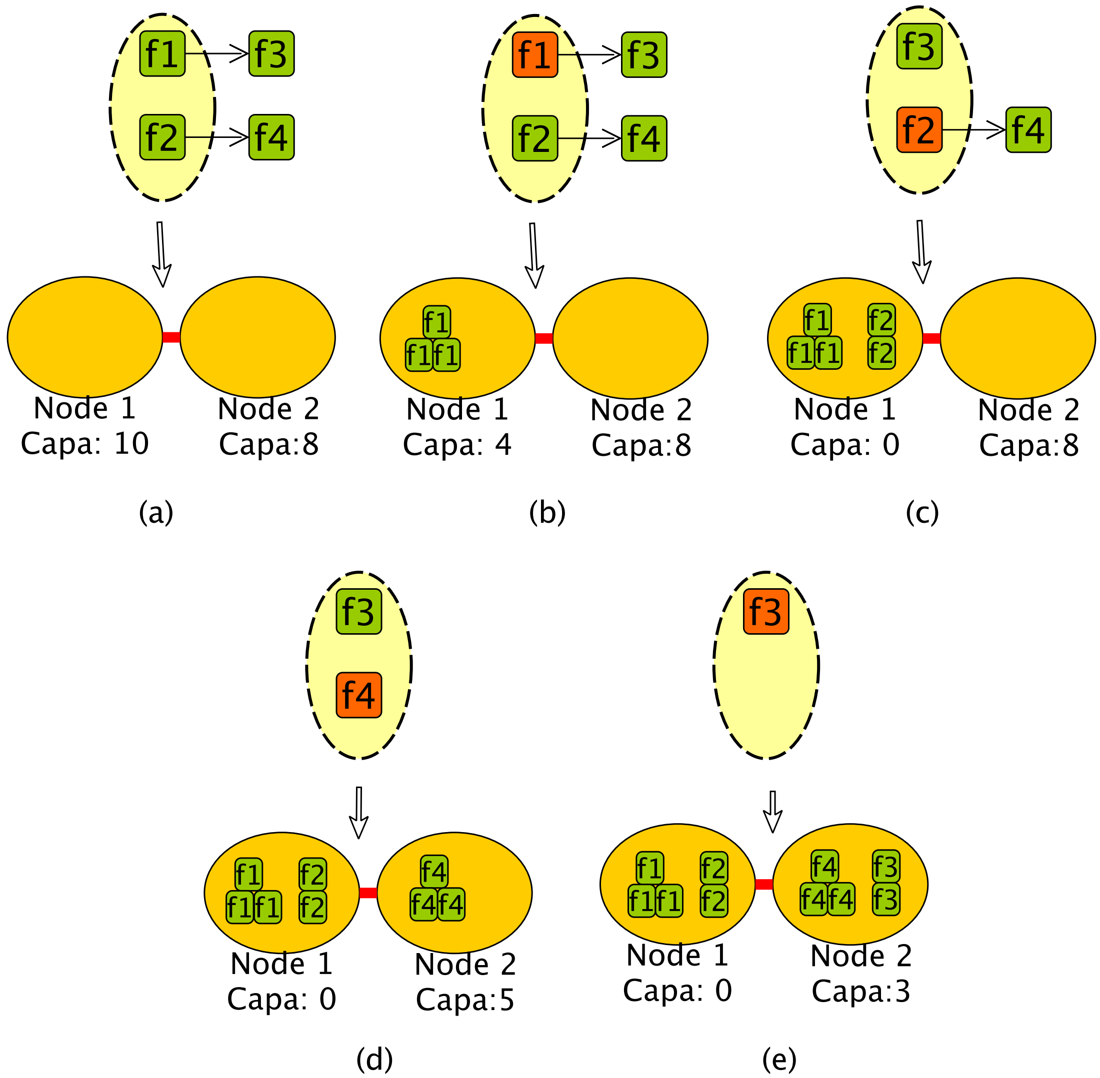


Figure 5.6. Replicas deployment example

Table 5.1. Summary of the proposed heuristic algorithms

Algorithms	Route selection	Replicas # and distribution	Replica deployment
Avail-bal algo.	*availability-based*	*path-balance*	*replica balance*
Avail-vul algo.	*availability-based*	*vulnerable-vnf-first*	*replica balance*
Vvpb algo.	*availability-based*	*vulnerable-vnf-path-balance*	*replica balance*
LC-bal algo.	*link-cost-based*	*path-balance*	*replica balance*
LC-vul algo.	*link-cost-based*	*vulnerable-vnf-first*	*replica balance*
Seg-vnf algo.	*availability-based*	*vulnerable-segment-vnf*	*replica balance*

in the 24-node U.S. mesh network. The request acceptance rate is define as the fraction of SFC requests for which the availability requirement can be met.

5.4.1 Experimental environment

In this section, we evaluate the performance of the proposed heuristic algorithms in the 24-node U.S. mesh network, as shown in Fig. 5.7. Each node represents a computing node, and they are connected by fiber links. Every fiber link accommodates 4 THz of bandwidth and a frequency slot occupies 12.5 GHz. Therefore, there are 320 wavelengths on each fiber link [68]. We randomly generate 100 SFCs with the availability requirement ranging from 0.99 to 0.9999. The traffic bandwidth of each SFC request ranges from 1 to 3 wavelengths. There are 10 types of VNFs with the required number of computing resource units for each VNF ranging from 1 to 3. The number of the VNFs in each SFC is determined randomly and ranges from 2 to 6. We consider three scenarios, each which has a different level of physical network availability, to evaluate the performance of the proposed algorithms. In the low availability level case, the availability of the physical links is randomly generated from 0.99 to 0.999, in the medium availability level case, the availability of the physical links ranges from 0.999 to 0.9999, and in the high availability level case, the availability of the physical links ranges from 0.9999 to 0.99999. For each of the different availability level cases, we evaluate the performance of the proposed algorithms for cases in which the availability of the VNFs is 0.99, 0.999, and 0.9999.

5.4.2 SFC request acceptance

We first evaluate the performance of the algorithms in terms of the SFC request acceptance rate in the physical network with low, medium, and high levels of availability. If one SFC's availability cannot be met even when the backup path and the VNF replicas are provided, the SFC request is blocked. For the physical networks with different levels

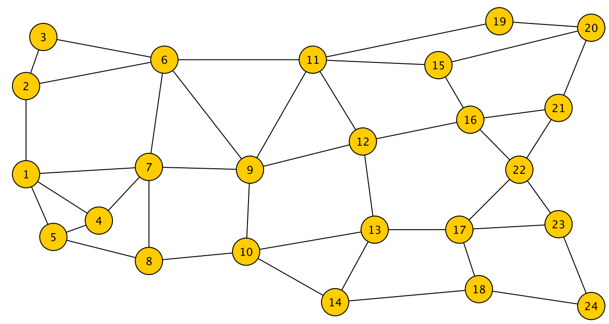
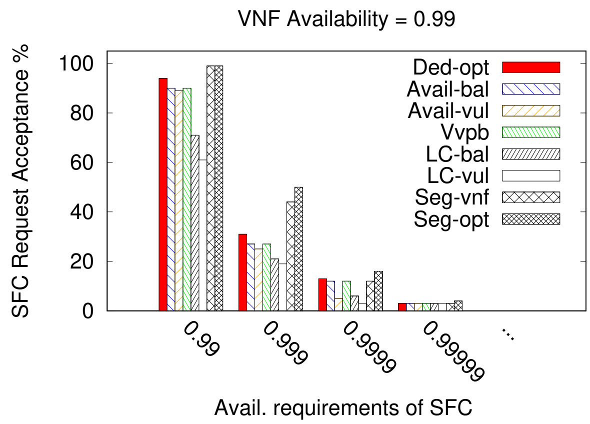


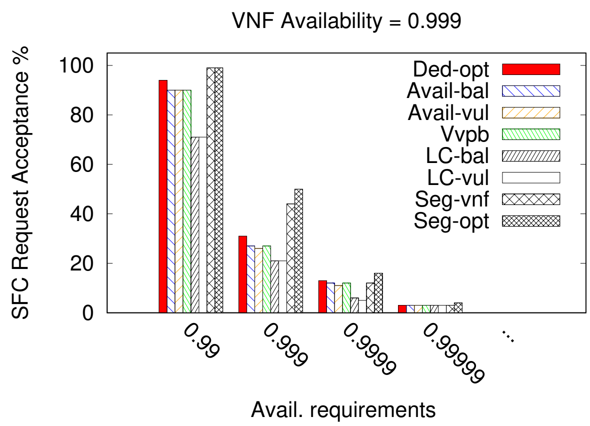
Figure 5.7. 24-node U.S. mesh network

of availability, the availability of each VNF is set to 0.99, 0.999 and 0.9999 for comparison. If we consider that dedicated path protection if needed, as shown in Fig. 5.8, we observe that *path-balance* based algorithms perform better than the greedy *vulnerable-vnf* based algorithms. In *path-balance* algorithms, we try to find a higher availability SFC mapping by distributing the replicas of the VNFs among the working chain and the backup chain based on the theoretical analysis in Lemma 1. A higher availability of the SFC means that we can remove some replicas to further reduce the computing resources while guaranteeing the SFC's availability. Comparing between *Avail-bal* and *LC-bal* (or *Avail-vul* and *LC-vul*) algorithms, we observe that the availability-based approach contributes to finding the paths with higher availability, which results in a lower blocking ratio. In addition, to show the efficiency of the proposed availability-based approach, we also find out the optimal acceptance rate with *Ded-opt* algorithm and *Seg-opt* algorithm under the low availability level of physical network with the availability from 0.99 to 0.999, as shown in Fig. 5.8.

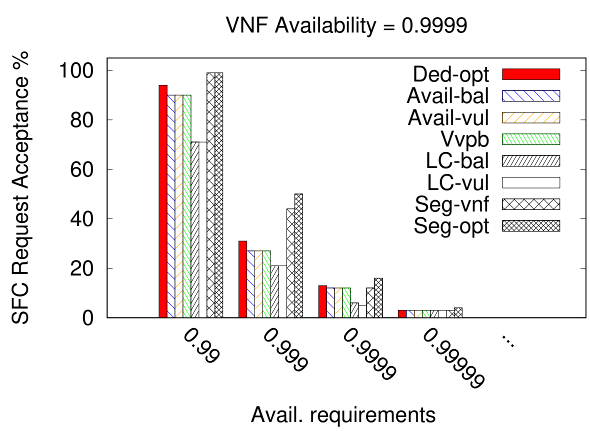
Furthermore, the *Seg-vnf* algorithm performs the best among the proposed algorithms to guarantee the availability of the SFC requests, especially in physical networks with low and medium availability levels. In the *Seg-vnf* algorithm, it is possible to provide protection only for the segments with vulnerable physical network components, therefore increasing the availability of the SFC, which results in a lower blocking ratio. We also observe that the



(a) VNF Availability = 0.99

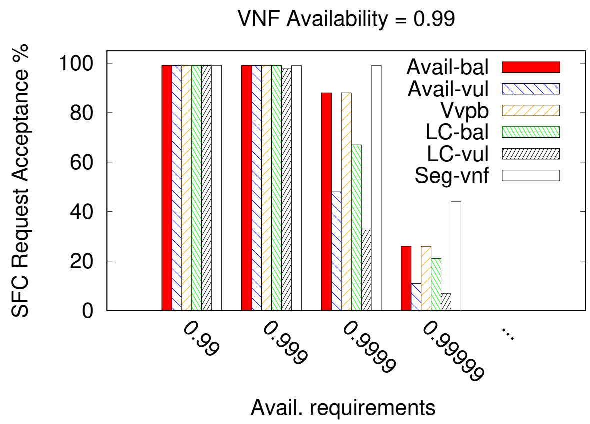


(b) VNF Availability = 0.999

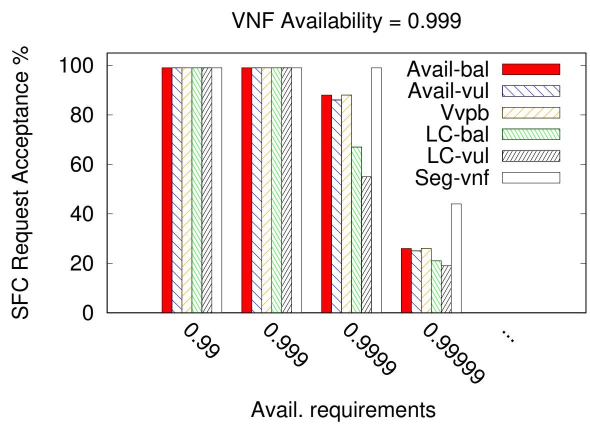


(c) VNF Availability = 0.9999

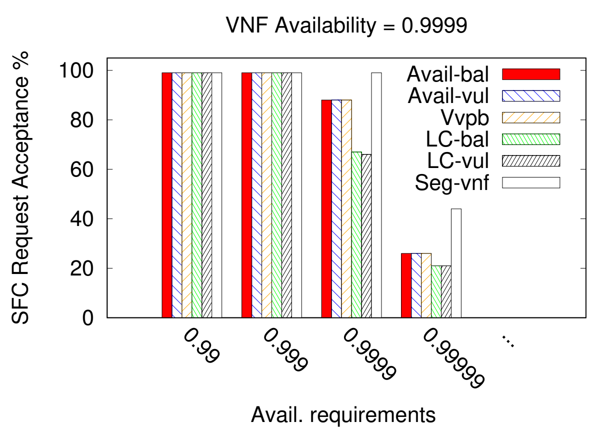
Figure 5.8. SFC requests acceptance rate in the low availability level of physical network (0.99-0.999)



(a) VNF Availability = 0.99

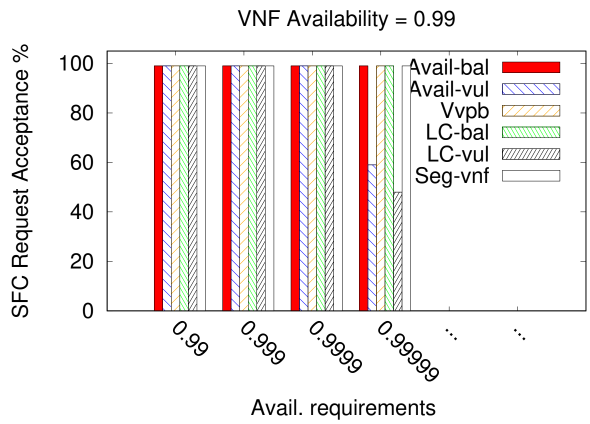


(b) VNF Availability = 0.999

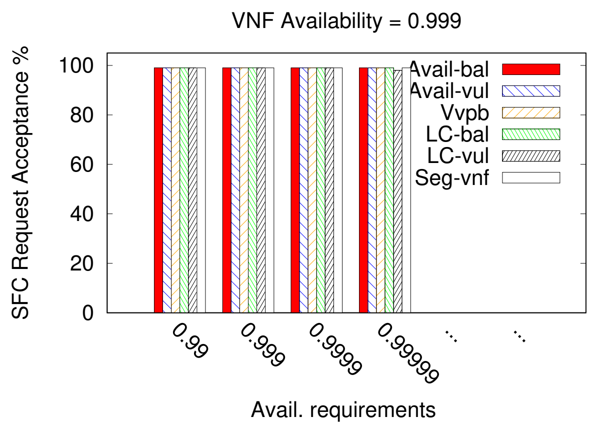


(c) VNF Availability = 0.9999

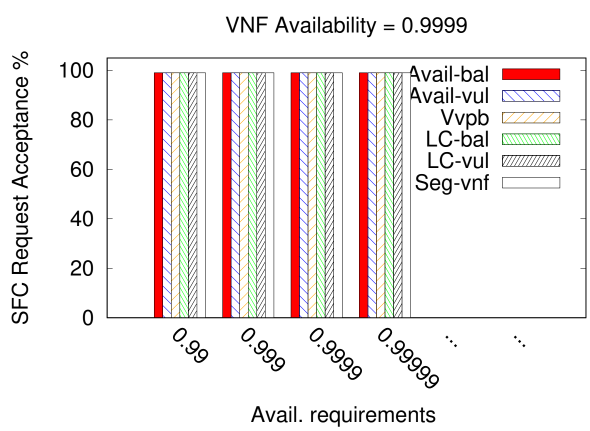
Figure 5.9. SFC requests acceptance rate in the medium availability level of physical network (0.999-0.9999)



(a)



(b)



(c)

Figure 5.10. SFC requests acceptance rate in the high availability level of physical network (0.9999-0.99999)

blocking of SFC requests decreases with an increase in the availability of the physical network (from low level case to medium level case to high level case). Therefore, the availability of the network components cannot be ignored when guaranteeing the SFC's availability, especially when the availability of the network component is not high enough. Most of the SFC requests are blocked when the availability of the VNFs and the physical network components are low, and the required availability of the SFC is high (0.9999 or 0.99999). The availability of SFC requests can be easily guaranteed when the availability of the physical network and the availability of VNFs is high.

5.4.3 Computing resource consumption

We next evaluate the performance of the proposed algorithms in terms of the total cost of computing resources. If there is no dedicated path protection or there is segment path protection, the cost of computing resources is calculated as follows:

$$C_{computing} = \sum N_{f_i} U_{f_i}, \tag{5.19}$$

where n_{f_i} is the total number of replicas for VNF f_i, and U_{f_i} is the computing resource units required for each VNF replica.

If dedicated path protection is provided, the total cost of computing resources for one SFC request is calculated as follows:

$$C_{computing} = \sum W_{f_i} U_{f_i} + B_{f_i} U_{f_i}, \tag{5.20}$$

where W_{f_i} and B_{f_i} are the total number of replicas for VNF f_i along the working path and the backup path, respectively, and U_{f_i} is the computing resource units required for each VNF replica.

We compare the total cost of computing resources when the availability requirement of each SFC is 0.99. The availability of the VNFs ranges from 0.9 to 0.9999. In this case, the availability of all SFC requests can be guaranteed when the availability level of the physical network is medium or high. We ignore the low availability level case because the availability requirement of some SFC requests cannot be guaranteed. We also find out the optimal(minimum) computing resources consumption by enumerating all possible working/backup path and replica allocation solutions. As shown in Fig. 5.11, we observe that the *path-balance replica* based algorithms (*Avail-bal* and *LC-bal*) as well as the *Seg-vnf* algorithm have almost similar performance, and consume the least computing resources because those algorithms can maximize the SFC's availability, and therefore reduce the number of replicas. In addition, the above algorithms can approximate to the optimal computing resources best. As shown in Fig. 5.11, the proposed *path-balance* based algorithms and *Seg-vnf* algorithm can reduce the total computing resources by more than 50% compared to the other approaches when the availability of each SFC request is 0.999 and 0.9999.

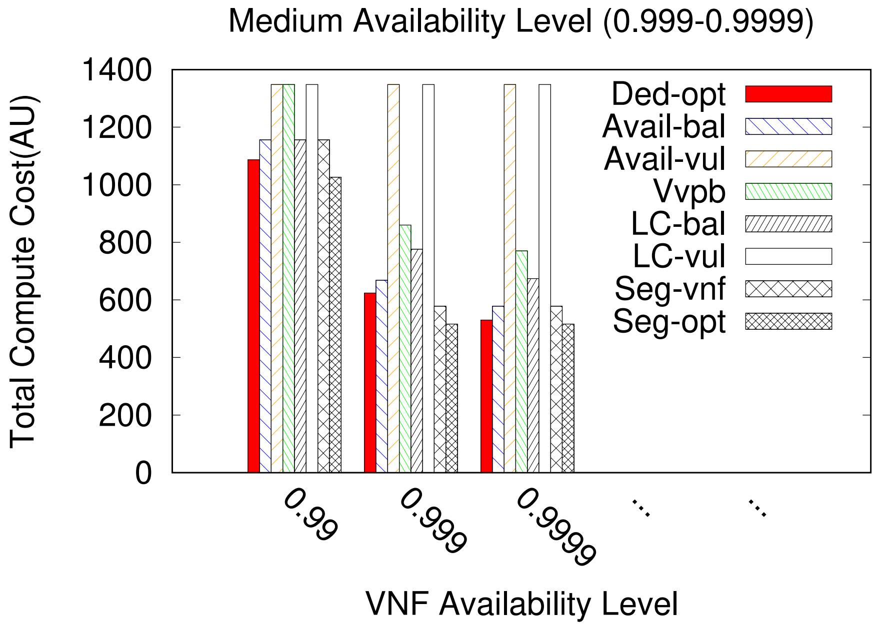
5.4.4 Link cost consumption

We finally evaluate the performance of the algorithms in terms of the total link cost. The total link cost is calculated as follows:

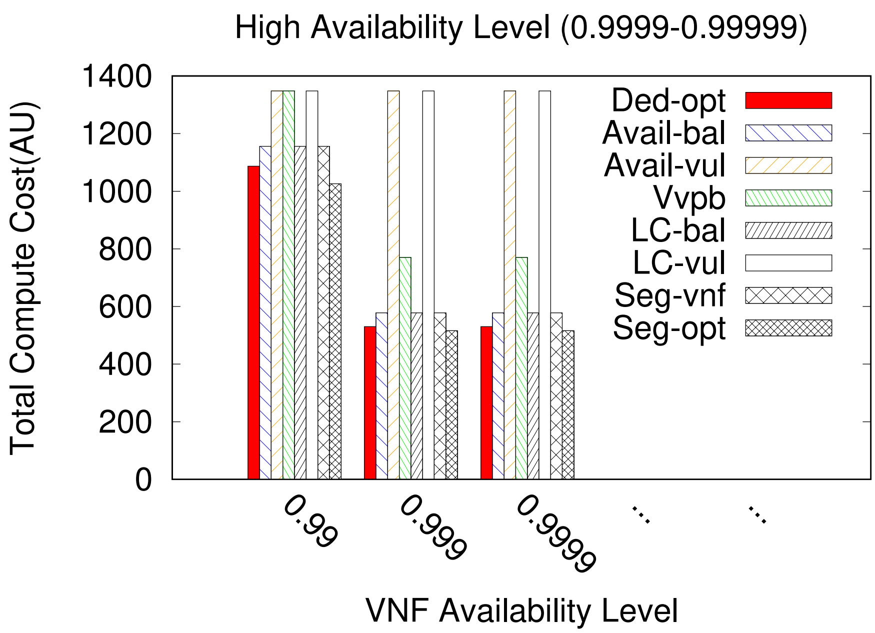
$$C_{link} = \sum W_i Bw + B_i Bw, \tag{5.21}$$

where Bw is the number of wavelengths of one SFC request, W_i is the length of the physical link i along the working path (or segment working path i in segment protection), B_i is the length of the physical link i along the backup path (or segment backup path i in segment protection).

We compare the results when the SFC's availability is 0.99 (there is no request blocking under the medium and high availability levels of the physical network). As shown in Fig. 5.12, the *link-cost-based* algorithms consume less link resources than the *availability-based* algorithm, and approximate the optimal value the best because they attempt to find the



(a) Availability of the physical network: 0.999-0.9999



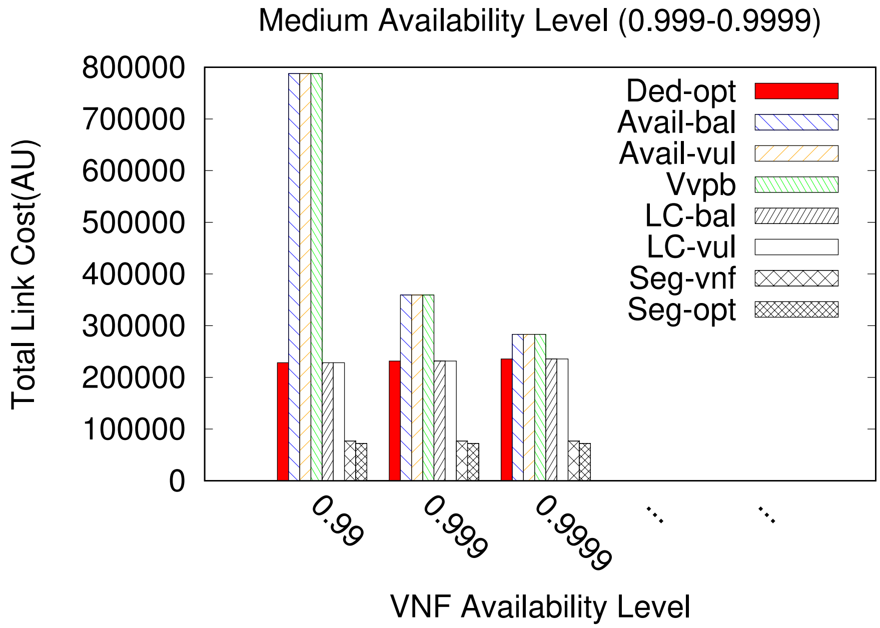
(b) Availability of the physical network: 0.9999-0.99999

Figure 5.11. Total computing resources when the availability requirement of each SFC request is 0.99

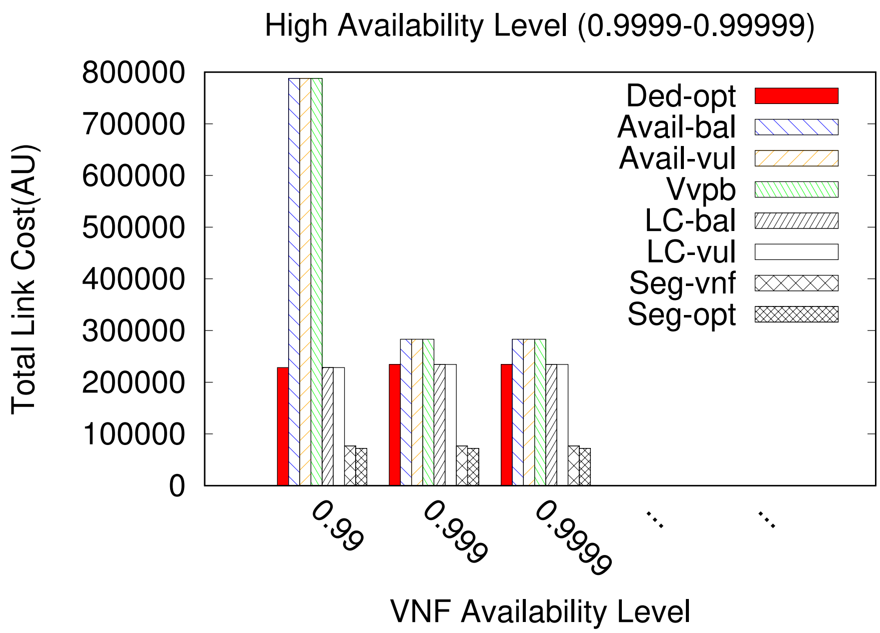
shortest cost paths, and the *Seg-vnf* algorithm performs the best in reducing the total link cost. Compared to dedicated path protection, the segment protection mechanism only provides protection for partial segments, and therefore can further reduce the total link cost when the availability of SFC requests can be guaranteed.

5.5 Conclusion

In this chapter, we consider the problem of availability-guaranteed service function mapping, and provide an efficient coordinated protection mechanism to guarantee each SFC's availability. By analyzing how to distribute the replicas of one VNF to the working path and the backup path in order to maximize the SFC's availability, we propose efficient heuristic algorithms to determine the number of the replicas for each VNF of the SFC, and then allocate the replicas to the physical node by considering the dependencies among different VNFs. To further reduce the SFC request blocking and the total computing and link resource consumption, we propose an efficient segment protection mechanism to guarantee each SFC's availability requirement. Numerical results show that the algorithms based on the *path-balance replica algorithm* perform better among the dedicated path protection algorithms, and the proposed *Seg-vnf* algorithm performs the best in terms of reducing the total computing resources as well as reducing the SFC blocking.



(a) Availability of the physical network: 0.999-0.9999



(b) Availability of the physical network: 0.9999-0.99999

Figure 5.12. Total physical link cost when the availability requirement of each SFC request is 0.99.

CHAPTER 6

GUARANTEED-AVAILABILITY NETWORK FUNCTION VIRTUALIZATION IN INTER-DATACENTER NETWORKS[1]

In this chapter, we consider the availability of the datacenter's network elements, and we propose a coordinated protection mechanism that adopts both backup path protection and SFC replicas distributed among datacenters to support high availability while reducing total cost [48].

6.1 Introduction

Network function virtualization (NFV) provides an efficient and flexible way to deploy service function chains (SFCs) consisting of a chain of connected network services that are deployed as software containers over general-purpose computing hardware [33]. However, software-based virtualized network functions (VNFs) bring new challenge for network operators in providing service availability guarantees. Traditionally, network-level protection mechanisms are considered separately from function-level VNF backup mechanisms.

However, a SFC's availability cannot be guaranteed if only network-level protection mechanisms or only function-level VNF backup mechanisms are considered. Furthermore, when SFCs are deployed in inter-datacenter networks, the physical network elements, such as core routers and inter-datacenter fiber links, as well as aggregation switches and intra-datacenter links within datacenters have a significant impact on the availability of the SFC [34, 70]. Therefore, the datacenter's infrastructure and the availability of its network elements should also be considered. In this chapter, we address the availability-guaranteed SFC mapping

[1]©IEEE. Reprinted, with permission, from Jian Kong, Inwoong Kim, Xi Wang, Qiong Zhang, Weisheng Xie, Hakki C. Cankaya, Nannan Wang, Tadashi Ikeuchi, and Jason P. Jue. "Guaranteed-Availability Network Function Virtualization in Inter-Datacenter Networks." In *Optical Fiber Communication Conference and Exhibition (OFC)*, 2018.

problem in inter-datacenter networks, while also considering the availability of a datacenter's internal network elements. We propose the concept of SFC replicas and develop a coordinated protection mechanism that adopts both backup path protection in the network and SFC replicas distributed among different datacenters to guarantee a SFC's availability.

6.2 Problem statement

We are given a physical inter-datacenter network $G(D_p, Inter_p)$, where D_p is a set of the datacenters, and $Inter_p$ is the set of the physical links connecting the datacenters. V_v is the set of the network elements (e.g., core routers, aggregation switches), $Intra_v$ is the set of intra-datacenter links, and R_v is the number of the computing resources units at datacenter $v \in D_p$. a_l is the availability of the network element l. We are also given a SFC request $S(F_s, B, SA)$, where F_s is a set of the VNFs in the SFC, CR_i is the number of computing resource units required by VNF $f_i \in F_s$, and a_i is the availability of the VNF f_i. B is the traffic bandwidth, and SA is availability requirement of the SFC.

The problem is to find the working path and backup path (if needed) for the SFC, the number of replicas for each VNF, and the placement of replicas in different datacenters along the working and backup path. The objective is to maximize the SFC request acceptance and to minimize the use of computing resources while guaranteeing the SFC's availability requirement. In this chapter, we assume that for one specific VNF, its replicas have the same availability and consume the same amount of computing resources.

6.3 Availability analysis and calculation

To calculate the availability of a SFC mapping, we consider each VNF's availability as well as the availability of all network elements used by the SFC. One strategy for maximizing availability is to place the different VNFs of a SFC close to each other within the same datacenter to minimize the datacenter internal consumed bandwidth and number of interconnect

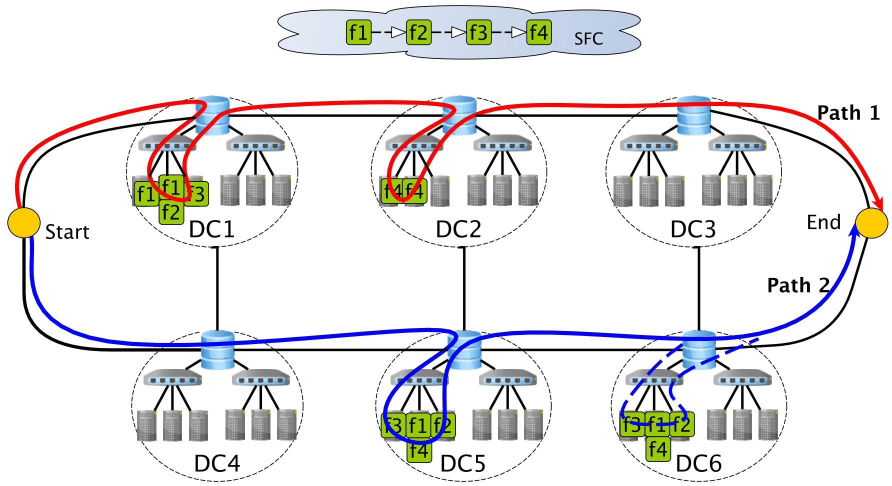


Figure 6.1. SFC in inter-datacenter network

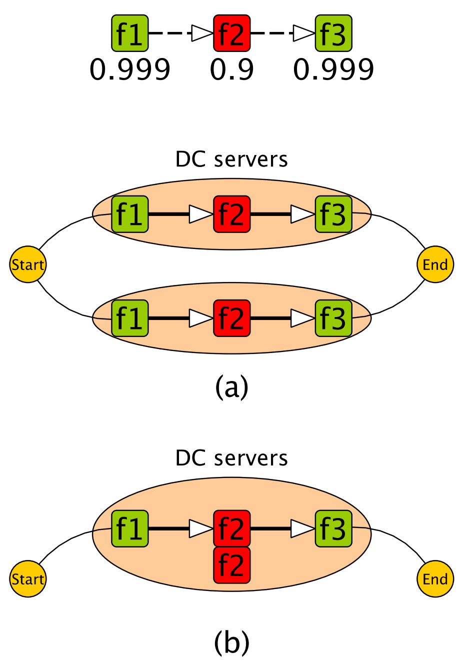


Figure 6.2. Chain replica instance

hops [34]. We initially consider the case in which replicas of a given VNF are also deployed close to each other at the same datacenter, but on different servers connected to the same aggregation switches. If a datacenter doesn't have sufficient resources, the remaining VNFs and their replicas are mapped to another datacenter along the path. As shown in Fig. 6.1, the SFC is mapped to *Path* 1 (red line), which spans several inter-datacenter links. VNFs f_1, f_2, and f_3 and their replicas are mapped to DC1, and f_4 is mapped to DC2. Θ is the set of inter-datacenter switches and links used by the SFC, and Ω is the set of datacenters used by the SFC. Φ_m is the set of VNFs mapped to datacenter m, and Δ_m is the set of intra-datacenter links used by the SFC in datacenter m. The availability of the SFC mapping is:

$$a_{SFC} = \prod_{i \in \Theta} a_{inter_i} \cdot \prod_{m \in \Omega} (\prod_{j \in \Delta_m} a_{intra_{m,j}} \cdot \prod_{i \in \Phi_m} (1 - (1 - a_i)^{n_{i,m}}))) < \prod_{i \in \Theta} a_{inter_i}, \tag{6.1}$$

where a_{inter_i} is the availability of the inter-datacenter link i, $a_{intra_{m,j}}$ is the availability of the intra-datacenter link j in datacenter m, a_i is the availability of NFV f_i, and $n_{i,m}$ is the number of replicas of NFV f_i at datacenter m.

Note that placing all replicas of a given VNF within the same datacenter may not be the best approach for maximizing availability, since all of the replicas may be simultaneously affected by failures within the datacenter. An alternate approach is to place replicas of a given VNF in different datacenters. When all of the VNFs of a SFC are placed within a single datacenter, we refer to this set of VNFs and their replicas as a single SFC replica. For example, in Fig. 6.1, we deploy two SFC replicas to *Path* 2 (blue line), one is mapped to DC5 and the other is mapped to DC6. We assume that the SFC is available only when any one of its SFC replicas is available. Thus, its availability is:

$$a_{SFC} = \prod_{i \in \Theta} a_{inter_i} \cdot (1 - \prod_{m \in \Omega} (1 - (\prod_{j \in \Delta_m} a_{intra_{m,j}} \cdot \prod_{i \in \Phi_m} a_i^{n_{i,m}}))) < \prod_{i \in \Theta} a_{inter_i}. \tag{6.2}$$

From Eq. 6.1 and Eq. 6.2, we observe that $a_{SFC} < \prod a_{inter_i}$ when no backup path is provided for the SFC. If the availability requirement of the SFC is high, path protection should be provided. The availability of the SFC mapping when there is dedicated path protection is: $a_{SFC} = a_w + a_b - a_w a_b$, where a_w is the availability of the SFC mapped to the working path, and a_b is the availability of the SFC mapped to the backup path.

6.4 SFC mapping heuristics

Based on the availability analysis of the SFC mapping, we observe that the availability of one SFC mapping depends on the number of the replicas of VNFs as well as the path(s) for the SFC. Therefore, we propose several heuristic algorithms to guarantee the SFC's availability. The main idea is to determine the working path with the highest availability, and then determine the replicas of the VNFs or the SFC replicas. If the availability requirement of the SFC is high, dedicated protection is provided to guarantee its availability.

6.4.1 Node availability

When determining a path for a SFC, it is desirable to find a path that contains links and nodes with high availability. By setting weights of links and nodes to -log a_i, where a_i is the availability of link or node i, we can apply Djikstra's algorithm to find a maximum-availability path. There is no single way to define a node's availability, since the availability depends on whether the SFC utilizes the datacenter at a node for its VNFs, in which case the availability will depend on the datacenter's internal elements, or the SFC merely bypasses the node, in which case the availability depends only on the availability of the node's core router. We consider three approaches for approximating a node's availability:

Evaluation 1: find the intra-datacenter path with the lowest availability from the core router to the servers, and set its availability as the node's availability, as shown in Fig. 6.3(a).

Evaluation 2: choose the availability of the core router that connects the datacenter to the other datacenters, as shown in Fig. 6.3(b).

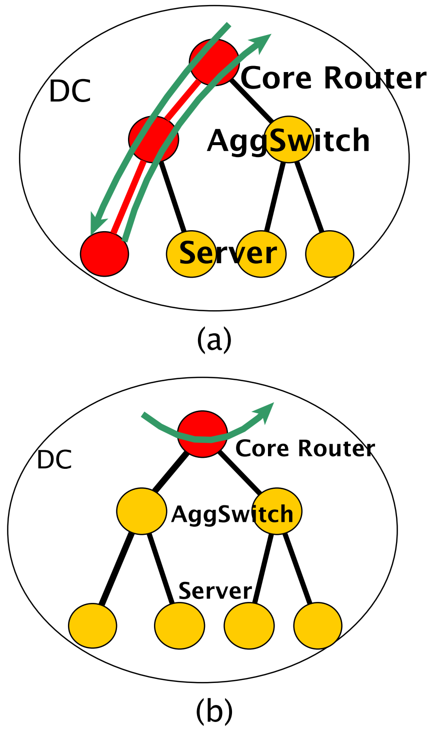


Figure 6.3. Datacenter availability evaluation

Evaluation 3: ignore the node's availability by setting it to 1.0.

6.4.2 SFC replica instance

Based on the definition and availability analysis of the SFC replicas, an efficient way improving the SFC's availability is to provide several SFC replicas, and then distribute them to different datacenters. However, we also need to determine the number of replicas of each VNF in each SFC replica. For example, given a SFC with an availability requirement of 0.985 and three VNFs, f_1,f_2 and f_3 with availabilities $a_1 = 0.999, a_2 = 0.99, a_3 = 0.999$, a possible solution is that each VNF has one replica, as shown in Fig. 6.2(a), and we need provide two SFC replicas to guarantee its availability. In this case, we obtain $a_{sfc} = 0.989$ and the total cost is 6 replicas. However, a more efficient solution is to provide more replicas for the VNFs with low availability within a SFC replica, as shown in Fig. 6.2(b). In this case, we obtain $a_{sfc} = 0.988$ and the total cost is 4 replicas.

A *SFC replica instance* is defined as an instance of a SFC replica that specifies the number of VNF replicas for each VNF in the SFC. For example, the SFC replica instances in Fig. 6.2(a) contain one replica for each VNF, while the SFC replica instance in Fig. 6.2(b) contains one replica each of VNFs f_1 and f_3, and two replicas of VNF f_2. We try to provide more replicas for the vulnerable VNFs such that each VNF has roughly the same availability within a SFC replica instance as follows. We first determine the VNF f_m with the highest availability a_m. Then, for each VNF f_i, we calculate the minimum number n_i of replicas that satisfies: $1.0 - (1.0 - a_i)^{n_i} \geq a_m$, and we set n_i as the number of replicas for f_i in the *instance*. We call this method the *balanced* SFC replica instance. For comparison, a *non-balanced* SFC replica instance provides one replica for each VNF in a SFC replica instance.

6.4.3 SFC-replica-based algorithm

We propose the following heuristic algorithm based on the SFC replica concept:

Step 1: Generate the availability-based directed graph G_a with weights set to $(-\log a_i)$.

Step 2: Determine the shortest cost path P_w(highest availability path) in G_a by Dijkstra's algorithm.

Step 3: Determine the *SFC replica instance* for this SFC.

Step 4: Find the node with the highest value evaluated in *Section* 4.1 and with sufficient resources from those nodes unused by this *SFC* along the path, and allocate one *SFC replica instance* to it.

Step 5: Calculate the availability of current SFC mapping, a_c. If $a_c \geq a_{req}$, go to **Step 9** to re-optimize the resources. Otherwise, provide one more *SFC replica instance* and allocate it to a node unused by the SFC (similar to **Step 4**) until $a_c \geq a_{req}$. If the total number of the *SFC replica instances* doesn't exceed a limitation K, go to **Step 9**. Otherwise, remove all the *SFC replica instances* mapped along P_w, and go to next step to provide dedicated protection.

Step 6: Determine the link-disjoint shortest path P_b as the backup path by removing the links used by P_w in G_a.

Step 7: Allocate one *SFC replica instance* on each of paths P_w and P_b to the nodes with the highest value evaluated in *Section* 4.1 and with sufficient resources from the nodes unused by this SFC.

Step 8: Calculate the availability of the SFC mapping, a_c. If $a_c \geq a_{req}$, go to **Step 9** to further reduce the total resources. Otherwise, allocate one more *SFC replica instance* to P_w or P_b until $a_{sfc} \geq a_{req}$. If the total number of *SFC replica instance* exceeds K, Stop here.

Step 9: Re-optimize the total computing resources: for each VNF f_i in a *SFC replica instance*, if f_i has more than one replica, remove one replica of f_i and check whether $a_c \geq a_{req}$. Repeat for each VNFs of all *SFC replica instances* until $a_{sfc} < a_{req}$.

6.4.4 VNF-cluster-based algorithm

We also propose a *vnf-cluster-based* heuristic algorithm for comparison. The main idea is to allocate all the replicas of one SFC's VNFs to one datacenter (node). Initially, no protection is provided and there is one replica for each VNF, and they are mapped to the datacenter with the highest evaluation availability and with sufficient resources. If $a_c < a_{req}$, determine the VNF f_m with the lowest availability, and provide one more replica for f_m. Repeat the above until $a_c \geq a_{req}$ or the total number of replicas exceeds the limitation. Otherwise, provide dedicated path protection, and increase the number of the replicas of each VNF until the SFC's availability is met.

6.5 Simulation results

In this section, we evaluate the performance of the proposed heuristic algorithms on a 75-node U.S. mesh network. We consider a three level tree architecture in datacenters. The availability of the network element (inter/intra datacenter links, router/switches) ranges from 0.999 to 0.99999. There are 10 types of VNFs with availability between 0.99 and 0.9999, and which require 1 unit of computing resources. We randomly generate 100 SFCs, and randomly determine the number of the VNFs ranging from 2 to 6 for each SFC. The availability requirement of each SFC is distributed between 0.99 and 0.999999. The number of the chain replicas is limited to $K = 10$ in the *chain replica* based algorithm, and $K =$ (10 · number of the VNFs) in the vnf*-cluster* algorithm.

We first compare the performance of the three node availability evaluation approaches in terms of the SFC request acceptance. As shown in Fig. 6.4, Evaluation Method 2 performs the best because the core router's failure affects all the SFC replicas at a given node, and therefore has more impact on the availability of the SFC. We then compare the SFC request acceptance and the total cost among *SFC replica* based algorithms and the vnf*-cluster* based algorithm. All algorithms deploy the second availability evaluation approach to determine the path(s). As shown in Fig. 6.5, the *SFC replica balance* based algorithm performs better than the other algorithms because it provides more replicas for the vulnerable VNFs, and distributes the SFC replicas to different DCs to improve the SFC's availability. We also compare the total cost used by the SFCs in Fig. 6.6. The *SFC replica balance* based algorithm performs better in reducing the total computing resources. Note that compared to the *non-balance* way, the *balance* way can guarantee more SFC requests' availability while consuming less total computing resources.

6.6 Conclusion

We consider the availability-guaranteed SFC mapping problem by considering the physical network elements' availability over the inter-datacenter networks. We propose a coordinated protection mechanism that adopts both backup path protection in the network and SFC replicas distributed among different datacenter to guarantee a SFC's availability. Numerical

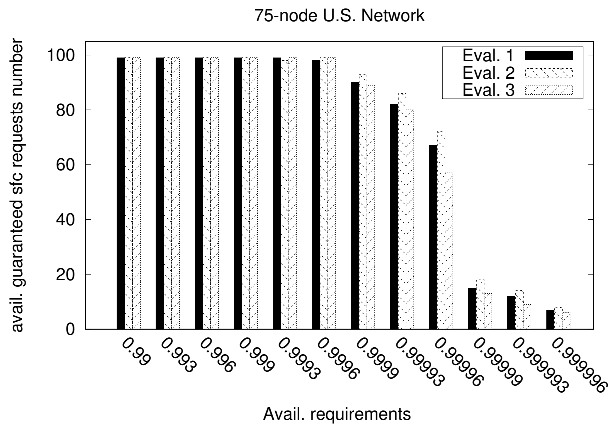


Figure 6.4. DC avail. evaluation

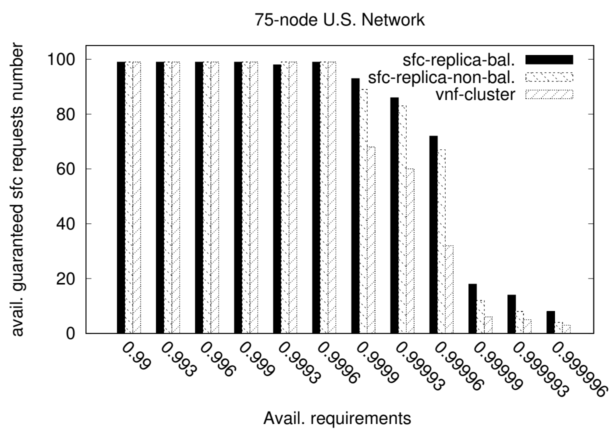


Figure 6.5. SFC request acceptance

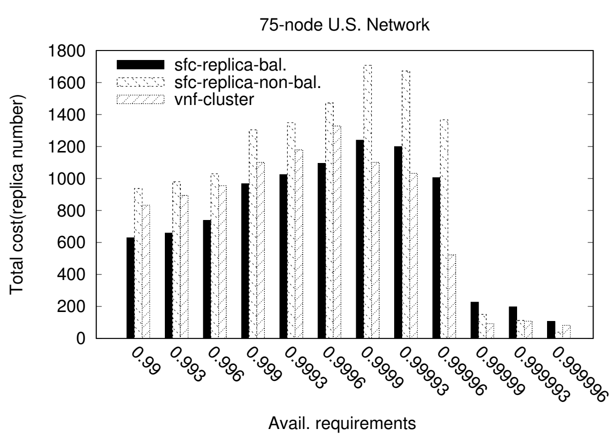


Figure 6.6. Total computing resources

results show that the proposed *SFC replica* based algorithm performs better in terms of reducing the SFC blocking and the total computing resources.

CHAPTER 7

CONCLUSION

In this dissertation, considering independent and correlated failures in the physical optical network, we address multiple availability-based virtual optical network problems by providing different protection mechanisms.

In Chapter 2, we analyze the availability of virtual optical networks mapped over a physical substrate and propose VON mapping algorithms with selective dedicated protection to minimize cost while meeting availability requirements.

In Chapter 3, we consider virtual optical network (VON) mapping with the objective of minimizing total network link cost while guaranteeing VON availability, where VON availability is supported by providing shared backup path protection for selected VON links. We develop a matrix-based approach for calculating the availability of a VON mapping with shared backup path protection. In order to efficiently evaluate the maximum availability of a VON mapping, we transform the problem to a group node-weighted Steiner tree problem and propose an efficient auxiliary-graph-based availability (AA) algorithm to find a VON mapping with high availability. Based on the availability evaluation, we propose a heuristic algorithm to map the VON, and numerical results show that our algorithms are effective in achieving high availability while reducing the total link cost and the blocking rate.

In Chapter 4, we address the high-availability VON mapping problem over physical optical networks in which physical links may suffer correlated failures. We first aim to find a VON mapping that provides the maximum availability without any protection, and then we provide a dedicated path protection mechanism to guarantee the availa bility of each VON if needed. We propose the Group Steiner tree problem under correlated cost (GSTP-CC), which can be used to determine the maximum availability of a VON mapping w hen no protection resources are provisioned. For a special case of this problem in which there are only nodal correlated failures, called the Group Steiner tree problem under noda l correlated failure (GSTP-NCF), we propose an approximate solution based on hierarchically well-separated trees. For the general GSTP-CC problem, we propose two heuristic algorit hms to find the maximum availability of the VON mapping under correlated failures. We then address the VON mapping under correlated failures by providing dedicated protection based on the GSTP-CC algorithms

In Chapter 5, we propose a coordinated protection mechanism that adopts both backup path protection in the network and VNF replicas at nodes to guarantee a SFCs availability. The proposed mechanism determines the number of replicas required for each VNF in the SFC, and allocates the replicas to physical nodes on the working and backup paths while maintaining ordered dependency among VNFs. To further improve the SFC request acceptance and reduce the total computing and link resource consumption, we also propose an efficient segment protection mechanism to guarantee each SFC's availability requirement. Simulation results show that the proposed algorithms contribute to reducing the SFC blocking and the cost of computing and link resources.

In Chapter 6, Considering the availability of the datacenter's network elements, we propose a coordinated protection mechanism that adopts both backup path protection and SFC replicas distributed among datacenters to support high availability while reducing total cost.

APPENDIX

PROOF OF THEOREM 4.6

In this section, we give the proof of Theorem 4.6 as follows.

A metric space is an ordered pair (M, d) where M is a set and d is a metric on M such that for any $x, y, z \in M$, the following holds:

1. $d(u, v) > 0$
2. $d(u, v) = 0 \iff u = v$
3. $d(u, v) = d(v, u)$
4. $d(u, v) \leq d(u, w) + d(w, v)$

It is obvious that the items 1, 2, 3 hold in the auxilary graph $G_a(N_s, w)$. We need to guarantee that:

$$d(u, v) < d(u, w) + d(w, v) \quad \Longleftarrow \quad \pi_0 > (\pi_0 + \pi_1)(\pi_0 + \pi_2) \tag{1}$$

where $d(u, v)$ is the correlated cost of $link_{u,w}$ and $link_{w,v}$.

Consider the two link Markov model as shown in Fig. 4.3, if $\pi_0 > \frac{\sqrt{5}+1}{2} \approx 0.618$, we can guarantee that:

$$\pi_0 > (\pi_0 + \pi_1)(\pi_0 + \pi_2) \tag{2}$$

We obtain the following equations from the Markov model:

$$\begin{cases} \pi_0 + \pi_1 + \pi_2 + \pi_3 &= 1 \\ (\lambda_0 + \lambda_1 + \lambda_{0,1})\pi_0 - \mu_0\pi_1 - \mu_1\pi_2 &= 0 \\ (\mu_0 + \lambda_1)\pi_1 - \lambda_0\pi_0 - \mu_1\pi_3 &= 0 \\ (\mu_1 + \lambda_0)\pi_2 - \lambda_1\pi_0 - \mu_0\pi_3 &= 0 \end{cases} \tag{3}$$

Solving the above equations yields

$$\begin{cases} \pi_1 &= \frac{\lambda_0\Delta+\lambda_0\lambda_{0,1}+\mu_1\lambda_{0,1}}{\Delta\mu_0}\pi_0 \\ \pi_2 &= \frac{\lambda_1\Delta+\lambda_1\lambda_{0,1}+\mu_0\lambda_{0,1}}{\Delta\mu_1}\pi_0 \\ \pi_3 &= \frac{\lambda_0\lambda_1\Delta+(\mu_0+\lambda_1)(\lambda_0\lambda_{0,1}+\mu_1\lambda_{0,1})}{\Delta\mu_0\mu_1}\pi_0 \end{cases} \tag{4}$$

where $\Delta = \mu_0 + \mu_1 + \lambda_0 + \lambda_1$.

We obtain:

$$1 + \frac{\pi_1}{\pi_0} + \frac{\pi_2}{\pi_0} + \frac{\pi_3}{\pi_0} = \frac{1}{\pi_0} \tag{5}$$

If $\pi_0 > (\pi_0 + \pi_1)(\pi_0 + \pi_2)$, we need:

$$\begin{aligned} & \tfrac{1}{\pi_0} > (1 + \tfrac{\pi_1}{\pi_0})(1 + \tfrac{\pi_2}{\pi_0}) \\ \Leftarrow\ & 1 + \tfrac{\pi_1}{\pi_0} + \tfrac{\pi_2}{\pi_0} + \tfrac{\pi_3}{\pi_0} > (1 + \tfrac{\pi_1}{\pi_0})(1 + \tfrac{\pi_2}{\pi_0}) \\ \Leftarrow\ & \tfrac{\pi_3}{\pi_0} > \tfrac{\pi_1}{\pi_0}\tfrac{\pi_2}{\pi_0} \\ \Leftarrow\ & \tfrac{\lambda_0\lambda_1\Delta+(\mu_0+\lambda_1)(\lambda_0\lambda_{0,1}+\mu_1\lambda_{0,1})}{\Delta\mu_0\mu_1} \\ & > \tfrac{\lambda_0\Delta+\lambda_0\lambda_{0,1}+\mu_1\lambda_{0,1}}{\Delta\mu_0}\tfrac{\lambda_1\Delta+\lambda_1\lambda_{0,1}+\mu_0\lambda_{0,1}}{\Delta\mu_1} \\ \Leftarrow\ & \mu_0\mu_1\lambda_{0,1}\Delta > \lambda_0\lambda_1\lambda_{0,1}\Delta + \lambda_{0,1}^2(\lambda_0+\mu_1)(\mu_0+\lambda_1) \\ \Leftarrow\ & a\mu_0\mu_1\lambda_{0,1}\Delta > \lambda_0\lambda_1\lambda_{0,1}\Delta \quad \&\& \\ & (1-a)\mu_0\mu_1\lambda_{0,1}\Delta > \lambda_{0,1}^2(\lambda_0+\mu_1)(\mu_0+\lambda_1) \end{aligned} \tag{6}$$

where $0 < a < 1$.

Assume that $\pi_0 > \frac{1}{1+\omega}$, therefore,

$$\frac{1}{\pi_0} < 1 + \omega \tag{7}$$

We can calculate π_0 from the above equations:

$$\begin{aligned} \tfrac{1}{\pi_0} &= 1 + \tfrac{\lambda_0}{\mu_0} + \tfrac{\lambda_1}{\mu_1} + \tfrac{\lambda_1\lambda_1}{\mu_0\mu_1} + \Theta, \\ &< 1 + \omega \end{aligned} \tag{8}$$

where $\Theta = \lambda_{0,1}\frac{(\mu_0+\mu_1)(\lambda_0+\mu_1)+(\lambda_1+\mu_0)\mu_0+(\lambda_0+\mu_1)\mu_1}{\Delta\mu_0\mu_1}$, and each item in the above equations less or equal than ω

If $a\mu_0\mu_1\lambda_{0,1}\Delta > \lambda_0\lambda_1\lambda_{0,1}\Delta$, we should guarantee:

$$\omega^2 < a, \tag{9}$$

Since:

$$\frac{\lambda_{0,1}}{\Delta}\frac{(\lambda_0+\mu_1)(\mu_0+\lambda_1)}{\mu_0\mu_1} < \Theta < \omega, \tag{10}$$

If we meet the condition:

$$\omega < 1 - a, \tag{11}$$

we obtain:

$$(1-a)\mu_0\mu_1\lambda_{0,1}\Delta > \lambda_{0,1}^2(\lambda_0+\mu_1)(\mu_0+\lambda_1), \tag{12}$$

Solve the above equations, we obtain:

$$\omega < \frac{\sqrt{5}-1}{2} \ and\ a = \frac{3-\sqrt{5}}{2}, \tag{13}$$

Therefore, Theorem 4.6 holds when:

$$\pi_0 > \frac{1}{1+\omega} = \frac{\sqrt{5}+1}{2} \approx 0.618. \tag{14}$$

REFERENCES

[1] Addis, B., D. Belabed, M. Bouet, and S. Secci (2015). Virtual network functions placement and routing optimization. In *CloudNet*, pp. 171–177.

[2] Agarwal, P. K., A. Efrat, S. K. Ganjugunte, D. Hay, S. Sankararaman, and G. Zussman (2013). The resilience of wdm networks to probabilistic geographical failures. *IEEE/ACM Transactions on Networking (TON) 21*(5), 1525–1538.

[3] Bartal, Y. (1996). Probabilistic approximation of metric spaces and its algorithmic applications. In *Foundations of Computer Science, 1996. Proceedings., 37th Annual Symposium on*, pp. 184–193. IEEE.

[4] Bartal, Y. (1998). On approximating arbitrary metrices by tree metrics. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pp. 161–168. ACM.

[5] Basta, A., W. Kellerer, M. Hoffmann, H. J. Morper, and K. Hoffmann (2014). Applying nfv and sdn to lte mobile core gateways, the functions placement problem. In *Proceedings of the 4th workshop on All things cellular: operations, applications, and challenges*, pp. 33–38. ACM.

[6] Beck, M. T. and J. F. Botero (2017). Scalable and coordinated allocation of service function chains. *Computer Communications 102*, 78–88.

[7] Belbekkouche, A., M. M. Hasan, and A. Karmouch (2012). Resource discovery and allocation in network virtualization. *IEEE Communications Surveys & Tutorials 14*(4), 1114–1128.

[8] Carpio, F., S. Dhahri, and A. Jukan (2017). Vnf placement with replication for loac balancing in nfv networks. In *Communications (ICC), 2017 IEEE International Conference on*, pp. 1–6. IEEE.

[9] Casazza, M., P. Fouilhoux, M. Bouet, and S. Secci (2017). Securing virtual network function placement with high availability guarantees. In *IFIP Networking Conference (IFIP Networking) and Workshops, 2017*, pp. 1–9. IEEE.

[10] Chen, X., M. Tornatore, S. Zhu, F. Ji, W. Zhou, C. Chen, D. Hu, L. Jiang, and Z. Zhu (2015). Flexible availability-aware differentiated protection in software-defined elastic optical networks. *Journal of Lightwave Technology 33*(18), 3872–3882.

[11] Cheriton, D. and R. E. Tarjan (1976). Finding minimum spanning trees. *SIAM Journal on Computing 5*(4), 724–742.

[12] Chowdhury, N. M. K., M. R. Rahman, and R. Boutaba (2009). Virtual network embedding with coordinated node and link mapping. In *INFOCOM 2009, IEEE*, pp. 783–791. IEEE.

[13] Cohen, R., L. Lewin-Eytan, J. S. Naor, and D. Raz (2015). Near optimal placement of virtual network functions. In *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pp. 1346–1354. IEEE.

[14] Cvijetic, M. and I. Djordjevic (2013). *Advanced optical communication systems and networks*. Artech House.

[15] Datta, P. and A. K. Somani (2004). Diverse routing for shared risk resource groups (srrg) failures in wdm optical networks. In *Broadband Networks, 2004. BroadNets 2004. Proceedings. First International Conference on*, pp. 120–129. IEEE.

[16] Demaine, E. D., M. Hajiaghayi, and P. N. Klein (2014). Node-weighted steiner tree and group steiner tree in planar graphs. *ACM Transactions on Algorithms (TALG) 10*(3), 13.

[17] Ding, W., H. Yu, and S. Luo (2017). Enhancing the reliability of services in nfv with the cost-efficient redundancy scheme. In *Communications (ICC), 2017 IEEE International Conference on*, pp. 1–6. IEEE.

[18] Doucette, J., M. Clouqueur, and W. D. Grover (2003). On the availability and capacity requirements of shared backup path-protected mesh networks. *Optical Networks Magazine 4*(6), 29–44.

[19] Fan, J., Z. Ye, C. Guan, X. Gao, K. Ren, and C. Qiao (2015). Grep: Guaranteeing reliability with enhanced protection in nfv. In *Proceedings of the 2015 ACM SIGCOMM Workshop on Hot Topics in Middleboxes and Network Function Virtualization*, pp. 13–18. ACM.

[20] Faraci, V. (2006). Calculating failure rates of series/parallel networks. *The Journal of Alion s, System Reliability Center*.

[21] Fischer, A., J. F. Botero Vega, M. Duelli, D. Schlosser, X. Hesselbach Serra, and H. De Meer (2011). Alevin-a framework to develop, compare, and analyze virtual network embedding algorithms. In *Open-Access-Journal Electronic Communications of the EASST*, pp. 1–12.

[22] Galdamez, C. and Z. Ye (2017, April). Resilient virtual network mapping against large-scale regional failures. In *2017 26th Wireless and Optical Communication Conference (WOCC)*, pp. 1–4.

[23] Gao, C., H. C. Cankaya, and J. P. Jue (2014). Survivable inter-domain routing based on topology aggregation with intra-domain disjointness information in multi-domain optical networks. *Journal of Optical Communications and Networking 6*(7), 619–628.

[24] Garey, M. R. and D. S. Johnson (1977). The rectilinear steiner tree problem is np-complete. *SIAM Journal on Applied Mathematics 32*(4), 826–834.

[25] Garg, N., G. Konjevod, and R. Ravi (2000). A polylogarithmic approximation algorithm for the group steiner tree problem. *Journal of Algorithms 37*(1), 66–84.

[26] Gong, L. and Z. Zhu (2014). Virtual optical network embedding (vone) over elastic optical networks. *Journal of Lightwave Technology 32*(3), 450–460.

[27] Goscien, R., K. Walkowiak, M. Klinkowski, and J. Rak (2015). Protection in elastic optical networks. *IEEE Network 29*(6), 88–96.

[28] Gour, R., J. Kong, G. Ishigaki, A. Yousefpour, S. Hong, and J. P. Jue (2017). Survivable routing in multi-domain optical networks with geographically correlated failures. In *GLOBECOM 2017-2017 IEEE Global Communications Conference*, pp. 1–6. IEEE.

[29] Gour, R., J. Kong, G. Ishigaki, A. Yousefpour, S. Hong, and J. P. Jue (2018). Finding survivable routes in multi-domain optical networks with geographically correlated failures. *Journal of Optical Communications and Networking 10*(8), C39–C49.

[30] Gu, L., X. Chen, H. Jin, and F. Lu (2018). Vnf deployment and flow scheduling in geo-distributed data centers. In *2018 IEEE International Conference on Communications (ICC)*, pp. 1–6. IEEE.

[31] Habib, M. F., M. Tornatore, F. Dikbiyik, and B. Mukherjee (2013). Disaster survivability in optical communication networks. *Computer Communications 36*(6), 630–644.

[32] Halpern, J. and C. Pignataro (2015). Service function chaining (sfc) architecture. Technical report.

[33] Han, B., V. Gopalakrishnan, L. Ji, and S. Lee (2015). Network function virtualization: Challenges and opportunities for innovations. *IEEE Communications Magazine 53*(2), 90–97.

[34] Herker, S., X. An, W. Kiess, S. Beker, and A. Kirstaedter (2015). Data-center architecture impacts on virtualized network functions service chain embedding with high availability requirements. In *Globecom Workshops (GC Wkshps), 2015 IEEE*, pp. 1–7. IEEE.

[35] Herker, S., X. An, W. Kiess, and A. Kirstadter (2013). Path protection with explicit availability constraints for virtual network embedding. In *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on*, pp. 2978–2983. IEEE.

[36] Herker, S., W. Kiess, X. An, and A. Kirstadter (2014). On the trade-off between cost and availability of virtual networks. In *Networking Conference, 2014 IFIP*, pp. 1–9. IEEE.

[37] Herrera, J. G. and J. F. Botero (2016). Resource allocation in nfv: A comprehensive survey. *IEEE Transactions on Network and Service Management 13*(3), 518–532.

[38] Hong, S., J. P. Jue, P. Park, H. Yoon, H. Ryu, and S. Hong (2015). Survivable virtual topology design in ip over wdm multi-domain networks. In *Communications (ICC), 2015 IEEE International Conference on*, pp. 5150–5155. IEEE.

[39] Hong, S., J. P. Jue, Q. Zhang, X. Wang, H. C. Cankaya, C. She, and M. Sekiya (2014). Virtual optical network embedding in multi-domain optical networks. In *Global Communications Conference (GLOBECOM), 2014 IEEE*, pp. 2042–2047. IEEE.

[40] Jiang, H., Y. Wang, L. Gong, and Z. Zhu (2015). Availability-aware survivable virtual network embedding in optical datacenter networks. *IEEE/OSA Journal of Optical Communications and Networking 7*(12), 1160–1171.

[41] Jinno, M., H. Takara, K. Yonenaga, and A. Hirano (2013). Virtualization in optical networks from network level to hardware level. *Journal of Optical Communications and Networking 5*(10), A46–A56.

[42] Kang, J., O. Simeone, and J. Kang (2017). On the trade-off between computational load and reliability for network function virtualization. *IEEE Communications Letters 21*(8), 1767–1770.

[43] Klein, P. and R. Ravi (1995). A nearly best-possible approximation algorithm for node-weighted steiner trees. *Journal of Algorithms 19*(1), 104–115.

[44] Kong, J., S. Hong, J. Jue, I. Kim, X. Wang, Q. Zhang, H. C. Cankaya, W. Xie, and T. Ikeuchi (2016). Availability-guaranteed virtual optical network mapping with selective path protection. In *Optical Fiber Communication Conference*, pp. W1B–4. Optical Society of America.

[45] Kong, J. and J. Jue (2016). Correlated-failure-aware resilient virtual optical network mapping. In *2016 Technical Report*, Number UTDCS-14-16. University of Texas at Dallas.

[46] Kong, J., J. Jue, I. Kim, X. Wang, Q. Zhang, H. C. Cankaya, W. Xie, and T. Ikeuchi (2016). Availability-guaranteed virtual optical network mapping with shared backup path protection. In *2016 IEEE Global Communications Conference*. IEEE.

[47] Kong, J., I. Kim, X. Wang, Q. Zhang, H. C. Cankaya, W. Xie, T. Ikeuchi, and J. P. Jue (2017). Guaranteed-availability network function virtualization with network protection and vnf replication. In *GLOBECOM 2017-2017 IEEE Global Communications Conference*, pp. 1–6. IEEE.

[48] Kong, J., I. Kim, X. Wang, Q. Zhang, W. Xie, H. C. Cankaya, N. Wang, T. Ikeuchi, and J. P. Jue (2018). Guaranteed-availability network function virtualization in inter-datacenter networks. In *Optical Fiber Communication Conference*, pp. W1D–1. Optical Society of America.

[49] Kong, J., N. Wang, J. P. Jue, I. Kim, X. Wang, Q. Zhang, H. C. Cankaya, W. Xie, and T. Ikeuchi (2017). Correlated-failure-aware von mapping. In *Optical Fiber Communications Conference and Exhibition (OFC), 2017*, pp. 1–3. IEEE.

[50] Lin, T., Z. Zhou, M. Tornatore, and B. Mukherjee (2016). Demand-aware network function placement. *Journal of Lightwave Technology 34*(11), 2590–2600.

[51] Lucerna, D., M. Tornatore, B. Mukherjee, and A. Pattavina (2012). Trading availability among shared-protected dynamic connections in wdm networks. *Computer Networks 56*(13), 3150–3162.

[52] Ma, C., J. Zhang, Y. Zhao, M. F. Habib, S. S. Savas, and B. Mukherjee (2015, November). Traveling repairman problem for optical network recovery to restore virtual networks after a disaster [invited]. *IEEE/OSA Journal of Optical Communications and Networking 7*(11), B81–B92.

[53] Maier, G., A. Pattavina, S. De Patre, and M. Martinelli (2002). Optical network survivability: protection techniques in the wdm layer. *Photonic Network Communications 4*(3-4), 251–269.

[54] Medhat, A. M., T. Taleb, A. Elmangoush, G. A. Carella, S. Covaci, and T. Magedanz (2017). Service function chaining in next generation networks: State of the art and research challenges. *IEEE Communications Magazine 55*(2), 216–223.

[55] Meixner, C. C., F. Dikbiyik, M. Tornatore, C. Chuah, and B. Mukherjee (2013, April). Disaster-resilient virtual-network mapping and adaptation in optical networks. In *2013 17th International Conference on Optical Networking Design and Modeling (ONDM)*, pp. 107–112.

[56] Neumayer, S. and E. Modiano (2010). Network reliability with geographically correlated failures. In *INFOCOM, 2010 Proceedings IEEE*, pp. 1–9. IEEE.

[57] Pages, A., J. Perello, S. Spadaro, and G. Junyent (2012). Strategies for virtual optical network allocation. *IEEE Communications Letters 16*(2), 268–271.

[58] Potharaju, R. and N. Jain (2013). Demystifying the dark side of the middle: a field study of middlebox failures in datacenters. In *Proceedings of the 2013 conference on Internet measurement conference*, pp. 9–22. ACM.

[59] Qu, L., C. Assi, K. Shaban, and M. Khabbaz (2017). A reliability-aware network service chain provisioning with delay guarantees in nfv-enabled enterprise datacenter networks. *IEEE Transactions on Network and Service Management 14*(3), 554–568.

[60] Rahnamay-Naeini, M., J. E. Pezoa, G. Azar, N. Ghani, and M. M. Hayat (2011). Modeling stochastic correlated failures and their effects on network reliability. In *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on*, pp. 1–6. IEEE.

[61] Richardson, T., Q. Stafford-Fraser, K. R. Wood, and A. Hopper (1998). Virtual network computing. *IEEE Internet Computing 2*(1), 33–38.

[62] Shakya, S., N. Pradhan, X. Cao, Z. Ye, and C. Qiao (2014). Virtual network embedding and reconfiguration in elastic optical networks. In *Global Communications Conference (GLOBECOM), 2014 IEEE*, pp. 2160–2165. IEEE.

[63] She, Q., X. Huang, and J. P. Jue (2010). How reliable can two-path protection be? *IEEE/ACM Transactions on Networking (TON) 18*(3), 922–933.

[64] Shen, G., H. Guo, and S. K. Bose (2016). Survivable elastic optical networks: survey and perspective. *Photonic Network Communications 31*(1), 71–87.

[65] Suurballe, J. W. and R. E. Tarjan (1984). A quick method for finding shortest pairs of disjoint paths. *Networks 14*(2), 325–336.

[66] Tornatore, M., M. Carcagní, and A. Pattavina (2010). Availability formulations for segment protection. *IEEE Transactions on Communications 58*(4).

[67] Tornatore, M., D. Lucerna, B. Mukherjee, and A. Pattavina (2012). Multilayer protection with availability guarantees in optical wdm networks. *Journal of Network and Systems Management 20*(1), 34–55.

[68] Wang, N. and J. P. Jue (2014). Holding-time-aware routing, modulation, and spectrum assignment for elastic optical networks. In *Global Communications Conference (GLOBECOM), 2014 IEEE*, pp. 2180–2185. IEEE.

[69] Wang, N., J. P. Jue, J. Kong, X. Wang, and Q. Zhang (2017). Bulk data flows scheduling over non-disjoint multiple paths in elastic optical networks. In *Photonics in Switching*, pp. PM3D–3. Optical Society of America.

[70] Wang, N., J. P. Jue, X. Wang, Q. Zhang, H. C. Cankaya, Q. She, W. Xie, and M. Sekiya (2015). Scheduling large data flows in elastic optical inter-datacenter networks. In *Global Communications Conference (GLOBECOM), 2015 IEEE*, pp. 1–6. IEEE.

[71] Wang, N., J. P. Jue, and R. Zhu (2016). Survivable bulk data-flow transfer strategies in elastic optical inter-datacenter networks. In *Global Communications Conference (GLOBECOM), 2016 IEEE*, pp. 1–6. IEEE.

[72] Wei, X., L. Guo, X. Wang, Q. Song, and L. Li (2008). Availability guarantee in survivable wdm mesh networks: A time perspective. *Information Sciences 178*(11), 2406–2415.

[73] Weston-Dawkes, J. (2014). Calculating availability bounds in complex systems and ip network topologies. In *Optical Fiber Communication Conference*, pp. W2A–11. Optical Society of America.

[74] Xie, W., J. P. Jue, Q. Zhang, X. Wang, Q. She, P. Palacharla, and M. Sekiya (2014). Survivable virtual optical network mapping in flexible-grid optical networks. In *Computing, Networking and Communications (ICNC), 2014 International Conference on*, pp. 221–225. IEEE.

[75] Yang, S., S. Trajanovski, and F. A. Kuipers (2015). Shortest paths in networks with correlated link weights. In *2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 660–665. IEEE.

[76] Ye, Z., X. Cao, J. Wang, H. Yu, and C. Qiao (2016). Joint topology design and mapping of service function chains for efficient, scalable, and reliable network functions virtualization. *IEEE Network 30*(3), 81–87.

[77] Yousefpour, A., G. Ishigaki, and J. P. Jue (2017). Fog computing: Towards minimizing delay in the internet of things. In *Edge Computing (EDGE), 2017 IEEE International Conference on*, pp. 17–24. IEEE.

[78] Yu, H., C. Qiao, J. Wang, L. Li, V. Anand, and B. Wu (2014, November). Regional failure-resilient virtual infrastructure mapping in a federated computing and networking system. *IEEE/OSA Journal of Optical Communications and Networking 6*(11), 997–1007.

[79] Yu, M., Y. Yi, J. Rexford, and M. Chiang (2008). Rethinking virtual network embedding: substrate support for path splitting and migration. *ACM SIGCOMM Computer Communication Review 38*(2), 17–29.

[80] Zhang, J., W. Wu, and J. Lui (2018). On the theory of function placement and chaining for network function virtualization. In *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 91–100. ACM.

[81] Zhang, J., K. Zhu, H. Zang, N. S. Matloff, and B. Mukherjee (2007). Availability-aware provisioning strategies for differentiated protection services in wavelength-convertible wdm mesh networks. *Networking, IEEE/ACM Transactions on 15*(5), 1177–1190.

[82] Zhu, R., Y. Zhao, H. Yang, Y. Tan, X. Yu, G. Gao, J. Zhang, N. Wang, and J. P. Jue (2016). Crosstalk-aware virtual optical network embedding (vone) in spatial division multiplexing enabled elastic optical networks with multi-core fibers. In *ECOC 2016; 42nd European Conference on Optical Communication; Proceedings of*, pp. 1–3. VDE.

BIOGRAPHICAL SKETCH

Jian Kong received his BS in Automation at Shandong University, Jinan, China in 2007 and his MS in Control Theory and Control Engineering at Beihang University, Beijing, China in 2010. He worked as a Software Integration Test Engineer at Huawei after his graduation in 2010. He also worked at Huahuan Electronics Co., Ltd. as a software engineer from 2013 to 2014. Since 2014, he has been pursuing his PhD in the Department of Electrical Engineering at The University of Texas at Dallas where his research focuses on the availability-based resilient virtual optical network. He is a member of the Advanced Network Research Lab (ANRL) under Dr. Jason Jue's supervision. He has been a research and teaching assistant from 2014 to 2018. During the spring and summer semesters 2017, he was a full-time Software Intern with Futurewei Technologies, Inc., and during summer 2018, he was a full-time Research Intern at Fujitsu Labs of America.

CURRICULUM VITAE

Jian Kong

December 18, 2018

Contact Information:

Department of Electrical Engineering
The University of Texas at Dallas
800 W. Campbell Rd.
Richardson, TX 75080-3021, U.S.A.

Email: jxk133130@utdallas.edu

Educational History:

B.S., Automation, Shandong University, China, 2007
M.S., Electrical Engineering, Beihang University, China, 2010

Availability-based Resilient Virtual Optical Network Mapping
Ph.D. Dissertation
Telecommunications Engineering, The University of Texas at Dallas
Advisors: Dr. Jason P. Jue

Employment History:

Research Intern, Fujitsu Laboratories of America, Inc., May 2018 – August 2018
Software Intern, Futurewei Technologies, Inc., January 2017 – August 2017
Software Engineer, Huahuan Electronics Co., Ltd., April 2013 – July 2014
System Test Engineer, Huawei Technologies Co., Ltd., August 2010 – November 2012