Attractor Selection-based Virtual Network Topology Control with Dynamic Threshold Reconfiguration for Managed Self-organization Network

Shohei Kamamura*, Yuki Koizumi†, Daisaku Shimazaki*, Takashi Miyamura*, Shin'ichi Arakawa†, Kohei Shiomoto‡, Atsushi Hiramatsu*, and Masayuki Murata†
*NTT Network Service Systems Laboratories, NTT Corporation
‡ NTT Service Integration Laboratories NTT Corporation
9-11, Midori-Cho 3-Chome Musashino-Shi, Tokyo, 180-8585 Japan
E-mail:{kamamura.shohei, shimazaki.daisaku, miyamura.takashi, shiomoto.kohei, hiramatsu.atsushi}@lab.ntt.co.jp
† Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka 565-0871, Japan
Email: {ykoizumi, arakawa, murata}@ist.osaka-u.ac.jp

Abstract—In the managed self-organization network concept, multiple virtual networks are constructed on a single physical network. They are controlled with a self-organization mechanism based on attractor selection algorithm while the physical resource management server arranges resources allocation for the total resources optimization. An objective of the self-organization mechanism is to keep a high applicability to network changes; it is robust but resources utilization is not necessarily optimized. As a result, the resource management server cannot know the correct states of each virtual network. For extending the applicability of the current self-organization mechanism, this paper proposes a novel virtual network reconfiguration algorithm by dynamically feeding back to the target value of maximum link utilization for attractor selection algorithm. We also show the effectiveness through extensive simulations.

***Index Terms*— Attractor selection, Dynamic reconfiguration, Managed self-organization network, Virtual network**

I. INTRODUCTION

Future network should flexibly deal with the unexpected network changes caused by the diversification of the network services, and economically accommodate multiple service networks.

We have studied the managed self-organization network [1] for satisfying above-mentioned future network requirements. One aspect of the managed self-organization network is network virtualization; single physical infrastructure accommodates multiple virtual networks (VNs). Each VN constructs virtual network topology (VNT) [2] on assigned resources to accommodate IP traffic over wavelength-routed WDM network. A VNT consists of IP routers and IP logical links. A wavelength path established on a physical layer corresponds to the IP link on IP layer. By dynamically reconfiguring the form of VNT in accordance with traffic demands, we can flexibly provide adaptive network to network changes. Another is a managed self-organization mechanism; each VN controls its network with self-organization mechanism while a central physical network (PN) manager arranges the resource allocation for each VN to optimize total resource utilization. We realize the self-organization behavior by attractor selection algorithm [3], which models behaviors where living organisms adapt to unknown changes in their surrounding environments and recover their conditions.

To balance a resources allocation in accordance with demands of each VN, the PN manager should know the accurate state of each VN. The design strategy of the current attractor selection algorithm is to achieve robustness; the solution is not necessarily optimal but has high applicability to network changes. The attractor selection algorithm uses maximum link utilization as a control objective, and continues the VNT reconfiguration until the algorithm gets the maximum link utilization, which drops below a threshold value (e.g., 50%). In other words, VNT reconfiguration stops as long as the maximum link utilization is less than the threshold, even if it is not optimal value. As a result, the PN manager cannot understand whether VN utilizes given resources or not, and then inaccurate resources reallocation occurs. Algorithms for VNT optimization are also proposed [4-6]. However, they require end-to-end traffic called traffic matrix while attractor selection algorithm only uses the link-by-link traffic. Traffic matrix

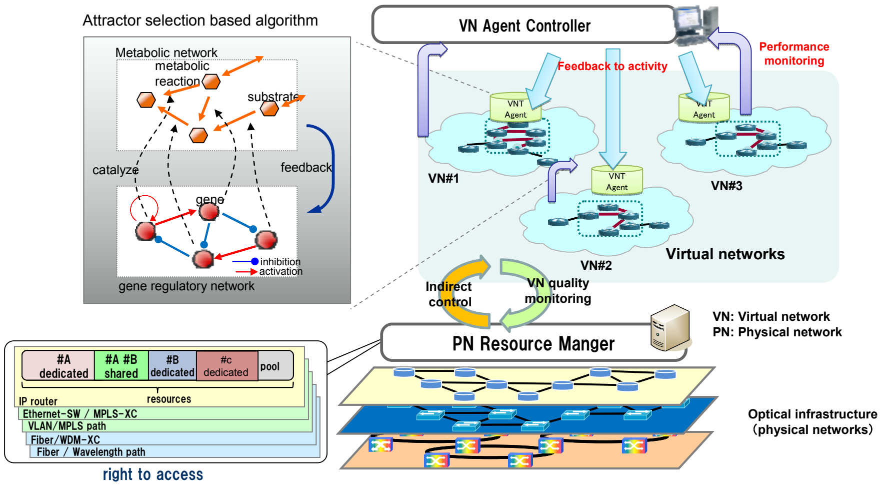


Fig. 1 Concept of managed self-organization network

fluctuates greatly, and accurate measurement is depends on traffic distributions [7].

In this paper, we extend the applicability of current attractor-based algorithm to handle a network optimization problem for the managed self-organization network. The key idea of our proposal is dynamic threshold reconfiguration algorithm. The algorithm restricts a feasible solution space by changing the threshold value. If resources are not efficiently used, the algorithm gradually reduces the threshold, and searches the solution that provides efficient link utilization. Inversely, if a behavior of attractor selection is unstable, the algorithm increases the threshold for network stabilization. We propose two algorithms: sequential search-based, which gradually decreases the threshold, and binary search-based. For the general search problem, binary search is faster than sequential search. On the other hand, the case for applying this network optimization problem, the binary search algorithm is not necessarily fast and is not stable because the accurate lower bound for binary search is unknown. With sequential search, we can get about 25% reduction compared to the current algorithm.

The contributions of our work are the dynamic threshold reconfiguration algorithm for attractor selection-based VNT control, and demonstration of its effectiveness. For the attractor selection algorithm, its applicability is extended to deal with the network optimization problem. For the managed self-organization network, PN manager get the accurate state of each VNs, and stocks more pooled resources for resource allocation.

The rest of the paper is organized as follows. In Section II, we describe the concept of managed self-organization network, and introduce the attractor selection-based VNT configuration algorithm. In Section III, our dynamic threshold reconfiguration algorithm and framework are presented. In section IV, we present performance evaluation. Finally, we conclude our discussion in Section V.

II. Managed Self-organization Network

In this section, we firstly introduce the concept of self-organization network and a VNT control algorithm based on attractor selection [3]. Then, we state our problem.

A. *Overview of Managed Self-organization Network*

The managed self-organization network (Fig. 1) is composed of a single PN and multiple VNs, which share the physical resource, e.g., optical cross connects, IP routers, fibers, and wavelength paths. As control systems, the PN manager manages the physical resource allocation to each VN, and the VN agent, which is assigned to each VN, controls the configuration of VN in accordance with traffic demand. In each VN, VNT [2] is constructed by utilizing assigned resources to accommodate IP traffic over wavelength-routed WDM network. A VNT consists of IP routers and IP logical links. A wavelength path, which is established on a physical layer, corresponds to the IP link on IP layer. We assume that we accommodate each service on the exclusive VN, and a few hundreds of VNs are accommodated on the single physical network.

The key concept of the managed self-organization network is that each VN is controlled by self-organization mechanism while the PN manager arranges resource allocation to avoid instability due to resource contention among VNs. Each VN is basically controlled by self-organization mechanism to handle unexpected network changes flexibly [3]. On the other hand, to control each VN independently causes the resource contention because the PN resources are shared with all VNs. To avoid the resource contention, the PN manager periodically monitors the performance of VNs. If the PN manager detects inadequate resource allocation, it re-allocates the resources for improving total resource utilization. The resource management model, and dynamic resource rearrangement algorithm were proposed in [8] and [1], respectively. They are not main scope of this paper.

B. *Self-organization based on Attractor Selection*

In this section, we present a self-organizing mechanism by VNT control based on attractor selection [3], which models behaviors where living organisms adapt to unknown changes in their surrounding environments and recover their conditions. The fundamental concept underlying attractor selection is that the system is driven by *stochastic* and *deterministic behaviors*, and these are controlled by *activity*, which is a simple feedback of current system conditions. While rule-based heuristic approaches cannot handle unexpected environmental changes, attractor selection has the capability of adapting to unknown changes with stochastic behavior. As one of models for an attractor selection, the mechanism for adaptability of a cell, which consists of a gene regulatory network and a metabolic network, is introduced in [9].

In the VNT control algorithm with attractor selection [3], the gene regulatory network and the metabolic reaction network are interpreted as a WDM network and an IP network, respectively. As the gene regulatory network in a cell controls the metabolic reaction network to adapt to changes in the availability of a nutrient outside the cell, the WDM network configures VNTs for the IP network to adapt to environmental changes. We place genes on all the candidates of feasible lightpaths l_i and expression level of each gene x_i determines whether the corresponding lightpath is established or not. The dynamics of x_i is formulated as follows,

$$\frac{dx_i}{dt} = \alpha \cdot \left(\varsigma\left(\sum_j W_{ij} \cdot x_j - \theta \right) - x_i \right) + \eta \, . \qquad (1)$$

The first term at the right hand side represents the deterministic behavior and the second term η represents stochastic behavior. We use white Gaussian noise whose mean value is zero as η as shown in [9,10]. Our method controls the deterministic and stochastic behaviors depending on the activity *α,* which expresses the condition of the configured VNT. The deterministic behavior controls x_i due to the effects of activation and inhibition from the other genes. The regulatory matrix W_{ij} indicates the relations of activation and inhibition among genes. The rate of increase in the expression level is given by the sigmoidal regulation function, $\varsigma(z) = 1/(1+\exp(-\mu z))$, where μ indicates the gain parameter of the sigmoid function. The range of x_i is [0, 1]. In the case that x_i is higher than a threshold 0.5, the lightpath l_i is established, otherwise it is not established.

The activity α is the index for the performance of IP network and formulated with maximum link utilization $u_{\max}$ as follows,

$$\alpha = \frac{\gamma}{1+\exp(\delta \cdot (u_{\max} - \zeta))} \, , \qquad (2)$$

where γ, ζ, and δ are constant values. The constant value γ specifies the range of activity, [0, γ]. By Eqn. (2), the activity rapidly approaches to 0 if $u_{\max}$ exceeds a threshold ζ, and its slope is determined by δ. When the $u_{\max}$ is lower than ζ, i.e., the current VNT configuration is suitable for the environment, and activity is high, deterministic behavior dominates the VNT configurations. When $u_{\max}$ is higher than ζ and thus activity is low, stochastic behavior dominates over deterministic behavior.

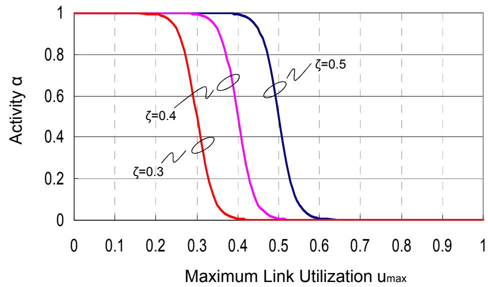


Fig. 2 Relationship between maximum link utilization and activity with different threshold value ($\gamma = 1, \delta = 50$)

While stochastic behavior is dominant in controlling VNTs, x_i fluctuates randomly due to noise and the system searches for a new VNT configuration. When the system conditions have recovered, deterministic behavior again controls VNTs. These two behaviors are controlled by activity. In this way, our approach adapts to environmental changes by selecting attractors using stochastic behavior, deterministic behavior, and activity.

C. Problem Statement

The strategy of the current VNT configuration algorithm with attractor-selection is to achieve robustness; the solution is not necessarily optimal but has high applicability to network changes. In other words, VNT reconfiguration stops as long as the maximum link utilization $u_{\max}$ is less than the threshold ζ, even if it is not optimal value. As a result, the PN manager cannot understand whether VN utilizes given resources or not, and then inaccurate resources reallocation possibly occurs.

From the above-mentioned observation, our problem statement is extending the current attractor selection-based VNT configuration algorithm for applying the performance optimization problem. Though algorithms for VNT optimization have already proposed [4-6], they require the end-to-end traffic for optimization, and their computation is exhaustive. The advantage of attractor selection-based algorithm is only requiring the link-by-link traffic, and its lightweight computation. Therefore, our focus is extending the applicability of the current attractor selection-based algorithm while its advantages are maintained.

III. Attractor Selection-based Virtual Network Topology Control with Dynamic Threshold Reconfiguration

Our strategy for handling a VNT optimization problem with the attractor selection is dynamically changing the threshold value ζ. Changing the threshold value equals to the restriction of a feasible solution space. Here, we focus on the relation between the activity α and threshold ζ. Figure 2 illustrates the relation between α and $u_{\max}$ with different ζ. When $u_{\max}$ approaches to ζ, α rapidly approaches to zero. That is, target value for $u_{\max}$ is changed by changing ζ. Then, by changing the ζ by PN manager, we can control the target value for each VN.

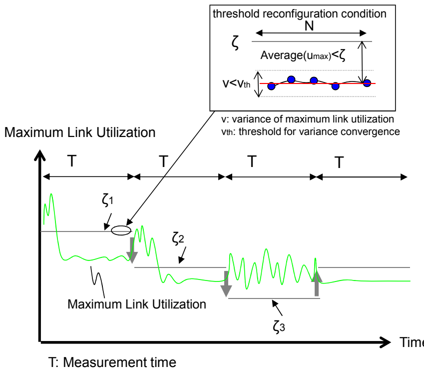


Fig. 3 Overview of dynamic threshold reconfiguration

The Key idea of our algorithm is dynamically changing the threshold ζ based on the current $u_{\max}$. Figure 3 illustrates an overview of our algorithm. With attractor selection-based algorithm, VNT reconfiguration stops if current $u_{\max}$ drops below ζ. If we can get such a VNT solution, we reduce ζ, and search more optimal solution, which achieves lower $u_{\max}$. When we assume that VNT configuration stochastically converges by T times configuration, we judge the solution convergence by past N times measurement of $u_{\max}$. If solution is converged, ζ is updated to lower value than the current $u_{\max}$. Inversely, if $u_{\max}$ does not converges constant value, our algorithm increases ζ, and search the solution, which realize the stable VNT. Then, we introduce our algorithms; sequential search-based algorithm and binary search-based algorithm.

A. Algorithms

Figure 4 and Fig. 5 are pseudo code for sequential search and binary search respectively, and the notation for them is illustrated in Tab. I. Each description is a single threshold configuration after over T-times configuration with attractor selection. If the *finish()* subroutine is called, alarm is notified to PN manager, and dynamic threshold configuration is not performed any more.

In the sequential search-based algorithm (Fig. 4), average and variance are checked whether their convergence conditions are satisfied; average is below ζ and variance is also below v_{th} (line 1). If above conditions are satisfied, current ζ and the current VNT ($VNT_{current}$) are saved to ζ_{stable} and VNT_{stable} respectively (line 2, 3). Then, new ζ is computed by subtracting minimum unit m from the current ζ. The minimum unit is sufficiently-small (e.g., 0.01), and then most relaxed restriction is given for next solution search. If above conditions (line 1) are not satisfied, recovery to the stable conditions is performed (line 5—8). Recovery is performed by restoring ζ_{stable} and VNT_{stable} to ζ and $VNT_{current}$. Then, threshold configuration finishes by *finish()* subroutine (line 8).

Then we describe the binary search-based algorithm using Fig. 5. Firstly, *flag* value, which is used for finish judging, is initialized as zero. Then convergence conditions are checked (line 2) in the same with sequential one. If convergence is finished, binary search for reducing ζ is performed (line 3—6). The current ζ is saved as ζ_{prev} (line 3), and new ζ is set to intermediate value between ζ and *bound* (line 4). The bound is lower limit for binary search, and then it is set to lower value than the current ζ or the current $u_{\max}$. The finish condition is checked using *flag* value (line 5). If *flag* is one, threshold configuration finishes (line 6). Then, if convergence conditions (line 2) are not satisfied, binary search for increasing ζ is performed (line 8—9). As upper bound, the *bound* is set to ζ_{prev}, which last achieves stable VNT (line 8). Then the new ζ is set to intermediate value between *bound* and ζ (line 9). When once VNT experiences unstable condition, *flag* value is set to one (line 10). If convergence conditions are satisfied in the next threshold configuration, it finishes by *finish()* subroutine. This specific finish judgment is for the fast convergence of dynamic threshold reconfiguration.

TABLE. I NOTATION FOR ALGORITHMS

u_{max}	maximum link utilization
ζ	current threshold for objective u_{max}
v	variance of maximum link utilization
v_{th}	threshold for variance convergence
bound	lower bound for binary search
$\mathrm{average}_N()$	return a average value between past N points
finish()	return a message to finish the threshold configuration

1 if $((\mathrm{average}_N(u_{max}) < \zeta)$ *and* $(v < v_{th}))$
2 $\quad \zeta_{stable} = \zeta$
3 $\quad \mathrm{VNT}_{stable} = VNT_{current}$
4 $\quad \zeta = average_N(u_{max}) - m$
5 else
6 $\quad \zeta = \zeta_{stable}$
7 $\quad \mathrm{VNT}_{current} = VNT_{stable}$
8 $\quad$ finish()
9 end

Fig. 4 Pseudo code for sequential search

B. Framework for Applying the Managed Self-organization Network

In the framework of the managed self-organization network, PN manager can know the correct state of each VN by optimizing the performance (e.g. maximum link utilization) of each VN. As a result, PN manager can efficiently allocate physical resources with its algorithm [1]. Figure 6 illustrates application of our algorithm to the managed self-organization framework. In this example, 50% resource utilization is desirable considering unexpected demand changes. Before the optimization, resource utilization of VN #A and #B is 50%, and it of VN #C is 80%. For reducing utilization of VN #C, PN manager should allocate resources to VN #C from the pooled resources. If multiple VNs, whose utilization is high such as VN

```
flag = 0
if ((average_N(u_max) < ζ) and (v < v_th))
    ζ_prev = ζ
    ζ = bound + (ζ − bound)/2
    if (flag)
        finish()
else
    bound = ζ_prev
    ζ = ζ + (bound − ζ)/2
    flag = 1
end
```

Fig. 5 Pseudo code for binary search

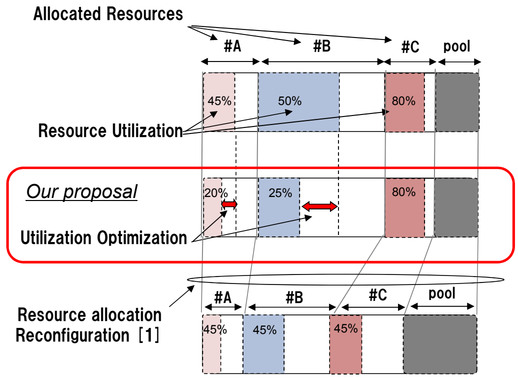


Fig. 6 Framework for applying the managed self-organizing network.

#C, exist, the pooled resources may be exhausted. On the other hand, with optimization, PN manager can know that VN #A and #B practically use their resources about quarter of allocated resources. Therefore, PN manager tears down the allocated resources from VN #A and #B, and such resources can be allocated for VN #C or pooled one. As a result, resource utilization of each VN becomes 50% without reduction of pooled resources. In other words, total required resources for managing each VN are reduced.

The dynamic threshold reconfiguration may be performed as the same cycle of resource rearrangement [1]. By performing the threshold reconfiguration before the resource rearrangement, PN manager can know the correct state of each VN. On the other hand, each VN is by ordinary controlled with only the current self-organization control mechanism. That is, injection of our algorithm to the managed self-organization framework does not disturb the one of advantages; it is basically controlled by the self-organization mechanism.

IV. Performance Evaluation

In this section, we demonstrate the effectiveness of our dynamic threshold reconfiguration algorithm through computer simulation.

A. Aims and Simulation Condition

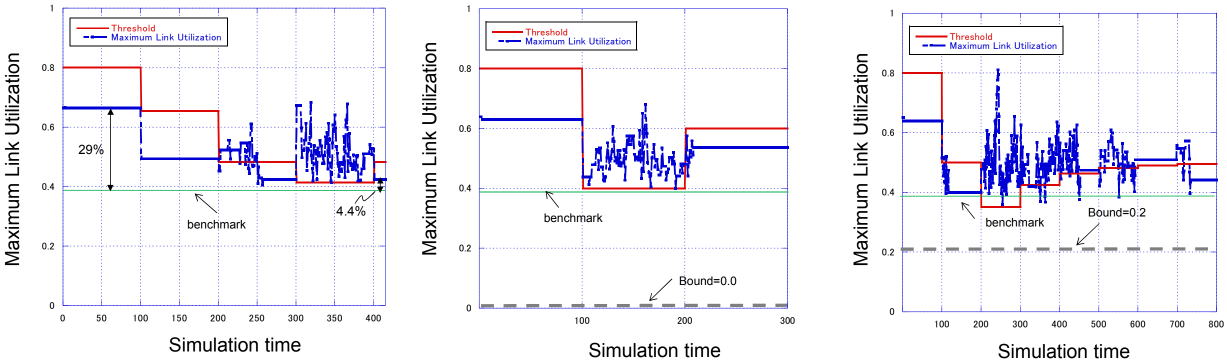
Our aim is to confirm the reduction effect of maximum link utilization with our dynamic threshold configuration. We compare our algorithm to the current attractor selection algorithm [10]. We also evaluate IMLTDA [6] method, which requires the traffic matrix, as benchmark of the nearly optimal solution. Evaluation indexes are maximum link utilization and a number of control cycles for the convergence. A physical network topology is European optical network (EON), whose number of nodes is 19 and number of links is 39 [3]. Traffic matrix model is in accordance with log-normal distribution [11]. Parameters for the attractor selection in Eqn. (1), (2) are set to $\theta = 0, \gamma = 1, \delta = 50, \mu = 20$. Number of attractor is about 50. Parameters for the dynamic threshold configuration are set to as follows; number of cycles $T = 100$, measurement times $N = 5$, minimum unit $m = 0.01$, initial threshold $\zeta = 0.8$, and threshold for maximum utilization variance $v_{th} = 0.001$. An initial border variable *bound* for binary search is set to 0 or 0.2. The attractor selection algorithm computes first VNT with random seed. Then we performed 30-times simulation with different random seed, and get the average, minimum, and maximum value of maximum link utilization and the number of cycles for convergence.

B. Reduction Effect of Maximum Link Utilization

Figure 7 shows the transition of maximum link utilization with different algorithms. While the performance difference between the current attractor selection-based algorithm and benchmark method is about 29%, with our algorithm reduces to about 4.4%; the attractor selection-based algorithm with our dynamic threshold configuration can be applied to the optimization problem.

In terms of reduction effect of maximum link utilization, the sequential search-based algorithm is the most efficient solution though the binary search is superior to the sequential one in the general search problem. This is because that an accurate lower bound is unknown In the VNT configuration problem. If we set the initial lower bound to enough lower value (e.g., *bound*=0.0), fluctuation range of threshold ζ becomes large. As a result, an extreme restriction is given to the problem; solution space is too small or too large. In Fig. 7 (b), *bound* is set to 0.0, and then the first changed threshold (ζ =0.4) is rather small, and the second changed threshold (ζ =0.6) is too large. As observation, rapid descent of threshold causes the unstable state of VNT, and then VNT converges under the loose restriction (ζ =0.6) in next solution search. If we increase the initial lower bound (*bound*=0.2), fluctuation range of threshold ζ becomes small (Fig. 7 (c)). However, the small fluctuation causes the slow recovery to the stable VNT when lower threshold is once set. In Fig. 7 (c), the second threshold change causes lower threshold about 0.35, which is lower than the benchmark value. Then binary search tries to increase threshold, but it is gradual. Then, the number of cycles for convergence increases. On the example of Fig. 7 (c), the number of cycles becomes eight.

Figure 8 shows the average, minimum, and maximum value of maximum link utilization and the number of cycles with 30-times simulation. In terms of average reduction of maximum link utilization, the sequential search gets the best solution. In addition, for the minimum value, algorithms excepting binary search (*bound*=0.0) get the value, which drops below the 0.4, and it is nearly equals to the solution with the benchmark. It also



(a) Sequential Search (b) Binary Search (bound=0.0) (c) Binary Search (bound=0.2)

Fig. 7 Evaluation Results (Numbers of cycles for convergence are (a) 5, (b) 3, (c) 8.

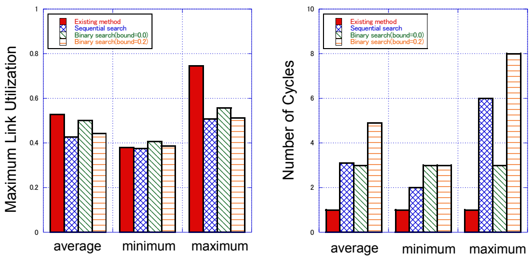


Fig. 8 maximum link utilization and number of cycles with 30-times simulation

means that attractor selection has potential to handle the optimization problem. On the other hand, the worst case with the current attractor selection algorithm causes the high maximum link utilization (about 0.78). That is, our dynamic threshold reconfiguration is essential for application to the optimization problem.

In terms of the number of cycles, binary search (*bound*=0.2) is the worst, and binary search (*bound*=0.0) is the best. The sequential search gets intermediate solution. As previously mentioned, fluctuation range of binary search (*bound*=0.0) becomes large. When the restriction for VNT configuration is relaxed, it rapidly converges. Then, this algorithm has applicability is for pruning the worst case with minimal iteration though it is not for the problem in this paper.

V. Concluding Remarks

This paper proposed a extension of current attractor-based algorithm to handle a network optimization problem. With our dynamic threshold reconfiguration algorithm, maximum link utilization is reduced about 25%. It also means that a flexibility for resource rearrangement by management server increases under the managed self-organization network. As future works, we will evaluate the effectiveness of our algorithm under the environment where multiple virtual networks exist.

Acknowledgement

This work was supported in part by the Strategic Information and Communications R&D Promotion Programme of the Ministry of Internal Affairs and Communications, Japan.

References

[1] T. Miyamura, S. Arakawa, Y. Ohsita, S. Kamamura, K. Shiomoto, M. Murata, “Dynamic Resource Allocation for Managed Self-Organization,” in Proceedings of APNOMS, Sep. 2011.

[2] K. Shiomoto, E. Oki, W. Imajuku, S. Okamoto, and N. Yamanaka, “Distributed virtual network topology control mechanism in GMPLS-based multiregion networks,” IEEE Journal on selected areas in communications, vol. 21, No. 8, pp. 1254-1262, Jan. 2003.

[3] 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, pp. 1720-1731, June 2010.

[4] 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, June 1996.

[5] A. Genc¸ata and B. Mukherjee, “Virtual-topology adaptation for WDM mesh networks under dynamic traffic,” IEEE/ACM Transactions on Networking, vol. 11, pp. 236–247, Apr. 2003.

[6] E. Leonardi, M. Mellia, and M. A. Marsan, “Algorithms for the logical topology design in WDM all-optical networks,” Optical Networks, vol. 1, pp. 35-46, 2000.

[7] Y. Zhang, M. Roughan, N. Duffield, A. Greenberg, “Fast Accurate Computation of Large-Scale IP Traffic Matrices from Link Loads,” in Proceedings of ACM SIGMETRICS 2003.

[8] T. Miyamura, S. Kamamura, K. Shiomoto, “Policy-based resource management in virtual network environment,” in proceedings of CNSM 2010 Oct. 2010.

[9] C. Furusawa and K. Kaneko, “A generic mechanism for adaptive growth rate regulation,” PLoS Computational Biology, vol. 4, p. e3, Jan. 2008.

[10] Y. Koizumi, T. Miyamura, S. Arakawa, K. Shiomoto, M. Murata, “Cooperation among Multiple Virtual Topologies Based on Attractor Superimposition,” in proceedings of ONDM 2010.

[11] A. Nucci, A. Sridharan, and N. Taft, “The Problem of Synthetically Generating IP Traffic Matrices: Initial Recommendations,” in proceedings of ACM/sigcomm Computer Communication Review, Vol. 35, Issue 3, Jul. 2005.