This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE Globecom 2011 proceedings.

A Proposed Topology Design and Admission Control Approach for Improved Network Robustness in Network Virtualization

Mika Mori *, Takuji Tachibana †, Kentaro Hirata *, and Kenji Sugimoto *
*Graduate School of Information Science, Nara Institute of Science and Technology, Ikoma, Nara 630–0192, Japan
Email: {mika-m,kent,kenji}@is.naist.jp
†Graduate School of Engineering, University of Fukui, Fukui, Fukui 910–8507, Japan
Email: takuji-t@u-fukui.ac.jp

***Abstract*—Worldwide, network virtualization has attracted attention as a promising network technology and is expected to be utilized in new-generation networks. In this paper, we propose a topology design and admission control approach for such network virtualization with improved network robustness of both the physical and virtual networks. In our proposed method, first we construct a new network graph that includes the performance of the nodes. We construct a new Laplacian matrix to evaluate the network robustness of the physical networks. Then, based on a user's request, a topology of a virtual network is determined with the KMB algorithm. This topology design is represented as a minimum Steiner tree problem. Moreover, by using admission control, the virtual network is created so as not to degrade the network robustness of the physical networks. If the network robustness of physical networks is degraded, the user's request is rejected and the virtual network is not created. With a simulation, we evaluated the performance of our proposed topology design and admission control approach. Numerical examples show that by using our proposed method, a large number of virtual networks can be provided for users. Moreover, the network robustness of the virtual networks can be improved. Our proposed method is effective for improving the network robustness of both the physical and virtual networks.**

I. Introduction

Currently, network virtualization has attracted attention worldwide as a promising network technology. Network virtualization is expected to be utilized in new-generation networks. In the literature, several kind of network virtualization technologies have been studied. Moreover, a testbed for network virtualization was constructed in the GENI initiative [1], Emulab [2], and PlanetLab [3].

By using network virtualization, multiple virtual networks can be created using part of the network resources over a physical network. When a virtual network is provided for a user, the user can utilize the provided virtual network independently of other users. Multiple virtual networks share several types of network resources such as bandwidth, CPU, and memory. Therefore, for network virtualization, the topology design and resource allocation for a virtual network are important issues. Currently, some topology design and resource allocation algorithms have been studied [4], [5].

On the other hand, in recent network management, one of the principal challenges is to satisfy the service level agreement (SLA) requirements of different users [6]. Improved network robustness increases the service availability for users, and hence network robustness is an important performance metric for satisfying SLA requirements. In terms of network virtualization, users utilize virtual networks that are created with resources in a physical network. Moreover, in some cases, users utilize the physical network with the remaining resources for packet transmission such as Internet. Therefore, for network virtualization, improved network robustness is indispensable for the virtual networks and the physical network. However, as far as the authors know, the improvement of network robustness has not been studied in detail. Especially, the network robustness of the physical networks has not been considered.

In this paper, we propose a topology design and admission control approach for network virtualization to improve the network robustness of the physical networks. In our proposed method, first we construct a new network graph that includes the performance of the nodes. With this graph, we construct a new Laplacian matrix to evaluate the network robustness of the physical networks. Next, based on a user's request, a topology of a virtual network is determined. This topology design is represented as a minimum Steiner tree problem, and we utilize a KMB algorithm to derive the nearest optimal solution. Then, using the proposed admission control approach, the virtual network is created. When the service provider can assure the needed level of network robustness of the physical networks with the remaining resources, the virtual network is created and provided to the user. If the network robustness of the physical networks is degraded, the user's request is rejected and the virtual network is not created. We evaluated the performance of our proposed topology design and admission control approach with a simulation.

The rest of this paper is organized as follows. Section II introduces related work about network virtualization and network robustness. Section III explains the proposed topology design and admission control approach for improved network robustness in network virtualization. Section IV shows some numerical examples and Sect. V describes some conclusions and future work.

978-1-4244-9268-8/11/$26.00

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE Globecom 2011 proceedings.

II. Related Work

A. Network virtualization

Several methods for the topology design of virtual networks have been proposed in the literature. [4] has proposed a network topology approach where the network topology is designed so as to minimize the load on nodes and links. To determine the network topology simply, [7] has proposed a new topology design approach where only a star topology is used. In [5], a topology design using a k-shortest path algorithm has been proposed.

In terms of resource allocation for virtual networks, [5] has proposed a resource allocation method to maximize revenue from the economic perspective. In [8], dynamic resource allocation has been proposed to utilize network resources effectively. The details of other methods have been introduced in [9].

B. Performance Evaluation in terms of Network Robustness

For transport networks, network robustness is an important performance metric for satisfying SLA requirements. To evaluate the network robustness for a transport network, [10] utilized a Laplacian matrix L for the corresponding network graph. Let W be the weight matrix of the graph and D be the diagonal matrix of weighted degrees of graph nodes. From W and D, the Laplacian matrix L is given by

$$L = D - W. \tag{1}$$

With the Laplacian matrix L, to evaluate the network robustness, [10] utilized the second-smallest eigenvalue of L. However, it has been reported that this metric cannot capture some robustness properties, for instance the robustness property for the number of nodes.

In [11], a new metric for network robustness has been proposed. Now, the number of nodes in the transport network is denoted as $n > 0$ and the Laplacian matrix is also denoted as L. A new metric τ called *network criticality* is given by

$$\tau = 2n\mathrm{Tr}(L^{+}). \tag{2}$$

In (2), the Laplacian matrix L is not regular and L^{+} is a pseudo inverse matrix of the Laplacian matrix L. Moreover, the operator Tr denotes the diagonal sum. In contrast to the second-smallest eigenvalue, when τ is small (large), the robustness of the network is high (low). Some simulation results have shown that τ can capture the robustness property in terms of the number of nodes. As a result, network criticality τ is effective to evaluate network robustness for large-scale networks.

III. Proposed Method

In this section, we explain our proposed method for improving network robustness in network virtualization. We focus on a simple network virtualization system where a service provider provides a virtual network to users on a physical network. Each user who would like to utilize a virtual network sends a request for a desired virtual network to the service provider. The service provider performs our proposed method to create and provide a virtual network.

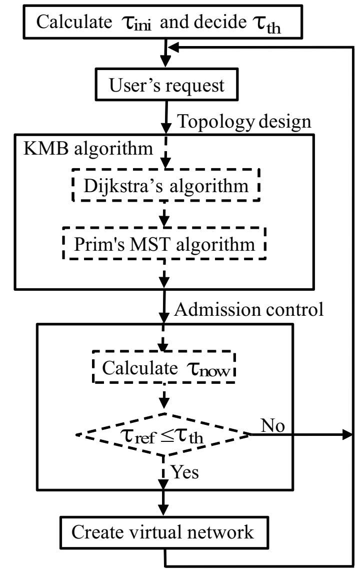
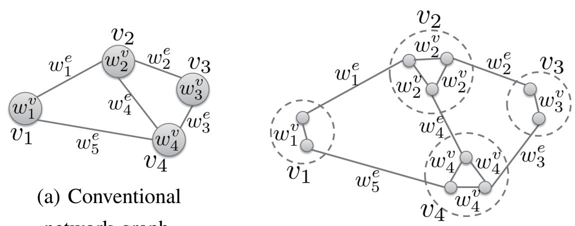


Fig. 1. A flowchart of our proposed algorithm.

A. Overview

Our proposed method consists of two parts, one for topology design and one for admission control (see Fig. 1). As shown in Fig. 1, after the provider receives an user's request, the service provider performs both procedures based on the request.

1) From the node performance and link performance in the physical network, the service provider constructs a new Laplacian matrix to evaluate the robustness of the physical network.
2) With the new Laplacian matrix, the service provider calculates network criticality τ according to (2). Let $\tau_{\text{ini}} = \tau$ denote the initial value of network criticality. Then, the service provider determines a threshold τ_{th} ($\tau_{\text{th}} > \tau_{\text{ini}}$). As shown in [9], τ_{th} should be determined experimentally, however, the discussion of this determination is omitted due to the page limitation.
3) An user sends a request for a virtual network to the service provider. This request includes any nodes that should be included in the desired network and the amount of network resources needed for the virtual network.
4) When the service provider receives the request, the provider performs our proposed method. At first, the provider determines a network topology for the virtual network with a KMB algorithm. Then the service provider determines the amount of network resources used so that it is equal to the requested amount.
5) Next, assuming that the virtual network was provided, the service provider evaluates the robustness of the remaining physical network by calculating the network criticality τ_{new}.



(a) Conventional network graph.

(b) Proposed network graph.

Fig. 2. Physical network graph.

6) According to the admission control parameters, the service provider determines whether the virtual network is actually created or not based on τ_{new} and τ_{th}. If the creation of virtual network is accepted, the service provider provides the virtual network to the user. Otherwise, the request is rejected. Go to step (3).

In the following subsections, we explain each step in detail.

B. Evaluate the network robustness of the physical networks

Figure 2(a) shows an original physical network $G = (V, E, W^V, W^E)$ where the service provider provides virtual networks to users. Let M denote the number of nodes and N denote the number of links in the physical network. Here, $V = \{v_1, \cdots, v_M\}$ and $E = \{e_1, \cdots, e_N\}$ denote a set of nodes and a set of links, respectively. In addition, let $w_i^v > 0$ be a weight that represents the performance of node v_i and let $w_i^e > 0$ be a weight that represents the performance of link e_i. The weights for nodes and links are adjusted preliminarily so that the node performance and the link performance can be represented in the same unit. (The discussion of this determination is out of the scope of this paper.) The two sets of weights are denoted as $W^V = \{w_1^v, \cdots, w_M^v\}$ for nodes and $W^E = \{w_1^e, \cdots, w_N^e\}$ for links.

For the original physical network, the service provider calculates network criticality τ to evaluate the network robustness. In general, τ is derived with a Laplacian matrix from (2). However, the Laplacian matrix includes only the weights for links and never includes the node weights. As a result, the performance of the nodes cannot be considered in the network robustness. Therefore, the conventional network criticality may not be appropriate in network virtualization. In this paper, we propose a new network graph and a new Laplacian matrix so that the performance of the nodes can be considered. Figure 2(b) shows how the new network graph is constructed.

Now, we focus on a node v_i $(i = 1, \cdots, M)$ whose weight is w_i^v. Let the number of links that are connected with v_i $(i = 1, \cdots, M)$ be $h_i \geq 1$. In this case, we divide the node v_i into h_i nodes $\{v_i^1, \cdots, v_i^{h_i}\}$. Then, we connect a new generated node for an original node, v_i^k $(k = 1, \cdots, h_i)$, with a new generated node for another node, v_j^l $(j \neq i)$, by using an original link. Moreover, h_i nodes for an original node v_i are connected with each other by using $h_i(h_i - 1)/2$ new links, i.e., a full-mesh network. The weights of these links are the same as the weight of the original node v_i, i.e., w_i^v. Thus, in a new network graph, the weight of each node is represented as the link weight.

For the new network graph, we construct a new Laplacian matrix L^V according to the conventional procedure. Now, we denote a weight matrix for this graph as W^V. Here, the size of W^V is $\sum_i^M h_i \times \sum_i^M h_i$ although the size of the conventional weight matrix is $N \times N$. In this case, W^V is given by

$$W^V = \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_{12} & \cdots & \cdots & \mathbf{B}_{1M} \\ \mathbf{B}_{21} & \mathbf{A}_2 & \mathbf{B}_{23} & \cdots & \mathbf{B}_{2M} \\ \mathbf{B}_{31} & \mathbf{B}_{32} & \mathbf{A}_3 & \cdots & \mathbf{B}_{3M} \\ \vdots & \vdots & \dots & \ddots & \vdots \\ \mathbf{B}_{M1} & \mathbf{B}_{M2} & \mathbf{B}_{M3} & \cdots & \mathbf{A}_M \end{bmatrix}, \quad (3)$$

$$\mathbf{A}_i = \begin{bmatrix} 0 & w_i^v & \cdots & w_i^v \\ w_i^v & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & w_i^v \\ w_i^v & \cdots & w_i^v & 0 \end{bmatrix}, \quad (4)$$

$$\mathbf{B}_{ij} = \begin{bmatrix} 0 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & \ddots & \cdots & \vdots \\ \vdots & w_{i-j}^e & \cdots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix}. \quad (5)$$

The size of A_i is $h_i \times h_i$, and this matrix corresponds to the weights of the new links generated inside the original node v_i. On the other hand, the size of matrix B_{ij} is $h_i \times h_j$, and this matrix denotes the link weight between the original nodes v_i and v_j. Moreover, the diagonal matrix D^V is derived from W^V, where the (i, j) element in D^V is equal to

$$D^V(i, j) = \begin{cases} \mathbf{A}_i + \sum_k \mathbf{B}_{ik}, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

With W^V and D^V, the new Laplacian matrix L^V is given by

$$L^V = D^V - W^V. \quad (7)$$

Finally, network criticality τ is derived as follows:

$$\tau = 2n\mathrm{Tr}(L^{V+}). \quad (8)$$

C. A user's request for a virtual network

When a user would like to utilize a virtual network, the user sends a request to the service provider. The request includes any nodes that should be included in the provided virtual network and the desired amount of resources to be used in the virtual network. Now let H $(0 < H \leq M)$ be the number of nodes in the user's request and $\{v_1^*, \cdots, v_H^*\}$ be a set of the nodes. Moreover, let $l > 0$ be the desired amount of resources. In this case, we represent the request of this user as $(v_1^*, \cdots, v_H^*, l)$.

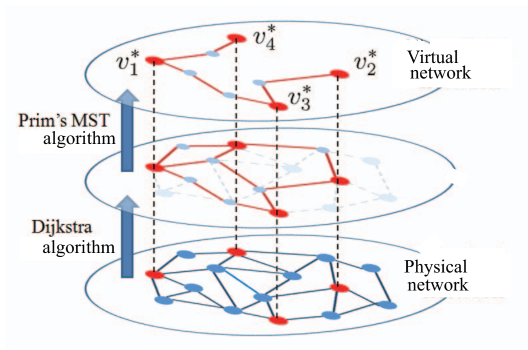


Fig. 3. Topology design with the KMB algorithm.

D. Construct topology design for a virtual network

When the service provider receives the request $(v_1^*, \cdots, v_H^*, l)$, the provider tries to create a virtual network according to the request. First, to consider the network robustness of both the physical and virtual networks, link weights of the current physical network are changed as follows:

$$w_i^e \leftarrow cost_i^e = \tau + w_i^e \frac{\partial \tau}{\partial w_i^e}. \tag{9}$$

$$w_i^v \leftarrow cost_i^v = \tau + w_i^v \frac{\partial \tau}{\partial w_i^v}, \tag{10}$$

Moreover, when R_j denotes a set of $h_j(h_j-1)/2$ links for original node v_j and link i is included in R_j, w_i^v is updated as follows:

$$w_i^v \leftarrow \sum_{i \in R_j} cost_i^v, \tag{11}$$

These costs $cost_i^v$ and $cost_i^e$ are increased when the network criticality τ_{now} increases. Note that τ_{now} increases as a virtual network is provided.

Next, for a network graph with the new weights, the service provider determines a topology of a virtual network that includes all requested nodes $\{v_1^*, \cdots, v_H^*\}$. Moreover, the provider would like to determine the topology so as to prevent the significant increase of the network criticality. This topology design is represented as a minimum Steiner tree problem. Because this problem is known as an NP-complete problem, we utilize the KMB algorithm [12] to obtain an approximate solution.

Figure 3 shows how a network topology is determined with the KMB algorithm. In our KMB algorithm, Dijkstra's algorithm and Prim's minimum spanning tree (MST) algorithm are utilized. First, for a network with costs (9) and (11), the minimum cost path is derived between each two nodes in $\{v_1^*, \cdots, v_H^*\}$ by using Dijkstra's algorithm. A full mesh network is constructed with nodes in $\{v_1^*, \cdots, v_H^*\}$. Then, for this full mesh network, a minimum spanning tree is derived by using Prim's MST algorithm. The obtained minimum spanning tree corresponds to the network topology of the virtual network.

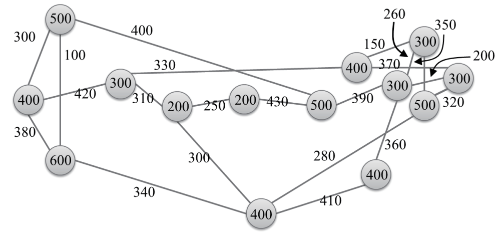


Fig. 4. Network topology 1.

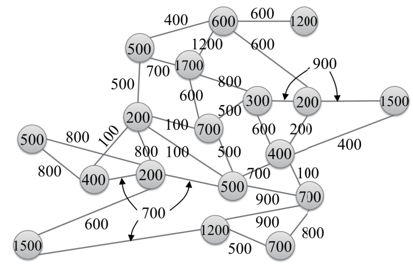


Fig. 5. Network topology 2.

E. Applying admission control

After the topology is determined, the service provider investigates whether the virtual network whose amount of resources is l can be provided with the remaining network resources. Now, let the amount of remaining resource for link (node) i be e_i. If e_k is smaller than l for any node and link k in the virtual network, the user's request is rejected immediately due to the lack of resources. Otherwise, the service provider performs the admission control analysis for network robustness.

If the virtual network is provided to the user, the remaining amount of resources becomes $e_k - l$ for any node and link k. Based on the remaining resources, τ_{new} is calculated from (2). We compare τ_{new} with threshold τ_{th}. If τ_{new} is smaller than or equal to τ_{th}, the user's request is accepted. That is, the virtual network is created and provided to the user. On the other hand, if τ_{new} is larger than τ_{th}, the request is rejected and the virtual network is not created. This is because the network robustness of the remaining physical network cannot satisfy the threshold.

IV. Numerical Examples

We evaluated the performance of our proposed method with a simulation. Figures 4 and 5 show two different physical networks.

For the utilization of virtual networks, users' requests arrive at each physical network according to a Poisson process with rate λ. For each user's request, the service provider performs our proposed method including topology design and admission control. In terms of the user's request, the number H of nodes that are requested by each user is selected from three to $\textit{the number of nodes} - 1$ at random and the nodes are selected at random in each network. In addition, the amount l of resources for a virtual network is selected from 1 to the minimum amount of resources at random.

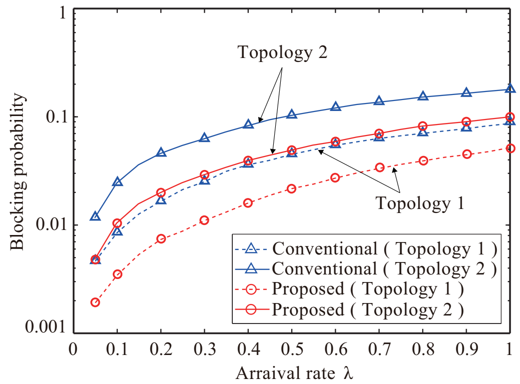


Fig. 6. Blocking probability vs. arrival rate.

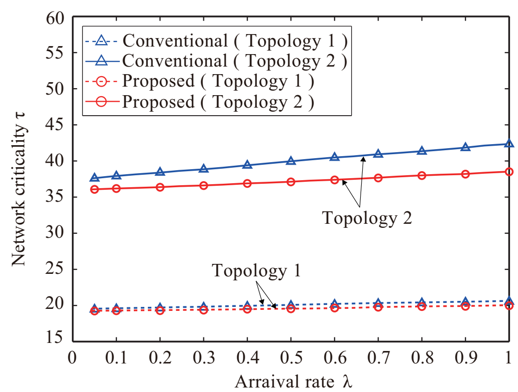


Fig. 7. Network robustness of virtual networks vs. arrival rate.

If the request of an user is accepted, a virtual network is provided to the user. We assume that the utilization time of each virtual network follows an exponential distribution with mean 1.0. When the utilization of a virtual network is completed, its network resources are returned immediately to the remaining physical network. For the performance comparison, we evaluated the performance of a method where a topology is determined based on shortest path without (9) and (11). In the following, we call this method the *conventional method*.

A. Impact of arrival rate λ

First, we investigate the effectiveness of the proposed method in terms of the blocking probability of a user's request. Figure 6 shows the blocking probability against the arrival rate λ in topology 1 and topology 2. In this figure, the threshold τ_{th} is equal to 20 for topology 1 and is equal to 30 for topology 2.

From this figure, we find that regardless of arrival rate λ, the blocking probability for our proposed method is smaller than that for the conventional method. Moreover, regardless of network topology, the blocking probability of our proposed method is also small. Therefore, the service provider can provide a larger number of virtual networks to users.

B. Network robustness of virtual networks

In this subsection, we evaluate the network robustness of virtual networks that are provided to users. Figure 7 shows the network criticality τ for virtual networks against λ in topology 1 and topology 2. In this figure, τ_{th} is the same value as in the previous subsection.

From this result, regardless of λ, τ for our proposed method is always smaller than that for the conventional method. In addition, we find that our proposed method is more effective in network topology 2 as is the case with the blocking probability. These results mean that our proposed method focuses on the network robustness of virtual networks in addition to physical networks.

V. Conclusions and Future Work

In this paper, we proposed a topology design and admission control approach for improved network robustness in network virtualization. In our proposed method, a new network graph and a new Laplacian matrix are constructed and the KMB algorithm is utilized for the topology design. Moreover, users' requests are rejected when the network robustness of the physical networks becomes low. We evaluated the performance of our proposed method with a simulation and compared its performance with the performance of a conventional method. From numerical examples, we found that a large number of virtual networks can be provided to users by using our proposed method without degrading the network robustness of the physical networks. Moreover, the network robustness of the virtual networks can be improved by using the proposed method. Therefore, our proposed method is effective for improving the network robustness of both the physical and virtual networks.

In our future work, we will study the design of physical networks to improve the network robustness. We will define this design as a semidefinite program (SDP).

References

[1] GENI, http://www.geni.net/.

[2] Emulab - Network Emulation Testbed, http://www.emulab.net/.

[3] PlanetLab, http://www.planet-lab.org/.

[4] Y. Zhu and Mostafa Ammar, "Algorithms for Assigning Substrate Network Resources to Virtual Network Components," in *Proc. IEEE INFOCOM 2006*, Apr. 2006, pp. 1-12.

[5] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking Virtual Network Embedding: Substrate Support for Path Splitting and Migration," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 17-29, Apr. 2008.

[6] R.S. Gracià, Y. Labit, J. D. Pascual, and P. Owezarski, "Towards an Efficient Service Level Agreement Assessment," in *Proc. IEEE INFOCOM 2009*, Apr. 2009, pp. 2581-2585.

[7] J. Lu and J. Turner, "Efficient Mapping of Virtual Networks onto a Shared Substrate," *Washington University, Technical Report*, WUCSE-2006-35, 2006.

[8] L. Peterson, A. Bavier, M.E. Fiuczynski, and S. Muir, "A Taxonomy of Rerouting in Circuit-Switched Networks," *IEEE Communications Magazine*, vol. 37, no. 11, pp. 116-122, Nov. 1999.

[9] A. Haider, R. Potter, and A. Nakao, "Challenges in Resource Allocation in Network Virtualization," in *Proc. the 20th ITC Specialist Seminar on Network Virtualization*, Hoi An, Vietnam, May 2009.

[10] M. Fiedler, "Algebraicc Connectivity of Graphs," *Czechoslovak Math. Journal*, vol. 23, no. 98, pp. 298-305, 1973.

[11] A. Tizghadam and A. Leon-Garcia, "Autonomic Traffic Engineering for Network Robustness" *IEEE Journal on Selected Areas in Communication*, vol. 28, no. 1, pp. 1-12, Jan. 2010.

[12] L. Kou, G. Markowsky, and L. Berman, "A Fast Algorithm for Steiner Trees," *Acta Informatica*, vol. 15, no. 2, pp. 141-145, June 1981.