Migration based Protection for Virtual Infrastructure Survivability for Link Failure

Hongfang Yu[1], Vishal Anand [2], Chunming Qiao [3], Hao Di[1]
[1]*School of Communication and Information Engineering, University of Electronic Science and Technology of China, China*
[2] Department of Computer Science, The College at Brockport, State University of New York, USA
[3] Department of Computer Science and Engineering, State University of New York at Buffalo, USA
Email: yuh2004@gmail.com ,vanand@brockport.edu, qiao@computer.org, haodi@uestc.edu.cn

Abstract: We propose a new migratory protection scheme that maps a virtual infrastructure to a substrate network using minimal resources to recover from a single substrate link failure. The efficiency of our solution is shown using simulation.
OCIS codes: (060.4257) Networks, network survivability

1. Introduction

Distributed computing over high-speed optical networks connecting multiple data centers is receiving an increasing amount of attention. Each distributed computing request may be represented using a graph called virtual infrastructure (VI) request consisting of a set of VI nodes and VI links. Each VI node needs to be mapped to a distinct facility node with required computing resources (e.g., CPU, memory and storage). A VI node also needs to communicate with other VI nodes to send e.g., intermediate results, and accordingly, each VI link also needs to be mapped to a physical path with required bandwidth. In a virtualized system, multiple VI requests can be mapped to the same substrate (consisting of physical nodes and links). How to map a VI request to a substrate in a resource efficient manner is a challenging research problem on its own [1-2], and how to perform survivable mapping in order to tolerate possible failures of the facility nodes and the substrate nodes/links is even more challenging [3-4].

Due to the shared nature of virtualization, even small failures of substrate network nodes and/or links can cripple many computations and communications, thus making survivability an important criterion. In this work we consider physical/substrate link failure recovery. Physical link failure-tolerant techniques have been widely studied in (optical) substrate networks. There are two commonly used approaches to protect optical link failures in a substrate network, namely shared protection and dedicated protection. In either approach, each primary path has a corresponding backup path, and shared protection is a networking domain approach that tries to optimize networking resources. However, like many prior works, shared path protection does not take into consideration the flexibility and capability of computing domain approaches when dealing with distributed computing applications. More specifically, it does not consider the possibility of relocating (or migrating) the computing task(s) whose communications are affected by the physical link failure. This has led to only locally efficient and effective approaches without overall optimality and reliability.

In this paper, we jointly optimize the networking and computing resources to tolerate link failures, and extend the shared protection scheme by incorporating a novel node migration technique, wherein upon failure a mapped VI node can be migrated and mapped onto another facility node to increase resource efficiency. Thus, instead of simply protecting a primary path by allocating bandwidth from an indicated source to an indicated destination, we *relocate* a VI node to another facility node which could be closer to the destination in terms of backup path length. This can result in an overall network optimization with a potential penalty in terms of extra cost for the facility nodes which need to run the relocated the computing task. We call this new approach as *migratory protection*. The major difference between this work and those in [3-4] is that here we adopt a different failure model and correspondingly, a new protection scheme.

This paper is first that 1) use the node migration principle in VI mapping for protection against link failures; 2) proposes a migratory protection scheme and the corresponding share rules; 3) compares the traditional shared protection with migratory shared protection in terms of various metrics under different environments. We show that the migratory protection approach can improve over traditional protection.

2. Problem Statement

Given: a substrate network $G_S=(N_S, E_S)$, where N_S is the set of substrate facility nodes, and E_S corresponds to the set of bidirectional fiber links and access links, and a VI request $G_V=(N_V, E_V)$, where N_V corresponds to the set of VI nodes, and E_V is the set of bidirectional communication demands among the VI nodes.
Question: how to find a mapping of the VI request on to the substrate network by jointly allocating computing and networking resources to recover from the failure of one physical link such that the sum of the computing and bandwidth resource cost is minimized?

To survive physical link failures, a simple approach is that each primary path that a VI link is mapped onto is protected by a link-disjoint backup path from the source to the destination of the corresponding primary path. These back-up paths can be shared as long as the corresponding primary paths are unlikely to fail at the same time. The corresponding survivable mapping solution includes: 1) a one-to-one node mapping from the VI nodes in N_V to the facility nodes in N_S, 2) mapping of each VI link $e \in E_V$ to a link-disjoint path pair in G_S, where one is the primary path, and the other is used as the backup path.

In this paper, we introduce a new approach called *migratory shared protection* that permits node migration to support protection. In this approach when the primary (substrate) path onto which a VI link is mapped fails, and if a link-disjoint backup path from the same (facility node) source and (facility node) destination cannot be found or is resource-inefficient, one end-node of this VI link is relocated (or migrated) to another (backup) facility node which is now link-disjoint and/or resource-efficient. Accordingly, all VI links (or connections) connected with the migrated VI node have to be remapped. These newly re-established paths originating from the new backup facility node form a tree, called a migratory backup tree. The migratory protection approach has more flexibility in supporting survivable VI mapping and results in a better performance than traditional protection scheme, e.g., it not only improves the resource utilization but also increases the success rate of finding a survivable mapping. Thus a survivable mapping solution with migratory protection includes: 1) a one-to-one node mapping from the VI nodes in N_V to the facility nodes in N_S, 2) mapping of each VI link $e \in E_V$ to a primary path from the original source node to the original destination node; 3) mapping of each VI link e to a link-disjoint backup path or migratory backup tree.

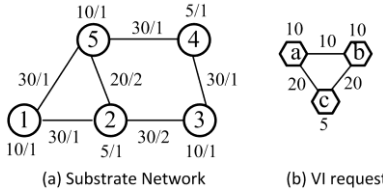

Fig. 1: A substrate network and a VI request

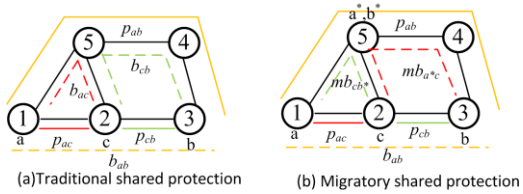

Fig. 2: Traditional shared protection and migratory shared protection example

3. Migratory shared protection

A. Motivation for migratory shared protection

Fig.1a shows a substrate network where the numbers x/y over the links indicate the available bandwidth of x units and the unit bandwidth cost of y, and the numbers over the nodes 1 through 5 represent the available computing resources and its unit cost at the substrate facility nodes. Fig.1b shows a VI request with three VI nodes a, b and c, and VI links, and the associated computing and communication requirements. Fig. 2a and Fig.2b illustrate the mapping results of traditional shared protection and migratory shared protection, respectively.

As shown in Fig.2a, the VI nodes *a*,*b* and *c* are mapped onto facility nodes 1,3 and 2, respectively The VI link (*a*,*b*) is protected by a link-disjoint path pair consisting of primary path p_{ab}= (1,5,4,3) shown in solid lines and backup path b_{ab}= (1,2,3), shown with dashed lines. Similarly the VI link (*a*,*c*) is protected by a link-disjoint path pair p_{ac}= (1,2) and b_{ac}= (1,5,2), and the VI link (*b*,*c*) is protected by a link-disjoint path pair p_{bc}= (2,3) and b_{bc}= (2,5,4). Note that since p_{ac} and p_{cb} are link-disjoint, their corresponding backup paths b_{ac} and b_{cb} can share the backup resources on fiber link (2, 5). The total resource cost equals to 245. From Fig.2a, we find that when the primary path used to map the corresponding VI link (*a*,*c*) fails, we can migrate the VI node *a* from facility node 1 to facility node 5, and use the shorter backup path (5,2) to recovery the failure, thus reducing the required backup resources. Based on this observation, we propose the migratory shared protection scheme as shown in Fig.2b.

First we define a migratory backup tree mb_{u*v}, which is link-disjoint with the primary path $p_{u,v}$ of the VI link (*u*,*v*) to protect the VI link (*u*,*v*) by migrating the end node *u*. As shown in Fig.2b, to tolerate the failure of the primary path p_{ac}= (1,2) that the VI link (*a*,*c*) is mapped onto, we migrate VI node *a* from facility node 1 to facility node 5, and establish a migratory tree mb_{a*c} from facility node 5 (which serves as the backup facility node for VI node a in this case) to facility nodes 2 and 3 onto which VI node *c* and *b* are mapped; Note that mb_{a*c} is link-disjoint with primary path p_{ac}. Similarly, for VI link (*c*,*b*), when the primary path p_{cb} fails, we migrate VI node *b* from facility node 3 to facility node 5, and establish a backup tree mb_{cb*} from facility node 5 (which serves as the backup facility node for VI node b in this case) to facility nodes 1 and 2 onto which VI nodes *a* and *c* are mapped; Note that mb_{cb*} is link-disjoint with the primary path p_{cb}. At the same time we note that mb_{a*c} only provides protection for primary path p_{ac} and thus can reuse the primary resources of physical links (5,4) and (4,3) on primary path p_{ab}, thus we only need to reserve backup resources on fiber link (5,1). We define this share strategy, e.g., sharing resource among the migratory backup tree and the corresponding migrated primary paths, as *intra-share*. In addition, for backup tree mb_{a*c} and mb_{cb*} , their corresponding primary paths p_{ac} and p_{cb} are link-disjoint, so their backup resources on fiber link (5,2) and backup facility node resources on facility node 5 can be shared. We define this sharing of backup

resources between different backup paths as *inter-share*. As shown in Fig.2b the total resources cost equals to 195, which is significantly lower than that needed by traditional shared protection.

Thus the policies for migratory protection are as follows: 1) Migratory backup tree or backup path must be link-disjoint with the corresponding primary path, 2) Migratory backup tree can share bandwidth resources with the corresponding migrated primary paths associated with the migrated VI node, 3) The computing resources at migratory backup facility nodes can be shared only when the primary paths that they protect are link-disjoint, and 4) Migratory backup trees and backup paths can share the backup bandwidth resource when their corresponding primary paths are disjoint.

B. A Three-step Heuristic Algorithm

We decompose the survivable mapping problem into three subproblems, 1) working/primary node mapping and working/primary link mapping (with no survivability consideration), 2) find a traditional shared backup path for each primary link, and 3) find migratory shared backup trees in order to improve performance. Firstly, we use our NSVIM* algorithm [3] to calculate the working node mapping and link mapping, then we use the heuristic algorithms in [5] to minimize the additional backup path resources. After obtaining the backup paths for each VI link we set the cost of links according to the share policies in migratory protection. We then find a migratory backup tree for each primary path with a minimum cost, and compare the performance between migratory protection and the traditional backup protection. If migratory protection can improve the performance, then we use the migratory backup tree as the recovery solution, otherwise we still use the traditional backup path as the recovery solution.

4. Simulation Results and Conclusion

We evaluate our algorithms in a substrate network with 27 node and 41 links. The computing capacity at facility nodes and bandwidth capacity on the links follow a uniform distribution from 100 to 300 units, and the computing and bandwidth requirements of VI requests follow a uniform distribution from 10 to 30 and 10 to 50 units, respectively.. The VI requests are generated randomly based on four main parameters: (i) the number of VI nodes in the VI request $|N|$, (ii) the average degree of VI nodes in the VI request, and (iii) the computing requirements of a VI request and (iv) bandwidth requirements of a VI request.

We compare the performance of using (i) migratory shared protection and (ii) traditional shared protection in terms of a) *total resource redundancy ratio*, which is the ratio of the difference in the total backup resource cost between migratory protection and traditional protection to the total working resource cost, and b) *node (computing) resource redundancy ratio*, similarly defined. We also show the effect of the size of the VI request (i.e., number of VI nodes in the VI request) in migratory protection in terms of 1) *number of migrated nodes*, which are migrated to new facility nodes and 2) *number of migrated paths* which is the total number of migrated primary paths.

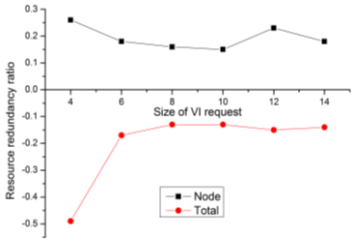

Fig.3: Sharing schemes comparison

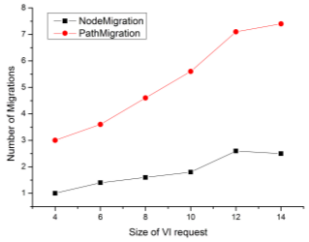

Fig.4: Number of Migrations

Fig. 3 shows the node resource redundancy ratio and total resource redundancy ratio of the migratory protection to the traditional protection. We can see that the node resource redundancy ratio is above 0, while the total resource redundancy ratio is below 0. This implies that although the migratory protection scheme requires more redundant node resources, it outperforms traditional protection in terms of total amount of redundant resources needed. In particular, the migratory protection saves approximately 15% to 50% resources when compared to the traditional protection. Fig.4 shows the number of node and path migrations in migratory protection approach. From the Fig., we note that as a performance tradeoff when using migratory protection, one to three VI nodes may need to be migrated along with 3 to 8 paths, while only one path needs to be migrated in conventional shared path protection.

References

[1] N. M. M. K. Chowdhury et al., "Virtual Network Embedding with Coordinated Node and Link Mapping", IEEE INFOCOM, 2009.
[2] M. Yu et al., "Rethinking virtual network embedding: Substrate support for path splitting and migration," SIGCOMM, vol.38, no.2, 2008.
[3] H. Yu et al. "On the Survivable Virtual Infrastructure Mapping Problem", IEEE ICCCN, Aug. 2010.
[4] X. Liu et al., "Robust Application Specific and Agile Private (ASAP) Networks Withstanding Multi-layer Failures," OFC/NFOEC, 2009.
[5] Yu Liu et al., "Spare capacity allocation in two-layer networks", IEEE JSAC, vol. 25, no. 5, June 2007: 974-986