Master's Thesis

Title

Adaptive and autonomous placement method of virtualized network functions based on biochemical reactions

Supervisor

Professor Masayuki Murata

Author

Koki Sakata

February 9th, 2018

Department of Information Networking

Graduate School of Information Science and Technology

Osaka University

Master's Thesis

Adaptive and autonomous placement method
of virtualized network functions based on biochemical reactions

Koki Sakata

Abstract

Due to the wide and rapid spread of smartphones and tablets, and the development of Internet of Things (IoT), the types of network services have diversified and the traffic amount has also increased rapidly. If network equipment implemented on dedicated hardware is used to cope with such changes, cost of capital investment and operation becomes enormous.

Network Function Virtualization (NFV) is considered as one possible technique for resolving such problems. In NFV, various Virtual Network Functions (VNFs) are placed on general-purpose servers. Multiple VNFs may share the resource of one server or one VNF is distributed to multiple servers to provide services throughout the network. A flow receiving NFV service may have a service chaining request indicated in the order of VNFs to be applied to the flow. Therefore, in order to efficiently operate the NFV system, placement of VNFs to servers, resource allocation to each VNF, and flow routes, are determined carefully. In addition, in order to quickly respond to environmental fluctuations, and to maintain the scalability of services, it is desirable the NFV network services is operated in a distributed manner without operator's intervention. One way to realize such behavior is to exploit a biochemical mechanism with high autonomous dispersibility and self organization.

Our research group has proposed a construction method of service space in virtualized network system based on biochemical-inspired tuple space model. This method realizes distributed execution of multiple services according to user requests, autonomous sharing of server resources among services on each server, relocation of services to appropriate servers, and load balancing among surrounding servers in various situations. Since biochemical reaction equations are defined independently in each tuple space, it is suitable for realizing autonomous and decentralized behavior. However, in the past research, application of this method to specific network services has not been studied.

In this thesis, we apply the service space construction method to NFV system. We introduce chemical substances and biochemical reaction equations to support the context of NFV system. We then perform computer simulation experiments based on the τ-leaping method to clarify the effectiveness of the proposed method, and confirm that the functions required for the NFV system can be realized with the proposed method.

Furthermore, we present that various processing patterns required by the NFV system can be realized by parameter tuning of the proposed method. Specifically, we conduct the mathematical analysis of the proposed method to reveal its fundamental behavior and the effect of control parameters. We finally show the simulation results to exhibit that we can control the degree of the load balancing among multiple servers by tuning of a single parameter of the proposed method.

Keywords

Network Function Virtualization (NFV)
Virtual Network Function (VNF)
Biochemical Mechanism
Tuple Space Model
Parameter Tuning

Contents

List of Figures

List of Tables

1 Introduction

Due to the wide and rapid spread of smartphones and tablets, and the development of Internet of Things (IoT) [1], the number of devices connected to the network are increasing. As a result, the types of network services have diversified, and the traffic amount has also increased rapidly. If network equipment implemented on dedicated hardware is used to cope with such changes, new dedicated hardware is required to launch a new network service, causing an increase in initial installation cost and operational cost. In addition, the product life of such dedicated hardwares has been shortened, resulting in that the cost of capital investment becomes enormous. It also results in low flexibility to deal with environmental fluctuations such as system failures and sudden demand changes. Moreover, as the network becomes more complicated, it becomes difficult to allocate human resource to design and operate the network.

Network Function Virtualization (NFV) is considered as one possible technique for resolving such problems [2]. In NFV, network functions that have been realized with dedicated hardware such as Firewall [3], Deep Packet Inspection (DPI) [4], Network Address Translation (NAT) [5], Evolved Packet Core (EPC) [6], and so on, are executed as a software on general-purpose high-volume servers [7–10]. Such softwarelized functions are called Virtual Network Functions (VNFs). Figure 1 shows the network environment based on NFV. Hereafter, we call this environment a NFV system. In NFV, various VNFs are placed on general-purpose servers. Multiple VNFs may share the resource of one server or one VNF is distributed to multiple servers to provide services throughout the network. Consequently, it is possible to suppress operational cost and capital investment cost by aggregating physical devices. It is also possible to flexibly cope with the environment fluctuations by reallocating server resources to VNFs and rerouting flows.

A flow receiving NFV service may have a service chaining request indicated in the order of VNFs to be applied to the flow. As depicted in Figure 1, flows arriving at the NFV system receives services realized by VNFs according to the service chaining requests and exit the system. Therefore, in order to efficiently operate the NFV system, placement of VNFs to servers, resource allocation to each VNF, and flow routes, and so on, are determined carefully according to the service chaining requests, traffic amount of the flows and server resource distribution. The authors in [11], tackles this problem by using a context-free language for denoting complex composition of VNF. However, the proposed method in [11] requires the advance knowledge of the service

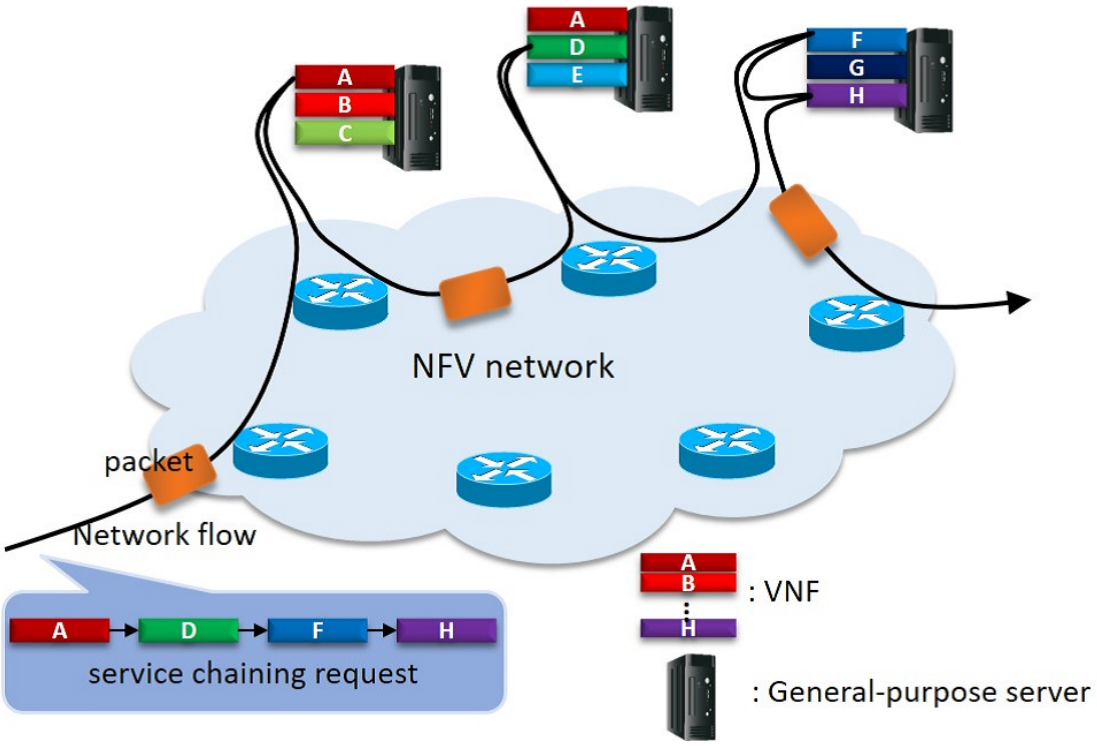


Figure 1: NFV system and service chaining

chaining requests of all flows. Since enormous calculation time is also required to obtain the result, it cannot be applied to environments where many flows with various service chaining requests arrives continuously. In general, in order to quickly respond to environmental fluctuations, and to maintain the scalability of services, centralized control system is not desirable and distributed control system is required [12]. One way to realize such behavior is to exploit a biochemical mechanism with high autonomous dispersibility and self organization [13–16].

Our research group has proposed a construction method of service space in virtualized network system based on biochemical-inspired tuple space model [17, 18]. In this method, in a virtual network system, a server is modeled as a tuple space, and service requests, service demands, and server resources are represented by chemical substances in the tuple space. The behavior of servers is described as biochemical reaction equations in the tuple space. Furthermore, by configuring a network by connecting multiple tuple spaces, movement and spread of services and requests in a network service environment composed of a number of servers can be described. This method realizes distributed execution of multiple services according to user requests, autonomous sharing of server resources among services on each server, relocation of services to appropriate servers, and load balancing among servers in various situations. Since biochemical reaction equations are defined and executed independently in each tuple space, it is suitable for realizing autonomous and decentralized behavior. However, in the past researches, the application of this method to specific network services such as NFV system has not been studied.

In this thesis, we apply the service space construction method to NFV system and clarify its effectiveness. Specifically, we exploit the method in order to adaptively, autonomously, and dispersively determine placement of VNFs on servers, resource allocation to each VNF, and flow routes, according to service chaining requests, the traffic amount of the flows and server resource distribution. We introduce chemical substances and biochemical reaction equations to support the context of NFV system. In particular, we describe various behaviors of NFV system, such as the execution of VNF to flows, the allocation of server resource to the flows, realization of service chaining, and coexistence of multiple VNFs on a single server. We then perform computer simulation experiments based on the τ-leaping method [19] to clarify the effectiveness of the proposed method, and confirm that the functions required for the NFV system can be realized with the proposed method.

Furthermore, we present that various processing patterns required by the NFV system can be

realized by parameter tuning of the proposed method. In detail, we show the load distribution on servers executing a VNF can be realized by adjusting a single control parameter of the proposed method. For that purpose, we conduct the mathematical analysis of the proposed method to reveal its fundamental behavior and the effect of control parameters. Based on the analysis results, we show the simulation results to exhibit that we can control the degree of the load balancing among multiple servers by tuning the parameter determining the biochemical reaction speed.

The rest of this thesis is organized as follows. In Section 2, we explain the tuple space model using biochemical reactions. In Section 3, we describe how to apply the model to NFV system. In Section 4, we show the simulation results to confirm the behavior of the proposed method. In Section 5, we show that parameter tuning method to control VNF services by mathematical analysis and computer simulation. Finally, in Section 6, we present a conclusion and some directions for future research.

2 Tuple Space Model using Biochemical Reactions

2.1 Tuple Space Model

A tuple space model is a model that implements a distributed system, proposed in [14,16]. Figure 2 depicts the tuple space model, where each component of a distributed system is modeled as a tuple space. Then, the state of the tuple space, that is, the information on the system component is expressed by the types and the quantity of the data records called tuples in the tuple space. A distributed system is controlled by performing operations such as "addition, deletion, movement of tuples" in the tuple space. The correspondence between the tuple space model and a distributed system is summarized in Table 1.

2.2 Biochemical Reactions in Tuple Space Model

A tuple space is defined as a cell where biochemical reactions take place. Then tuples in the tuple space correspond to chemical substances, the types of the tuples correspond to the types of chemical substances, and the amount of the tuples corresponds to the concentration of the chemical substances. By defining biochemical reaction equations in the tuple space and increasing/decreasing tuples using the result of reactions, it is possible to give regularity to increase/decrease of tuples, and thereby various behavior can be defined. A reaction rate of a biochemical reaction equation is determined by the product of the concentration of each reactant and the reaction rate coefficient defined in each biochemical reaction equation. For example, in the following reaction, if the concentrations of X and Y are x and y, respectively, the reaction rate is axy.

$$X \mid Y \xrightarrow{a} Z$$

Due to this property, when the concentration of the reactants in the biochemical reaction changes, the reaction rate autonomously changes. That is, by using the tuple space model using biochemical reactions, it is possible to perform autonomous control depending on the concentration of reactants. Figure 3 shows an image diagram of the tuple space model using biochemical reactions.

Table 1: Correspondence between tuple space model and distributed system

Distributed System	Tuple Space Model
System Component	Tuple Space
System Information	Types and Quantity of Tuples
System Control	Add, Delete and Move Tuples

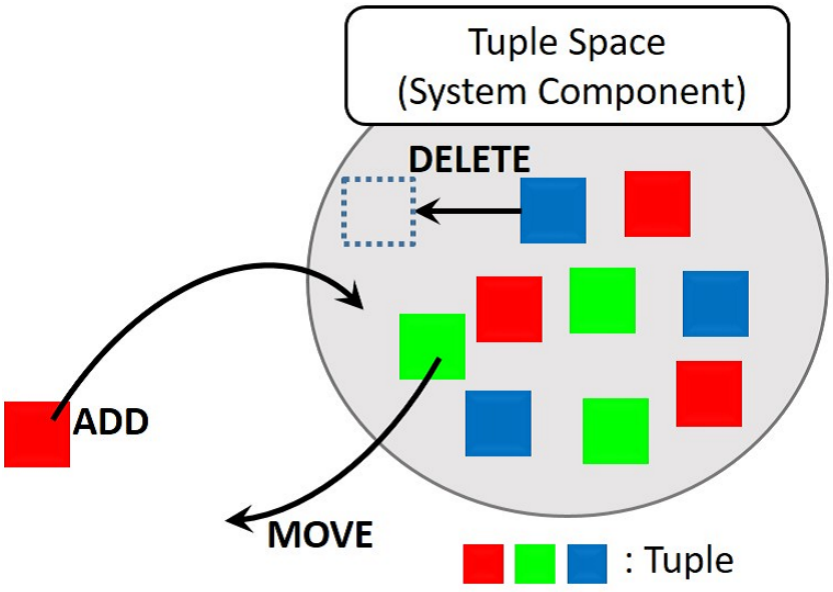


Figure 2: Tuple space model

2.3 Networked System with Tuple Space Model

Biochemical reaction mechanisms can include not only biochemical reactions within cells in organisms, but also biochemical reactions between cellular tissues in organisms. Mimicking this property, we construct a network by connecting multiple tuple spaces. By defining biochemical reaction equations describing diffusion and movement of tuples among tuple spaces, network behaviors can be described. Since biochemical reactions in each tuple space occur independently, we can describe autonomous decentralized behaviors in a networked system by using this model. Figure 4 shows an image diagram of the network model of tuple space using such biochemical reactions.

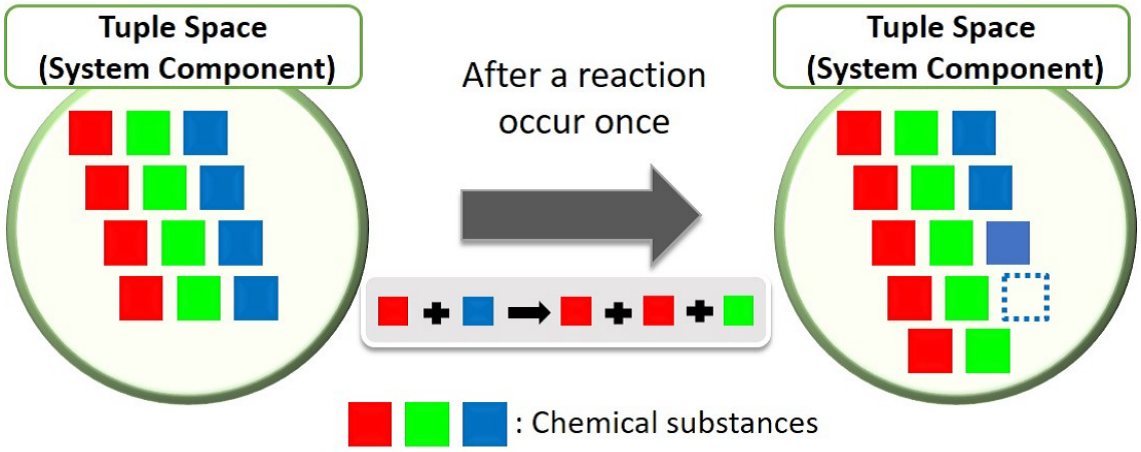


Figure 3: Biochemical reactions in tuple space

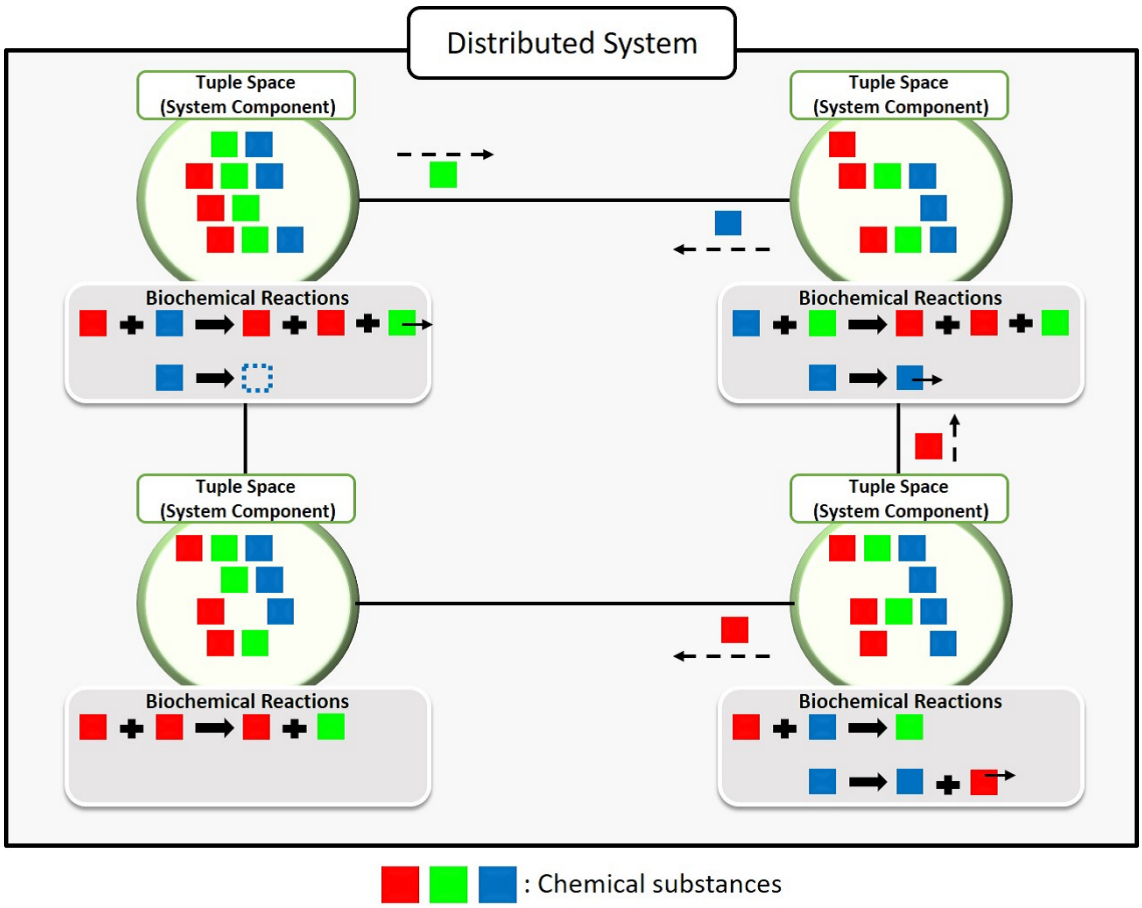


Figure 4: Networked tuple space

3 Application of Tuple Space Model using Biochemical Reactions to NFV System

In this section, we consider the application method of the tuples space model using biochemical reaction, described in Section 2 to NFV system. In particular, we define biochemical reaction equations to adaptively, autonomously, and dispersively determine placement of VNFs on the servers, the resource allocation to each VNF, and flow routes, according to service chaining requests, the traffic amount of the flows, and the amount of server resource. In this thesis, it is assumed that VNF is executed for individual packets constituting a flow.

3.1 Overview

Figure 5 depicts an image diagram of the application of tuple space model to the NFV system. A tuple space, which means a component of the distributed system, corresponds to a general-purpose server in the NFV system. VNFs, packets and server resources on the general-purpose servers are expressed by chemical substances in the tuple space. Then, behaviors such as execution of VNFs to the packets, resource allocation to each VNF, movement of packets, are described by biochemical reaction equations. Table 2 shows the correspondence between the tuple space model using biochemical reactions and the NFV system.

3.2 Service Function Chaining

The service chaining request c of a flow where $f_1, f_2, f_3..., f_{end}$ are VNFs to be executed to the flow is represented as follows.

$$c = \{f_1, f_2, f_3, ..., f_{end}\}$$

When VNF f_1 is executed to the flow with service chaining request c, c changes as follows.

$$c \leftarrow c \backslash \{f_1\} = \{f_2, f_3, ..., f_{end}\}$$

A VNF which is executed first in c is represented as $f^1(c)$. Hereafter, the subscript f of chemical substances represents VNFs, and subscript c represents service chaining requests.

Table 2: Correspondence between tuple space model and NFV system

Tuple Space Model	NFV System
Tuple Space	general-purpose server
Chemical Substance	Demand of VNF， Packet Flow， Server Resources, and so on
Biochemical Reactions	Execution of VNF to the packet, Server Resource Allocation to each VNF Move Packets in the Network, and so on

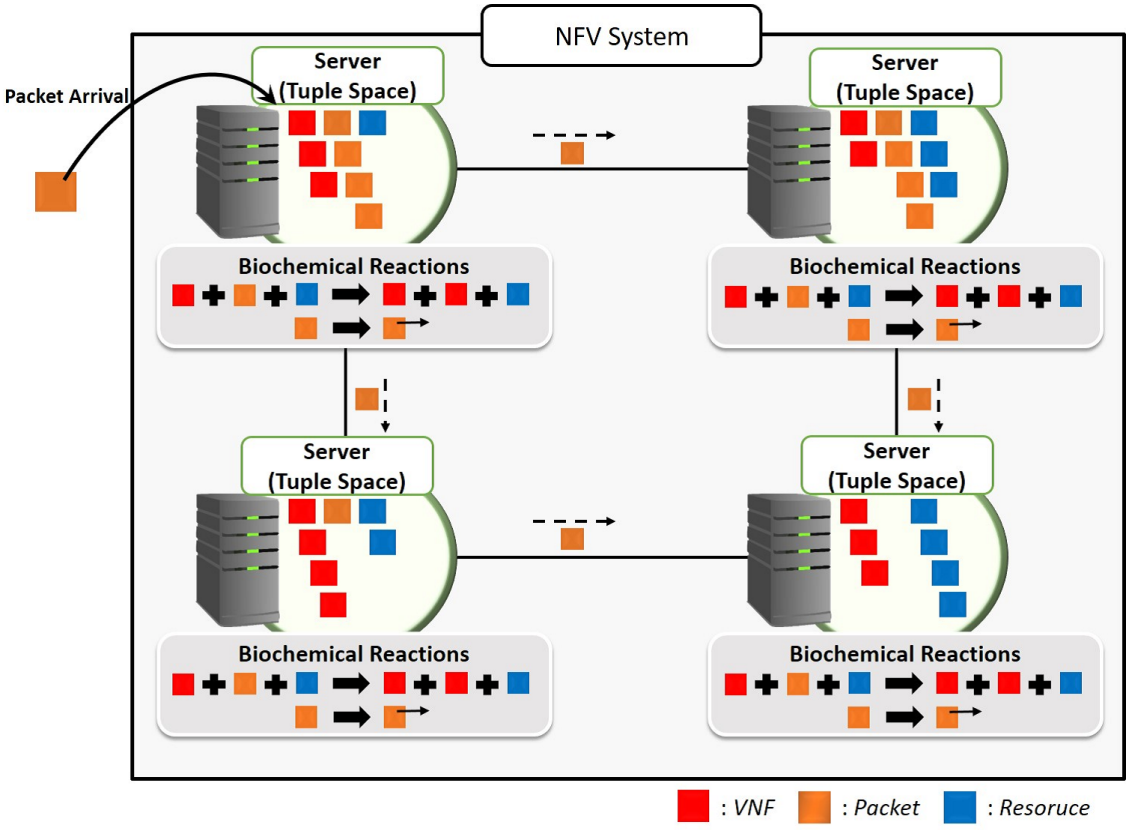


Figure 5: Application of tuple space model to NFV system

3.3 Application of VNFs to Packet Flow

We first explain biochemical reaction equations in a tuple space, that is, a server. It is desirable that placement of VNFs on servers and resource allocation to each VNF are determined according to the demand of VNFs. It is also required that VNFs with low demand has low priority in the server and VNFs with high demand would have high priority to be executed. When a VNF is executed on a packet of a flow whose service chaining request is composed of multiple VNFs, the service chaining request of the packet changes so that the executed VNF is removed. When the executed VNF is the last one in the service chaining request, the packet would disappear. Then, we consider the packet of the flow to which execution of all the VNFs has been completed disappear. These behaviors are realized by Reactions (1) and (2).

$$VNF_{f^1(c)}|PKT_c \rightarrow \begin{cases} VNF_{f^1(c)}|VNF_{f^1(c)}|PKT_{c\backslash\{f^1(c)\}} \\ |toserve(VNF_{f^1(c)}, PKT_c) \ \ (c\backslash\{f^1(c)\} \neq \emptyset) \\ \\ VNF_{f^1(c)}|VNF_{f^1(c)} \\ |toserve(VNF_{f^1(c)}, PKT_c) \ \ (c\backslash\{f^1(c)\} = \emptyset) \end{cases} \quad (1)$$

$$VNF_f \rightarrow 0 \quad (2)$$

In the above equations, substance VNF_f represents VNF f, indicating large value of the concentration of VNF_f means the execution of VNF f is highly demanded. The existence of VNF_f in a server means that the corresponding server can execute VNF f. Substance PKT_c represents a packet constituting a flow with service chaining request c. Substance $toserve(VNF_f, PKT_c)$ represents that VNF f has been executed to a packet of a flow with service chaining request c. Reaction (1) includes two behaviors in a server. One is that a VNF is executed to a packet of a flow on a server. Another is that the concentration of VNF increases as a result of the packet processing to represent the increase of the demand for the corresponding VNF. Also, when a VNF is executed to a packet of a flow, the service chaining request changes to $c\backslash\{f^1(c)\}$ so that remaining requests in its service chaining request are executed. On the other hand, PKT_c disappears when the packet has no remaining request in its service chaining request. Reaction (2) represents the decay of VNFs.

3.4 Limitation of Server Resource

The execution rate of a biochemical reaction is determined in proportion to the product of the concentration of each reactant of the reaction. According to this, for example, in Reaction (1), as the concentrations of VNF and PKT increase, the reaction rate increases without limitation. However, in the actual server, the execution speed of a VNF is limited by resources such as CPU capacity and memory size. Therefore, using only Reaction (1) is not suitable for describing the behavior of the actual NFV system. In the proposed method, this problem is resolved by introducing enzyme-catalytic reactions, one of biochemical reactions [20, 21]. In enzyme-catalyzed reactions, the reaction rate can be controlled by the concentration of the enzyme which does not affect the reaction itself [22]. The basic equation of the enzyme-catalyzed reaction is shown in Equation (3), where E is an enzyme, S is a substrate, ES is an enzyme-substrate complex, and P is a product.

$$E + S \leftrightarrows ES \rightarrow E + P \tag{3}$$

By extending the original reaction, Reaction (1) in Reactions (4) and (5), we can describe constraints of the execution speed of VNFs by the server resource.

$$RSRC|VNF_f \leftrightarrows RS\_VNF_f \tag{4}$$

$$RS\_VNF_{f^1(c)}|PKT_c \quad \leftrightarrows \quad MEDIATE_c \rightarrow \begin{cases} VNF_{f^1(c)}|VNF_{f^1(c)}|PKT_{c\backslash\{f^1(c)\}}|RSRC|toserve(VNF_{f^1(c)}, PKT_c) \\ (if c\backslash\{f^1(c)\} \neq \emptyset) \\ \\ VNF_{f^1(c)}|VNF_{f^1(c)}|RSRC|toserve(VNF_{f^1(c)}, PKT_c) \\ (if c\backslash\{f^1(c)\} = \emptyset) \end{cases} \tag{5}$$

The concentration of substance $RSRC$, RS_VNF_f, and $MEDIATE_c$ respectively represent the amount of available server resources, the amount of server resources allocated to VNF f, and the amount of server resources allocated to the packets constituting the flow with service chaining c. Figure 6 shows an image diagram of the application of enzyme-catalyzed reaction to NFV system.

Note that these equations are defined for each VNF in a server. This means that when multiple VNFs coexist on one server, server resource is shared among them according to the demand of each VNF.

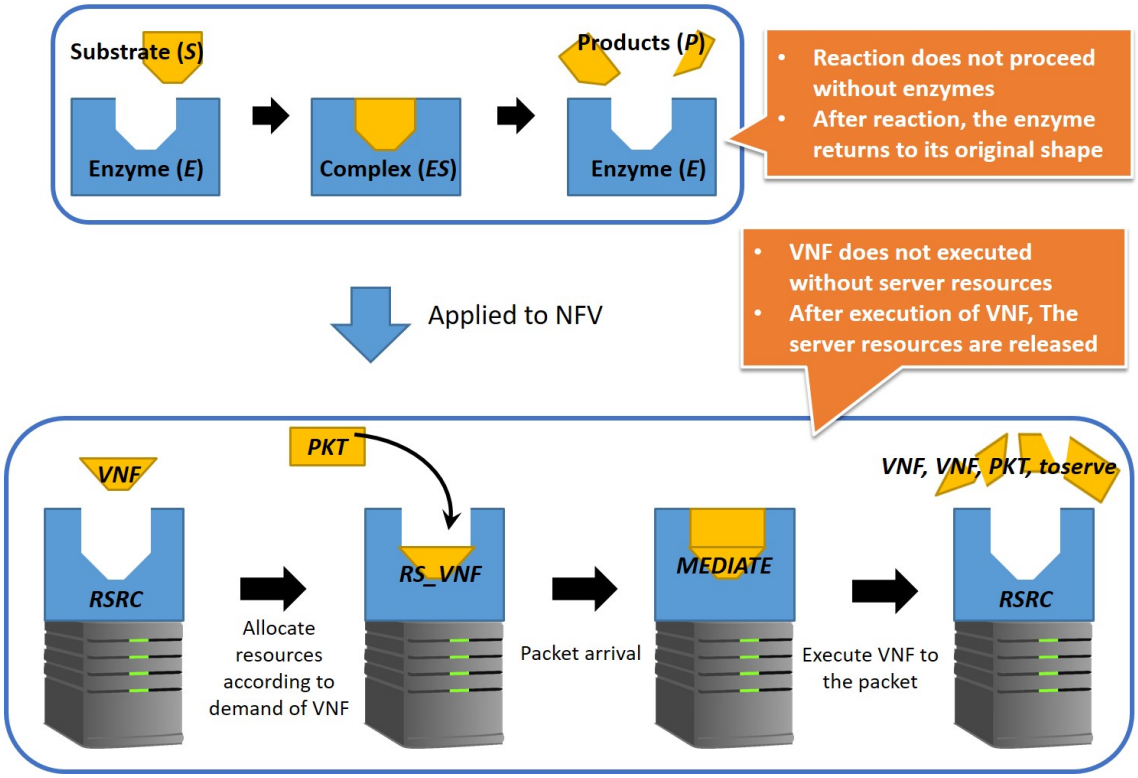


Figure 6: Application of enzyme-catalyzed reactions to NFV system

3.5 Diffusion of VNFs

In order to describe the NFV system, composed of multiple general-purpose servers, we configure a network with connecting multiple tuple spaces. Then, we can describe movement of tuples between tuple spaces. Diffusion of VNFs to other interconnected servers in the NFV system is described by introducing Reaction (6).

$$VNF_f \quad \rightarrow \quad VNF_f^{\rightsquigarrow} \tag{6}$$

By this reaction, substance VNF is diffused to the surrounding servers at a rate proportional to its concentration. When the destination server of the diffusion has packets for the VNF, they are processed according to Reactions (4) and (5), resulting the evolution of the VNF at the server. On the other hand, when the destination server has no corresponding packet, the VNF would decay according to Reaction (2). These behavior means that the diffused VNFs grow only at the servers with the demand for the VNFs.

3.6 Routing Packet Flow

When packets remains unprocessed while the corresponding VNF cannot be executed on a server, these packets should be routed to other servers which can process the corresponding VNF. It is desirable that the destination of these packets should be servers which have plenty of remaining server resources and the corresponding VNFs can be executed. We exploit a gradient field to determine the moving direction of packets in the networked tuple spaces. We construct a gradient field based on the demand of VNF and the available resource on each server. We then determine flow routes according to the gradient field and forward packets. Reactions (7)-(11) are introduced to realize such behaviors.

$$VNF_f \,|\, RSRC \quad \rightarrow \quad VNF_f \,|\, RSRC \,|\, GRAD_f \tag{7}$$

$$VNF_f \,|\, RS\_VNF_f \quad \rightarrow \quad VNF_f \,|\, RS\_VNF_f \,|\, GRAD_f \tag{8}$$

$$GRAD_f \quad \rightarrow \quad 0 \tag{9}$$

$$GRAD_f \quad \rightarrow \quad GRAD_f^{\rightsquigarrow}(GRAD_f^{-}) \tag{10}$$

$$PKT_c \quad \rightarrow \quad PKT_c^{\rightsquigarrow}(GRAD_f^{+}) \tag{11}$$

$GRAD$ is a substance for establishing a gradient field in the NFV system. Reactions (7), (8) mean that $GRAD$ is generated at a rate proportional to the concentration of VNF, $RSRC$, and RS_VNF. Reaction (9) means that $GRAD$ decays at a rate proportional to its concentration. Reaction (10) means that $GRAD$ spreads to surrounding servers with less concentrations of $GRAD$. Therefore, the gradient field is constructed so that the server providing the VNF becomes a summit with the largest concentrations of $GRAD$, and surrounding servers have smaller concentrations of $GRAD$ according to the distance (e.g. hop count) from the summit. Reaction (11) describes the movement of PKT to surrounding servers with larger concentrations of $GRAD$.

When there are multiple of surrounding servers with gradients larger than a serve, the movement direction of substance PKT is determined at a probability proportional to the "slope" to each surrounding server. Specifically, we utilize Equation (12) for determining S_i, which is the speed at which PKT moves to server i.

$$S_i = PKT \cdot r_{mp} \cdot \frac{GRAD_i}{GRAD_{SUM}} \tag{12}$$

where $GRAD_i$ means the concentration of $GRAD$ at server i, $GRAD_{SUM}$ means the sum of the concentrations of $GRAD$ at surrounding servers with higher concentration, and r_{mp} means the reaction rate coefficient of Equation (11). Figure 7 depicts the movement of packets of a flow with service chaining request $c = \{f_0, f_1, f_2\}$.

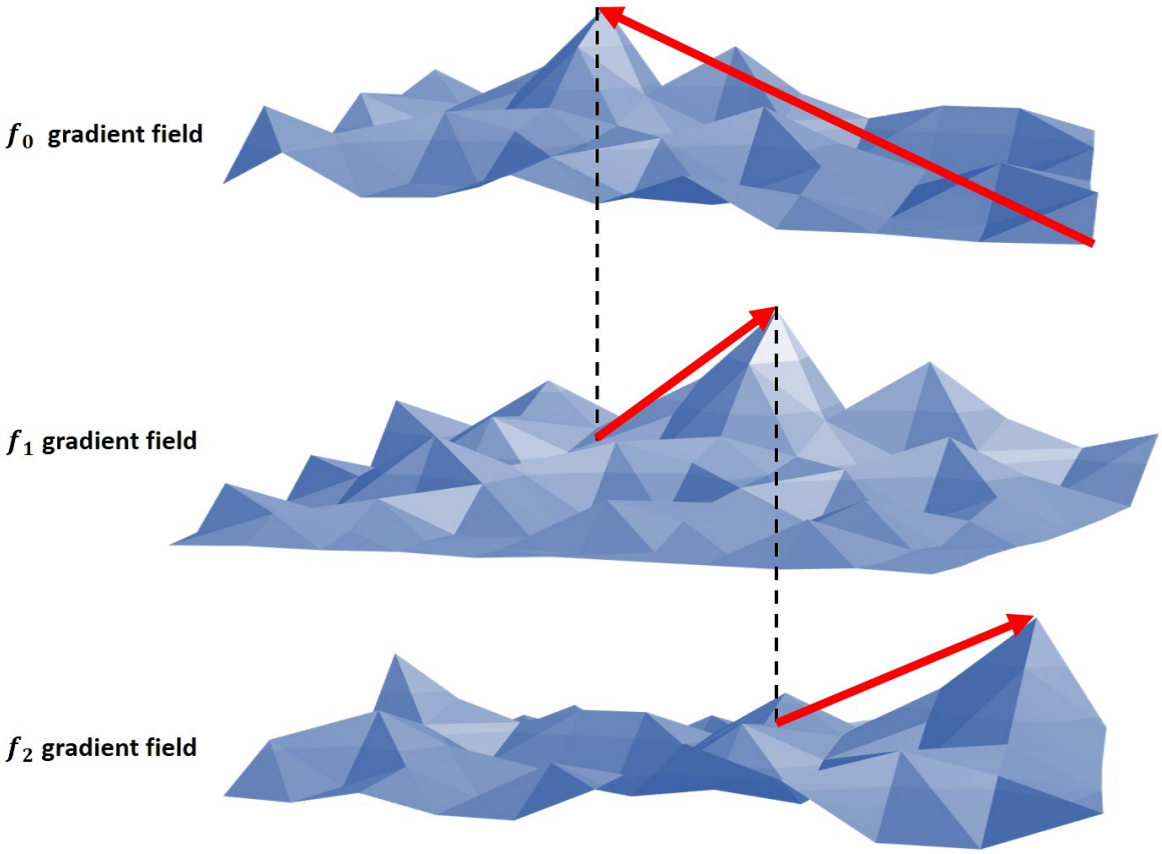


Figure 7: Movement of packet according to the gradient fields

3.7 Summary

We summarize all reaction equations in the NFV system in Reactions (13)-(21), where F is the set of all VNFs provided by the NFV system, C is the set of all service chaining requests that may exist in the NFV system, and $v_1, v_2, u_1, u_2, w, d, r_{mf}, r_{rg}, r_{dg}, r_{mg}$, and r_{mp} are the reaction rate coefficients for Reactions (13) – (21), respectively.

$$\forall f \in F,\ RSRC|VNF_f \underset{u_1}{\overset{v_1}{\leftrightarrows}} RS\_VNF_f \tag{13}$$

$$\forall c \in C,\ RS\_VNF_{f^1(c)}|PKT_c \underset{u_2}{\overset{v_2}{\leftrightarrows}} MEDIATE_c \xrightarrow{w} \begin{cases} VNF_{f^1(c)}|VNF_{f^1(c)}|PKT_{c\backslash\{f^1(c)\}}|RSRC|toserve(VNF_{f^1(c)}, PKT_c) \\ (ifc\backslash\{f^1(c)\} \neq \emptyset) \\ \\ VNF_{f^1(c)}|VNF_{f^1(c)}|RSRC|toserve(VNF_{f^1(c)}, PKT_c) \\ (ifc\backslash\{f^1(c)\} = \emptyset) \end{cases} \tag{14}$$

$$\forall f \in F,\ VNF \xrightarrow{d} 0 \tag{15}$$

$$\forall f \in F,\ VNF_f \xrightarrow{r_{mf}} VNF_f^{\rightsquigarrow} \tag{16}$$

$$\forall f \in F,\ VNF_f \,|\, RSRC \xrightarrow{r_{rg}} VNF_f \,|\, RSRC \,|\, GRAD_f \tag{17}$$

$$\forall f \in F,\ VNF_f \,|\, RS\_VNF_f \xrightarrow{r_{rg}} VNF_f \,|\, RS\_VNF_f \,|\, GRAD_f \tag{18}$$

$$\forall f \in F,\ GRAD_f \xrightarrow{r_{dg}} 0 \tag{19}$$

$$\forall f \in F,\ GRAD_f \xrightarrow{r_{mg}} GRAD_f^{\rightsquigarrow}(GRAD_f^-) \tag{20}$$

$$\forall c \in C,\ PKT_c \xrightarrow{r_{mp}} PKT_c^{\rightsquigarrow}(GRAD_f^+) \tag{21}$$

4 Simulation Experiments

In this section, we perform computer simulation experiments to clarify that by the biochemical reactions defined in section 3, it is possible to adaptively, autonomously, and dispersively decide placement of VNFs on servers, the resource allocation to each VNF, and flow routes, according to service chaining requests, the traffic amount of the flows and server resource distribution in the NFV system.

4.1 τ-Leaping Method

In order to simulate the proposed method, we exploit τ-leaping method in [19, 23], which is one of stochastic simulation algorithms that can capture the inherent stochasticity in many biochemical systems such as the chemically reacting system and the cell system. We briefly explain the procedures of τ-leaping algorithm for a server in the proposed method as follows.

(1) Set τ for the time step of the simulation

(2) Calculate the reaction rates of biochemical reactions by the product of reactants concentration and reaction rate coefficients

(3) Generate a Poisson random variable for each chemical reaction whose mean is the product of corresponding reaction rate and time τ

(4) Execute biochemical reactions at the number of times derived in (3), and update the concentration of each substance

(5) Progress simulation time by τ

(6) Repeat (2) - (5) until the simulation end time

As the value of τ increases, the simulation can proceed faster while the results becomes different from the actual behavior. So τ should be carefully chosen. There is an algorithm that decides the optimal τ [24], but it is complicated, takes time to implement, and takes a long calculation time when executing the program. Therefore, we perform some simulation experiments with some values of τ and determined that τ is 0.6 msec and simulation time is one second.

4.2 Network Topology for simulation experiments

The network topology used for simulation experiments is shown in Figure 8. The topology consists of three nodes and three links, and each node is called node 0-2 as depicted in the figure. Although we have confirmed the effectiveness of the method in a larger topology, we use this simple topology to ease the explanation. The propagation delay time of the link is ignored to confirm the fundamental behavior of the propose method. We assume that each node has one server accommodating VNFs.

4.3 Parameter Settings

The concentration at the start of simulation of each chemical substance in Reactions (13)-(21) is determined as follows. We set the initial values of the concentration of VNF_{f_0} of node 1 and the concentration of VNF_{f_1} of node 2 to 2,000. All node has 1,000 of the initial concentration of $RSRC$. Assumed that a VNF is executed to each packet of a flow, the initial concentration of $RSRC$ corresponds to the capacity of executing VNFs at roughly 83 Kpps, which becomes roughly 1 Gbps with packet size of 1,500 Bytes. The initial value of the concentrations of other chemical substances are set to 0.

Unless otherwise specified, the reaction rate coefficients of Reactions (13)-(21) is determined as $r_{u1} = 0.0003,\ r_{v1} = 0.278,\ r_{u2} = 0.1,\ r_{v2} = 0.001,\ r_w = 0.05,\ r_d = 0.01,\ r_{mf} = 0.003,$ $r_{rg} = 0.0001,\ r_{dg} = 0.03,\ r_{mg} = 0.005,\ r_{mp} = 0.3.$

In this simulation, in the NFV system, all VNFs provided are VNF f_0 and VNF f_1, and all service chaining requests that may exist are $c_0 = \{f_0, f_1\}$ and $c_1 = \{f_1\}$.

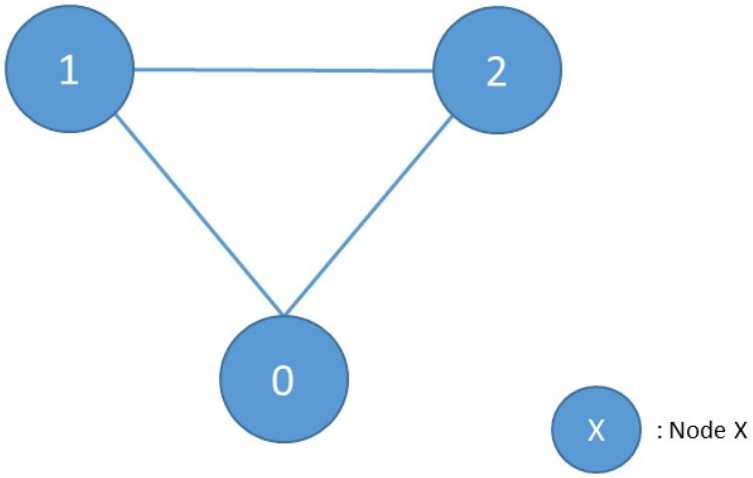


Figure 8: Network topology for simulation experiments

4.4 Simulation Results and Discussion

We perform computer simulation experiments based on three scenarios and show and discuss the results to clarify the effectiveness of the proposed method.

4.4.1 Scenario 1: Decision of Packet Flow Route

In this scenario, the packets of the flow with service chaining request c_0 arrive at node 0 at 10 packets per time step, corresponding to 16.7 Kpps. By setting the reaction rate coefficient r_{mf} of Reaction (5) to 0, we prevent VNF from diffusing. Figure 9 depicts the simulation environment of this scenario. By this scenario, we show that gradient field is configured based on the demand of VNF and the available resource on each server, packets are forwarded to the appropriate node providing the VNF, and VNFs are executed according to the service chaining requests.

Figure 10 shows the temporal change in the concentration of $GRAD$ at each node. From this figure, it can be seen the gradient field of $GRAD_{f_0}$, which is the gradient substance of VNF_{f_0}, is formed with node 1 as the summit, and the gradient field of $GRAD_{f_1}$, which is the gradient substance of VNF_{f_1}, is formed with node 2 as the summit. Figure 11 shows the average number of $toserve$ generated at each node. From this figure, it can be seen that VNF f_0 is executed on node 1,and VNF f_1 is executed on node 2. From these results, it is clarified the packets of the flow are forwarded to the appropriate note providing corresponding VNFs and the VNFs are executed to the packets of the flow.

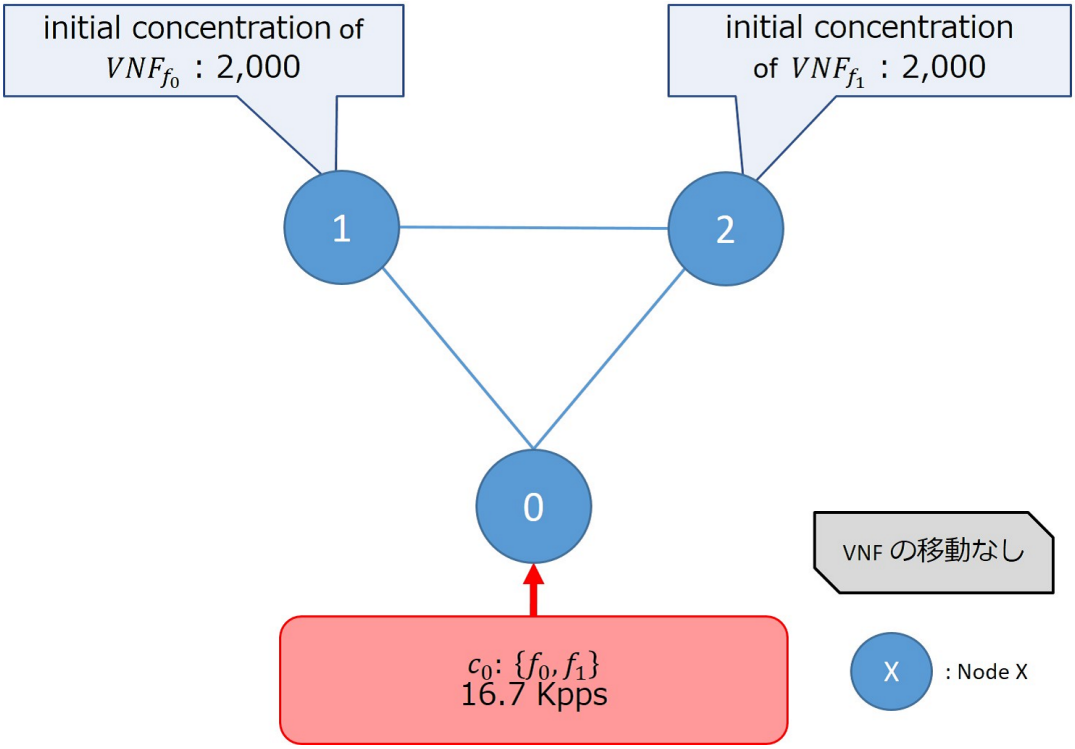
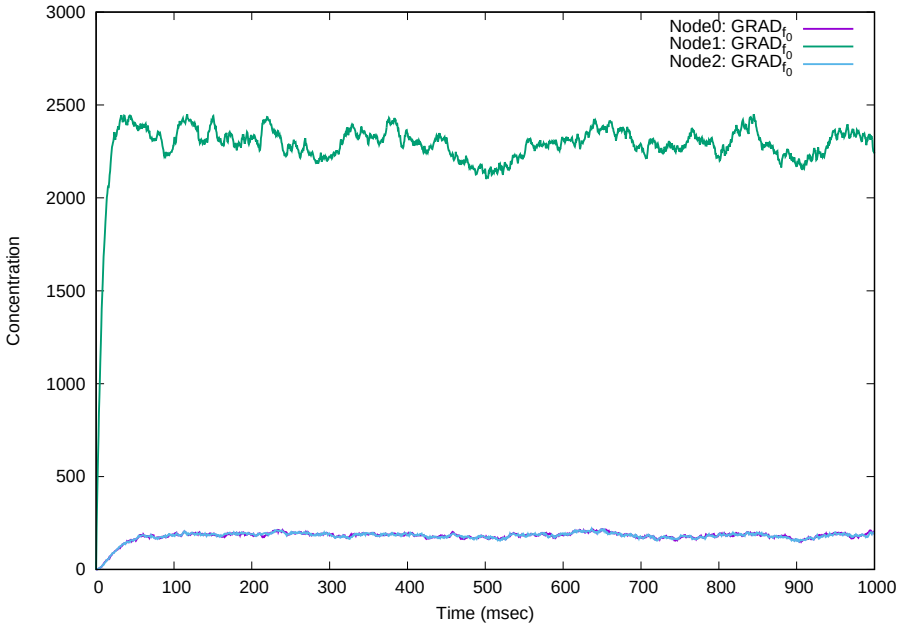
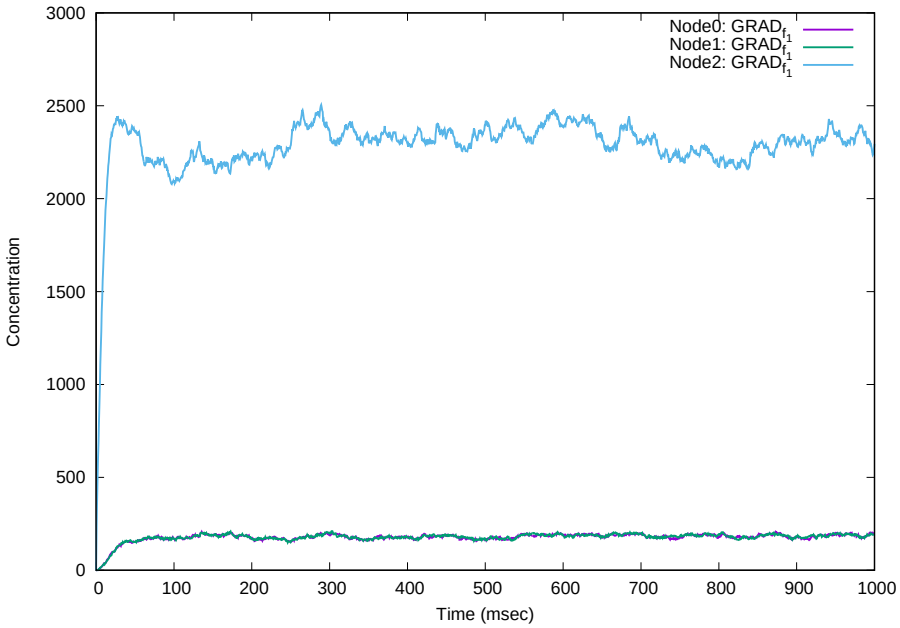


Figure 9: Scenario 1: Simulation environment

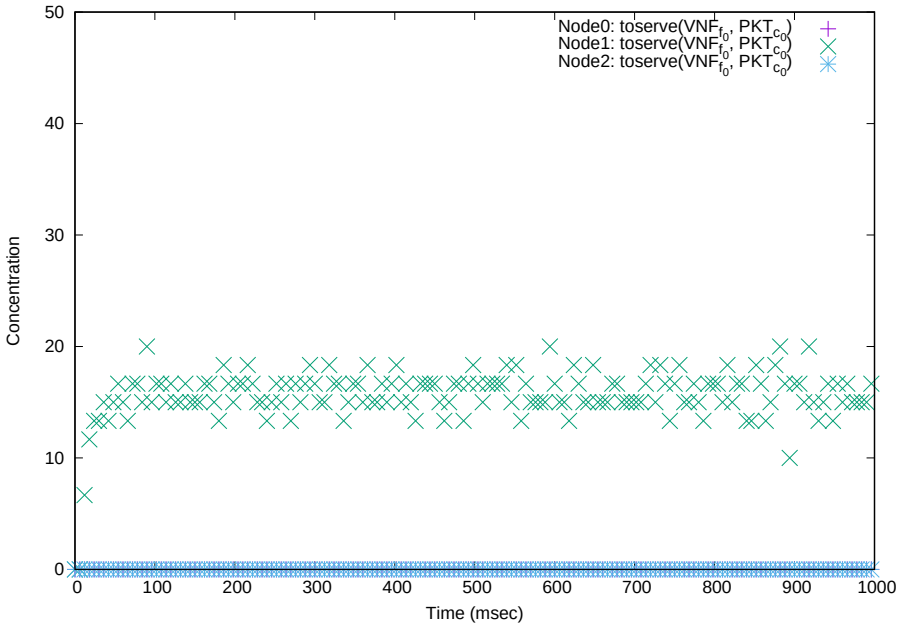


(a) $GRAD_{f_0}$

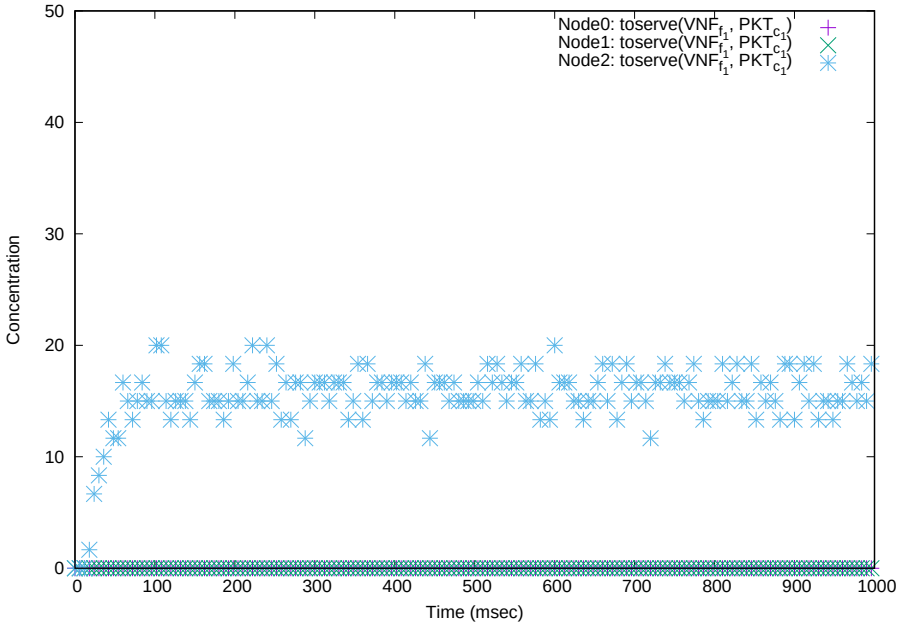


(b) $GRAD_{f_1}$

Figure 10: Scenario 1: Temporal change in the concentration of $GRAD$



(a) $toserve(VNF_{f_0}, PKT_{c_0})$



(b) $toserve(VNF_{f_1}, PKT_{c_1})$

Figure 11: Scenario 1: Average number of $toserve$

4.4.2 Scenario 2: Placement and Resource Allocation of VNFs

In this scenario, the packets of the flow with service chaining request c_0 arrive at node 0 at 10 packets per time step. Unlike Scenario 1, where the reaction rate coefficient r_{mf} of Reaction (5) is set to 0, the VNF diffusion occurs in this scenario. Figure 12 shows the simulation environment of this scenario. By this scenario, we exhibit the relocation of the VNF to an appropriate node and resource sharing among VNFs.

Figure 13 shows the temporal change in the concentration of VNF at each node. This figure shows that VNF_{f_0} and VNF_{f_1} is moving from node 1 and 2, respectively, to node 0 where the flow packets arrive. This is because when the VNF diffuses, the VNF grows on the node because packets requesting the VNF arrives. Figure 14 shows the average number of $toserve$ generated at node 0, and Figure 15 shows the temporal change in the concentration of $GRAD$ at each node. Immediately after the start of the simulation, $GRAD_{f_0}$ forms the gradient field with node 1 as the summit, and $GRAD_{f_1}$ forms the gradient field with node 2 as the summit. Then, the packets arriving at node 0 are forwarded to node 1, VNF f_0 is applied, forwarded to node 2, and executed VNF f_1 is applied. As time passes, however along with the movement of VNF_{f_0} and VNF_{f_1}, node 0 becomes the summit of both gradient fields. Consequently, both VNFs are at node 0. Figure 16 shows the temporal change in the concentration of resource-related substances at each node. From the figure, it is confirmed that resource sharing between VNF f_0 and VNF f_1 is performed appropriately at node 0. From the above results, it is clarified that the VNF is relocated to appropriate node and the server resource is shared among the VNFs.

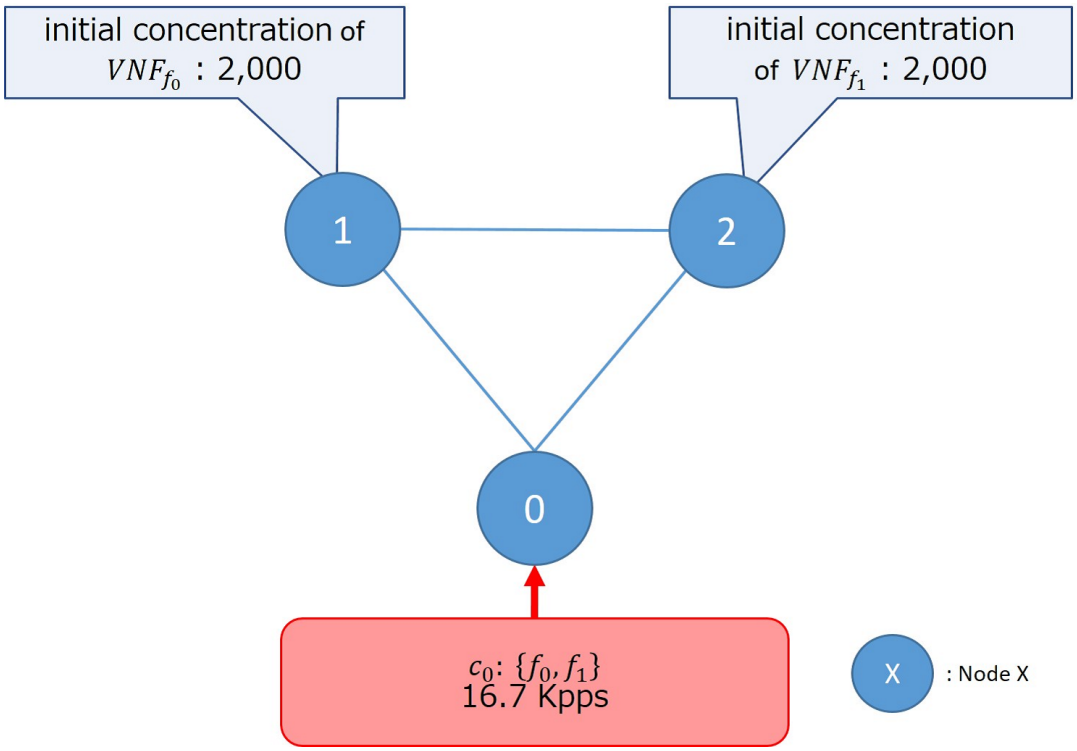
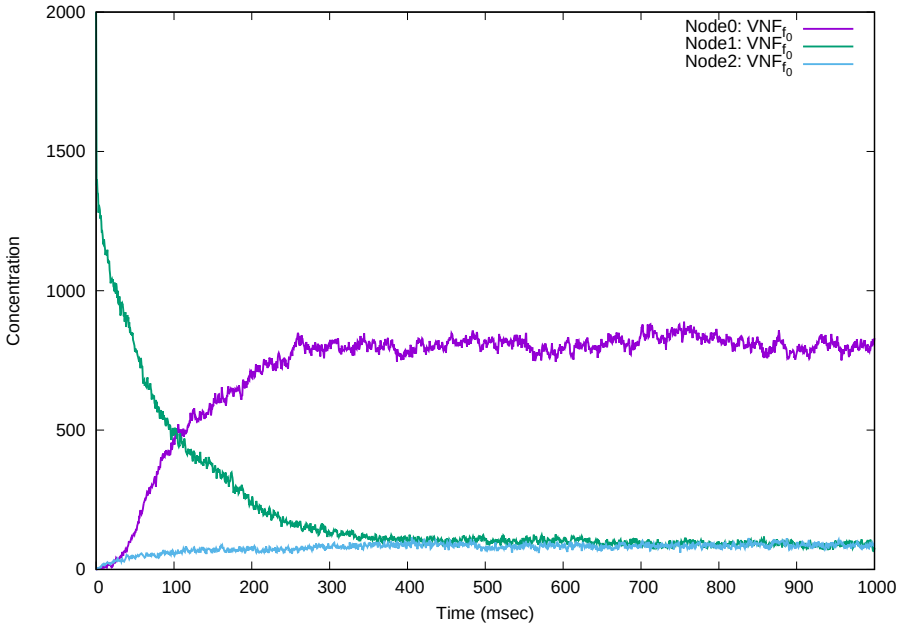
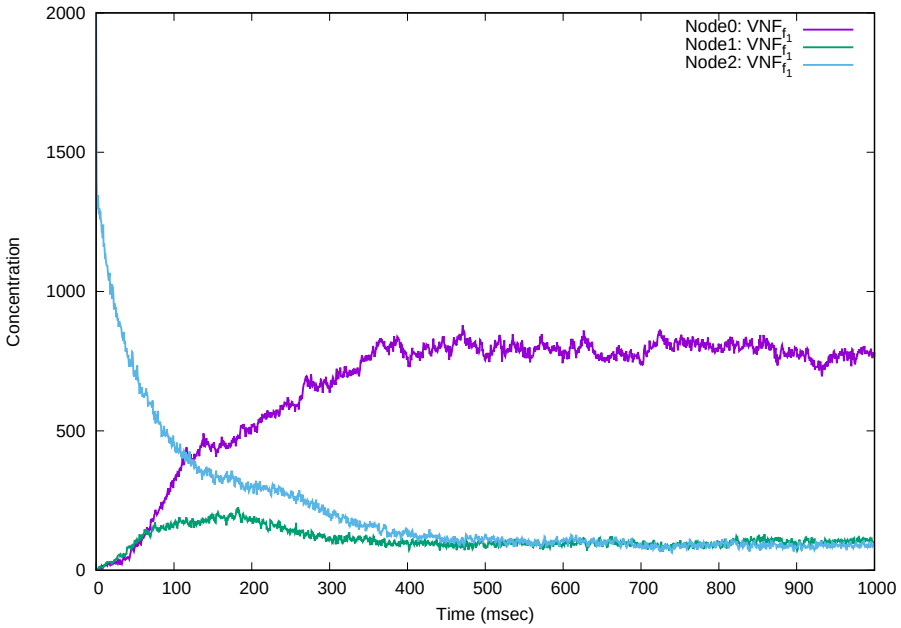


Figure 12: Scenario 2: Simulation environment

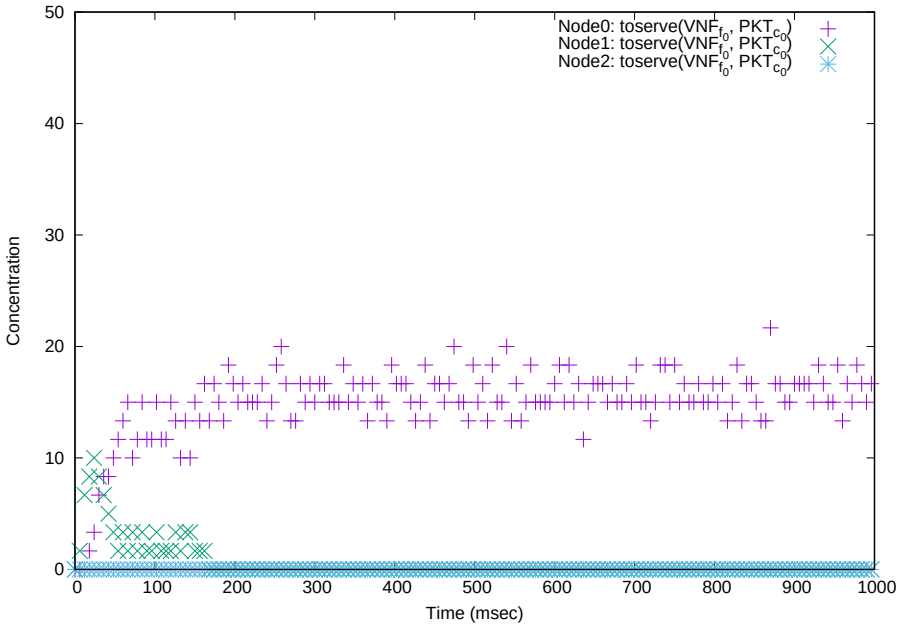


(a) VNF_{f_0}

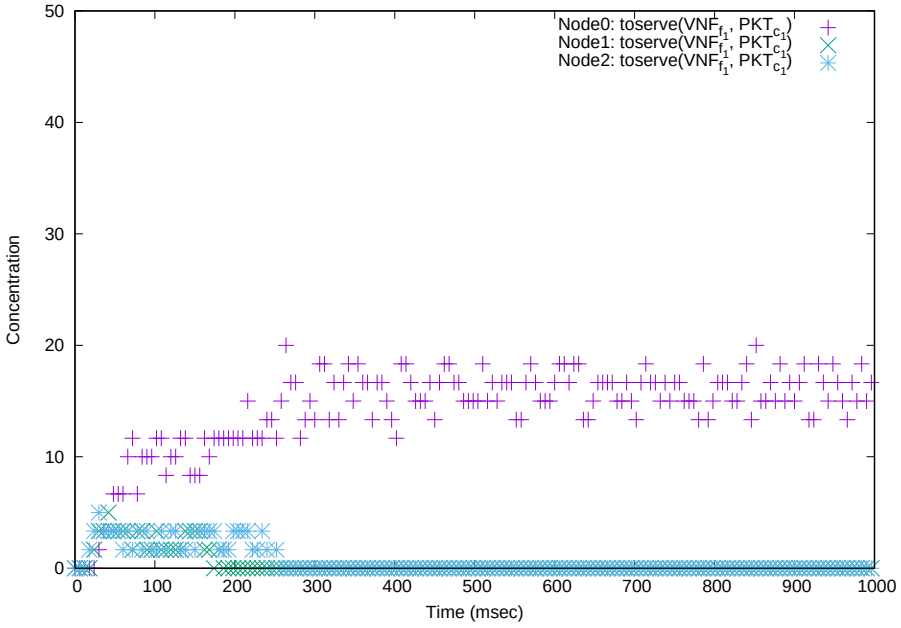


(b) VNF_{f_1}

Figure 13: Scenario 2: Temporal change in the concentration of VNF

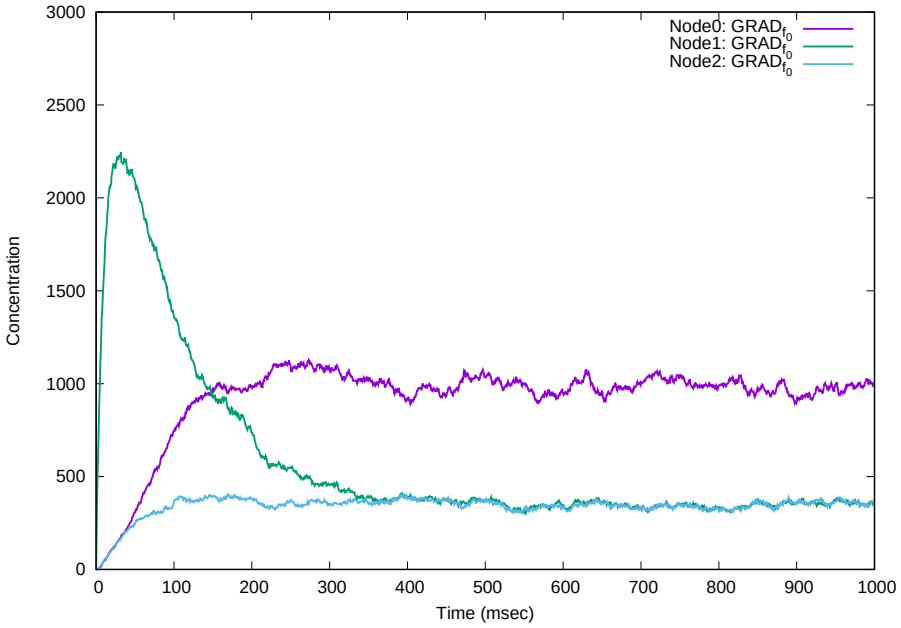


(a) $toserve(VNF_{f_0}, PKT_{c_0})$

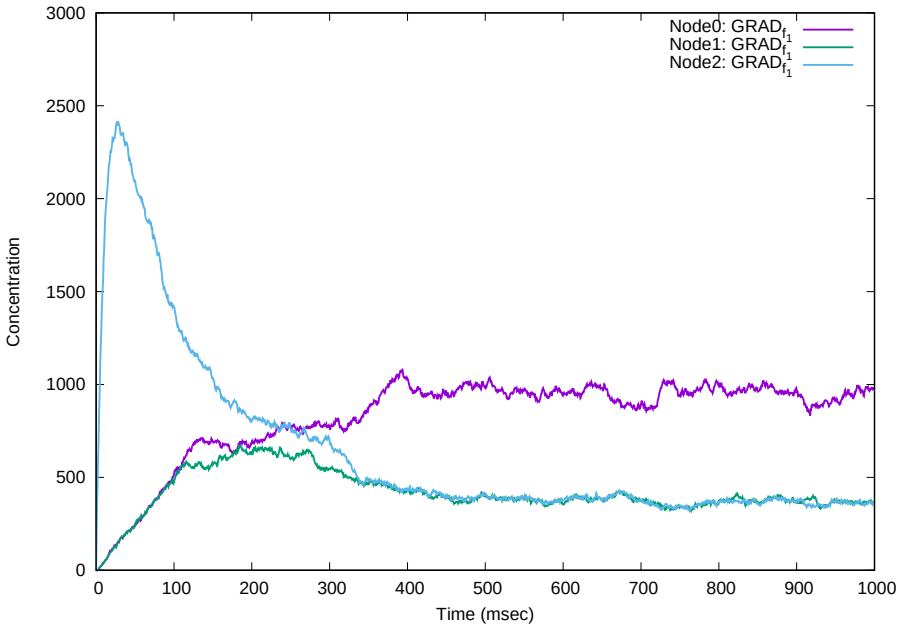


(b) $toserve(VNF_{f_1}, PKT_{c_1})$

Figure 14: Scenario 2: Average number of $toserve$ at node 0



(a) $GRAD_{f_0}$



(b) $GRAD_{f_1}$

Figure 15: Scenario 2: Temporal change in the concentration of $GRAD$

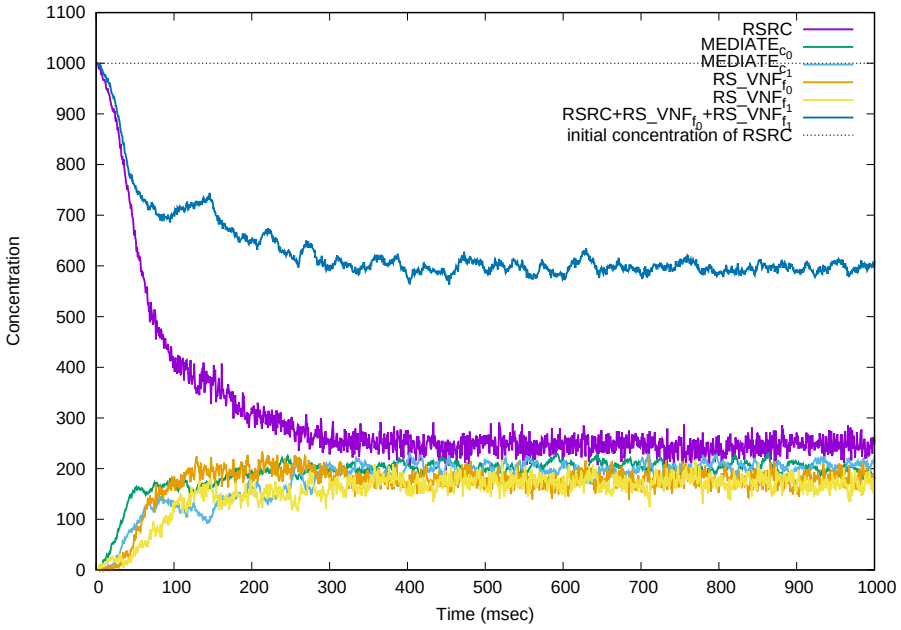


Figure 16: Scenario 2: Temporal change in the concentration of resource-related substances

4.4.3 Scenario 3: Distributed Execution of VNFs

In this scenario, the packets of the flow with service chaining request c_0 arrive at node 0 at 30 packets per time step. This means all packets cannot be executed at one server, and distributed execution is required. Figure 17 shows the simulation environment.

Figure 18 shows the average number of $toserve$ generated at each node. This result exhibits that VNFs are executed to the packets at all nodes, and the execution rate of VNFs of node 0 is highest since the packets arrive at node 0. This is because the packets that can not be processed at node 0 move to nodes 1 and 2 and the VNF is executed to the packets.

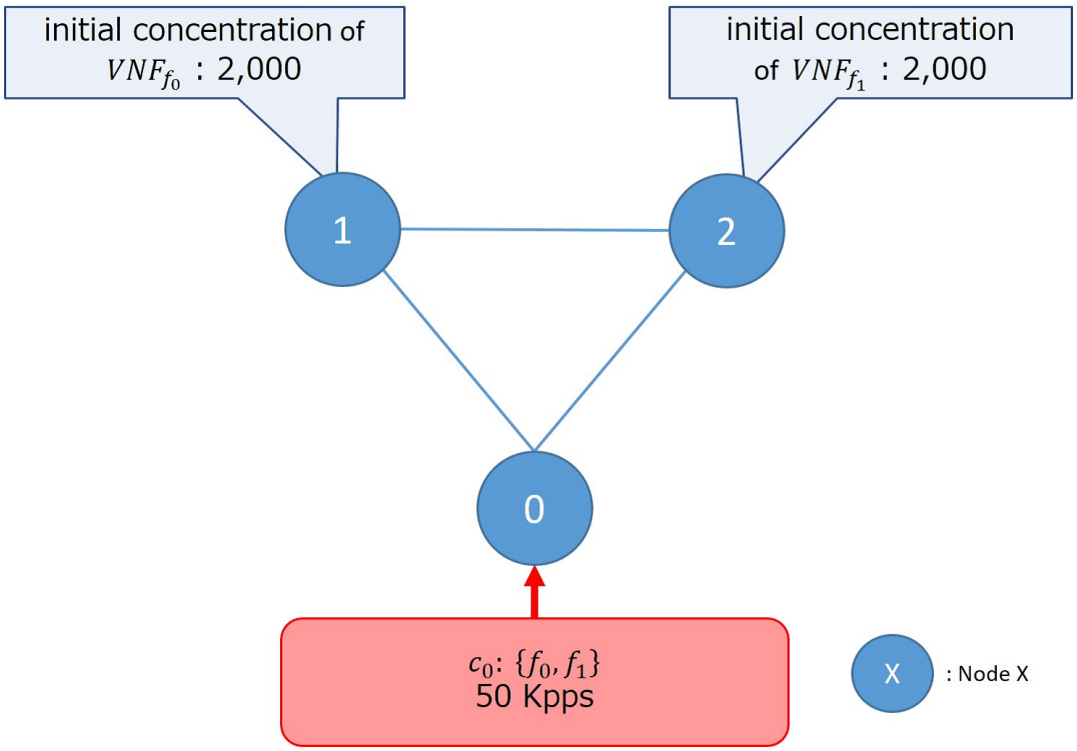
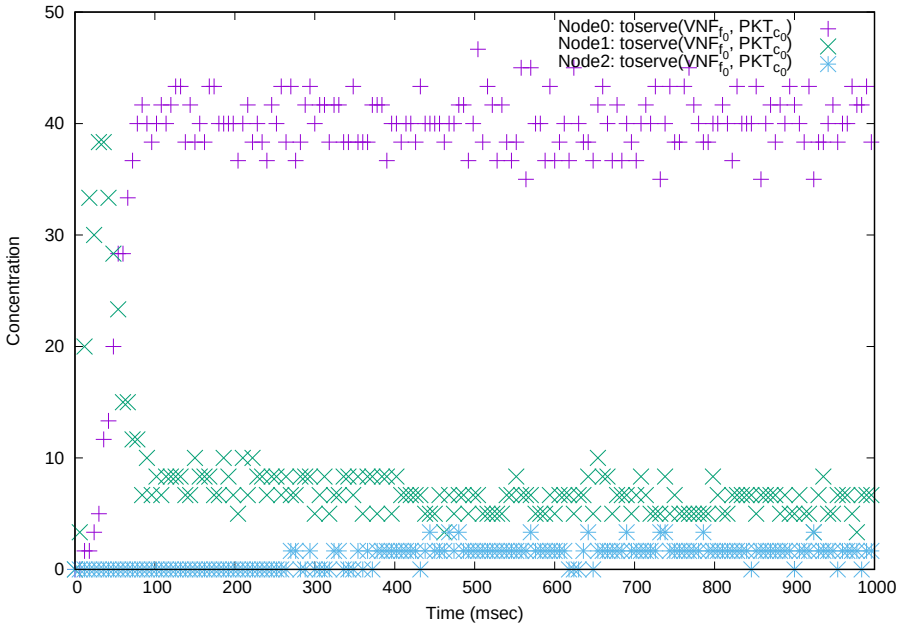
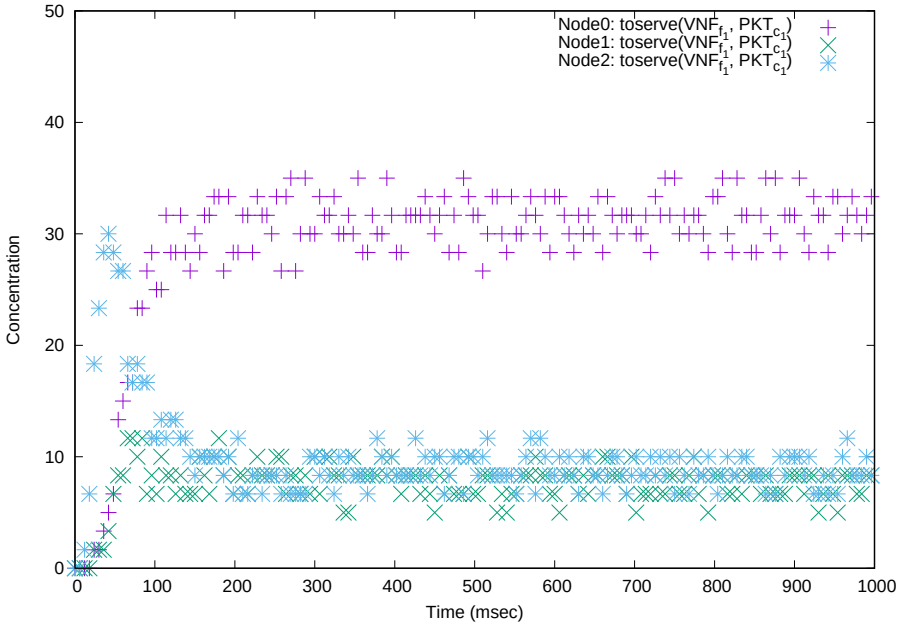


Figure 17: Scenario 3: Simulation environment



(a) $toserve(VNF_{f_0}, PKT_{c_0})$



(b) $toserve(VNF_{f_1}, PKT_{c_1})$

Figure 18: Scenario 3: Average number of $toserve$

5 Parameter Tuning to Control VNF Services

5.1 Overview

In the proposed method, as shown in Subsection 3.6, the flow path is determined according to the gradient field When the shape of the gradient field, that is, the ratio of the gradient between nodes changes, the flow route also changes.

When the difference in ratio of gradient between nodes is small, VNF processing is distributed at multiple nodes. This makes it possible to increase the redundancy, for example in the NFV system. On the other hand, when the difference in ratio of the gradient between nodes is large, VNF processing is concentrated on a small number of nodes with large value of the gradient. This makes it possible, for example, to make an unnecessary node sleep, and power consumption reduction is expected, while the redundancy may degrade.

One possible method for changing the shape of the gradient field is tuning the reaction rate coefficients of Reactions (7)-(10), which realize generation, diffusion, decay of $GRAD$. We first conduct the mathematical analysis of the proposed method to reveal its fundamental behavior and the effect of control parameters on the generation of the gradient field. We then present the simulation results to confirm that we can control the degree of the load balancing among multiple servers by tuning the parameter determining the biochemical reaction speed.

5.2 Mathematical Analysis

We define V, R, R_V, M, G as the concentration of VNF, $RSRC$, RS_VNF, $MEDIATE$, $GRAD$ as a function of time, respectively, and G_{in} is the concentration of $GRAD$ which moves from neighboring nodes. Then, the temporal change in the concentration of $GRAD$ of a node can be calculated as follows, where r_{mg} is regarded as 0 when there is no node with a lower gradient.

$$\frac{dG}{dt} = VRr_{rg} + VR\_Vr_{rg} - Gr_{dg} - Gr_{mg} + G_{in} \tag{22}$$

$$= V(R + R\_V)r_{rg} - G(r_{dg} + r_{mg}) + G_{in} \tag{23}$$

After a sufficient time has elapsed, assuming $\frac{dG}{dt} = 0$, Equation (24) is derived, where G^* is determined to be one value by the relation of gradient with other nodes.

$$G^* = \frac{1}{r_{dg} + r_{mg}}\{V^*(R + R\_V)r_{rg} + G_{in}\} \tag{24}$$

Assuming a simple topology consisting of two nodes and one link, we derive the convergence value of $GRAD$ of each node and the ratio of gradient between the nodes. We call each node node 1 and node 2, and assume the gradient of node 1 is greater than that of node 2. Denoting that the concentration of substance X of node n is represented as X_n, the convergence value of $GRAD$ of each node is derived from Equation (24) and the following equations are obtained.

$$G_1^* = \frac{V_1^*(R_1^* + R\_V_1^*)r_{rg}}{r_{dg} + r_{mg}} \tag{25}$$

$$G_2^* = \frac{V_2^*(R_2^* + R\_V_2^*)r_{rg} + G_{in2}^*}{r_{dg}} \tag{26}$$

Since the concentration of $GRAD$ diffusing from node 1 is $G_1 r_{mg}$ and the diffusion destination is node 2, G_{in2} can be expressed by the following equation.

$$G_{in2} = G_1 r_{mg} \tag{27}$$

Therefore, Equation (26) can be converted as follows.

$$G_2^* = \frac{V_2^*(R_2^* + R\_V_2^*)r_{rg} + G_1^* r_{mg}}{r_{dg}} \tag{28}$$

The ratio of the gradient of node is obtained as follows.

$$\frac{G_1^*}{G_1^* + G_2^*} = \frac{G_1^*}{\frac{(r_{dg}+r_{mg})G_1^*}{r_{dg}} + \frac{V_2^*(R_2^*+R\_V_2^*)r_{rg}}{r_{dg}}} \tag{29}$$

$$= \frac{\frac{V_1^*(R_1^*+R\_V_1^*)r_{rg}}{r_{dg}+r_{mg}}}{V_1^*(R_1^* + R\_V_1^*)r_{rg} + V_2^*(R_2^* + R\_V_2^*)r_{rg}} \tag{30}$$

$$= \frac{V_1^*(R_1^* + R\_V_1^*)r_{dg}}{\{V_1^*(R_1^* + R\_V_1^*) + V_2^*(R_2^* + R\_V_2^*)\}(r_{dg} + r_{mg})} \tag{31}$$

From this equation, we can observe that the smaller the value of r_{mg}, G_1^* becomes smaller compared with G_2^*.

5.3 Scenario 4: Evaluation Results

From the above discussion, by tuning the value of r_{mg}, we can adjust the difference of ratio of gradient between nodes. We conduct simulation experiments to confirm this property. Figure 19 depicts the network topology, the packet rates, and the initial concentration of substances for the simulation in this subsection. The simulation environment is the same with that of Section 4 except

the initial concentration of VNF_{f_1} of node 2. In this scenario, we only set the initial value of the concentration of VNF_{f_0} of node 1 to 2,000.

The topology consists of three nodes and three links, and each node is called node 0-2 as depicted in the figure.

The concentration at the start of simulation of each chemical substance in Reactions (13)-(21) is determined as follows. We set the initial values of the concentration of VNF_{f_0} of node 1 to 2,000. All node has 1,000 of the initial concentration of $RSRC$. Assumed that a VNF is executed to each packet of a flow, the initial concentration of $RSRC$ corresponds to the capacity of executing VNFs at roughly 83 Kpps, which becomes roughly 1 Gbps with packet size of 1,500 Bytes. The initial value of the concentrations of other chemical substances are set to 0.

The reaction rate coefficients of Reactions (13)-(21) are determined as $r_{u1} = 0.0003,\ r_{v1} = 0.278,\ r_{u2} = 0.1,\ r_{v2} = 0.001,\ r_w = 0.05,\ r_d = 0.01,\ r_{mf} = 0.003,\ r_{rg} = 0.0001,\ r_{dg} = 0.03,\ r_{mg} = 0.005,\ r_{mp} = 0.3$. We conduct the simulation experiments while changing the value of r_{mg} from 5×10^{-1} to 5×10^{-6}.

In this simulation, in the NFV system, the VNF provided is VNF f_0 , and the service chaining request that may exist is $c_0 = \{f_0\}$.

Figure 20 shows the temporal change in the concentration of $GRAD_{f_0}$ with various values of r_{mg}. From this figure, it is clarified that as r_{mg} becomes small, the difference in the gradient between the nodes 1 and 2 becomes large. Figure 21 shows the average number of $toserve(VNF_{f_0}, PKT_{c_0})$ generated at each node. From (a) to (f), the difference in gradient ratio becomes larger, and more VNFs are executed at node 1.

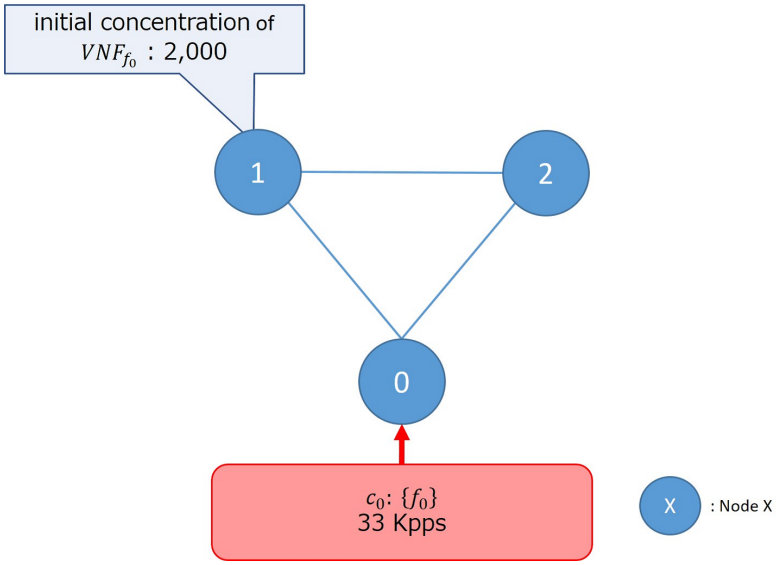
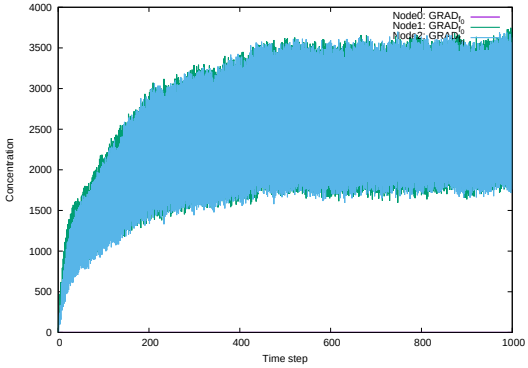
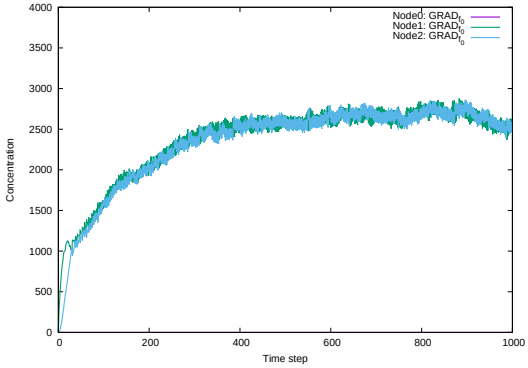


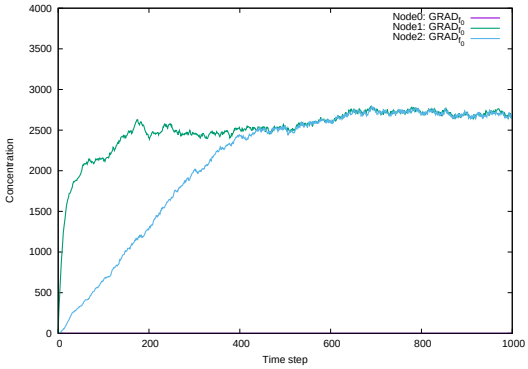
Figure 19: Scenario 4: Simulation environment



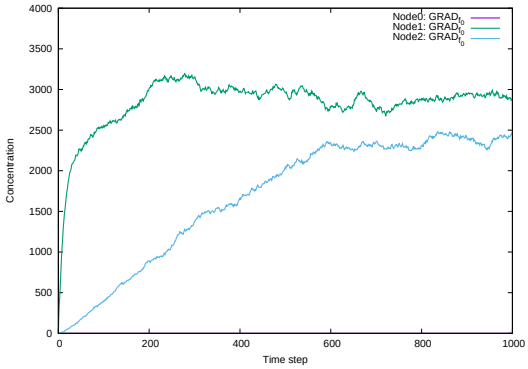
(a) $r_{mg} = 0.5$



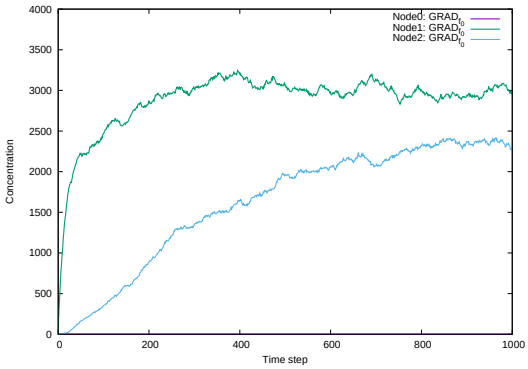
(b) $r_{mg} = 0.05$



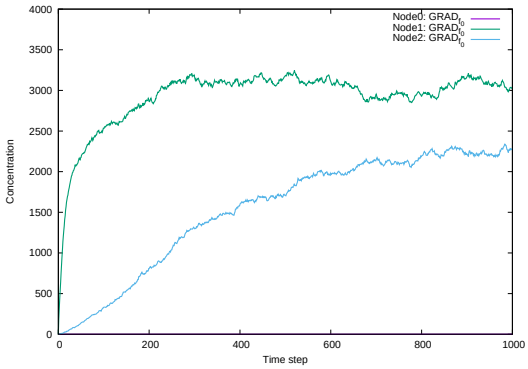
(c) $r_{mg} = 0.005$ (default)



(d) $r_{mg} = 0.0005$

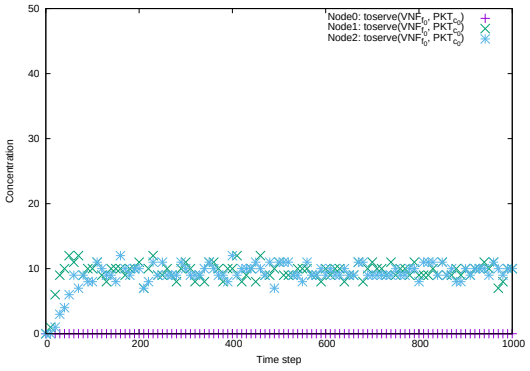


(e) $r_{mg} = 0.00005$

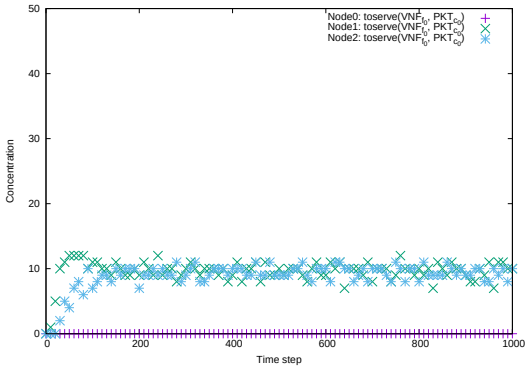


(f) $r_{mg} = 0.000005$

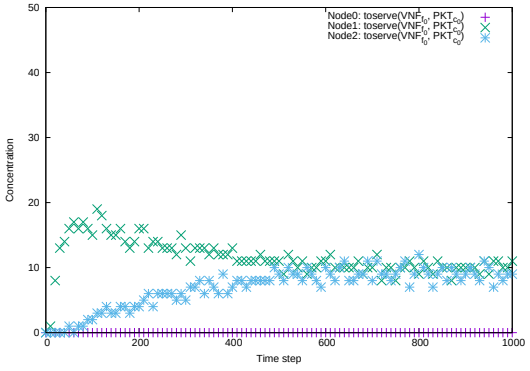
Figure 20: Scenario 4: Temporal change in the concentration of $GRAD_{f_0}$



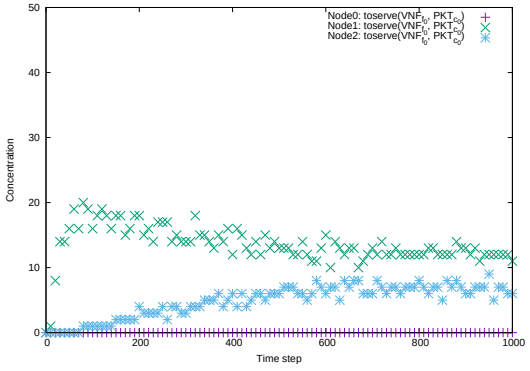
(a) $r_{mg} = 0.5$



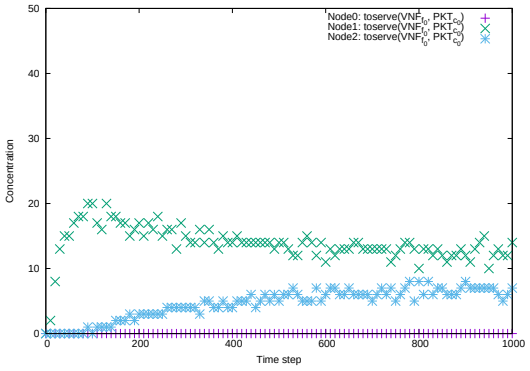
(b) $r_{mg} = 0.05$



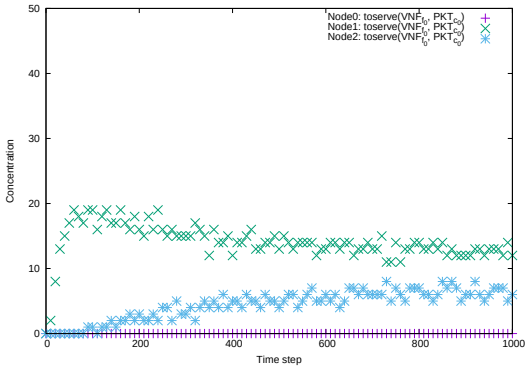
(c) $r_{mg} = 0.005$ (default)



(d) $r_{mg} = 0.0005$



(e) $r_{mg} = 0.00005$



(f) $r_{mg} = 0.000005$

Figure 21: Scenario 4: Average number of $toserve$

6 Conclusion and Future Work

In this thesis, we exploited the construction method of service space in virtualized network system based on biochemical-inspired tuple space model, to create the NFV system. Specifically, we defined chemical substances and biochemical reaction equations to describe the behavior of NFV system such as the execution of VNF to flows, the allocation of server resource to the flows, realization service chaining, coexistence of multiple VNFs on a single server, and decision of packet flow routes. We then performed computer simulation experiments based on three scenarios and clarify the effectiveness of the proposed method, and confirmed that the functions required for the NFV system can be realized with the proposed method. Furthermore, we exhibited that various processing patterns required by the NFV system could be realized by parameter tuning of the proposed method.

For future work, we plan to conduct more detailed performance evaluation of the proposed method with larger and more actual scenarios. We also would like to extend the proposed method to describe the effect of the propagation delay time between nodes, the link bandwidth and evaluate it on a large scale network, and heterogeneous resource utilization by various kind of VNFs. In addition, it is also important to design and implement the NFV system based on the proposed method and to conduct experimental evaluation to confirm the effectiveness of the proposed method in real systems.

Acknowledgments

There are so many people to thank for helping me during my master ' s degree studies at Osaka University. I would like to express my deepest gratitude to my supervisor, Professor Masayuki Murata of Osaka University for his insightful advice, guidance and encouragement. Furthermore, I would like to express the deepest appreciation to Associate Professor Go Hasegawa of Osaka University, for his elaborated guidance and invaluable firsthand advice. I would like to appreciate to Associate Professor Shin ' ichi Arakawa, Assistant Professor Yuichi Ohsita, Assistant Professor Daichi Kominami, Appointed Assistant Professor Naomi Kuze and Appointed Assistant Professor Tatsuya Otoshi of Osaka University for beneficial comments and suggestions on this study. Finally, I truly thank my friends and colleagues in Graduate School of Information Science and Technology of Osaka University, for their great encouragement and support.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, pp. 2347–2376, June 2015.

[2] ETSI, "Network functions virtualisation - white paper 3." available at `https://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf`.

[3] K. Ingham and S. Forrest, "A history and survey of network firewalls," *University of New Mexico, Tech. Rep*, Jan. 2002.

[4] M. Finsterbusch, C. Richter, E. Rocha, J.-A. Muller, and K. Hanssgen, "A survey of payload-based traffic classification approaches," *IEEE Communications Surveys & Tutorials*, vol. 16, pp. 1135–1156, Oct. 2013.

[5] D. Wing, "Network address translation: Extending the internet address space," *IEEE Internet computing*, vol. 14, pp. 66–70, June 2010.

[6] M. Olsson, S. Rommer, C. Mulligan, S. Sultana, and L. Frid, *SAE and the Evolved Packet Core: Driving the mobile broadband revolution*. Academic Press, Aug. 2009.

[7] J. Carapinha and J. Jimenez, "Network virtualization - a view from the bottom," in *Proceedings of VISA 2009*, pp. 73–80, June 2009.

[8] D. King and C. Ford, "A critical survey of network functions virtualization," in *Proceedings of the Accounting and Finance 2013*, pp. 1–21, July 2013.

[9] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 18, pp. 236–262, Sept. 2015.

[10] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and S. Davy, "Design and evaluation of algorithms for mapping and scheduling of virtual network functions," in *Proceedings of Network Softwarization (NetSoft), 2015 1st IEEE Conference on*, pp. 1–9, IEEE, June 2015.

[11] S. Mehraghdam, M. Keller, and H. Karl, "Specifying and placing chains of virtual network functions," *Networking and Internet Architecture*, vol. 1058, pp. 1–7, June 2014.

[12] C. Kesselman, C. Lee, and B. Lindell, "A distributed resource management architecture that supports advance reservations and co-allocation," in *Proceedings of International Workshop on Quality of Service 1999*, pp. 27–36, June 1999.

[13] Marco Piraccini, "Biotucson: biochemical extension of tucson to support self-organising coordination," Master's thesis, University of Bologna, Mar. 2013.

[14] Mirko Viroli, Matteo Casadei, Sara Montagna, Franco Zambonelli, "Spatial coordination of pervasive services through chemical-inspired tuple spaces," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 6, pp. 1–24, June 2011.

[15] M. Viroli, M. Casadei, and Matteo, "Chemical-inspired self-composition of competing services," in *Proceedings of the ACM Symposium on Applied Computing 2010*, pp. 2029–2036, July 2006.

[16] M. Viroli and M. Casadei, "Biochemical tuple spaces for self-organising coordination," in *Proceedings of Coordination Models and Languages*, pp. 143–162, Springer, June 2009.

[17] Shun Sakurai, "Construction method of service space in virtualized network system based on chemical-inspired tuple space model," Master's thesis, Osaka University, Feb. 2015.

[18] Go Hasegawa and Masayuki Murata, "Biochemically-inspired method for constructing service space in virtualized network system," in *Proceedings of ICIN 2016*, Mar. 2016.

[19] D. T. Gillespie, "Stochastic simulations for chemical kinetics," *Annual Review of Physical Chemistry*, vol. 58, pp. 35–55, Oct. 2006.

[20] K. A. Johnson and R. S. Goody, "The original michaelis constant: Translation of the 1913 michaelis–menten paper," *Biochemistry*, vol. 50, no. 39, pp. 8264–8269, 2011.

[21] R. N. Goldberg, Y. B. Tewari, and T. N. Bhat, "Thermodynamics of enzyme-catalyzed reactions," *Bioinformatics*, vol. 20, no. 16, pp. 2874–2877, 2004.

[22] W. W. Cleland, "The kinetics of enzyme-catalyzed reactions with two or more substrates or products," *Science Direct*, vol. 67, pp. 173–187, May 1962.

[23] H. Li, Y. Cao, and D. T. Gillespie, “Algorithms and software for stochastic simulation of biochemical reacting systems,” *Biotechnology Progress*, vol. 24, pp. 56–61, Feb. 2008.

[24] Cao, Y., D. Gillespie, and L. Petzold, “Efficient stepsize selection for the tau-leaping method,” *The Journal of Chemical Physics*, vol. 124, pp. 44–109, Jan. 2006.