Hierarchical design of an attractor structure for VNT control based on attractor selection

Toshihiko Ohba, Shin'ichi Arakawa, Yuki Koizumi, Masayuki Murata
Graduate School of Information Science and Technology
Osaka University
1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
{t-ohba, arakawa, ykoizumi, murata}@ist.osaka-u.ac.jp

***Abstract*—Our research group has proposed a VNT control method that is adaptive to traffic changes. The method is based on a dynamical system, called attractor selection, which models behavior where living organisms adapt to unknown changes in their surrounding environments and recover their conditions. One of important things of our VNT control method is how to determine attractors, i.e., VNT candidates, because a VNT configured by our VNT method finally converges on one of the VNT candidates. However, since the number of VNT candidates is limited, it is crucial that the limited number of attractors have diversity so that various kinds of VNTs are searched by attractor selection. In this paper, we propose a method to decide the VNT candidates. Our approach prepares the VNT candidates whose bottleneck links (lightpaths) are different to each other. However, this approach has a problem that it takes a heavy computational time for large-scaled networks. We therefore propose a method that divides a network into clusters for which our algorithm can be applied. Evaluation results show that the VNT candidates prepared by our method can suppress maximum link utilization than the ones prepared in a random manner or by an existing heuristic algorithm.**

I. Introduction

Emergence of new Internet services such as video on demand and cloud computing services causes large fluctuations of traffic demands. Since new Internet services are developed one after another, a network should have flexibility to accommodate new and existing services.

One of approaches to achieve the flexibility is to deploy a flexible infrastructure, such as SDN (software-defined network) [1] or WDM (Wavelength Division Multiplexing)-based networks, that can change a virtual network, i.e., connectivity of network equipment and/or bandwidth allocation, in a dynamical manner. Then, network operators conduct traffic engineering over the flexible infrastructure so that the virtual network can adapt to the large fluctuations of traffic demands.

Many researches have investigated methods to accommodate the virtual network (see [2] and references therein). For example, a method proposed in [3] creates an optimal virtual network in terms of embedding costs, i.e., resources needed to respond to virtual network requests, by solving a mixed integer linear program (MILP). However, the method constructs virtual networks using given virtual network requests. That is, the method does not take into account that traffic demands fluctuate largely. Another example can be found in research of carrying IP-packet over WDM networks [4]–[6]. IP-packet over WDM network consists of two layers, WDM network and IP network (Fig. 1). In the WDM network, optical cross-connects (OXCs) are interconnected by optical fibers. A set of optical channels, called lightpaths, are established between IP routers via OXCs. Lightpaths and IP routers form a virtual network topology (VNT) and it accommodates IP traffic on the WDM network. IP packets as electric signals are converted into optical signals and OXCs switch optical signals in the WDM network. Lightpaths using different wavelengths can be multiplexed on a single optical fiber. Refs. [7] investigates a method to reconfigure a VNT for IP-packet over WDM networks, which aims to minimize the average hop distance or the maximum link utilization for time-varying traffic by solving a MILP. This approach assumes that traffic demand matrices are available in a priori. Then, a management node, which collects information for VNT control, calculates and configures a VNT. However, with the growth of the Internet, the amount of information necessary for VNT control and computational time to construct the optimal VNT increase. That is, when traffic demands fluctuate largely, it is difficult to quickly configure a VNT following traffic changes. Therefore, it is important to achieve a method of controlling a VNT that is adaptive to changing traffic demands in a shorter period.

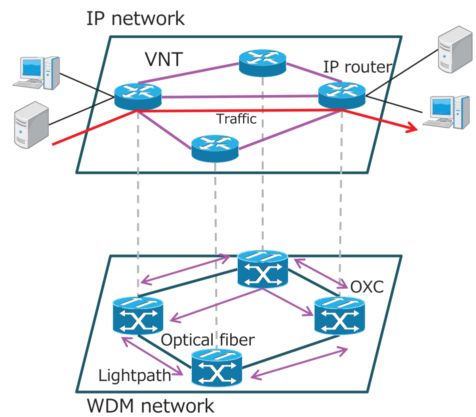


Fig. 1. IP-packet over WDM network

Our research group has proposed a VNT control method that is adaptive to traffic changes [8], [9]. This method is based on a dynamical system, called attractor selection model, which models behavior where living organisms adapt to unknown changes in their surrounding environments and recover their conditions. The behavior of the system driven by attractor selection is described as

$$\frac{d\mathbf{x}}{dt} = \alpha \cdot f(\mathbf{x}) + \eta, \tag{1}$$

where $\mathbf{x}$ represents a state of the system, α is the conditions of the system, $f(\mathbf{x})$ defines an attractor structure. The attractor structure is a part of equilibrium points in the solution space. η represents a noise term. The basic mechanism of VNT control consists of deterministic behavior and stochastic behavior. The behavior is controlled by α, which indicates the conditions of the IP network. When the current conditions of the IP network are poor, α becomes small. Then, the stochastic behavior is dominant in controlling the system. When the current conditions of the IP network are good, α becomes large. Then, the influence of the stochastic behavior becomes a little and the state of the system converges on an attractor defined by $f(\mathbf{x})$. Note that this method only uses the conditions of the IP network for VNT control while most VNT control methods need traffic matrices. We have shown in Refs. [8], [9] that this method can configure a VNT adaptively against the fluctuations in network environments such as traffic changes and node failure.

In Refs. [8], [9], we design the function $f(\mathbf{x})$ in a random manner. However, a main problem of our attractor selection model is how to design $f(\mathbf{x})$ since it determines the attractive state of the dynamical system. If we do not design $f(\mathbf{x})$ properly, the state of the system takes a long time to reach a solution where its conditions are good. For example, when $f(\mathbf{x})$ has only one attractor, i.e., a VNT candidate in case of IP-packet over WDM network, and it is tuned for the current traffic demand, the system cannot reach a solution immediately against unknown traffic changes. A challenge of our attractor selection model is how to design $f(\mathbf{x})$ under fluctuations in network environments. One extreme approach is to prepare all the VNT candidates as the attractors. However, the number of VNT candidates is limited to 10$\sim$15% of the number of lightpath candidates according to the property of Hopfield Network [10]. Therefore, we propose a method to decide VNT candidates, which classifies various VNT candidates into groups and selects an attractor from each group. Note that although we explain the method with IP-packet over WDM network as a target, this approach can be widely applied to virtual network designs.

The rest of this paper is organized as follows. In Section II, we explain our VNT control method based on attractor selection. We then propose a method to decide VNT candidates in Section III. We also propose a method which divides a network into clusters so that VNT candidates are decided for large-scaled networks in Section IV. In Section V, we evaluate performance of the VNT candidates prepared by our method. We conclude this paper in Section VI.

II. VNT Control Based on Attractor Selection

In this section, we explain the VNT control method based on attractor selection proposed in [8], [9].

A. Overview of VNT Control Based on Attractor Selection

A dynamic system driven by attractor selection model adapts to unknown changes in its surrounding environments [11]. In attractor selection model, attractors are a part of equilibrium points in the solution space where the system conditions are preferable. A basic mechanism of attractor selection model consists of deterministic behavior and stochastic behavior. The behavior of the dynamic system driven by attractor selection is described as Eq.(1). A state of the system is represented as $\mathbf{x} = (x_1, ..., x_i, ..., x_n)$ (n is the number of state variables). $f(\mathbf{x})$ represents the deterministic behavior and η represents the stochastic behavior. The behavior is controlled by activity α, which is simple feedback of the system conditions. When the current system conditions are suitable for the environment and α takes a large value, the deterministic behavior drives the system to the attractor. When the current system conditions are poor, i.e., when α takes a small value, the stochastic behavior is dominant in controlling the system. While the stochastic behavior dominates over the deterministic behavior, the state of the system fluctuates randomly due to noise η and the system searches for a new attractor where the system conditions are preferable. In this way, attractor selection adapts to environmental changes by selecting attractors using properly the deterministic behavior and the stochastic behavior according to the activity.

Our VNT control method considers the state of the system $\mathbf{x}$ as a state of all lightpaths that form a VNT and uses the conditions of the IP network as the activity. Our VNT control method then configures VNTs so that the comfort of the IP network gets improved when the conditions of the IP network become uncomfortable due to fluctuations of traffic demands.

B. Activity

Our VNT control method uses maximum link utilization on the IP network as a performance metrics. Although it is necessary to collect load information on all links (lightpaths) in the IP network, this can be retrieved in much shorter time than traffic demand matrices used by existing VNT control methods. We convert the maximum link utilization on the IP network, u_{max}, into the activity α using the following expression Eq.(2). The activity is in a range [0, γ]. The constant number θ is a threshold for VNT control. When the maximum link utilization is more than the threshold θ, the activity rapidly approaches to 0 and our VNT control method searches for a new attractor so that the comfort of the IP network gets improved. δ is also a constant number, which determines an inclination of the function.

$$\alpha = \frac{\gamma}{1 + \exp(\delta \cdot (u_{max} - \theta))} \tag{2}$$

C. Dynamics of VNT Control

Our VNT control method decides whether or not to set up a lightpath l_i according to a state variable $x_i (\in \mathbf{X})$. Dynamics of the state variable x_i is defined as

$$\frac{dx_i}{dt} = \alpha \cdot \left(\varsigma \left(\sum_j W_{ij} x_j \right) - x_i \right) + \eta. \tag{3}$$

α indicates the conditions of the IP network as mentioned in Section II-B. The term $\varsigma(\Sigma_j W_{ij} x_j) - x_i$ represents the deterministic behavior. $\varsigma(z) = \tanh(\frac{\mu}{2} z)$ is a sigmoidal regulation function. The first term is calculated using a regulatory matrix W_{ij}, which defines an attractor structure. The second term η

represents the stochastic behavior and is white Gaussian noise. After x_i is updated on the basis of Eq.(3), we decide whether or not to set up the lightpath l_i. Specifically, we set a threshold to 0 and if x_i is equal to or more than the threshold, we set up the lightpath l_i and otherwise tear down the lightpath l_i.

D. Construction of Attractor Structure

The regulatory matrix W_{ij} that the first term of Eq.(3) has determines an attractor structure. We set the regulatory matrix so that it has a number of VNT candidates as attractors. Assuming that we set m VNT candidates as attractors and one of the candidates is represented as $\mathbf{x}^{(k)} = (x_1^{(k)}, x_2^{(k)}, ..., x_n^{(k)})(1 \leq k \leq m)$, the regulatory matrix that has m attractors is calculated as

$$\mathbf{W} = \mathbf{X}^{+}\mathbf{X}, \tag{4}$$

where $\mathbf{X}$ is a matrix that has $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(m)}$ in each row and $\mathbf{X}^{+}$ is the pseudo inverse matrix of $\mathbf{X}$.

III. Design Method of Attractor Structure

A. Requirements and approach to deciding VNT candidates

Since a VNT configured by our VNT control method based on attractor selection finally converges on one of attractors, either of the attractors should accommodate the current traffic demand. Although the solution space is 2^{n^2} (n is the number of nodes), the number of VNT candidates that can be kept as attractors is limited to $10 \sim 15\%$ of the number of lightpath candidates n^2 [10]. Therefore, a problem to properly decide VNT candidates as attractors comes down to the problem to select $0.1n^2$ VNT candidates with diversity from 2^{n^2} solution space. By preparing VNT candidates with diverse characteristics, various kinds of VNTs are searched by attractor selection, which makes our VNT control method more adaptive against traffic changes.

For the problem, we focus on characteristics of VNT candidates. We take an approach to classifying VNT candidates into groups on the basis of their characteristics and selecting an attractor from each VNT candidates group. Fig. 2 briefly illustrates this approach. In Fig. 2, each circle represents a VNT candidate and circles surrounded with dotted lines represent groups of VNT candidates that have similar characteristics. Then, red colored circles are VNT candidates selected as attractors from VNT candidates groups. We pick up a VNT candidate as an attractor so that each attractor has different characteristics to each other.

B. Algorithm to decide VNT candidates

We develop an algorithm that selects attractors for our VNT control. The goal of our algorithm is to select $0.1n^2$ attractors from 2^{n^2} solution space. An outline of our algorithm is as follows:

1) Enumerate isomorphic VNT candidates of VNT g.
2) Classify the enumerated VNT candidates on the basis of their characteristics.
3) Select an attractor from each group of the VNT candidates.

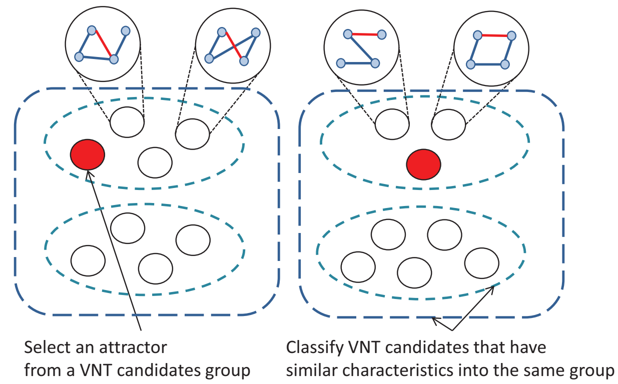


Fig. 2. Approach to deciding VNT candidates

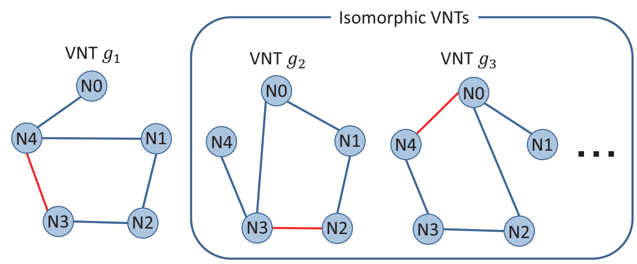


Fig. 3. Examples of isomorphic VNTs

In our algorithm, a VNT g is given in advance. We use a heuristic method to configure the VNT g with a traffic demand matrix T. Note that although we use the traffic demand matrix T to decide VNT candidates, we do not use T for our VNT control method. We explain the detail of the algorithm below.

C. Enumeration of VNT candidates

We enumerate isomorphic VNTs of g. The isomorphic VNTs are generated by exchanging all the nodes of the VNT g. Fig. 3 illustrates examples of isomorphic VNTs. In, Fig. 3, the VNT g_1 consists of five nodes $N0, N1, ..., N4$ and the VNT g_2 and g_3 is one of isomorphic VNTs of g_1. The isomorphic VNT g_2 is generated by shifting $N0$ of the VNT g_1 to $N1$, $N1$ to $N2$, $N2$ to $N3$, $N3$ to $N4$, $N4$ to $N0$. The isomorphic VNT g_3 is generated by shifting $N0$ of the VNT g_1 to $N4$, $N4$ to $N3$, $N3$ to $N2$, $N2$ to $N1$, $N1$ to $N0$. However, VNT candidates that do not meet restrictions on resources in a physical network, such as the number of router ports each node has, are excluded. Thus, the number of the enumerated VNT candidates is $n!$ at most.

In Fig. 3, we assume that a VNT g_1 is configured by a heuristic method with a traffic demand matrix T_1 and traffic loads on the red colored link between the nodes $N3$ and $N4$ is highest. Since the VNT g_1 is configured by a heuristic method with the traffic demand matrix T_1, the VNT g_1 can accommodate T_1. Let us assume that a traffic demand matrix T_2 is generated by exchanging all the elements of T_1 . Since we exchange the nodes of the VNT g_1 in order to generate the isomorphic VNT g_2, traffic loads on the link between the nodes $N2$ and $N3$ is highest and the VNT g_2 can accommodate T_2.

That is, it is expected that any of the isomorphic VNTs can accommodate changing traffic demands, unless each value of a traffic demand matrix is too large. Hereafter, we denote G as the set that includes the VNT g and the enumerated VNT candidates.

D. Classification of the VNT candidates

We classify the VNT candidates that belong to G into groups on the basis of their characteristics. We use Edge Betweenness Centrality, which is the number of shortest paths that go through the link, as characteristics of VNT candidates. Then, we classify the VNT candidates that have different bottleneck links each other into different groups. A bottleneck link has the largest value of Edge Betweenness Centrality among links that form a VNT candidate. When each VNT candidate selected as an attractor has a different bottleneck link, it is expected that any of VNT candidates selected as attractors accommodates various traffic demands. Note that in our VNT control method based on attractor selection, the maximum link utilization indicates the comfort of the IP network. It is likely that a link whose link utilization is high has a high value of Edge Betweenness Centrality. Therefore, we classify the VNT candidates that have the same bottleneck links into the same group. A formal definition of the VNT candidates group is below.

- $p = (s, d)$: an identifier of a node pair, a source node s and a destination node d
- l_p : a link (lightpath) established between the node pair p
- $C(g_i, l_p)$: a value of Edge Betweenness Centrality of the link l_p in a VNT candidate g_i

The VNT candidates group G_p that is expected to have the bottleneck link l_p is described as

$$G_p = \{g_i | g_i \in G, C(g_i, l_p) = \max_q C(g_i, l_q)\}, \qquad (5)$$

using the above notations.

In this way, we divide the set of the VNT candidates enumerated in Section III-C. The number of groups is at most n^2 since the number of lightpath candidates is n^2. However, because the number of VNT candidates that can be kept as attractors is $0.1n^2$, we further merge the VNT candidates groups.

We merge the VNT candidates groups with a condition when traffic loads of their bottleneck links are highly correlated. The condition is satisfied when a correlation of traffic loads is high between two links connected via a node whose degree is low. When traffic flows from a source node s to a destination node d via a node a whose degree is low, a part of traffic that flows on a link $l_{(s,a)}$ also flows on a link $l_{(a,d)}$. That is, if the link $l_{(s,a)}$ is a bottleneck link, it is likely that a traffic load on the link $l_{(a,d)}$ is also high. Therefore, we consider that VNT candidates that belong to a group $G_{(s,a)}$ and $G_{(a,d)}$ have similar characteristics. Based on this heuristic, we merge the VNT candidates groups as

$$G_{(s,d)} \leftarrow G_{(s,a)} \cup G_{(a,d)} \cup G_{(s,d)}, \qquad (6)$$

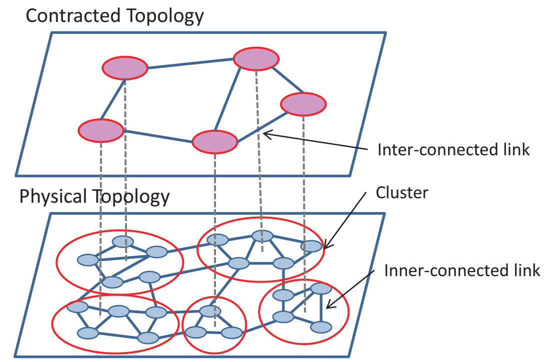


Fig. 4. Contraction of physical topology

where the degree of the node a is low. In Eq.(6), we also regard VNT candidates included in the group $G_{(s,d)}$, which have a bottleneck link $l_{(s,d)}$, have similar characteristics to the group $G_{(s,a)}$ and $G_{(a,d)}$. The reason is that it is likely that a traffic load on the link $l_{(s,d)}$ is high when the link $l_{(s,a)}$ and $l_{(a,d)}$ are bottleneck links. We select the node a, s and d in ascending order of the degree, since a correlation of traffic loads on the link $l_{(s,a)}$ and $l_{(a,d)}$ is high when the degree of the node a is low. However, since each group has different VNT candidates, we select the node a, s and d on the basis of the average value of the degree among all the VNT candidates in the group. We repeatedly merge the VNT candidates groups until the number of VNT candidates groups become about $0.1n^2$.

E. Selection of attractors

We finally select an attractor from each of the VNT candidates group. We select a VNT candidate as an attractor whose maximum of values of Edge Betweenness Centrality is lowest among the VNT candidates group, since the smaller the value of Edge Betweenness Centrality is, the more likely it is that the maximum link utilization is suppressed.

IV. HIERARCHICAL DESIGN METHOD OF ATTRACTOR STRUCTURE

A. Problem of the method to decide VNT candidates

The method to decide VNT candidates shown in Section III have a problem that it takes a heavy computational time for large-scaled networks. This is because the number of enumerated VNT candidates explodes as the number of nodes n increases. With our ordinal PC, we can decide VNT candidates for up to 10-nodes networks with a realistic calculation time. Therefore, we take an approach to contracting a physical network and applying the method in Section III to the contracted network topology. Specifically, we divide a physical network into clusters each of which has several nodes, as shown in Fig. 4. We consider the clusters as nodes and apply the method in Section III to the contracted network topology.

B. Algorithm to decide VNT candidates in a hierarchical manner

An outline of our method to decide VNT candidates in a hierarchical manner is as follows.

Step.1 Divide a physical network into clusters and decide a clustering structure that has several layers.
Step.2 Construct VNT candidates in clusters at the bottom layer
Step.3 Construct VNT candidates at upper layers following the method in Section III.
Step.4 Connect lightpaths between clusters to nodes in the clusters.

We explain the detail of the algorithm below.

Step.1 Cluster division of a physical network

We divide a physical network into c clusters. When the number of vertexes in a cluster is more than c, we divide the cluster into clusters recursively until the number of vertexes in a cluster is equal to or less than c, i.e., the division leads to making a clustering structure that has several layers. An upper layer consists of clusters that have nodes at a lower layer. For example, in a three-layer network, the top layer consists of clusters that have nodes at the middle layer. Then, nodes at the middle layer are clusters that have nodes at the bottom layer (Fig. 6(b)).

Step.2 Construction of VNT candidates inside clusters at the bottom layer

We construct VNT candidates inside clusters at the bottom layer. We construct a VNT candidate that has full-mesh topology in a cluster at the bottom layer. This is because a cluster can adapt to traffic changes in a cluster and can keep connectivity when network failure occurs.

Step.3 Construction of VNT candidates at upper layers

We enumerate VNT candidates with diversity at upper layers following the method in Section III. However, we do not need to merge VNT candidates groups. We prepare $c(c-1)/2$ VNT candidates groups at most, since we set up lightpaths bidirectionally. This is because we do not need to set up single directional lightpaths in order to decide a limited number of VNT candidates with diverse characteristics. We construct VNT candidates at upper layers as follows:

Step.3-1 Calculate a VNT candidates using a heuristic method and enumerate isomorphic VNT candidates of it.
Step.3-2 Classify the enumerated VNT candidates into $c(c-1)/2$ groups at most on the basis of Edge Betweenness Centrality.
Step.3-3 Select an attractor from each group of the VNT candidates.

Step.4 Connection between clusters

We establish lightpaths between clusters in the VNT candidates at upper layers constructed in Step.3. That is, we connect nodes inside the clusters. We establish lightpaths between clusters from the kth layer to the $k+1$ layer, i.e., from an upper layer to a lower layer. We establish lightpaths between nodes in clusters as follows:

- C_x^k : the xth cluster at the kth layer
- V_x^k : nodes that belong to C_x^k
- $l_{i,j}^k$: a lightpath bidirectionally established between C_i^k and C_j^k
- k_u : the number of lightpaths connected to a node u (the degree of the node u)

The probability we establish the lightpath $l_{i,j}^k$ between $u(\in V_i^k)$ and $v(\in V_j^k)$ is described as

$$P_{u,v} = (k_u k_v)^{-1}. \tag{7}$$

Eq.(7) intends to balance traffic loads. Since it is likely that a larger amount of traffic flows via a node as the degree of the node increases, we connect nodes whose degree is low.

V. PERFORMANCE EVALUATION

A. Adaptability of VNT candidates by the method in Section III

In this section, we evaluate adaptability of VNT candidates by the method in Section III. We use a 10-nodes network that has a ring topology as a physical network. Each node has five router ports, i.e., five transmitters and five receivers. We configure a VNT candidate using I-MLTDA [12] as a heuristic method with a traffic demand matrix whose elements follow a log-normal distribution. We decide 10 VNT candidates by following the algorithm in Section III. For the evaluation, we use 1,000 patterns of traffic demands between each node pair according with a log-normal distribution. For a comparison purpose, we introduce a heuristic method named HLDA [12], which configures a VNT on the basis of a given traffic demand matrix. We also compare a method that constructs a VNT candidate by establishing lightpaths between randomly chosen node pairs. The number of VNT candidates is the same for all methods.

Fig. 5 shows the distribution of the maximum link utilization for each traffic pattern. The horizontal axis shows the maximum link utilization and the vertical axis shows the CCDF of the maximum link utilization. In Fig. 5, we can see that the maximum link utilization of the VNT candidates by our method is lower than that of the VNT candidates prepared by other methods. That is, the method in Section III can decide VNT candidates that suppress traffic loads against various traffic demands, compared with the other methods.

B. Adaptability of VNT candidates by the method in Section IV

In this section, we evaluate adaptability of VNT candidates by the method in Section IV. Here, we use a 25-nodes network and each node has 10 router ports. We consider the two-layer network so that every cluster has five nodes, as shown in Fig. 6(a). Topology at the top layer is regarded to be composed of five nodes with three router ports and we decide seven VNT candidates by following Step.3 of our algorithm. The reason why the number of VNT candidates is seven is that the enumerated VNT candidates are classified into seven groups, i.e., only seven lightpaths become bottleneck links among the enumerated VNT candidates. Thus, we do not merge the VNT candidates groups. VNT candidates in clusters at the bottom layer have full-mesh topology. When a lightpath is established

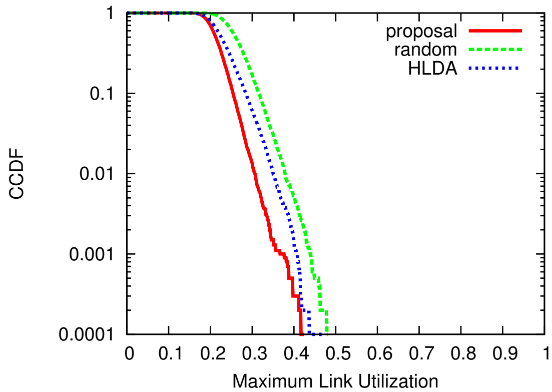
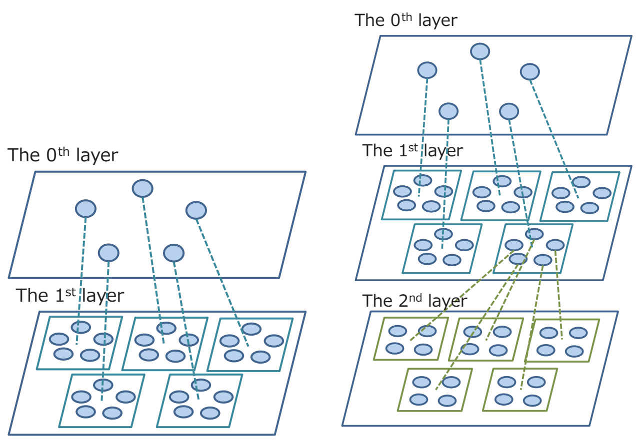


Fig. 5. CCDF of the maximum link utilization (10-nodes)



(a) the 25-nodes network (b) the 100-nodes network

Fig. 6. The clustering structure of the networks

between two nodes at the top layer, five bidirectional lightpaths are established between the corresponding clusters at the bottom layer. In this way, we connect seven VNT candidates at the top layer and one VNT candidate at the bottom layer. Finally, we decide seven VNT candidates.

Fig. 7 shows the distribution of the maximum link utilization for each traffic pattern. Traffic demands and methods for comparison are similar to those described in Section V-A. In the figure, we can see that our method can decide VNT candidates that suppress the maximum link utilization than the other methods. In other words, the method in Section IV can decide VNT candidates that suppress traffic loads against various traffic demands, compared with the other methods. That is, the method in Section IV can decide better VNT candidates than the other methods for larger networks where we cannot apply the method in Section III.

C. Adaptability of the VNT control method

In this section, we evaluate the VNT control method based on attractor selection using VNT candidates by our method as attractors. We set the target maximum link utilization θ

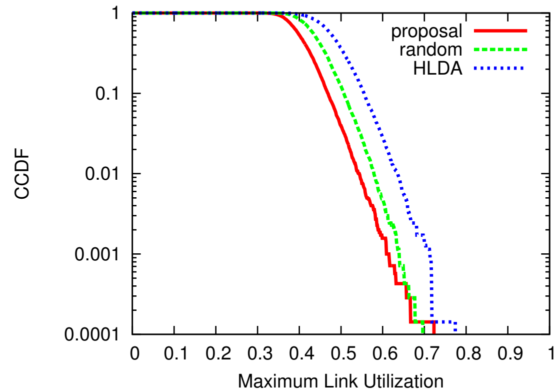


Fig. 7. CCDF of the maximum link utilization (25-nodes)

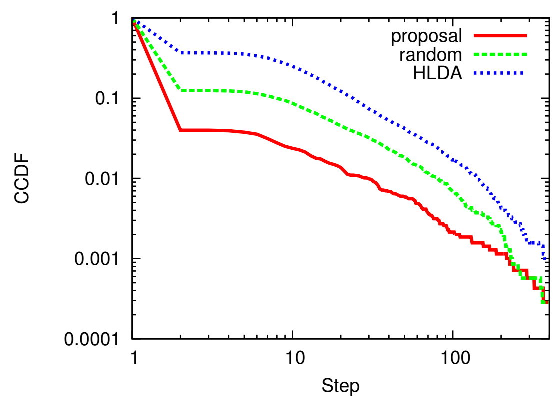


Fig. 8. CCDF of the steps of the VNT control (25-nodes)

in Eq.(2) to 0.5 and the VNT control succeeds when the maximum link utilization is suppressed less than 0.5 during 10 successive steps of the VNT control. We evaluate steps of the VNT control needed until the VNT control succeeds. At each step, the method collects load information on all lightpaths, calculates the activity α, and reconfigures a VNT. We set γ to 1 and δ to 50 in Eq.(2).

Fig. 8 shows the distribution of the steps of the VNT control in the 25-nodes network. Traffic demands and methods for comparison are similar to those described in Section V-A. The horizontal axis shows the steps of the VNT control and the vertical axis shows the CCDF of the steps. We can find that the VNT control that uses the VNT candidates by our method as the attractors needs less steps of the control. Since the VNT candidates by our method can suppress the maximum link utilization, as shown in Section V-B, the VNT control method finds an attractor, i.e., a VNT that can accommodate traffic, in a shorter time.

D. Scalability of the method to decide VNT candidates

In this section, we evaluate adaptability of VNT candidates by our method in a larger network. Here, we use a 100-nodes network and each node has 32 router ports. We consider the three-layer network as shown in Fig. 6(b). Topology at the top layer and topology in clusters at the middle layer are considered to be composed of five nodes with three router

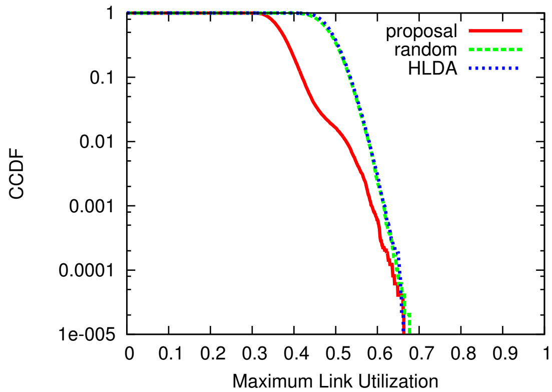


Fig. 9. CCDF of the maximum link utilization (100-nodes)

ports, and we construct seven VNT candidates for each layer. At the middle layer, since there are seven VNT candidates in each cluster and the number of clusters is five, the number of VNT candidates at the middle layer is 7^5, counting all combinations. Therefore, we use the same VNT candidate in all the clusters at the middle layer. That is, we decide seven VNT candidates at the middle layer. VNT candidates in clusters at the bottom layer have full-mesh topology. When a lightpath is established between two nodes at an upper layer, five bidirectional lightpaths are established between the corresponding clusters at a lower layer. In this way, we connect seven VNT candidates at the top layer, seven VNT candidates at the middle layer and one VNT candidate at the bottom layer. Finally, we decide 49 VNT candidates.

Fig. 9 shows the distribution of the maximum link utilization for each traffic pattern. Traffic demands and methods for comparison are similar to those described in Section V-A. However, since the total traffic demand is excessive in the 100-nodes network, we use one third of traffic demands used in the 10-nodes and the 25-nodes network. In Fig. 9, we can see that the maximum link utilization of the VNT candidates by our method is less than that of the VNT candidates by the other methods. We find that our method can decide better VNT candidates for three-layer networks, and we believe that our method can decide better VNT candidates for networks that have more than four layers. When we evaluate the VNT control method based on attractor selection in the 100-nodes network using the VNT candidates by our method as the attractors, it is expected that we obtain similar results in Section V-C. This is because that both the VNT candidates by our method in the 25-nodes and the 100-nodes network can suppress the maximum link utilization than the ones by the other methods, as shown in Fig. 7 and 9.

VI. Conclusion

We proposed a method to decide VNT candidates for the VNT control method based on attractor selection. We showed that the VNT candidates by our method can accommodate various traffic demands, and the VNT control that uses them as the attractors can find a solution in much shorter time.

In the future work, we decide VNT candidates by dividing a large-scaled network into clusters on the basis of the physical network topology, such as modularity of the geographical region.

Acknowledgment

A part of this work was supported by the National Institute of Information and Communications Technology (NICT).

References

[1] M.-K. Shin, K.-H. Nam, and H.-J. Kim, “Software-defined networking (SDN): A reference architecture and open APIs,” in *Proceedings of ICT Convergence (ICTC), 2012 International Conference on*. IEEE, Oct. 2012.

[2] A. Fischer, J. F. Botero, M. Till Beck, H. De Meer, and X. Hesselbach, “Virtual network embedding: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 1888–1906, Feb. 2013.

[3] I. Houidi, W. Louati, W. Ben Ameur, and D. Zeghlache, “Virtual network provisioning across multiple substrate networks,” *Computer Networks*, vol. 55, no. 4, pp. 1011–1023, Mar. 2011.

[4] S. Arakawa, M. Murata, and H. Miyahara, “Functional partitioning for multi-layer survivability in IP over WDM networks,” *IEICE Transactions on Communications*, vol. 83, no. 10, pp. 2224–2233, Oct. 2000.

[5] N. Ghani, S. Dixit, and T. Wang, “On IP-over-WDM integration,” *IEEE Communications Magazine*, vol. 38, no. 3, pp. 72–84, May 2000.

[6] Y. Ohsita, T. Miyamura, S. Arakawa, S. Ata, E. Oki, K. Shiomoto, and M. Murata, “Gradually reconfiguring virtual network topologies based on estimated traffic matrices,” *IEEE/ACM Transactions on Networking*, vol. 18, pp. 177–189, Feb. 2010.

[7] G. Agrawal and D. Medhi, “Lightpath topology configuration for wavelength-routed IP/MPLS networks for time-dependent traffic,” in *Proceedings of GLOBECOM*, Nov. 2006.

[8] Y. Koizumi, T. Miyamura, S. Arakawa, E. Oki, K. Shiomoto, and M. Murata, “Adaptive virtual network topology control based on attractor selection,” *IEEE/OSA Journal of Lightwave Technology*, vol. 28, no. 11, pp. 1720–1731, Jun. 2010.

[9] Y. Minami, S. Arakawa, Y. Koizumi, T. Miyamura, K. Shiomoto, and M. Murata, “Adaptive virtual network topology control in WDM-based optical networks,” in *Proceedings of INTERNET*, Sep. 2010.

[10] Y. Baram, “Orthogonal patterns in binary neural networks,” *NASA Technical Memorandum 100060*, Mar. 1988.

[11] C. Furusawa and K. Kaneko, “A generic mechanism for adaptive growth rate regulation,” *PLoS Computational Biology*, vol. 4, no. 1, p. e3, Jan. 2008.

[12] D. Banerjee and B. Mukherjee, “Wavelength-routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study,” *IEEE/ACM Transactions on Networking (TON)*, vol. 8, no. 5, pp. 598–607, Oct. 2000.