Enhancing Virtual Infrastructure to Survive Facility Node Failures

Hongfang Yu[1], Vishal Anand[2], Chunming Qiao[3], Gang Sun[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, gangsun@uestc.edu.c

Abstract: We propose *1*-redundant and *K*-redundant schemes for the design and mapping of survivable virtual infrastructure to recover from facility node failures while minimizing network costs. The efficiency of our solutions is compared using simulation.

OCIS codes: (060.4257) Networks, network survivability

1. Introduction

Network *virtualization* [1] can help diversify the Internet by allowing various virtual networks to coexist on the same physical substrate network. Optical networks [2] are a natural choice for the substrate network because of their high speed, enormous bandwidth and protocol transparency. 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 focus on the failure recovery of the facility nodes to jointly minimize the total amount of computing and bandwidth resources (or cost).

A virtual infrastructure (VI) request consists of a set of VI nodes, with each node requiring some computing resources (e.g., CPU, memory and storage) at a *separate* facility node. A VI node also needs to communicate with other VI nodes to send intermediate results thus imposing strict connectivity requirements among the VI nodes in terms of topology, bandwidth, and delay guarantees. Thus, given a VI request we need to find a *one-to-one mapping* of the VI request onto the substrate facility nodes and physical paths.

In this work we use a two-step approach to fully restore a VI from any single facility node failure; in the first step the *reliable VI graph design* stage, the original VI graph (or request) is augmented to form a reliable VI graph with redundant VI nodes and links that have sufficient computational and bandwidth resources. In the second step, the *reliable VI graph mapping* stage, the reliable VI graph is mapped to the substrate network to minimize the total cost with guaranteed resources to tolerate any facility node failure. More specifically, we proposed two new approaches whereby an N-node VI is first enhanced to a *1*-redundant and *K*-redundant VI with N+1 and N+K nodes, respectively, in addition to an appropriate number of redundant virtual links. In the subsequent mapping of the enhanced VI to the substrate network, maximal amount of sharing of the computing and communication bandwidth among the nodes and links in the enhanced VI is exploited. Accordingly, it is possible that some N+k ($1 \leq k \leq K$) substrate nodes are chosen by the algorithm when mapping the enhanced VI with N+K nodes. How to enhance the VI and then maximize sharing in the subsequent mapping of the enhanced VI are both open problems.

More recently the works in [3-5] have considered VI survivability but do not use the two-step paradigm and accordingly were not concerned with the problems such as how to enhance a VI and how to map the enhanced VI with sharing among the virtual nodes and links. The work in [6] also uses a two-step approach to tolerate concurrent facility node and substrate link failure, but their enhanced VI graph design requires the VI to be duplicated two to three times, and consequently a different mapping methodology (especially in terms of the share strategy) is used.

2. Network Model and Problem Statement

A. Network Model

We model the substrate network as an undirected graph G_S=(N_S, E_S)=($N_F \cup N_X$, E_S), where N_S is the set of substrate nodes, and E_S corresponds to the set of bidirectional fiber links and access links. N_S consists of N_F and N_X, where N_F is the set of facility nodes and N_X is the set of optical switches. For each facility node $n \in N_F$, the available computing capacity is ε_n and the unit computing cost is c_n. For each link $(u,v) \in E_S$, the available bandwidth capacity is w_{uv}, and unit bandwidth cost is c_l. In this paper, we assume c_n=1 for all facility node n, and c_l= g for all fiber link l, where g is the ratio of the unit cost of computing resources to that of bandwidth resources.

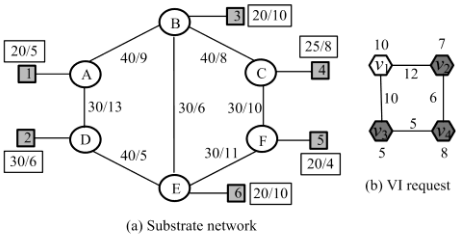
Fig. 1a shows a substrate network with 6 facility nodes (shaded squares), which are connected to 6 switching nodes (unshaded circles) that perform the computations, storage etc. The numbers over the links represent the available bandwidth and the cost of a bandwidth unit, and the numbers in the rectangles represent the available computing resources and cost of computing resource at the facility nodes.

A VI request is modeled as an undirected graph G_V= (N_V, C, E_V), where N_V corresponds to the set of VI nodes, C is the set of *critical* VI nodes (i.e., nodes that need a backup) and E_V is the set of bidirectional communication demands among the VI nodes. The computing resources required by a VI node u$\in N_V$ is denoted by ε_u. Similarly the bandwidth resources required by a communication demand $(u,v)\in E_V$ is denoted by α_{uv}. Fig. 1b shows a VI request with four VI nodes (critical nodes are shown shaded) and VI links, and associated computing and communication requirements.

B. Problem Statement

Given: a substrate network G_S=(N_S, E$_S$), a VI request G_V=(N_V, C, E_V).

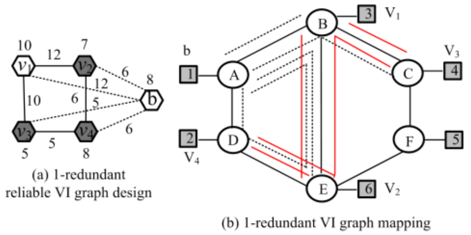
Question: how to design a reliable VI graph or enhanced VI graph and find a mapping of the same on to the substrate network by jointly allocating computing and networking resources to recover from the failure of one facility node such that the sum of the computing and bandwidth resource cost is minimized?



(a) Substrate network

(b) VI request

Fig. 1: Substrate network and a VI request



(a) 1-redundant reliable VI graph design

(b) 1-redundant VI graph mapping

Fig. 2: A *1*-redundant VI graph design and mapping example

3. Reliable VI Graph Design and Mapping

A. 1-redundant VI graph design

In this solution we transform the original non-survivable VI graph G_V to the *1*-redundant VI graph $G^1{}_V$ with redundancy by adding *one additional* VI node *b* that is used if any of the critical VI nodes in set *C* fails. The redundant node *b* should have connections (called redundant VI links) with all neighbors of all critical VI nodes. Formally, the set of redundant links that are added to G_V is $E^1{}_B$= $\{(b,v)\ |\exists(r,v)\in E_V,\forall c\in C,\ v\in N_V\}$, and $G^1{}_V$ can be denoted as $G^1{}_V$= ($N^1{}_V$, $E^1{}_V$), where $N^1{}_V$=$N_V\cup b$, $E^1{}_V$=$E_V\cup E^1{}_B$.

We call the set of redundant VI links that would be used simultaneously upon the failure of a certain VI node *v* as a *migratory association backup-link group*, denoted by *BG(v)*. We illustrate the working of the *1*-redundant solution in Fig. 2a that shows a VI request graph with three critical VI nodes and a redundant VI node *b*. Since *b* should connect with all neighbors of v_2, v_3 and v_4, four redundant VI links (v_1,*b*), (v_2,*b*), (v_3,*b*) and (v_4,*b*) are added. The computing resource requirement on redundant node *b* is 8 units, which is the maximum of the computing resources of all the critical nodes. Next we calculate the network bandwidth resource requirement as follows. When either of nodes v_2 or v_3 fails, the failed node will be migrated to *b* and redundant VI link (v_1,*b*) will be used to transport the communication traffic between v_2 and v_1 or between v_3 and v_1.The bandwidth requirement α_{v1b} on redundant VI link (v_1,*b*) is max($\alpha_{v1v2,}\alpha_{v1v3}$)=max(12,10)=12 units. Similarly, α_{v2b} is 6 units, α_{v3b} is 5 units, and α_{v4b} is 6 units.

B. 1-redundant VI graph Mapping

While mapping the 1-redundant VI graph it should be noted that as only one facility node *v* may fail at any one time not all redundant VI links belonging to different *BG*(*u*) will be used simultaneously, and hence we can share the physical link resources when mapping them onto the substrate network. When we share the resources among different backup paths belonging to different *BG*(*v*)*s*, we call such a sharing strategy as *backup share*. Also note that since we are considering facility node failures, none of the physical network nodes or links fails, and hence we can also share the bandwidth link resources between the original working path and its associated backup path. We call such a sharing strategy between working and backup paths as *cross share***.** We define the set of working VI links that would be simultaneously migrated to the backup VI links upon the failure of a VI node *v* as a *migratory association working-link group*, denoted by *WG*(*v*) .

Fig. 2b illustrates the mapping and sharing strategies discussed above for the mapping of the *1*-redundant reliable VI graph designed in Fig. 2a. Fig. 2b shows a substrate network where the VI nodes v_1, v_2, v_3 and v_4 are mapped onto facility nodes 3, 6, 4 and 2 respectively. As shown in Fig. 2b, original working VI links are mapped onto solid working paths and redundant VI links are mapped onto dashed backup paths. Redundant VI link (v_1,*b*) is mapped onto path A-B, redundant VI link (v_2,*b*) is mapped onto path A-B-E. Note that redundant link (v_1,*b*) is only used when critical node v_2 or v_3 fails; while redundant link (v_2,*b*) is only used when critical node v_4 fails. Hence the corresponding redundant paths A-B and A-B-E can share the backup bandwidth on fiber link A-B. This sharing is an example of backup share. Further, assume that redundant VI link (v_3,*b*) is mapped onto path A-B-C, and original working link (v_3,v_4) is mapped onto path C-B-E-D. When critical node v$_3$ fails, original VI link (v_3,v_4) in *WG*(v_3) is migrated to redundant link (v_3,*b*) in *BG*(v_3), so the corresponding redundant path A-B-C can reuse the bandwidth released by original working path C-B-E-D on fiber link B-C. This sharing is an example of cross share.

The *K*-redundant scheme is similar to the *1*-redundant scheme; we first design a *K*-redundant reliable VI graph (similar to Fig. 2a), in which we permit each critical node to have a corresponding backup node. Then we map the *K*-redundant VI graph onto the substrate network, such that no more than *K* backup facility nodes are used, where $K=|C|$. The actual number *k* of backup facility nodes used in the mapping stage could be anywhere from 1 to *K*.

C. Heuristic Algorithm

We formulate the problem of mapping the *1* and *K*-redundant VI graphs with the objective of minimizing costs as an MILP whose details we omit due to space limitation. The basic idea of our algorithm is to first use the D-ViNE algorithm [1] to find the working mapping for the original VI request. We then fix this working mapping and solve the simplified MILP using CPLEX to obtain only the backup solution.

4. Simulation Results and Conclusion

We compare the working of our algorithms for 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.

We compare the performance of using (i) both cross and backup share (labeled as "share"), (ii) only backup share ("bshare") and (iii) no share ("noshare") using the *redundancy ratio* performance metric, which is the ratio of the total backup resource cost to the total working resource cost. We also compare the performances of the *1* and *K*-redundant solutions using: 1) *node cost ratio* and 2) *link cost ratio*, which are the ratios of the node redundancy cost, and link redundancy cost incurred by the *K*-redundant solution to that incurred by the *1*-redundant solution.

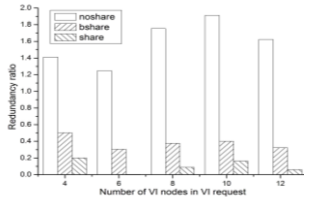


Fig.3: Sharing schemes comparison

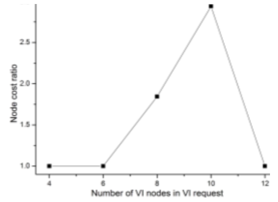


Fig.4: Node cost ratio comparison

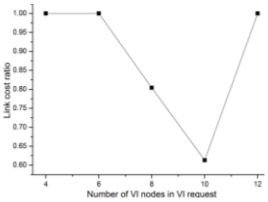


Fig.5: Link cost ratio comparison

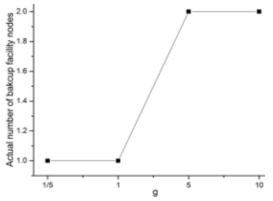


Fig.6: Effect of g (N=8)

Fig. 3 shows the redundancy ratio of the various sharing schemes of the *1*-redundant solution with the increasing size of VI request. The Fig. shows that the redundancy ratio of *noshare* is considerably high, while *bshare* with resource sharing among backup paths significantly reduces the redundancy ratio by requiring fewer redundant resources to tolerate any facility node failure. Furthermore, *share* further decreases the redundancy ratio and cost by permitting the sharing of resources between backup and working paths.

Figs. 4 and 5 show the node cost ratio and link cost ratio of the *K*-redundant solution to the *1*-redundant solution when g=5. From the Figs. we note that the node cost ratio is no less than 1.0, while the link cost ratio is no more than 1.0. This is because the backup facility nodes used by the *K*-redundant solution for failure recovery would be no less than the *1*-redundant solution. While the corresponding link cost ratio decreases due to increase in bandwidth sharing with the increase in the number of used backup facility nodes. Fig.6 shows the effect of different *g* on the actual number of backup facility nodes for a given VI request with 8 VI nodes (for e.g., when g is 5, the number of backup facility nodes used is 2).Fig.6 shows that when the value of *g* is smaller, i.e., node computing cost is more than the link communication cost, the *K*-redundant solution uses only one backup facility node. On the other hand as shown in Fig, 4 and 5 when g increases, the *K*-redundant solution uses more than one backup facility nodes to reduce the total cost of resources.

Our results show that the proposed backup and cross share strategies have a significant impact in conserving backup resources and improving resource utilization. We also find that under majority of the circumstances the *K*-redundant solution is more efficient than the *1*-redundant solution especially when communication costs are higher than the node computing costs.

5. References

[1] N. M. M. K. Chowdhury et al., "Virtual Network Embedding with Coordinated Node and Link Mapping", IEEE INFOCOM, 2009.

[2] B. Mukherjee, "Optical WDM Networks", Springer, 2006.

[3] H. Yu et al., "Survivable Virtual Infrastructure Mapping in a Federated Computing and Networking System under Single Regional failures", accepted by IEEE Globecom 2010.

[4] H. Yu et al., "On the Survivable Virtual Infrastructure Mapping Problem", IEEE ICCCN, Aug. 2010.

[5] M. R. Rahman et al., "Survivable Virtual Network Embedding", Lecture Notes in Computer Science, Apr. 2010.

[6] X. Liu et al., "Robust Application Specific and Agile Private (ASAP) Networks Withstanding Multi-layer Failures," OFC/NFOEC, 2009.