Master's Thesis

Title

Fast and Accurate Virtual Network Reconfiguration using Two-Pathway Bayesian Attractor Model: Design and Evaluation

Supervisor

Associate Professor Shin'ichi Arakawa

Author

Koki Nishida

January 29th, 2021

Department of Information Networking

Graduate School of Information Science and Technology

Osaka University

Master's Thesis

Fast and Accurate Virtual Network Reconfiguration using Two-Pathway Bayesian Attractor Model: Design and Evaluation

Koki Nishida

Abstract

Our research group studies the virtual network (VN) reconfiguration method that applies the Bayesian Attractor Model (BAM), which models the cognitive and decision-making behavior of the human brain, as a method that does not use the traffic matrix. The method holds some traffic situations and its corresponding virtual network topologies (VNTs) that work well under those traffic situations in advance. Then, the method identifies the current traffic situation using Bayesian inference and selects the corresponding VNT. However, it is known that when the method holds many traffic situations in advance to emphasize the accuracy of identification, it takes more time to identify the traffic situation. On the contrary, when the method emphasizes the speed of the traffic situation identification and holds fewer traffic situations, it may fail to configure the VNT that works well against the changes of traffic situations.

In this thesis, to achieve the VN reconfiguration with both accuracy and speed, we focus on the cognitive mechanism of the human brain to perform cognition that satisfies both speed and accuracy. As the knowledge of the brain science, it is known that the human brain contains two cognitive pathways; a cognitive pathway that emphasizes speed and a cognitive pathway that emphasizes accuracy. By incorporating such the two-pathway mechanisms into the BAM-based VN reconfiguration method, it is expected to achieve both speed and accuracy. To achieve this, we develop a VN reconfiguration method having two BAM mechanisms; a fast-pathway-BAM (FP-BAM) for the fast decision-making and a slow-pathway-BAM (SP-BAM) for the accurate decision-making. However, since we cannot obtain the internal parameters of the actual human brain, we need to design the parameters of each BAM suitable for VN control. More importantly, we need to have the design method of VNTs suitable for the role and parameter settings of each BAM. Our thesis presents the design principles of parameters of each BAM and the design method of VNTs based on the design principles. In our design method, for SP-BAM, we prepare a lot of VNTs that perform the best in a certain traffic situation using a traditional heuristic algorithm. Then, for FP-BAM, we prepare a few of VNTs that perform to some extent in more patterns of traffic situations, based on frequency of virtual links used to accommodate more traffic situations. Our simulation results show that FP-BAM quickly identifies traffic situations with 70% fewer steps than SP-BAM. The results also show that VNTs of SP-BAM has higher probability to form a virtual link between the node pairs taking high traffic volume, and at maximum, 45% higher than that of VNTs of FP-BAM. By operating FP-BAM and SP-BAM in parallel, our VN reconfiguration quickly relaxes the traffic congestion against traffic changes and configures more efficient VNTs after the accumulated traffic observations.

Keywords

Bayesian Attractor Model (BAM)

Virtual Network (VN)

VN Reconfiguration

Virtual Network Topology (VNT)

Fast Pathway

Slow Pathway

Contents

List of Figures

List of Tables

1 Introduction

In recent years, the demand for traffic is diversifying and the amount of traffic is increasing because various services such as video distribution services and cloud services have been remarkably developed on the Internet. Research works are being conducted on network virtualization problems that construct a virtual network (VN) on top of physical infrastructure and that flexibly reconfigure the virtual network topology (VNT) against fluctuations in the traffic demand [1–6].

There are several existing methods for reconfiguring VNT in real time against fluctuations in the traffic demand, and most of them acquire the traffic matrix and optimize it [7–10]. However, there remains challenges in applying the approach to the ever-changing traffic demand and reconfiguring a fast and efficient VNT. First, because of the difficulty to acquire the traffic matrix in real time, it is difficult to reconfigure the VNT based on the traffic matrix within a short period. Also, the amount of traffic between node pairs fluctuates significantly compared to the fluctuation of the traffic amount in the entire network, so the method may overly reconfigure the VNT when traffic demand accidentally increases or decreases. It is possible to use the time average of the amount of traffic between node pairs to input the existing method, but in this case, the method cannot quickly reconfigure VNT for the ever-changing traffic demand.

Our research group studies the VN reconfiguration method that applies the Bayesian Attractor Model (BAM) [11], which models the cognitive and decision-making behavior of the human brain, as a method that does not use the traffic matrix [12]. In this method, BAM holds some traffic situations as attractors. BAM identifies the current traffic situation by observing the amount of traffic on the real network, accumulating evidence, and calculating the attractor corresponding to the current traffic situation. In Ref. [12], the method quickly reconfigures the VNT by limiting the observing traffic information to the amounts of incoming and outgoing traffic at each node. BAM updates the probability density of each attractor by receiving the observed value of the amounts of incoming and outgoing traffic at each node as an input for each unit time and estimates the traffic situation. In this method, BAM holds some traffic situations and VNTs, each of which corresponds to the traffic situation and works well under those traffic situations, in advance.

When BAM identifies the current traffic situation to one of several traffic situations holding in advance using Bayesian inference, the method configures the corresponding VNT that works well for the traffic situation identified by BAM.

In the VN reconfiguration method using BAM, BAM reconfigures the VNT that works well corresponding to more traffic situations as the number of templates of traffic situations holding in advance, that is, the number of attractors increases. However, the time required to identify the traffic situation increases, as the number of attractors increases. When BAM holds many traffic situations in advance to emphasize the accuracy of identification, it can configure the VNT that works well against the changes of traffic situations, but it takes more time to identify the traffic situation and it impairs the speed of identification. When BAM emphasizes the speed of the traffic situation identification and assumes fewer traffic situations, it identifies the traffic situation quickly against the fluctuation of the traffic situation, but it may not configure the VNT that works well against the changes of traffic situations. Therefore, the challenge of VN reconfiguration method using BAM is to achieve both accuracy and speed.

In this thesis, we focus on the cognitive mechanism of the human brain to perform cognition that satisfies both speed and accuracy using BAM. As the knowledge of brain science, it is known that the human brain contains two cognitive pathways, a cognitive pathway that emphasizes speed, which we call Fast Pathway (FP), and a cognitive pathway that emphasizes accuracy, which we call Slow Pathway (SP), and recognizes by switching the cognitive pathway. By incorporating such the two-pathway mechanisms into our VN reconfiguration method, it is expected to achieve both accuracy and speed. To achieve this, we develop a VN reconfiguration method having two BAM mechanisms; a fast-pathway-BAM (FP-BAM) for the fast decision-making and a slow-pathway-BAM (SP-BAM) for the accurate decision-making. However, since we cannot obtain the internal parameters of the actual human brain, we need to design the parameters of each BAM suitable for VN control. More importantly, we need to have the design method of VNTs suitable for the role and parameter settings of each BAM. Our thesis presents the design principles of parameters of each BAM and the design method of VNTs based on the design principles. In our design method, for SP-BAM, we prepare a lot of VNTs that perform the best in a certain traffic situation using a traditional heuristic algorithm. Then, for FP-BAM, we prepare a few of VNTs that perform to some extent in more patterns of traffic situations, based on frequency of virtual links used to accommodate more traffic situations. By operating FP-BAM and SP-BAM in parallel, our VN reconfiguration quickly relaxes the traffic congestion against traffic changes and configures more efficient VNTs after the accumulated traffic observations.

This thesis is organized as follows. Section 2 describes virtual network reconfiguration method using BAM. Section 3 describes the principles and design of VN reconfiguration method using BAM which emphasizes accuracy and BAM which emphasizes speed, in parallel. Section 4 describes evaluations of our VN reconfiguration method by simulations. Finally, Section 5 describes the conclusions and future works.

2 Virtual Network Reconfiguration using Bayesian Attractor Model

2.1 Bayesian Attractor Model (BAM)

In this thesis, we use Bayesian Attractor Model (BAM) for virtual network (VN) control.

2.1.1 Outline of Bayesian Attractor Model

BAM is a model of the mechanism by which the human brain accumulates evidences extracted from sensory information from the outside world and makes cognitive and decision-making based on the evidences. BAM has a D-dimensional state variable $\mathbf{z}$, which changes in value against the attractor dynamics [13] that converges to the state ϕ_i $(i = 1, \cdots, D)$ as evidence accumulates. In addition, BAM has candidates μ_i $(i = 1, \cdots, D)$, which are average patterns of observed values, and each candidate μ_i corresponds to the state ϕ_i. At time t, BAM makes Bayesian estimates the posterior distribution $p(\mathbf{z}_t|\mathbf{X}_{1:t})$ defined on the state space using the Unscented Kalman Filter (UKF) [14], where $\mathbf{z}_t$ is the D-dimensional state variable at time t and $\mathbf{X}_{1:t} = \mathbf{x}_1, \cdots, \mathbf{x}_t$ is an observation value obtained at each time up to time t. $p(\mathbf{z}_t = \phi_i|\mathbf{X}_{1:t})$ represents the posterior belief for the state ϕ_i. Ref. [11] introduces the posterior belief $p(\mathbf{z}_t = \phi_i|\mathbf{X}_{1:t})$ as the confidence measure for making a decision for the candidate μ_i. After estimation, if $p(\mathbf{z}_t = \phi_i|\mathbf{X}_{1:t}) \geq \lambda$ holds, BAM adopts candidate μ_i as the correct candidate, and if none of $i = 1, \cdots, D$ holds, BAM continues observation. Thus, the BAM accumulates observation values and makes a decision when the confidence for the decision is large enough (Fig. 1).

2.1.2 Bayesian inference in the BAM

BAM has a generative model for Bayesian inference by the decision maker, i.e., the brain. The generative model calculates the likelihood of observations under all possible candidates that the decision maker considers. More precisely, the generative model predicts a probability distribution over observation values based on the current state variable and its attractor dynamics. The generative model defines a change in the state variable from

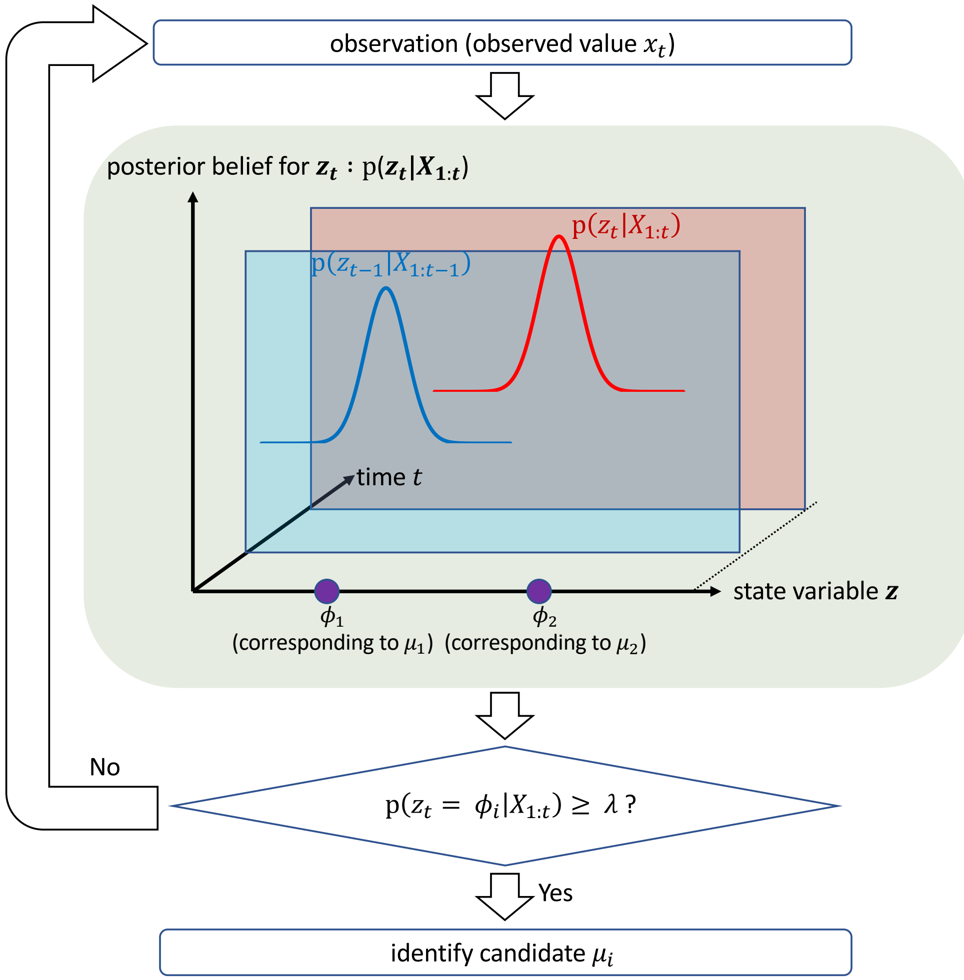

Figure 1: Bayesian Attractor Model (BAM)

one time step to the next as,

$$\mathbf{z}_t - \mathbf{z}_{t-\Delta t} = \Delta t \cdot f(\mathbf{z}_{t-\Delta t}) + \sqrt{\Delta t} \cdot \mathbf{w}_t, \tag{1}$$

where $f(\mathbf{z})$ is a function defining an attractor dynamics [13] for the vector of state variables $\mathbf{z}$. The noise variable $\mathbf{w}_t$ follows the normal distribution $N(\mathbf{0}, \mathbf{Q})$, where $\mathbf{Q} = (q^2/\Delta t \cdot \mathbf{I})$ is the isotropic covariance of the noise and q is called "dynamical uncertainty".

The generative model predicts a probability distribution over observation values, given the state variable $\mathbf{z}$. The equation for the prediction is,

$$\begin{aligned}\mathbf{x} &= \mathbf{M} \cdot \sigma(\mathbf{z}) + \mathbf{v} && (2)\\ &= [\mu_1, \cdots, \mu_D] \cdot \sigma(\mathbf{z}) + \mathbf{v} && (3)\end{aligned}$$

$$= \quad \sigma(z_1) \cdot \mu_1 + \sigma(z_2) \cdot \mu_2 + \cdots + \sigma(z_D) \cdot \mu_D + \mathbf{v}, \tag{4}$$

where $M = [\mu_1, \cdots, \mu_D]$ contains the averages of observation values that correspond to candidates and $\sigma(\mathbf{z})$ is the sigmoid function that maps all valuables $z_j \in \mathbf{z}$ to values between 0 and 1. Due to the winner-take-all mechanism of attractor dynamics [13], the state ϕ_i is mapped to a vector $\sigma(\phi_i)$ where one element is approximately 1 and the other elements are approximately 0. Thus, the linear combination $\mathbf{M} \cdot \sigma(\mathbf{z})$ associates each state ϕ_i with the candidate μ_i. The noise variable $\mathbf{v}$ follows the normal distribution $N(\mathbf{0}, \mathbf{R})$, where $\mathbf{R} = r^2 \cdot \mathbf{I}$ is the expected isotropic covariance of the noise on the observations and r is called "sensory uncertainty".

At time t, BAM infers the posterior distribution $p(\mathbf{z}_t|\mathbf{X}_{1:t})$ of the state variable $\mathbf{z}_t$ using the generative model and the unscented Kalman filter (UKF) [14]. The UKF is a statistical sampling method that approximates the posterior distribution $p(\mathbf{z}_t|\mathbf{X}_{1:t})$ with a normal distribution. We describe the flow of the Bayesian inference in BAM. First, the generative model predicts the posterior distribution of the state variable at time t and approximates it with a normal distribution $N(\hat{\mathbf{z}}_t, \hat{\mathbf{P}}_t)$, where $\hat{\mathbf{P}}_t$ represents the isotropic covariance of the predicted state variable $\hat{\mathbf{z}}_t$. Second, the generative model predicts the possibility distribution of the corresponding observation values at time t and approximates it with a normal distribution $N(\hat{\mathbf{x}}_t, \hat{\Sigma}_t)$, where $\hat{\Sigma}_t$ represents the isotropic covariance of the predicted observation values $\hat{\mathbf{x}}_t$. Finally, BAM calculates the observation residual between the predicted observation values $\hat{\mathbf{x}}_t$ and the actual observation values $\mathbf{x}_t$, $\epsilon_t = \mathbf{x}_t - \hat{\mathbf{x}}_t$. And BAM updates the estimation of the state variable $\bar{\mathbf{z}}_t$ and its posterior isotropic covariance $\bar{\mathbf{P}}_t$ via a Kalman gain $\mathbf{K}_t$ as follows;

$$\bar{\mathbf{z}}_t = \hat{\mathbf{z}}_t + \mathbf{K}_t \cdot \epsilon_t, \tag{5}$$

$$\bar{\mathbf{P}}_t = \hat{\mathbf{P}}_t + \mathbf{K}_t \cdot \hat{\mathbf{C}}_t^T. \tag{6}$$

The Kalman gain represents the relative importance of the observation residual and is given by,

$$\mathbf{K}_t = \hat{\mathbf{C}}_t \cdot \hat{\Sigma}_t^{-1}, \tag{7}$$

where $\hat{\mathbf{C}}_t$ is the covariance matrix between the predicted state variable $\hat{\mathbf{z}}_t$ and the predicted observation values $\hat{\mathbf{x}}_t$. In this way, BAM approximates the posterior distribution $p(\mathbf{z}_t|\mathbf{X}_{1:t})$ with a normal distribution $N(\bar{\mathbf{z}}_t, \bar{\mathbf{P}}_t)$.

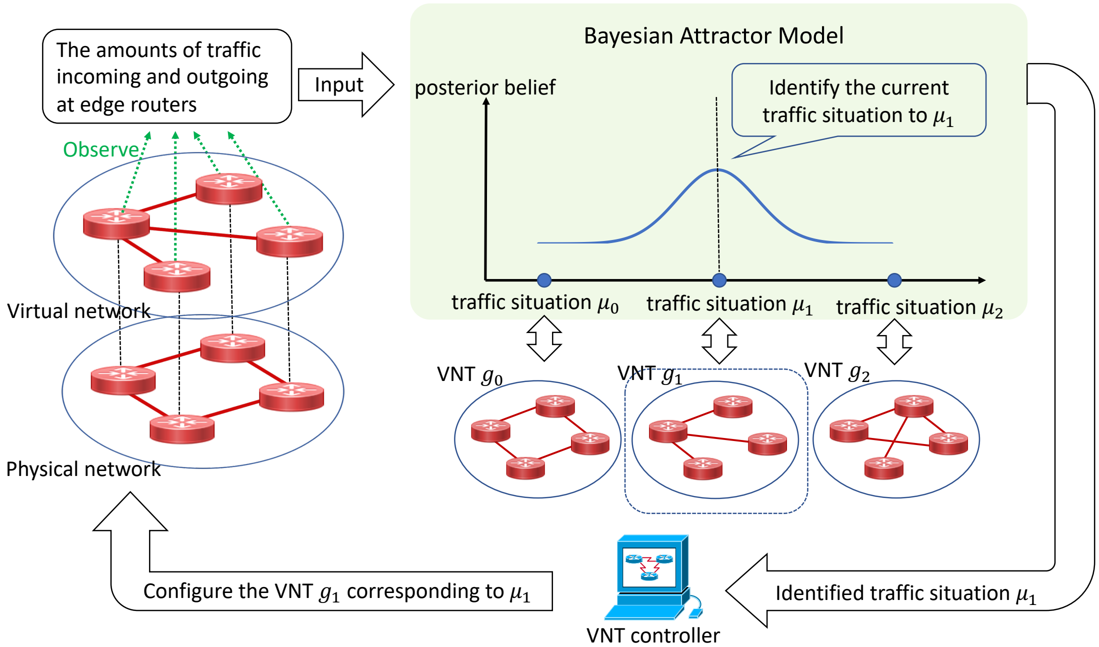

Figure 2: VN Reconfiguration using BAM

2.2 VN Reconfiguration using BAM

We consider applying BAM to estimate the traffic situation of the network and configuring a VN suitable for the estimated situation. Figure 2 shows a schematic diagram of VN Reconfiguration using BAM. BAM estimates a traffic situation by observing the amounts of incoming and outgoing traffic at edge routers (hereinafter referred to as the traffic amount). We make BAM hold specific traffic situations μ_i ($i = 1, \cdots, D$) and VN candidates g_i ($i = 1, \cdots, D$) that is suitable for specific traffic situations μ_i in advance. Every time BAM observes the traffic amount, it updates the confidence that the current traffic situation matches specific traffic situations μ_i. BAM identifies the current traffic situation when the confidence factor of a specific traffic situation μ_i exceeds the threshold, and we configure the VN candidate g_i suitable for the identified traffic situation μ_i.

2.3 Parameters of BAM

We can make setting for several variable parameters for BAM. The number of attractors BAM holds is one of the parameters. The more attractors BAM holds, the more traffic situations BAM can envision. However the more attractors BAM holds, the longer BAM will take to identify the current state from the attractors. Conversely, reducing the number of attractors BAM holds will reduce the time BAM takes to identify the current state from the attractors, but will reduce the traffic situations BAM can envision. There is a trade-off between the speed and accuracy of BAM estimation.

Other parameters are dynamics uncertainty and sensory uncertainty. Dynamics uncertainty represents the expected state noise at the attractor level, which can be interpreted as the propensity to switch between estimations. The higher the dynamics uncertainty, the more likely BAM switches the identified traffic situation between traffic situation candidates. Sensory uncertainty represents the expected isotropic covariance of the noise on the observation value. The higher the sensory uncertainty, the more noise BAM expects to be included in the observation value. In this thesis, we use the covariance of the last five observation values as sensory uncertainty.

3 Virtual Network Reconfiguration Method using Two-Pathway BAM

3.1 Two Pathways: Fast Pathway and Slow Pathway

We must set the parameters of BAM properly to apply BAM to estimate the traffic situation of the network and to configure a VN suitable for the estimated situation. However, it is difficult to establish the VN reconfiguration method that satisfies both speed and accuracy because there is a trade-off between speed and accuracy.

In this thesis, we focus on the cognitive mechanism of the human brain to perform cognition that satisfies both speed and accuracy using BAM. As the knowledge of brain science, it is known that the human brain contains two cognitive pathways, a cognitive pathway that emphasizes speed (FP) and a cognitive pathway that emphasizes accuracy (SP), and recognizes by switching the cognitive pathway [15]. We work to establish the VN reconfiguration method that satisfies both speed and accuracy by incorporating this mechanism into the algorithm.

By operating BAM, which we set the parameters that emphasizes the identification speed over accuracy, and BAM, which we set the parameters that emphasizes the identification accuracy over speed, in parallel in the traffic situations cognition, we establish the VN reconfiguration method that satisfies both speed and accuracy.

When traffic fluctuations occur, we first configure the VNT corresponds to the traffic situation identified by FP-BAM, and then reconfigure the VNT corresponds to the traffic situation later identified by SP-BAM with higher confidence. This algorithm realizes the VN reconfiguration method that shows good performance in both control speed and control accuracy (Fig. 3).

3.2 Principles to Design Parameters for Two-Pathway BAM

We must design the parameters setting of each BAM for VN control, since we cannot represent the parameters of the actual human brain. We need to determine guidelines on how we set each BAM parameters in the VN reconfiguration method using FP-BAM and SP-BAM. Therefore, we first clarify the principles and role of each BAM.

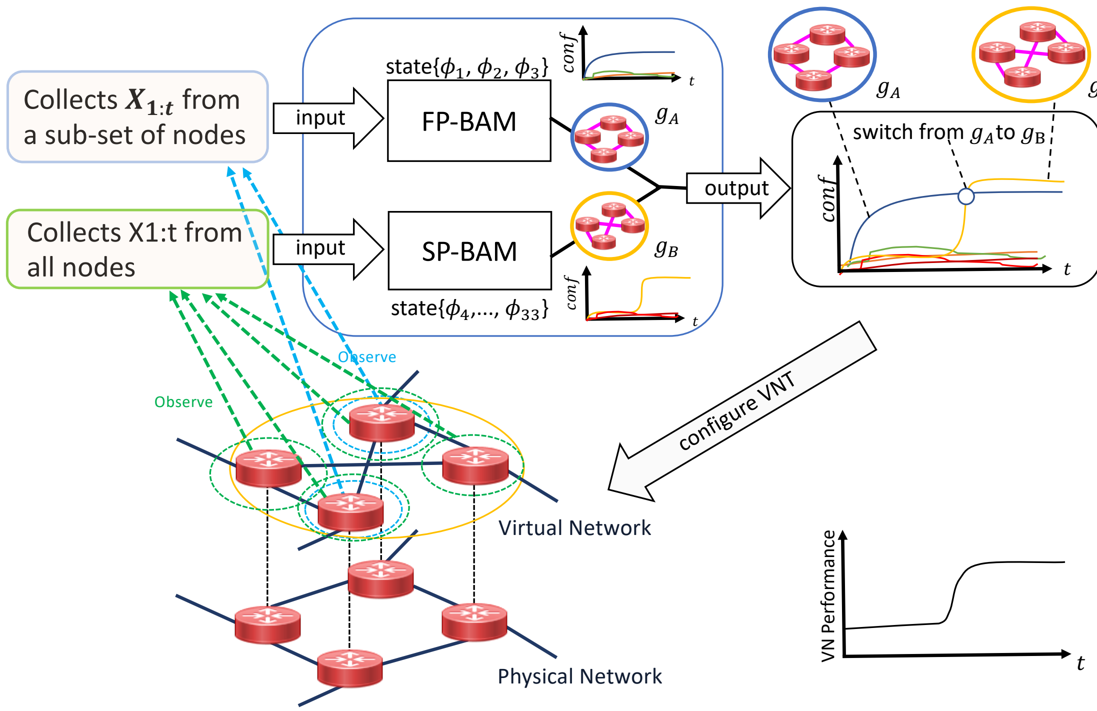

Figure 3: VN Reconfiguration method using two-pathway BAM

For FP-BAM, when network congestion occurs, it aims to quickly eliminate the congestion. For this reason,

FP1-1 Quickly reconfigure the VN in response to traffic fluctuations.

FP1-2 Reconfigure the VNT with a large number of virtual links between nodes to eliminate congestion.

FP-BAM should also set the number of attractors and the amount of information acquired to emphasize cognitive speed, thus,

FP2-1 Keep the number of attractors to a minimum to make quick cognition.

FP2-2 Limit the amount of information to collect information quickly.

For SP-BAM, it aims to reconfigure the VNT that is more suitable for the traffic situations. For this reason,

SP1-1 Reconfigure the VNT by observing the traffic situation in detail.

Table 1: Parameters setting of FP-BAM

Parameter	Setting	Reason (Principle)
Number of attractors BAM holds	fewer	To make quick cognition (Based on **FP2-1)**
Amount of information acquired	limited (Get the traffic amount only from a limited number of nodes)	To make quick decisions by limiting the amount of information (Based on **FP2-2)**
Threshold of confidence for identification	lower	To make quick cognition (Based on **FP1-1)**
Maximum number of links for each node	larger	To follow traffic fluctuations and avoid network congestion (Based on **FP1-2)**
Dynamic uncertainty	larger	To quickly switch attractors (Based on **FP1-1)**

SP1-2 Reconfigure the VNT with fewer virtual links between nodes to reduce power consumption.

SP-BAM should also set the number of attractors and the amount of information acquired to emphasize cognitive accuracy, thus,

SP2-1 Keep a large number of attractors to make accurate cognition.

SP2-2 Maximize the amount of information acquired to make accurate cognition.

3.3 Virtual Network Reconfiguration Method using Two-Pathway BAM

3.3.1 Parameters settings

In consideration of the above, the parameters setting guidelines for each BAM are shown in Table 1,2. By setting the parameters as shown in Table 1,2, this method should work as follows. When the fluctuation of the traffic amount is small, the traffic situation does not change significantly and the optimum VN for the traffic situation does not change so

Table 2: Parameters setting of SP-BAM

Parameter	Setting	Reason (Principle)
Number of attractors BAM holds	larger	To make accurate cognition (Based on **SP2-1)**
Amount of information acquired	maximized (Get the traffic amount from all nodes)	To make accurate decisions by maximizing the amount of information (Based on **SP2-2)**
Threshold of confidence for identification	higher	To make accurate cognition (Based on **SP1-1)**
Maximum number of links for each node	fewer	To reduce power consumption (Based on **SP1-2)**
Dynamic uncertainty	smaller	To accurately switch attractors (Based on **SP1-1)**

much, so that we can reconfigure the VN as intended as described above. When traffic fluctuations occur, FP-BAM identifies the traffic situation in response to the traffic fluctuations quickly and reconfigures the VNT, and SP-BAM continues long-term observations in parallel. When both FP-BAM and SP-BAM identify the traffic situation, SP-BAM reconfigures the VNT with a small number of virtual links to save resources. When the VNT corresponding to the attractor held by SP-BAM with a small amount of resources cannot accommodate a large amount of traffic and the maximum link utilization exceeds the numerical target, we attempt to reduce the maximum link utilization by reconfiguring the VNT corresponding to the attractor held by FP-BAM with a large number of virtual links.

3.3.2 Switching between two pathways

We set the state transition of the VN reconfiguration method using two-pathway BAM as shown in Fig. 4 to reconfigure VN as described above. There are eight states, State 1, State 2-A, State 2-B, State 3-A, State 3-B, State 4-A, State 4-B, and State 5.

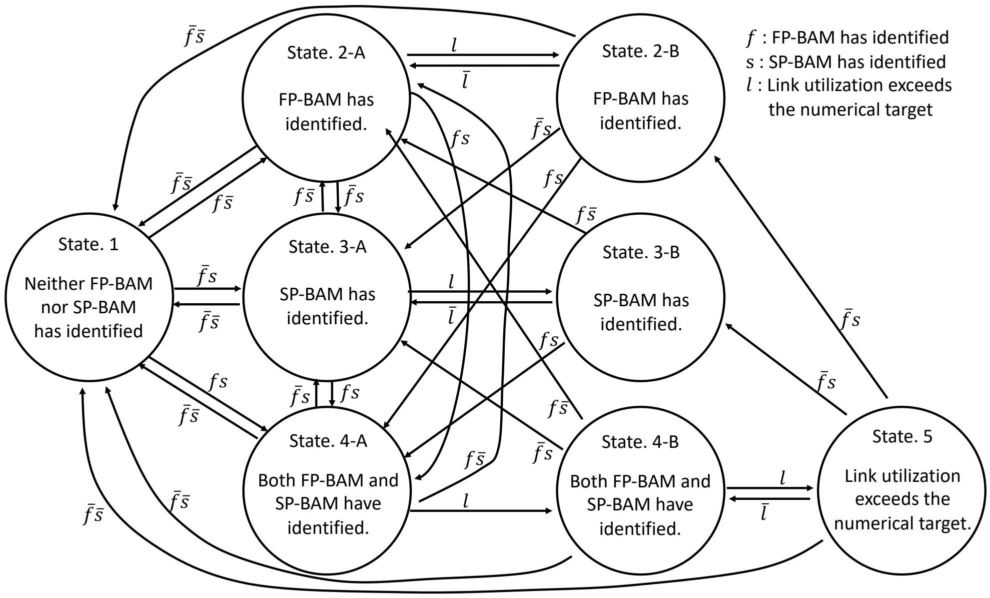

Figure 4: State transition diagram of VNT control using two-pathway BAM

State 1 is the initial state in which both FP-BAM and SP-BAM have not identified. The states are divided depending on the state in which only FP-BAM has identified, the state in which only SP-BAM has identified, and the state in which both FP-BAM and SP-BAM have identified into State 2, State 3, and State 4.

State 2 is divided into State 2-A and State 2-B, and State 3 is divided into State 3-A and State 3-B depending on whether the link utilization is within the numerical target. State 4 in which both FP-BAM and SP-BAM have identified is divided into State 4-A, which the VNT that SP-BAM configures achieves the numerical target, and State 4-B, which FP-BAM configure the VNT with a large number of virtual links since the VNT that SP-BAM configures do not achieve the numerical target. State 5 is the state which the numerical target is not achieved even with the VNT that FP-BAM configure.

When only FP-BAM has identified, transition to State 2-A, and when the link utilization exceeds the numerical target, transition to State 2-B. When only SP-BAM has identified, transition to State 3-A, and when the link utilization exceeds the numerical target, transition to State 3-B. When both FP-BAM and SP-BAM have identified, transition to State 4-A and configure the VNT that SP-BAM identified, which has a small number of virtual links. When the link utilization exceeds the numerical target with the VNT that SP-BAM configured, transition to State 4-B and try to reduce the link utilization by configuring the VNT corresponding to FP-BAM with a large number of virtual links. When even the VNT that FP-BAM configured exceeds the numerical target, transition to State 5.

3.3.3 VNT calculation

SP-BAM holds n_s attractors, and FP-BAM holds n_f less attractors than SP-BAM based on **FP2-1**. We consider how to calculate the VNT suitable for the traffic situation of the attractor each BAM holds. We want to calculate the suitable VNT for the traffic matrix. Each BAM aims at the following.

- FP-BAM wants to configure the VNT to adapt to various traffic situations with a small number of attractors.
- SP-BAM wants to configure the VNT that forms links between node pairs that efficiently flow traffic with a small number of virtual links.

SP-BAM uses HLDA, which will be described later, to efficiently flow traffic by forming virtual links in order from node pairs with the largest amount of traffic. FP-BAM reduces the number of nodes it observes to acquire information quickly. We would like to greatly change the traffic situation FP-BAM assumes depending on the traffic amount of the fewer nodes that it observes. However, it is unclear how FP-BAM can assume greatly different traffic situations. Therefore, we consider extracting attractors with significantly different traffic situations from SP-BAM, which has a large number of attractors. We consider grouping the attractors of SP-BAM based on the traffic amount of the node FP-BAM observes and imitating an attractor from each group with different traffic situations. However, there is no guideline as to which attractor we should imitate from each group with different traffic situations. Therefore, we consider calculating an attractor that has the characteristics of attractors in each group with different traffic amounts. One attractor can be calculated from each group, and the characteristics of the attractors are different each group. From the above, first determine the VNT corresponding to attractors of SP-BAM.

Since we generate a wide variety of traffic matrices for attractors SP-BAM holds, we generate the traffic matrices $A_T^{s(k)}$ for the attractor with index k ($k = 1, \cdots, n_s$) that SP-BAM holds by the following procedure. $A_T^{s(k)}(s, d)$ represents the amount of traffic transmitted from the source node s to the destination node d. $A_T^{s(k)}(s, d) = 0$ when $s = d$. Generate a random number p ($0 \leq p \leq 1.0$) for each node-pair ij,

$$T(i,j) = \begin{cases} \mu + \sigma \cdot N(0,1) & \text{if } 0 \leq p \leq 0.4, \\ 2\mu + \sigma \cdot N(0,1) & \text{if } 0.4 < p \leq 0.75, \\ 3\mu + \sigma \cdot N(0,1) & \text{if } 0.75 < p \leq 1.0. \end{cases} \tag{8}$$

$$A_T^{s(k)}(i,j) = \begin{cases} 0 & \text{if } i = j, \\ T(i,j) & \text{if } i \neq j. \end{cases} \tag{9}$$

SP-BAM holds the amounts of incoming and outgoing traffic of each node based on these traffic matrices as attractors and uses them for cognition. We calculate VNT by a simple heuristic algorithm, HLDA (Heuristic Logical topology Design Algorithm) [16, 17] for the traffic matrix corresponding to the attractor. The outline is shown below.

We calculate n_s VNTs, $A_V^{s(k)}$ for $k = 1, \cdots, n_s$, by setting $A_V^{s(k)}(a, b) = 1$ when a virtual link is assigned from node a to node b of VNT corresponding to index k attractor, and by setting $A_V^{s(k)}(a, b) = 0$ when no virtual link is assigned from node a to node b of VNT corresponding to index k attractor. The matrices $Out(N)$ and $In(N)$ represents the number of transmitter and receiver resources allocated to the virtual link of each node. $Out(c)$ represents the number of assigned transmitters on node c, and $In(c)$ represents the number of assigned receivers on node c. The parameters $OutLimit$ and $InLimit$ represents the maximum number of transmitters and receivers of each node that can allocate resources to virtual links. Each element of $Out(N)$ and $In(N)$ cannot allocate resources beyond $OutLimit$ and $InLimit$.

Algorithm 1 shows the HLDA algorithm. From the 2nd line to the 12nd line, assign virtual links between adjacent nodes. In the 13th line, calculate the node pair with the maximum traffic amount. From the 15th line to the 20th line, assign virtual links between the node pair with the maximum traffic amount when resources are remained, and calculate the node pair with the next maximum traffic amount. Repeat lines 15th to 20th until all node pairs are checked.

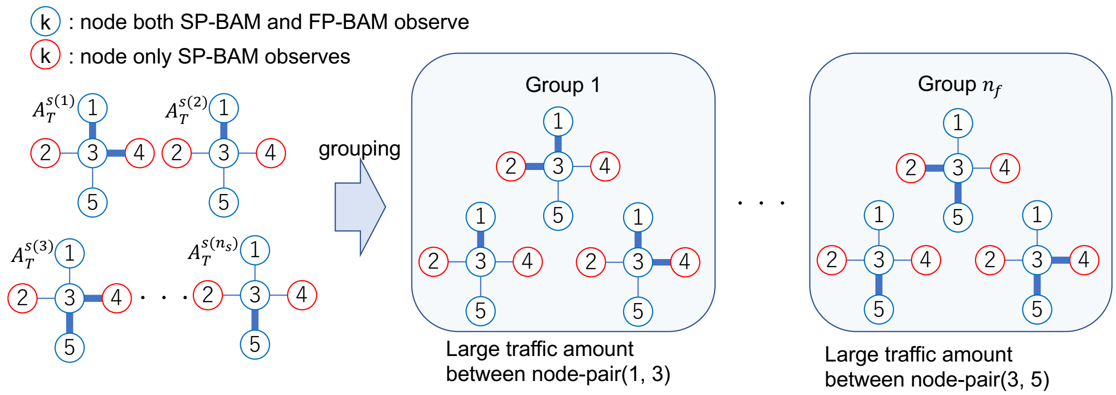

Figure 5: Classifying n_s attractors of SP-BAM into n_f groups

FP-BAM aims to quickly eliminate congestion when a traffic fluctuation occurs. While FP-BAM holds a small number of attractors, it has the requirement to handle a wide range of traffic situations. Therefore, FP-BAM needs to hold attractors whose traffic situation trends are significantly different for each attractor.

We roughly classify n_s SP-BAM attractors into n_f type groups based on the tendency of traffic situations, and use the characteristics of each group to create n_f attractors that FP-BAM holds (Fig. 5). We use the following method as a method for roughly classifying SP-BAM attractors.

Since FP-BAM limits the amount of information based on design principle **FP2-2**, FP-BAM observes only the traffic amount of a limited number of nodes. In addition, FP-BAM generates a VNT that has a larger maximum number of virtual links for each node than SP-BAM based on design principle **FP1-2**. We calculate feature quantities from the traffic matrix of each SP-BAM attractor and classify them into n_f groups based on the tendency of the feature quantity. We construct a feature quantity space using the traffic amount between nodes with a large average amount of traffic as the feature quantity of each attractor. We classify SP-BAM attractors to groups using the k-means method for feature quantities. Algorithm 2 shows a method to classify n_s SP-BAM attractors into n_f groups. From the 1st line to the 6th line, extract the traffic amount of the nodes FP-BAM observes. From the 8th line to the 9th line, group into n_f groups uniformly. From the 11th line to the 12th line, calculate the center of gravity C_k of each group k. From the 13th line to the 14th line, regroup to n_f by distance to center of gravity. Repeat lines

11th to 15th until grouping does not change.

As described above, we generate the traffic matrices $A_T^{s(k)}(i,j)(k = 1, \cdots, n_s)$ for the attractor with Index k that SP-BAM holds by the following procedure. Generate a random number p $(0 \leq p \leq 1.0)$ for each i, j,

$$T(i,j) = \begin{cases} \mu + \sigma \cdot N(0,1) & \text{if } 0 \leq p \leq 0.4, \\ 2\mu + \sigma \cdot N(0,1) & \text{if } 0.4 < p \leq 0.75, \\ 3\mu + \sigma \cdot N(0,1) & \text{if } 0.75 < p \leq 1.0. \end{cases} \tag{10}$$

$$A_T^{s(k)}(i,j) = \begin{cases} 0 & \text{if } i = j, \\ T(i,j) & \text{if } i \neq j. \end{cases} \tag{11}$$

SP-BAM attractor with index k represents the amount of traffic flowing from node a to node b in $A_T^{s(k)}(a,b)$. FP-BAM observes only t $(1 \leq t \leq N)$ nodes (N_1, N_2, …, N_t) with a large amount of traffic to make a quick decision. Then, we classify n_s SP-BAM attractors into n_f groups based on the traffic amount of t nodes that FP-BAM observes.

We roughly classify SP-BAM attractors based on the features of traffic matrices by focusing on the traffic amount flowing between arbitrary nodes of t nodes (Fig. 5). Since there are $\frac{t(t-1)}{2}$ combinations of t nodes, information focusing on the traffic amount of SP-BAM attractor between each node of t nodes is mapped to the $\frac{t(t-1)}{2}$-dimensional feature quantity space. $F(N, \frac{t(t-1)}{2})$ represents the feature quantities of SP-BAM attractor focusing on the t nodes of the traffic matrix. SP-BAM attractor of index k is located at $(F_k(1), F_k(2), ..., F_k(\frac{t(t-1)}{2}))$ on the $\frac{t(t-1)}{2}$-dimensional feature quantity space.

We calculate one VNT of FP-BAM attractor for each of the classified groups. When observing the current traffic situation $T(i,j)$, FP-BAM emphasizes the following points.

1. FP-BAM identifies the current traffic situation as one of FP-BAM attractors A^f.

2. When FP-BAM identifies the current traffic situation as $A^{f(1)}$, VNT $V^{f(1)}$ corresponding to $A^{f(1)}$ eliminates network congestion.

Since the principle of FP-BAM is to eliminate congestion when the traffic situation fluctuates, we generate virtual links between node pairs with large amounts of traffic for calculating VNT.

We need to form virtual links between node pairs required for a VNT can accommodate the traffic matrix of more attractors in the group. Therefore, we preferentially form virtual links between the node pairs that have a large number of times in common with VNTs of SP-BAM attractors in the group, which we call Frequency-based method. As long as physical resources remain, we form virtual links in sequence between the node pair that have the largest number of times in common with VNTs of SP-BAM attractors in the group. When there are multiple node pairs that have the same number of times in common with VNTs of SP-BAM attractors in the group and it becomes impossible to form a virtual link for physical resources confliction, we calculate VNTs using brute-force. When physical resources are surplus after brute force, we form virtual links in order from the node pair with the smallest index until the physical resources are exhausted. After a conflict, we try multiple thresholds of brute force. We perform brute force between the node pairs that have more than the threshold number of times virtual links are formed. Suppose a physical resource conflict occurs when the number of times in common with VNTs of SP-BAM attractors in the group is c. We implement Frequency-based-1 method in which a threshold is set to maximum value c, and Frequency-based-2 method in which a threshold is set to maximum value c -1 , and Frequency-based-all method in which a threshold value is set to 1 and performs brute force among all node pairs. In addition, as an algorithm for comparison, we also consider Brute-Force method which exclude node pairs with a small number of times in common with VNTs in the group from the candidates and performs brute force. We evaluate how much better the performance of the method is compared to Centroid-based method that calculate simple VNT using HLDA for the traffic matrix of the center of gravity in the group. Algorithm 3 shows the algorithm of Frequency-based method. From the 3rd line to the 11th line, function $CalculateTiebreak(L_k)$ calculates the maximum value c. From the 12th line to the 15th line, perform brute force between the node pairs that have more than each threshold number of times virtual links are formed.

Since FP-BAM aims to respond quickly to avoid congestion, we generate a traffic matrix with a randomly increased amount of traffic based on the traffic matrix of the center of gravity of the group, and evaluate whether the VNT accommodate the traffic matrix.

We use the center of gravity of the traffic matrices in the group as the traffic matrix of

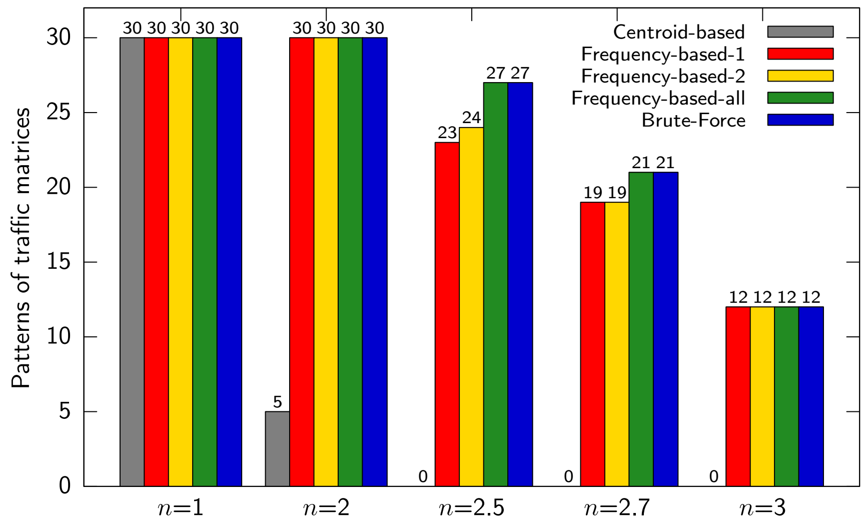

Figure 6: Patterns of traffic matrices accomodated for each VNT calculation method

attractors FP-BAM holds. We generate a traffic matrix $A_T^{f'(k)}$ with a randomly increased traffic amount by adding $n \cdot \mu \cdot |N(0,1)|$ to the traffic matrix of the center of gravity using a normal distribution and evaluate whether the VNT accommodates it. We generate a traffic matrix $A_T^{f'(k)}$ by 30 times with the normal distribution of different seeds and compare the number of times the VNT accommodates it. That is,

$$A_T^{f'(k)}(i,j) = \left\{ \; A_T^{f(k)}(i,j) + n \cdot \mu \cdot |N(0,1)|. \right. \tag{12}$$

In Fig. 6, the horizontal axis shows n, and the vertical axis shows how many times VNT accommodates out of the 30 traffic matrices generated with n ($n = 1$ to 3). Brute-Force has a huge number of candidate node pairs and cannot complete the calculation, so it shows the result obtained with the best VNT configured by 10 hours after the algorithm execution.

The difference between the elements of the traffic matrix of SP-BAM attractor is about 2 μ at most. Therefore, it is considered that the traffic matrix, which is obtained by adding $2\mu \cdot |N(0,1)|$ to each element of the traffic matrix $A_T^{f(1)}$ of FP-BAM attractor, is grouped into other groups and FP-BAM does not recognize X_n as $A^{f(1)}$. Therefore, we think that it is more than enough to accommodate all traffic matrices when $n = 2$, but we use

Frequency-based-all in consideration of the number of accommodations when it is larger than $n = 2$.

We calculate the VNT of FP-BAM attractor by following the steps below.

1. Calculate the VNT for the traffic matrix of all SP-BAM attractors in the group using HLDA.

    - Generate a VNT with the maximum number of virtual links on each node set to the same as parameter of FP-BAM.

2. Count the number of virtual link formations between each node pair for all VNTs of SP-BAM in the group.

    - Preferentially forms virtual links between node pairs that have a large number of times in common with VNTs in the group based on FP-BAM principle of eliminating network congestion.

3. Form virtual links in order from the node pair that have the largest number of times in common with VNTs in the group. Test if the generated VNT can accommodate the traffic matrix for each attractor of SP-BAM in the group.

    - When conflicts occur between node pairs that have the same number of times in common, use brute force among all the remaining nodes to form virtual links and generate VNTs.

    - Hold the VNT, which can accommodate the most traffic matrices in the group, as FP-BAM attractor.

    - Use the center of gravity of each group used in the k-means method as the traffic matrix of FP-BAM attractor.

By following the above steps, FP-BAM holds VNTs that work well in a variety of traffic situations, and quickly configure VNT for traffic fluctuations.

Algorithm 1 Calculate VNT (HLDA)

Input: $A_T^{HLDA}(double \times double)$

Output: $A_V^{HLDA}(int \times int)$

$A_V^{HLDA}(N \times N) \Leftarrow 0$
for $i = 1 \ldots N-1$ **do** ▷ Assign virtual links between adjacent nodes
  if $Out(i) < OutLimit$ and $In(i+1) < InLimit$ and $A_V^{HLDA}(i, i+1) = 0$ **then**
    $A_V^{HLDA}(i, i+1) \Leftarrow 1$
    $Out(i) \Leftarrow Out(i) + 1$
    $In(i+1) \Leftarrow In(i+1) + 1$
    $A_T^{HLDA}(i, i+1) \Leftarrow 0$
  if $Out(i+1) < OutLimit$ and $In(i) < InLimit$ and $A_V^{HLDA}(i+1, i) = 0$ **then**
    $A_V^{HLDA}(i+1, i) \Leftarrow 1$
    $Out(i+1) \Leftarrow Out(i+1) + 1$
    $In(i) \Leftarrow In(i) + 1$
    $A_T^{HLDA}(i+1, i) \Leftarrow 0$
$S_{\max}, D_{\max} \Leftarrow \arg\ \max_{S,D}(A_T^{HLDA}(S, D))$ ▷ calculate the maximum node pair.
while $A_T^{HLDA}(S_{\max}, D_{\max}) \neq 0$ **do** ▷ Repeat until all node pairs are checked
  if $Out(S_{\max}) < OutLimit$ and $In(D_{\max}) < InLimit$ and $A_V^{HLDA}(S_{\max}, D_{\max}) = 0$ **then**
  $A_V^{HLDA}(S_{\max}, D_{\max}) \Leftarrow 1$
  $Out(S_{\max}) \Leftarrow Out(S_{\max}) + 1$
  $In(D_{\max}) \Leftarrow In(D_{\max}) + 1$
$A_T^{HLDA}(S_{\max}, D_{\max}) \Leftarrow 0$
$S_{\max}, D_{\max} \Leftarrow \arg\ \max_{S,D}(A_T^{HLDA}(S, D))$

Algorithm 2 Feature extraction and grouping

Input: $A_T^{s(1 \ldots n_s)}(double \times double)$

Output: $S_{1 \ldots n_f}(int), C_{1 \ldots n_f}(int)$

for $n = 1 \ldots n_s$ **do**
    $i \Leftarrow 1$
    for $a = 1 \ldots t-1$ **do**
        for $b = a+1 \ldots t$ **do**
            $F_n(i) \Leftarrow A_T^{s(n)}(N_a, N_b) + A_T^{s(n)}(N_b, N_a)$
            $i \Leftarrow i+1$
$t \Leftarrow 0$
for $k = 1 \ldots n_f$ **do** ▷ Group into n_f uniformly
    $S_k^{(t)} \Leftarrow \{F_n : n \bmod n_f = k\}$ ▷ $S_k^{(t)}$ represents the group of index k at step t
while $\{\exists i \mid S_i^{(t)} \neq S_i^{(t-1)}\}$ **do**
    for $k = 1 \ldots n_f$ **do**
        $C_k \Leftarrow \frac{1}{|S_k^{(t)}|} \sum_{F_n \in S_k^{(t)}} F_n$ ▷ Calculate the center of gravity C_k of each group k
    for $k = 1 \ldots n_f$ **do** ▷ Regroup to n_f by distance to center of gravity
        $S_k^{(t+1)} \Leftarrow \{F_n : |F_n - C_k|^2 \leq |F_n - C_j|^2 \ \forall j, 1 \leq j \leq n_f\}$
    $t \Leftarrow t+1$

Algorithm 3 The calculation method of $V^{f(k)}$ (Frequency-based)

Input: $L^k_{ij}(int \times int), n^s_k(int)$

Output: $V^{f(k)}_{ij}(int \times int)$

n^s_k represents the number of SP-BAM attractors classified into group k to calculate FP-BAM attractor with index k.

Define $L^k(N \times N)$, which $L^k(i,j)$ represents the number of times a virtual link is formed between (i,j) using HLDA for all traffic matrices of SP-BAM in the group.

function CalculateTiebreak(L_k)

  for $c = n^s_k \ldots 0$ **do**

    for each node-pair(i, j) **do**

      if $L^k(i,j) == c$ **then**

        if Physical resource exists to form a virtual link between (i,j) **then**

          $V^{f(k)}(i,j) \leftarrow 1$

        else

          $V^{f(k)} \leftarrow V_{keep}$

          return c

    $V_{keep} \leftarrow V^{f(k)}$

Brute force between node pairs (i,j) where $L_k(i,j)$ is threshold or higher.

Frequency-based-1 uses l for threshold.

Frequency-based-2 uses $l-1$ for threshold.

Frequency-based-all uses 1 for threshold.

Figure 7: A 12-node network

4 Evaluations of the Virtual Network Reconfiguration Method

4.1 Simulation Settings

In the simulation, we use a 12-node network in the shape of Japan (Fig. 7). We simulate VN reconfiguration method using the two-pathway BAM on a 12-node network, and grasp the tendency of appropriate parameter setting in each BAM. After grasping appropriate parameter in a 12-node network, we perform VN reconfiguration simulation for 19 nodes EON to confirm that the parameter setting for each BAM are valid even on different topology. In a 12-node network, FP-BAM observes only three nodes, Tokyo (node 3), Osaka (node 8), and Fukuoka (node 11), to reduce the amount of information (Fig. 7). In a 19-node network, FP-BAM observes only four nodes, node 2, node 7, node 12, and node 15, to reduce the amount of information (Fig. 8).

We give traffic fluctuations and measure the speed and accuracy of VN reconfiguration. In the simulation, we cause one traffic fluctuation per trial. We simulate the trial 300 times and evaluate the average. We generate the traffic situation by the following formula during the simulation. We change the seed of the normal distribution N_{fluc} in the following

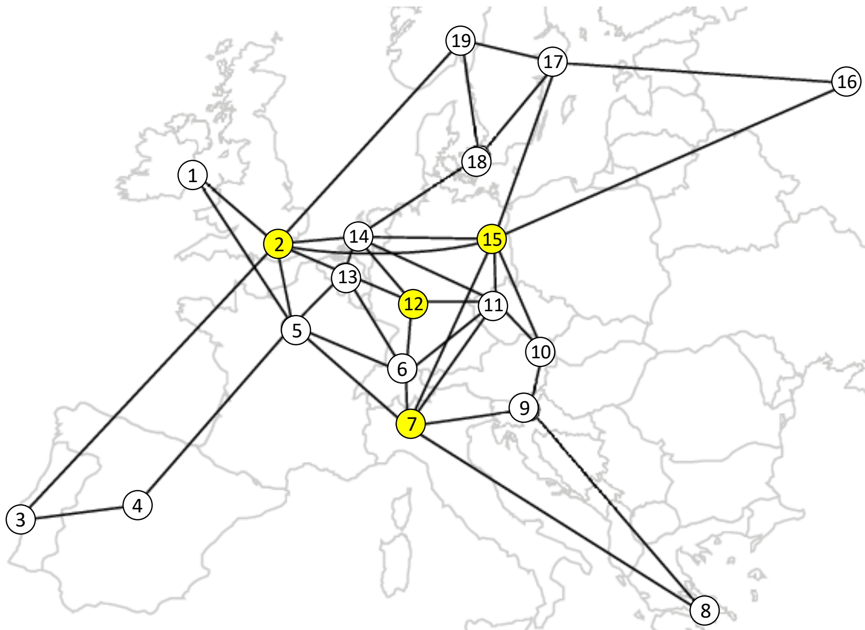

Figure 8: EON (European Optical Network)

equation each time a traffic fluctuation occurs.

$$T_{step}(i,j) = \begin{cases} 0 & \text{if } i = j, \\ |2\mu + \mu \cdot N_{fluc}(0,1)| & \text{if } i \neq j. \end{cases} \tag{13}$$

During the t_{fluc} steps after the traffic fluctuation occurs, we add a little noise to the traffic situation for each step based on the traffic matrix generated by the above equation, and continue to generate roughly the same traffic situation. We change the seed of the normal distribution N_{step} in the following formula for each step.

$$T_{step}(i,j) = \begin{cases} 0 & \text{if } i = j, \\ |2\mu + \mu \cdot N_{fluc}(0,1) + \sigma_{noise} \cdot \mu \cdot N_{step}(0,1)| & \text{if } i \neq j. \end{cases} \tag{14}$$

The numerical target of the maximum link utilization in each step is not to exceed 0.8. For FP-BAM, we set the maximum number of virtual links for each node to two in addition to the degree of each node on the physical network. For SP-BAM, we set the maximum number of virtual links for each node to one in addition to the degree of each node on the physical network.

4.2 Evaluation Results

4.2.1 Results of 12-node network

We simulate VN reconfiguration method using Two-Pathway BAM on a 12-node network.

Impact of the parameter q (dynamics uncertainty)

We evaluate the appropriate parameter q (dynamics uncertainty) in FP-BAM and SP-BAM. The dynamics uncertainty represents "ease of transition between attractors". Therefore, in FP-BAM, which emphasizes speed to respond quickly to traffic fluctuations, it is intuitively correct to set q larger. In SP-BAM, which emphasizes accuracy to switch the attractor with sufficiently confidence, it is intuitively correct to set q smaller.

Suppose a traffic fluctuation occurs and the traffic situation becomes A. Prepare $\delta_A(t)$ and when BAM has identified one of attractors at step t, $\delta_A(t)$ is set to 1, otherwise set to 0. We simulate 300 trials with $t_{fluc} = 50$ and calculate the average number of steps required for FP-BAM to identify the traffic situation against traffic fluctuations. We simulate setting σ_{noise} to 0.1 and changing the parameter q from 0.4 to 1.0 in 0.1 increments. Fig. 9 shows the cumulative complementary distribution of the number of steps required for FP-BAM to identify the traffic situation against traffic fluctuations. In Fig. 9, the horizontal axis shows the number of observation steps (t), and the vertical axis shows the probability that attractor identification is not completed within t step as a cumulative complementary distribution. We calculate the value on the vertical axis by taking the average of $\delta_A(t)$ with 300 traffic fluctuations within step t and subtracting it from 1. That is, it is desirable that the probability on the vertical axis decreases as the number of observation steps on the horizontal axis increases. The vertical axis shows the value obtained by taking the natural logarithm.

There is not much difference depending on the value of q, but the identification is a little fastest when we change q from 0.7 to 0.9. Since q is a parameter that indicates the ease of transition between attractors, it is desirable for FP-BAM that emphasize speed to set q larger to respond quickly to traffic fluctuations. We conclude that it is appropriate for FP-BAM to set $q = 0.9$ among $q = 0.7$ to 0.9 with an equivalent performance to emphasize speed.

Next, we evaluate the parameter q in SP-BAM. Since the confidence value may fluc-

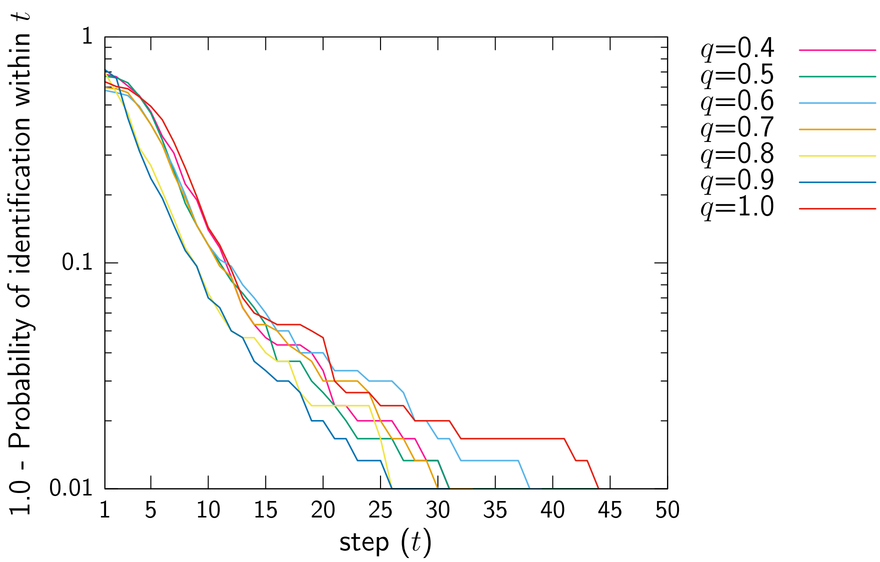

Figure 9: Cumulative complementary distribution of the number of steps to identify for FP-BAM: 12-node network

tuate excessively in noisy situations, as an evaluation for whether SP-BAM emphasizes accuracy than speed, examine whether the confidence value does not fluctuate excessively with respect to other attractors after identifying the traffic situation.

In the traffic fluctuation, we generate the traffic situation based on the traffic matrix of one of SP-BAM attractor (the index of the attractor is k in the formula below) and add noise as shown in the following formula;

$$T_{step}(i,j) = \begin{cases} 0 & \text{if } i = j, \\ A_T^{s(k)}(i,j) + \sigma_{noise} \cdot N_{step}(0,1) & \text{if } i \neq j. \end{cases} \tag{15}$$

We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 200$. We calculate the ratio of 1) the case when SP-BAM continues to identify the correct attractor, 2) the case when SP-BAM identifies the correct attractor but confidence value fluctuates to other attractors on the way, and 3) the case when SP-BAM cannot identify the correct attractor, to the traffic situation generated after each traffic fluctuation. In SP-BAM, which emphasizes accuracy, it is desirable that the ratio of the case when SP-BAM continues to identify the correct attractor is high. When we set the σ_{noise} to 0.4, the identification ratio for each q is as shown in Fig. 10. When we set the σ_{noise} to 0.8, the identification ratio for each q is as shown in Fig. 11.

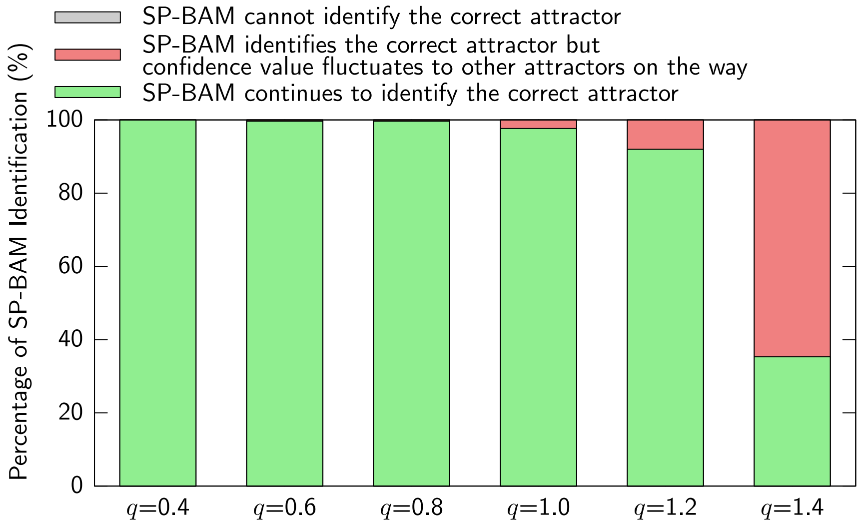

Figure 10: Identification ratio of SP-BAM: 12-node network, $\sigma_{noise} = 0.4$

When we set q too large, the ratio of the case when SP-BAM identifies the correct attractor but confidence value fluctuates to other attractors on the way increases, but when we set the value of q too small, the ratio of the case when SP-BAM cannot identify the correct attractor increases. When we set q to 0.6, the ratio of the case when SP-BAM continues to identify the correct attractor is the highest. We conclude that it is appropriate for SP-BAM to set q to 0.6 to emphasize accuracy.

From the above, we set the parameter q to 0.9 for FP-BAM and 0.6 for SP-BAM. This result is in line with the principles we have expected in advance, as q represents the ease of attractor transitions.

Impact of the number of attractors

We evaluate the speed of FP-BAM by the number of attractors. Suppose a traffic fluctuation occurs and the traffic situation becomes A. Prepare $\delta_A(t)$ and when BAM has identified one of attractors at step t, $\delta_A(t)$ is set to 1, otherwise set to 0.

We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 50$. We calculate the rate at which $\delta_A(t)$ becomes 1 when a total of 300 trials traffic fluctuations occur. We generate the traffic matrix A_T of the traffic situation A after the traffic fluctuation by the following formula. We change the seed of the normal distribution N_{fluc} in

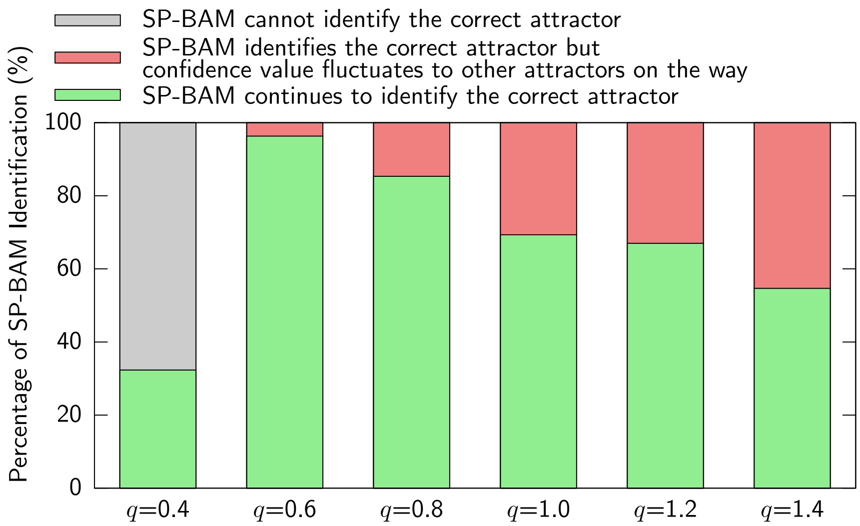

Figure 11: Identification ratio of SP-BAM: 12-node network, $\sigma_{noise} = 0.8$

the following formula for each trial.

$$A_T(i,j) = \begin{cases} 0 & \text{if } i = j, \\ |2\mu + \mu \cdot N_{fluc}(0,1)| & \text{if } i \neq j. \end{cases} \tag{16}$$

In Fig. 12, the horizontal axis shows t [step], and the vertical axis shows the probability that attractor identification is not completed within t step as a cumulative complementary distribution. We calculate the value on the vertical axis by taking the average of $\delta_A(t)$ with 300 traffic fluctuations within step t and subtracting it from 1. That is, it is desirable that the value on the vertical axis becomes smaller as the number of observation steps on the horizontal axis increases.

It is shown that the smaller the number of attractors FP-BAM holds, the faster FP-BAM identifies the traffic situation. Since FP-BAM has a small number of attractors when the number of FP-BAM attractors is 3, the identification is delayed by a few percent compared to when the number of FP-BAM attractors is 6 or 9, but the percentage of identification when FP-BAM holds 3 attractors is the highest from 0 to 15 steps, so FP-BAM is able to identify the traffic situation quickly. Regarding SP-BAM, although the number of steps required for identification of SP-BAM increases as the number of attractors increases, the possibility of identification at 50 steps is higher when the number

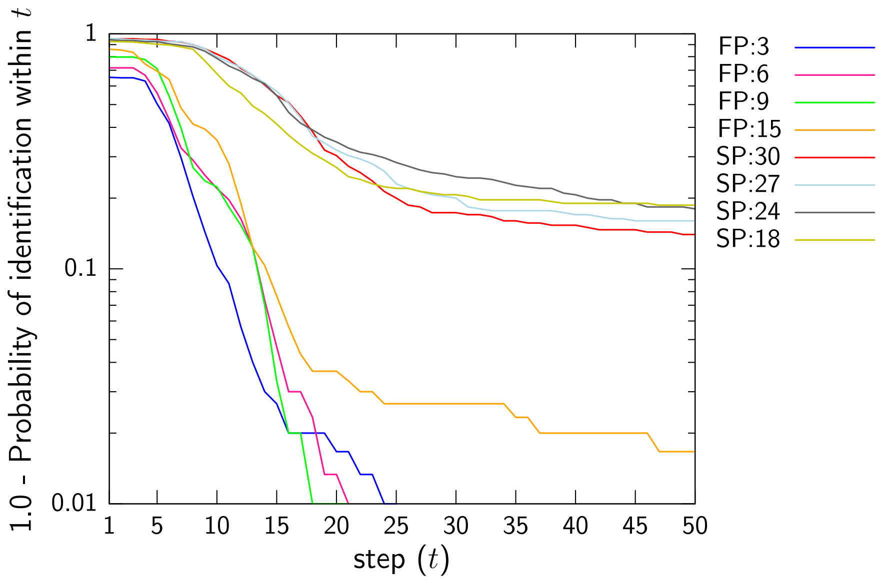

Figure 12: Cumulative complementary distribution of the number of steps to identify for FP-BAM and SP-BAM: 12-node network

of attractors is large.

Speed of identification of FP-BAM

We set the number of attractors to 33 for VN reconfiguration method using one BAM. To set the total number of attractors 33 as well, we set the combination of n_f and n_s as $(n_f, n_s) = (3, 33), (6, 27), (9, 24)$ and compare with the method using only SP-BAM.

We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 50$ and evaluate average steps required for identification. In Fig. 13, the horizontal axis shows the number of attractors, and the vertical axis shows the average steps required for identification.

FP-BAM, which require fewer steps for identification, are operated in parallel to quickly reconfigure the VNT in response to traffic fluctuations.

The accuracy of SP-BAM

We evaluate the accuracy of SP-BAM. To evaluate the accuracy, we evaluate whether the VNT that SP-BAM configures after the traffic fluctuation is suitable for the post-

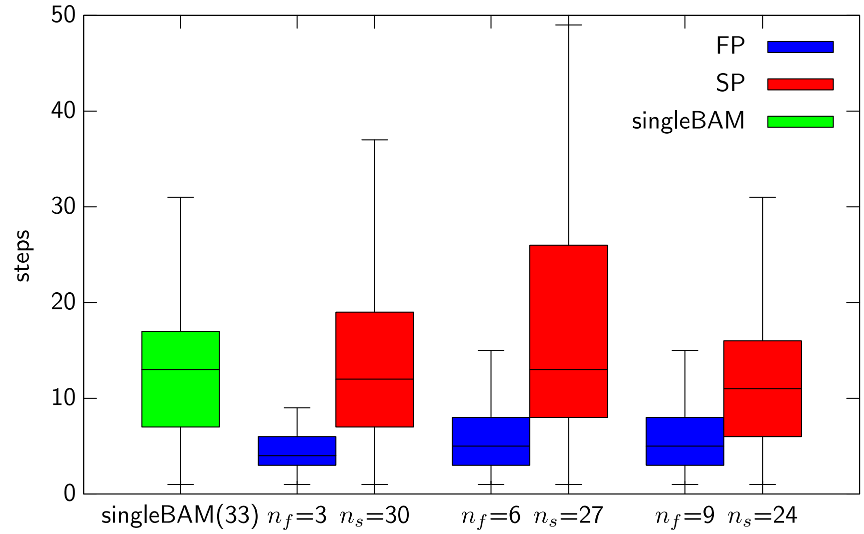

Figure 13: Box-plot of number of steps required for identification: 12-node network

fluctuation traffic matrix, that is, whether virtual links are formed between the node pairs with a large amount of traffic.

We rank the node pairs in descending order of the traffic amount in the traffic matrix after the traffic fluctuation. When SP-BAM identifies the traffic situation and configures the VNT, we evaluate the accuracy of SP-BAM based on whether links are formed between node pairs at the top group of the traffic amount ranking for the configured VNT. For comparison, we also calculate the accuracy when FP-BAM identifies the traffic situation and configures the VNT. The detailed definition of the evaluation method is described below.

The number of attractors is set to $n_f = 3$ for FP-BAM and $n_s = 30$ for SP-BAM. Let $A_V^{f(k)}$ and $A_V^{s(k)}$ be the VNT identified either by FP-BAM or SP-BAM 50 steps after the traffic fluctuation, and we calculate whether $A_V^{f(k)}$, $A_V^{s(k)}$ forms a virtual link between node pair for all of the traffic rankings. We perform 300 trials which a traffic fluctuation occurs and evaluate the percentage that $A_V^{f(k)}$, $A_V^{s(k)}$ forms virtual links between node pairs with large traffic amount. 21 cases in which SP-BAM failed to identify the traffic situation are excluded. In Fig. 14, the horizontal axis shows the ranking r of the traffic amount between node pair, and the vertical axis shows each percentage that FP-BAM and SP-BAM forms a virtual link between the node pair of the traffic amount corresponding to the ranking r.

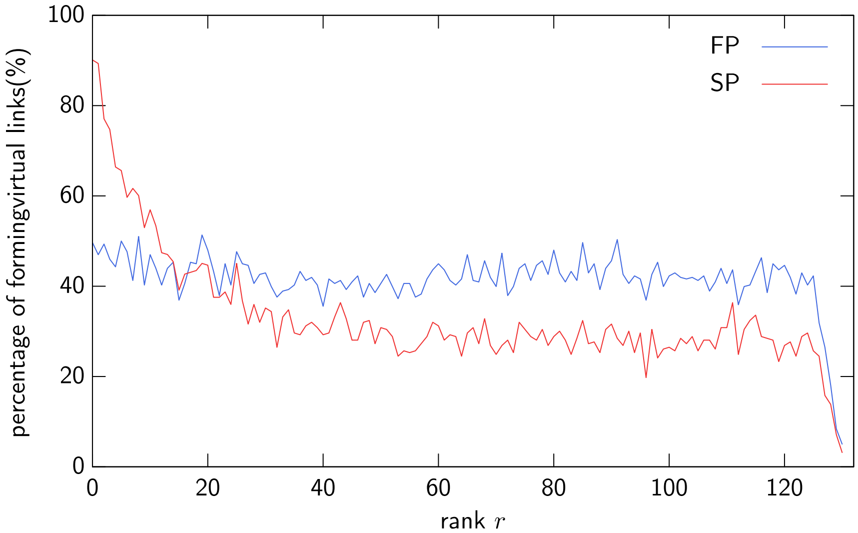

Figure 14: Relation betwen the traffic amount and existence probability of virtual link: 12-node network

The percentage of virtual links that are formed between the node pairs with the higher traffic amount ranking is higher in SP-BAM than in FP-BAM. It can be said that SP-BAM accurately configures the VNT suitable for the traffic situation.

Performance of the VN reconfiguration method with FP-BAM and SP-BAM in parallel

We set the combination of n_f and n_s as $(n_f, n_s) = (3, 33), (6, 27), (9, 24)$ and compare with the method using only SP-BAM. We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 50$ and evaluate the average number of steps where the link utilization exceeds the numerical target.

In Fig. 15, the horizontal axis shows the number of attractors, and the vertical axis shows the average number of steps where the link utilization exceeds the numerical target.

The smaller the number of attractors FP-BAM holds and the larger the number of attractors SP-BAM holds, the average number of steps where the link utilization exceeds the numerical target is suppressed. The reason for this is the result shown in Fig. 13 and Fig. 14. Figure 13 shows that the smaller the number of attractors of FP-BAM is the fewer steps required for the identification is, and Figure 14 shows that SP-BAM preferentially

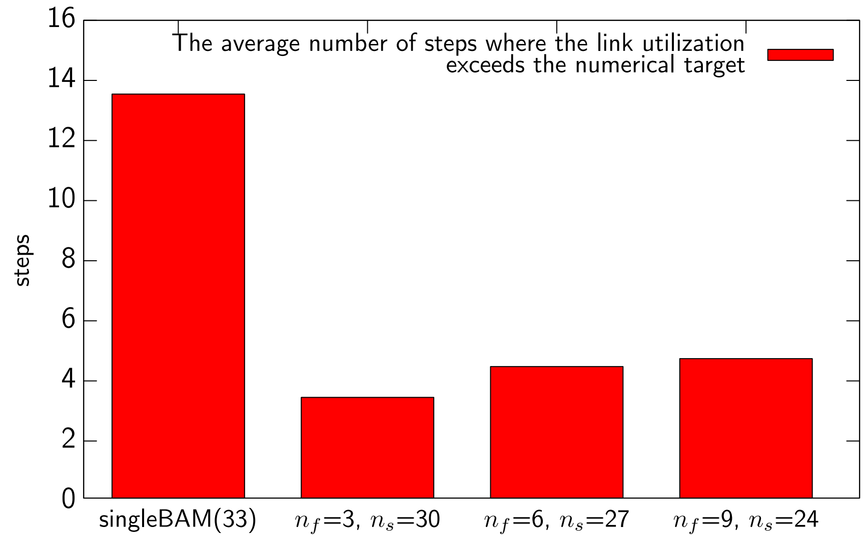

Figure 15: Average number of steps where the link utilization exceeds the threshold: 12-node network

forms virtual links between node pairs with higher traffic. The speed of identification of FP-BAM and the accuracy of SP-BAM shown by these figures reduces the average number of steps where the link utilization exceeds the numerical target.

4.2.2 Results of 19-node network

Next, we simulate VN reconfiguration method using Two-Pathway BAM on a 19-node network.

Impact of the parameter q (dynamics uncertainty)

We evaluate the appropriate parameter q (dynamics uncertainty) in FP-BAM and SP-BAM for a 19-node network in the same way as a 12-node network. We simulate 300 trials with $t_{fluc} = 50$ and calculate the average number of steps required for FP-BAM to identify the traffic situation against traffic fluctuations. We simulate setting σ_{noise} to 0.1 and changing the parameter q from 0.4 to 1.0 in 0.1 increments. In Fig. 16, the horizontal axis shows the number of observation steps (t), and the vertical axis shows the probability that attractor identification is not completed within t step as a cumulative complementary distribution. That is, it is desirable that the probability on the vertical axis decreases as the number of observation steps on the horizontal axis increases. The vertical axis shows

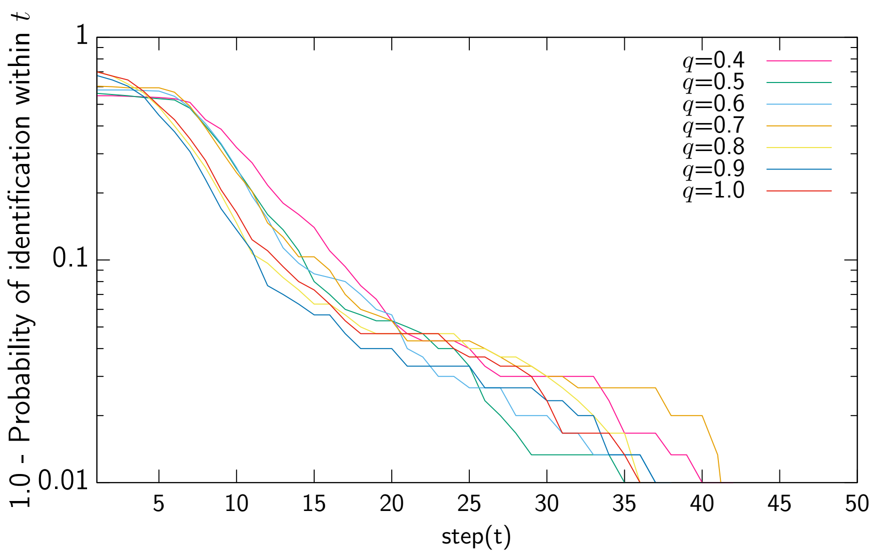

Figure 16: Cumulative complementary distribution of the number of steps to identify for FP-BAM: 19-node network

the value obtained by taking the natural logarithm.

We got a result similar in a 12-node network. Since q is a parameter that indicates the ease of transition between attractors, it is desirable for FP-BAM that emphasize speed to set q larger to respond quickly to traffic fluctuations. We conclude that it is appropriate for FP-BAM to set $q = 0.9$ among $q = 0.7$ to 0.9 with equivalent performance to emphasize speed.

Next, we evaluate the parameter q in SP-BAM for a 19-node network in the same way as a 12-node network. We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 200$. We calculate the ratio of 1) the case when SP-BAM continues to identify the correct attractor, 2) the case when SP-BAM identifies the correct attractor but confidence value fluctuates to other attractors on the way, and 3) the case when SP-BAM cannot identify the correct attractor, to the traffic situation generated after each traffic fluctuation. In SP-BAM, which emphasizes accuracy, it is desirable that the ratio of the case when SP-BAM continues to identify the correct attractor is high. When we set the σ_{noise} to 0.4, the identification ratio for each q is as shown in Fig. 17. When we set the σ_{noise} to 0.8, the identification ratio for each q is as shown in Fig. 18.

We got a result similar in a 12-node network. When we set q too large, the ratio of

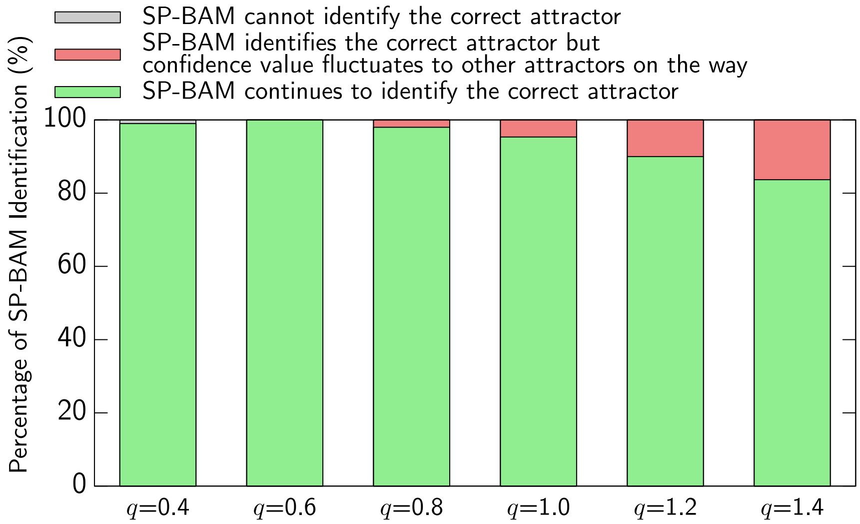

Figure 17: Identification ratio of SP-BAM: 19-node network, $\sigma_{noise} = 0.4$

the case when SP-BAM identifies the correct attractor but confidence value fluctuates to other attractors on the way increases, but when we set the value of q too small, the ratio of the case when SP-BAM cannot identify the correct attractor increases. When we set q to 0.6, the ratio of the case when SP-BAM continues to identify the correct attractor is the highest. We conclude that it is appropriate for SP-BAM to set q to 0.6 to emphasize accuracy.

From the above, we set the parameter q to 0.9 for FP-BAM and 0.6 for SP-BAM also in a 19-node network. This result is in line with the principles we have expected in advance, as q represents the ease of attractor transitions.

Impact of the number of attractors

We evaluate the speed of FP-BAM by the number of attractors for a 19-node network in the same way as a 12-node network. We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 50$. In Fig. 19, the horizontal axis shows t (step), and the vertical axis shows the probability that attractor identification is not completed within t step as a cumulative complementary distribution. That is, it is desirable that the value on the vertical axis becomes smaller as the number of observation steps on the horizontal axis increases.

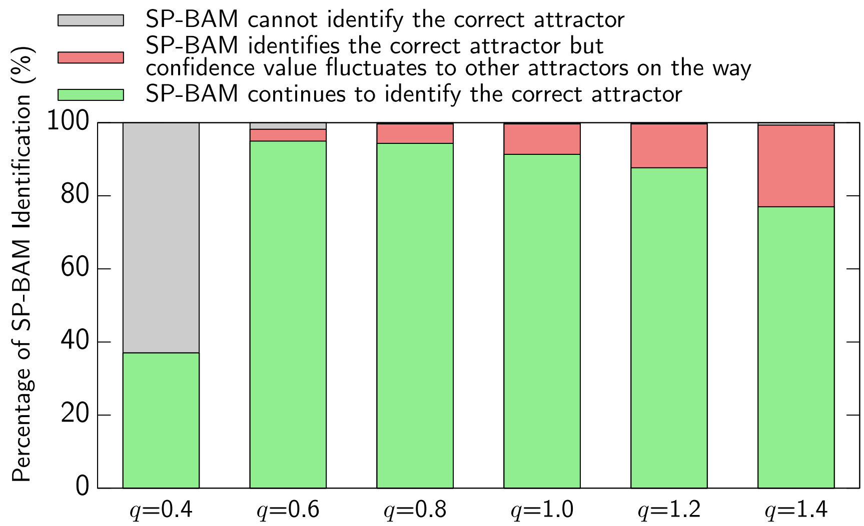

Figure 18: Identification ratio of SP-BAM: 19-node network, $\sigma_{noise} = 0.8$

It is shown that the smaller the number of attractors FP-BAM holds, the faster FP-BAM identifies the traffic situation also in a 19-node network.

Speed of identification of FP-BAM

We set the number of attractors to 33 for VN reconfiguration method using one BAM. To set the total number of attractors 33 as well, we set the combination of n_f and n_s as $(n_f, n_s) = (3, 33), (6, 27), (9, 24)$ and compare with the method using only SP-BAM.

We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 50$ and evaluate average steps required for identification. In Fig. 20, the horizontal axis shows the number of attractors, and the vertical axis shows the average steps required for identification.

FP-BAM, which require fewer steps for identification, are operated in parallel to quickly reconfigure the VNT in response to traffic fluctuations.

The accuracy of SP-BAM

We evaluate the accuracy of SP-BAM. To evaluate the accuracy, we evaluate whether the VNT that SP-BAM configures after the traffic fluctuation is suitable for the post-fluctuation traffic matrix, that is, whether virtual links are formed between the node pairs

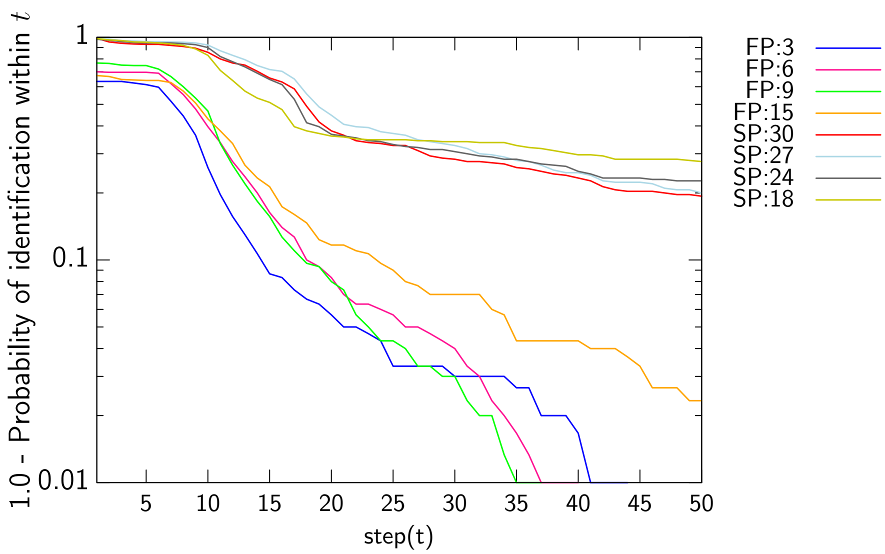

Figure 19: Cumulative complementary distribution of the number of steps to identify for FP-BAM and SP-BAM: 19-node network

with a large amount of traffic.

We rank the node pairs in descending order of the traffic amount in the traffic matrix after the traffic fluctuation. When SP-BAM identifies the traffic situation and configures the VNT, we evaluate the accuracy of SP-BAM based on whether links are formed between node pairs at the top group of the traffic amount ranking for the configured VNT. For comparison, we also calculate the accuracy when FP-BAM identifies the traffic situation and configures the VNT. The detailed definition of the evaluation method is described below.

The number of attractors is set to $n_f = 3$ for FP-BAM and $n_s = 30$ for SP-BAM. Let $A_V^{f(k)}$ and $A_V^{s(k)}$ be the VNT identified either by FP-BAM or SP-BAM 50 steps after the traffic fluctuation, and we calculate whether $A_V^{f(k)}$, $A_V^{s(k)}$ forms a virtual link between node pair for all of the traffic rankings. We perform 300 trials which a traffic fluctuation occurs and evaluate the percentage that $A_V^{f(k)}$, $A_V^{s(k)}$ forms virtual links between node pairs with large traffic amount. 21 cases in which SP-BAM failed to identify the traffic situation are excluded. In Fig. 21, the horizontal axis shows the ranking r of the traffic amount between node pair, and the vertical axis shows each percentage that FP-BAM and SP-BAM forms a virtual link between the node pair of the traffic amount corresponding to the ranking r.

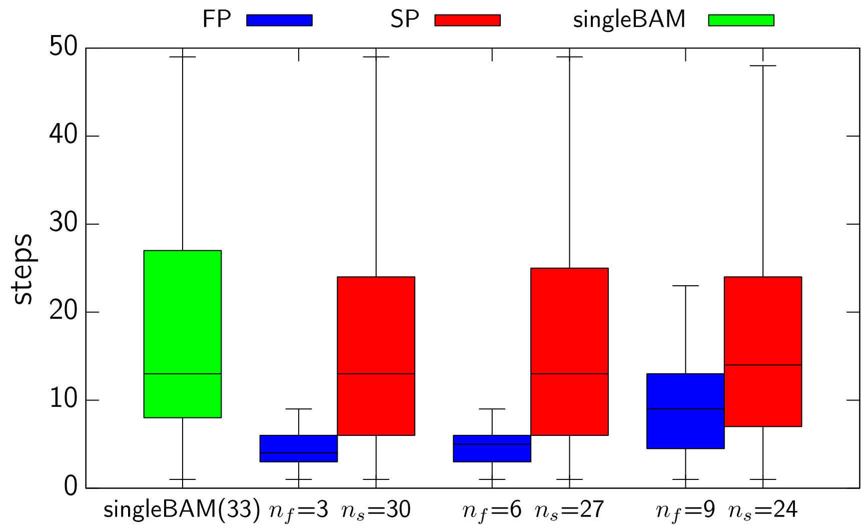

Figure 20: Box-plot of number of steps required for identification: 19-node network

The percentage of virtual links that are formed between the node pairs with the higher traffic amount ranking is higher in SP-BAM than in FP-BAM. It can be said that SP-BAM accurately configures the VNT suitable for the traffic situation.

Performance of the VN reconfiguration method with FP-BAM and SP-BAM in parallel

We set the combination of n_f and n_s as $(n_f, n_s) = (3, 33), (6, 27), (9, 24)$ and compare with the method using only SP-BAM. We simulate 300 trials which traffic fluctuation occurs one time with $t_{fluc} = 50$ and evaluate the average number of steps where the link utilization exceeds the numerical target.

In Fig. 22, the horizontal axis shows the number of attractors, and the vertical axis shows the average number of steps where the link utilization exceeds the numerical target.

We got a result similar in a 12-node network. The smaller the number of attractors FP-BAM holds and the larger the number of attractors SP-BAM holds, the average number of steps where the link utilization exceeds the numerical target is suppressed.

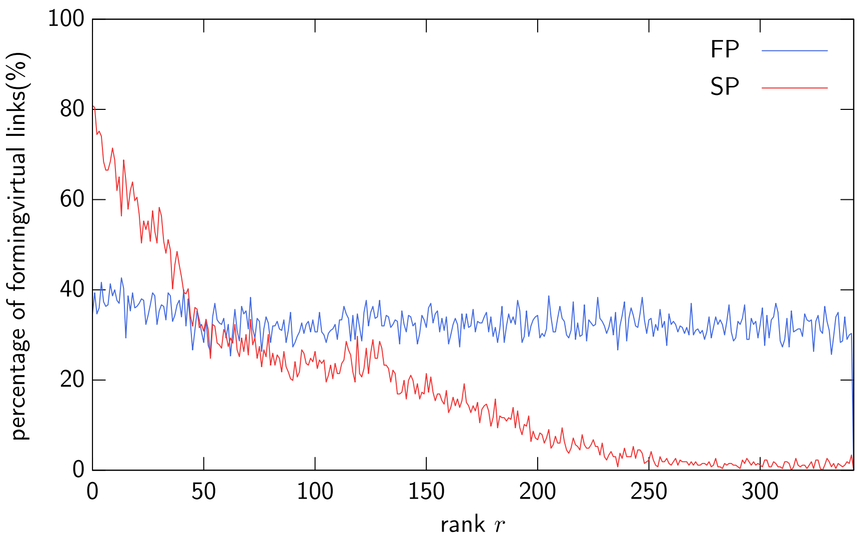

Figure 21: Relation betwen the traffic amount and existence probability of virtual link: 19-node network

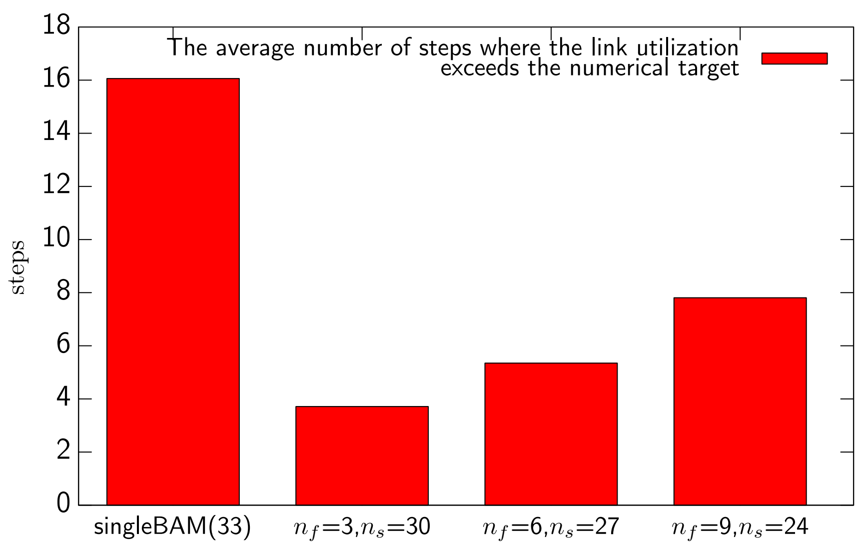

Figure 22: Average number of steps where the link utilization exceeds the threshold: 19-node network

5 Conclusion

We showed the VN reconfiguration method using the speed-emphasized BAM and the accuracy-emphasized BAM in parallel were able to quickly reconfigure a VNT to eliminate congestion and later reconfigure a high-performance VNT. In Chapter 4.2 we simulated on two networks to provide guidelines for parameters on different topologies. We would like to simulate what the result will be depending on conditions such as the magnitude of traffic fluctuations. Moreover, since we evaluated only by simulation, it is a future task to evaluate using the actual traffic on the actual network.

In the actual environment, we would like to think about the replacement of attractors. When there is an attractor that has not been used for a long time, the traffic situation that the attractor assumes is not appropriate for the current environment. It is meaningless to assume a traffic situation that is not appropriate for the current environment, so it is better to replace the unused attractor and assume a different traffic situation. The traffic situation may change suddenly, but it is possible to grasp the tendency statistically. For example, the amount of traffic between a certain node pair and in a certain area tends to increase during the day, the amount of traffic is generally low at night, etc.

We may reconfigure the VN more appropriately by replacing the attractors BAM holds in consideration of changes in the traffic situation on the real network over the time of the day. In addition to FP-BAM and SP-BAM, we consider increasing the number of BAMs that operate in parallel on the assumption that the BAM replace attractors it holds. We think that in the daytime, BAM holds attractors that corresponding VNT forms virtual links between nodes that tend to have a lot of traffic amount, and at night when the amount of traffic is low, BAM holds attractors that corresponding VNTs save the number of virtual links. By increasing the number of BAMs that operate in parallel, BAMs assume more traffic situations. We consider operating additional BAMs, which replace attractors in consideration of the tendency of the traffic situation on a monthly basis (a traffic amount between a specific node pair tends to increase at the beginning of the month, traffic amount between node pairs is small overall at the end of the month, etc.) or the tendency of the traffic situation on a yearly basis (a traffic amount between a specific node pair tends to increase in summer, etc.), in parallel. On the other hand, it is hard to control VN while combining cognitive results when multiple BAMs identify traffic situations, so it is necessary to carefully add a new BAM.

Acknowledgments

I am deeply grateful to everyone who supported me during my master's degree studies. First, my deepest heartfelt appreciation goes to Professor Masayuki Murata of Osaka University. He gave me a lot of sharp advice during my research and kindly guided me. I would like to express my gratitude to Associate Professor Shin'ichi Arakawa of Osaka University for taking the time to provide appropriate advice and guidance. I received generous guidance about BAM from Specially Appointed Associate Professor Alparslan Onur of Osaka University. I want to thank Associate Professor Yuichi Ohsita of Osaka University, Assistant Professor Daichi Kominami of Osaka University and Specially Appointed Assistant Professor Tatsuya Otoshi of Osaka University for their valuable comments and suggestions on this study. Special thanks to members in Murata Lab for making my days enjoyable.

References

[1] R. Casellas, A. Giorgetti, R. Morro, R. Martinez, R. Vilalta, and R. M. noz, “Virtualization of disaggregated optical networks with open data models in support of network slicing,” *IEEE/OSA Journal of Optical Communications and Networking*, vol. 12, pp. A144–A154, Feb. 2020.

[2] Y. Ou, M. Davis, A. Aguado, F. Meng, R. Nejabati, and D. Simeonidou, “Optical network virtualisation using multitechnology monitoring and SDN-enabled optical transceiver,” *IEEE/OSA Journal of Lightwave Technology*, vol. 36, pp. 1890–1898, May 2018.

[3] D. Zad Tootaghaj, F. Ahmed, P. Sharma, and M. Yannakakis, “Homa: An efficient topology and route management approach in SD-WAN overlays,” in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, Jul. 2020, pp. 2351–2360.

[4] Y. Jararweh, M. Al-Ayyoub, A. Darabseh, E. Benkhelifa, M. Vouk, and A. Rindos, “Software defined cloud: Survey, system and evaluation,” *Future Generation Computer Systems*, vol. 58, pp. 56–74, May 2016.

[5] P. Bhaumik, S. Zhang, P. Chowdhury, S. S. Lee, J. Lee, and B. Mukherjee, “Software-defined optical networks (SDONs): A survey,” *Photonic Network Communications*, vol. 28, no. 1, pp. 4–18, Jun. 2014.

[6] A. Hakiri, A. Gokhale, P. Berthou, D. C. Schmidt, and T. Gayraud, “Software-defined networking: Challenges and research opportunities for future Internet,” *Computer Networks*, vol. 75, Part A, pp. 453–471, Dec. 2014.

[7] V.Gkamas, K.Christodoulopoulos, and E.Varvarigos, “A joint multi-layer planning algorithm for IP over flexible optical networks,” *IEEE/OSA Journal of Lightwave Technology*, vol. 33, pp. 2965–2977, Jul. 2015.

[8] K. Assis, S. Peng, R. Almeida, H. Waldman, A. Hammad, A. Santos, and D. Simeonidou, “Network virtualization over elastic optical networks with different protection schemes,” *IEEE/OSA Journal of Optical Communications and Networking*, vol. 8, pp. 272–281, Apr. 2016.

[9] F. Morales, M. Ruiz, L. Gifre, L. M. Contreras, V. L. opez, and L. Velasco, "Virtual network topology adaptability based on data analytics for traffic prediction," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 9, pp. A35–A45, Jan. 2017.

[10] A. Gencata and B. Mukherjee, "Virtual-topology adaptation for WDM mesh networks under dynamic traffic," *IEEE/ACM Transactions on Networking (TON)*, vol. 11, pp. 236–247, Apr. 2003.

[11] S. Bitzer, J. Bruineberg, and S. J. Kiebel, "A bayesian attractor model for perceptual decision making," *PLoS Computational Biology*, vol. 11, p. e1004442, Aug. 2015.

[12] T. Ohba, S. Arakawa, and M. Murata, "Bayesian-based virtual network reconfiguration for dynamic optical networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, pp. 440–450, Apr. 2018.

[13] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," in *Proceedings of the national academy of sciences*, vol. 81, May 1984, pp. 3088–3092.

[14] S. S. Haykin *et al.*, "Kalman filtering and neural networks," *Wiley Online Library*, 2001.

[15] I. Fujita and T. Doi, "Seeing the world in three dimensions, fast-but-rough and slow-but-accurate," *Atlas of Science*, Jan. 2017.

[16] R. Ramaswami and K. N. Sivarajan, "Design of logical topologies for wavelength-routed optical networks," *IEEE Journal on Selected Areas in Communications*, vol. 14, pp. 840–851, Jun. 1996.

[17] 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, pp. 598–607, Oct. 2000.