Computer Networks 62 (2014) 137–151

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Design of reliable virtual infrastructure with resource sharing

Hao Di [a], Vishal Anand [c], Hongfang Yu [a,*], Lemin Li [a], Gang Sun [a,b], Dan Liao [a,b]

[a] Key Lab of Optical Fiber Sensing and Communications, Ministry of Education, University of Electronic Science and Technology of China, China
[b] Institute of Electronic and Information Engineering in Dongguan, China
[c] Department of Computer Science, The College at Brockport, State University of New York, USA

ARTICLE INFO

Article history:
Received 29 September 2012
Received in revised form 26 August 2013
Accepted 2 September 2013
Available online 1 December 2013

Keywords:
Network virtualization
Virtual infrastructure mapping
Virtual infrastructure reliability
Mathematical optimization
Cross sharing
Backup sharing

ABSTRACT

Network virtualization technology allows multiple virtual infrastructures (VIs) or virtual networks customized to suit user requirements on a shared substrate network without the need for much additional infrastructure. However, in such a scenario, even a single failure in the substrate can cause large disruptions as it will affect all the VIs mapped onto it. Thus, the problem of guaranteeing survivability and reliability of VI is important, and even more complex than in single layer networks. In this paper, we study the reliable VI mapping problem that enables efficient resource sharing by using the cross sharing scheme that reuses the primary bandwidth for backup VI links. In addition, we also study the sharing of backup resources across multiple different VI requests.

We first formulate the reliable VI mapping problem as a mixed integer programming (MIP) problem to minimize the VI mapping cost. We then present our reliable VI mapping (RVIM) algorithm for efficient bandwidth sharing using the cross and backup sharing schemes. Finally, we present the VI combination (ComVI) algorithm to generate shared backup VI nodes and links for multiple VI requests in a cost efficient manner. Through simulation we show that our RVIM algorithm can reduce the mapping costs when compared to other VI mapping algorithms, and backup resource sharing using ComVI can reduce the total mapping cost efficiently.

1. Introduction

Network virtualization [1,2] enables the coexistence of multiple networking approaches on the same underlying substrate network architecture. For example, multiple virtual overlay networks with diverse topologies, as well as bandwidth and resilience properties can be created using the same physical substrate network. Network virtualization can help diversify the Internet and overcome its ossification, and also stimulate the development and deployment of new network technologies and advanced applications. Optical networks using wavelength division multiplexing (WDM) [3] are a natural choice for the substrate network because of their high speed, enormous bandwidth and protocol transparency, such as federated computing and network systems in [4].

In a network virtualization environment, a service request with specific computing and network resource requests for execution can be modeled as virtual infrastructure (VI) request, e.g., using the MapReduce [5] model to do distributed computing on clusters of computers. Each such VI request is a collection of VI nodes connected together by a set of VI links to form a virtual topology [6]. To satisfy a VI request, the request should be allocated or mapped onto the facility nodes and paths in the substrate network, while satisfying the user's resource requirement subject to the resource constraints of the substrate network.

Since multiple VI requests are mapped onto the same underlying network, the problem of cost or resource

* Corresponding author. Tel.: +86 013880292298.
E-mail addresses: dihao@uestc.edu.cn (H. Di), vanand@brockport.edu (V. Anand), yuhf@uestc.edu.cn (H. Yu), gangsun@uestc.edu.cn (G. Sun), liaodan@uestc.edu.cn (D. Liao).

1389-1286/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.bjp.2013.09.022

efficient VI mapping is important. The work in [7–10] focuses on how to map VI nodes and links in a cost efficient manner without considering VI reliability or survivability. VI survivability is the capability of tolerating network failures by using backup resources that are assumed to be reliable [11–15]. Different from survivability, VI reliability [16,17] is defined as the probability of critical VI nodes of a VI request to continue to remain in operation even after the failure of VI nodes (primary or backup) due to facility node failures. In a shared virtualization environment reliable VI design is of utmost importance since multiple VI requests are mapped onto a shared substrate, and the failure of even a single substrate entity can affect multiple VI requests and cause large service disruptions. The works in [11–15] guarantee VI survivability under region, node and link failures, respectively. In this work, we assume that each VI node may fail with a given probability that is equal to the failure probability of the facility node (server) onto which it is mapped. We define VI reliability as the probability of all VI nodes of a VI request remaining in operation. To guarantee this reliability it is necessary to reserve resources for backup VI nodes and links.

In this study, we aim to minimize the cost of reliable VI mapping (i.e., the cost of resources allocated for VI requests). To guarantee the desired reliability of the VI requests, multiple backup VI nodes are needed. Furthermore, by protecting all primary VI nodes together by shared backup VI nodes we can minimize the required number of backup VI nodes. Thus, for reliable VI mapping, all the primary VI nodes are protected together to reduce the required backup computing resources. In addition, backup VI links are also necessary since the end nodes (i.e., primary VI nodes) of the primary VI links may be recovered (i.e., replaced) by backup VI nodes. We consider the *cross sharing* and *backup sharing* techniques to minimize the mapping cost. Backup sharing is the bandwidth sharing between the backup VI links that can be exploited since not all the backup links (or bandwidth) are used at the same time. Cross sharing is the reuse of the bandwidth reserved for unused primary VI links by the corresponding backup VI links. The work in [16,17] consider backup bandwidth sharing for reliable VI mapping, but ignore cross bandwidth sharing. In [17] the mapping of backup VI nodes and links without cross bandwidth sharing is modeled and solved by the D-ViNE algorithm [7], which relaxes the binary variables to obtain a linear program (LP). The work in [16] reduces the required backup bandwidth by sharing the backup bandwidth between the backup VI links that are not used simultaneously. Thus only the maximum of the required bandwidths of the backup VI links is used, and once the VI request is supplemented by backup VI nodes and links it is mapped using the graph isomorphism technique [8]. In [14], both backup bandwidth sharing and cross bandwidth sharing are considered, but no more than one node failure is assumed at any one time, thus the backup bandwidth related to each primary VI node can be shared with each other. In this work we allow more than one backup nodes to be used for recovery to guarantee the desired VI reliability. Accordingly the backup bandwidth sharing is more complex than that in [14] since sharing the backup bandwidth related to a particular primary VI node will affect the feasibility of sharing other backup bandwidth. After expanding the primary VI with the backup VI nodes and links, in our reliable virtual infrastructure mapping (RVIM) algorithm, the primary VI nodes and links are mapped by the Improved-vnmFlib algorithm [9], and then the backup VI nodes and links are mapped by the backup component mapping (BCM) sub-algorithm that uses backup and cross sharing.

We also study the reliable mapping for multiple VIs, using shared backup VI nodes and links. We remove the assumption that primary VI nodes in different VIs cannot be mapped onto the same facility nodes to minimize the overall mapping cost by combining multiple VIs into a single VI. The generation of shared backup VI nodes and links depends on the mapping of primary VIs, e.g., the primary VI nodes mapped onto a same facility node are to be recovered together. And the total required backup computing capacity depends on the number of required backup VI nodes and the capacity of a backup VI node (which is the maximum of all primary VI nodes protected by it). We can reduce the total used backup computing capacity by combining the primary VI nodes intelligently i.e., mapping the primary VI nodes in *different* VIs onto the same facility node. In this study, we combine/aggregate the primary VI nodes and links of multiple VIs before mapping using the ComVI algorithm to generate the shared components in a cost efficient manner. After combining multiple VIs into a single VI, the combined VI and shared backup components are mapped using the RVIM algorithm.

The rest of the paper is organized as follows. Section 2 describes the related works. Section 3 formulates the reliable VI mapping problem as a mixed integer programming (MIP) problem. Section 4 describes the RVIM algorithm. Section 5 describes the reliable mapping for multiple VIs and the ComVI algorithm. Section 6 presents the simulation results, and Section 7 concludes the paper.

2. Related work

There are a number of researches on the VI mapping problem, but not survivability or reliability [7–10]. These studies present the algorithms to determine how VI nodes and links are mapped onto the shared substrate while satisfying a set of constraints such as resource and QoS requirements. The algorithms search for possible VI mappings under finite substrate resources, while minimizing the cost of resources allocated for VIs (i.e., mapping cost). In [7], the VI mapping problem is formulated as a mixed integer programming (MIP) problem with coordinated node and link mapping. In the presented D-ViNE and R-ViNE algorithms, the binary variables are relaxed to the real-valued ones, and after solving the relaxed LP they are obtained using deterministic and randomized rounding (i.e., D-ViNE and R-ViNE), finally re-run the same LP with the rounded values. In [8], the vnmFlib algorithm is based on graph isomorphism, in which VI links are mapped onto substrate paths with a limited hop. And the mapping cost depends on the selected hop limit. In [9], we improve the vnmFlib algorithm to be more time and cost efficient. In the Improved-vnmFlib algorithm, we minimize the