社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

生化学反応に着想を得た動的かつ自律分散的なVNF配置手法

黒川　稜太[†]　　長谷川　剛[††]　　村田　正幸[†]

[†] 大阪大学大学院情報科学研究科　〒560-0871 大阪府吹田市山田丘 1–5
[††] 大阪大学サイバーメディアセンター　〒560–0043 大阪府豊中市待兼山町 1–32
E-mail: †{r-kurokw,murata}@ist.osaka-u.ac.jp, ††hasegawa@cmc.osaka-u.ac.jp

あらまし　ネットワーク機能仮想化技術 (Network Function Virtualization: NFV) では, 様々な仮想ネットワーク機能 (Virtual Network Function: VNF) が汎用サーバ上に配置される．NFV に基づくネットワークシステムを運用するためには, サーバへの VNF の配置, 各 VNF へのサーバ資源の割り当て, 及びネットワークフローの経路を動的に決定することが求められる．さらに, NFV システムは, 需要変動, 及びシステム障害などの環境変動に素早く対応するために, 自律分散的に動作することが望ましい．我々の研究グループでは, 生化学反応に着想を得たタプル空間モデルに基づくサービス空間構築手法を提案しており, そのモデルを NFV システムへ適用した場合の基本的な性能が評価されている．本報告では, NFV システムにおける様々な状況を想定したシミュレーションを実行し, 提案手法が NFV システムの動的な状況に対応できることを示す．

キーワード　ネットワーク機能仮想化技術, サービスチェイニング, 生化学機構, タプル空間モデル, Network Service Header

1. Introduction

In Network Function Virtualization (NFV), network functions on dedicated hardware are achieved by software, and deployed and executed on general-purpose servers [1]. The network functions achieved by software are called Virtual Network Functions (VNFs). Figure 1 shows an NFV system. In NFV, multiple VNFs may share the resource on a single server or one VNF may be distributed to multiple servers to provide services throughout the network [2, 3]. As a result, it is possible to suppress operational and capital expenditures by aggregating physical servers. It is also possible to flexibly respond to environmental fluctuations by reallocating server resources to VNFs, migrating VNFs, and rerouting flow packets.

A flow receiving NFV service may have a Service Function Chaining (SFC) request that describes the order of VNFs to be applied to the flow. In Figure 1, a flow arriving at the NFV system receives NFV services in accordance with the SFC request and exits the system. Therefore, to efficiently operate the NFV system, placement of VNFs to servers, resource allocation to each VNF, and flow routes are determined adaptively in accordance with the SFC requests, traffic amount of the flows, and amount of server resource. In addition, to quickly respond to environmental fluctuations such as system failures and changing demands, and to maintain the scalability of the NFV services, a distributed control is more feasible than a centralized one [4]. One way to achieve such behaviors is to exploit a biochemical mechanism with autonomous dispersibility and self organization [5].

Our research group has proposed a construction method of service space in virtualized network system based on biochemically-inspired tuple space model [6]. In this method, a server is considered as a tuple space, and service requests, service demands and server resources are expressed as chemical substances in tuple spaces. The behaviors in the virtualized network system are then described by biochemical reaction equations in tuple spaces. Furthermore, by configuring a network by connecting multiple tuple spaces, the movement and spread of services and requests in a network system composed of multiple servers are represented. Since biochemical reaction equations are defined and executed independently in each tuple space, it is suitable for achieving autonomous and decentralized behaviors. We consider that one of possible application of the above method is an NFV system. To operate an NFV system, the above method has been extended to handle flow routes in accordance with SFC requests, and server resource limitation. By including these behaviors in the method, it is possible to get closer to the actual NFV service. The basic behaviors of the extended method have been confirmed with computer simulation. However, the evaluation assuming various situations in the NFV system has not been performed.

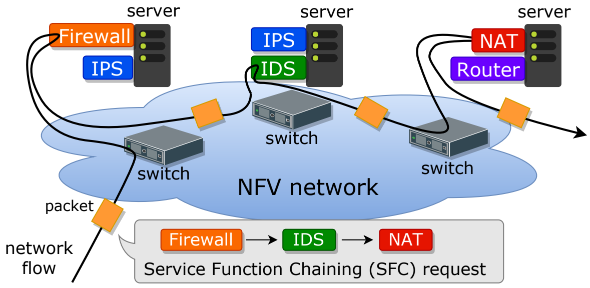


Figure 1　NFV system

In this report, we assess the performance of the NFV system based on the service space construction method. First, we briefly summarize how to apply the service space construction method to the NFV system, explained in [6]. In particular, we describe various behaviors in the NFV system, such as the execution of VNFs to flow packets, server resource allocation to each VNF, diffusion of VNFs, packet forwarding, and coexistence of multiple VNFs on a single server, by biochemical reaction equations in tuple spaces. Then, we perform computer simulation experiments assuming various situations in the NFV system, such as time variation of traffic amount and a sudden network failure. Through the simulation experiments, we confirm that the proposed method can cope with dynamical changes in the NFV system.

2. VNF Control based on Tuple Space Model with Biochemical Reactions

In this section, we summarize the tuple space model using biochemical reactions and how to apply the model to NFV system, described in [6].

2.1 Tuple Space Model

A tuple space model in [6] is one of the models that describes a distributed system. A component of the distributed system is modeled as a tuple space. In a tuple space, biochemical reactions occur. Then, tuples in the tuple space correspond to chemical substances, and the amount of tuples corresponds to the concentrations of chemical substances. The concentrations of tuples can be increased and decreased by defining and executing biochemical reactions in tuple spaces.

A reaction rate of a biochemical reaction is determined by the product of the concentration of each reactant and the rate coefficient defined in the biochemical reaction equation. For example, we consider that the following reaction equation is defined, which defines X and Y as reactants, Z as a product, and a as a reaction rate coefficient.

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

If the concentrations of reactants X and Y are respectively x and y, the reaction rate is axy. Due to this property, the reaction rates in biochemical reactions are controlled by the concentrations of reactants and the rate coefficients defined in biochemical reaction equations.

In addition, a network can be configured by connecting multiple tuple spaces. It is possible to achieve the interaction among multiple tuple spaces by defining biochemical reactions that describe the diffusion and movement of tuples among tuple spaces. Since biochemical reactions in each tuple space occur independently, autonomous and decentralized behaviors in networked system can be described.

2.2 Application to NFV System

To apply the tuple space model to NFV system, a tuple space is associated with a server that deploys and executes VNFs. Tuples in the tuple spaces correspond to demands of VNFs, flow packets, server resources, and so on. The behaviors in the NFV system are described by biochemical reaction equations in tuple spaces. Biochemical reaction equations are defined to adaptively and autonomously determine placement of VNFs on the servers, the resource allocation to each VNF, and flow routes in accordance with SFC requests, traffic amount of the flows, and the amount of server resources.

An SFC request for a flow, represented by a series of VNFs, $f_1, f_2, f_3, ..., f_{end}$ is described as follows.

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

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

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

A VNF that is executed at first in c is represented by $f^1(c)$. In this report, the subscript f, c and t of chemical substances represent a VNF, an SFC request, and a server, respectively. In what follows, we present biochemical reaction equations that achieve various behaviors for the NFV system.

2.2.1 Resource Allocation and Execution of VNFs

It is desirable that placement of VNFs on servers and resource allocation to each VNF are determined in accordance with demands of VNFs. It is required that VNFs in low demand have low priority in the server and those in high demand have high priority to be executed. When a packet of a flow with an SFC request c arrives at a server, VNF $f^1(c)$ is applied to the packet. Then, when c is composed of multiple VNFs, the SFC request c changes so that the executed VNF is deleted from c. On the other hand, when c is composed of one VNF, the packet disappears. The above behaviors, as well as realizing the server resource limitation by enzyme-catalyzed reactions mechanism [7], are described by Reaction Equations (1)-(4).

$$RSRC_t \mid VNF_f \underset{r_{u_1}}{\overset{r_{v_1}}{\leftrightarrows}} RS\_VNF_f \tag{1}$$

$$RS\_VNF_{f^1(c)} \mid PKT_c \underset{r_{u_2}}{\overset{r_{v_2}}{\leftrightarrows}} MEDIATE_c \tag{2}$$

$$MEDIATE_c \xrightarrow{r_w} \begin{cases} VNF_{f^1(c)} \mid VNF_{f^1(c)} \mid PKT_{c\backslash\{f^1(c)\}} \mid RSRC_t \\ \mid toserve(VNF_{f^1(c)}, PKT_c) \quad (c\backslash\{f^1(c)\} \neq \emptyset) \\ VNF_{f^1(c)} \mid VNF_{f^1(c)} \mid RSRC_t \\ \mid toserve(VNF_{f^1(c)}, PKT_c) \quad (c\backslash\{f^1(c)\} = \emptyset) \end{cases} \tag{3}$$

$$VNF_f \xrightarrow{r_{ds}} 0 \tag{4}$$

In the above Equations, substance VNF_f indicates the VNF to be applied for a flow. A VNF with a large concentration value means that its execution is highly demanded. Substance PKT_c represents a packet constituting a flow with c. Substance $toserve(VNF_{f^1(c)}, PKT_c)$ indicates result of applying the VNF to a packet of a flow with c. The concentrations of substances $RSRC_t$, RS_VNF_f, and $MEDIATE_c$ respectively represent the amount of available resources of a server t, the amount of server resources allocated to VNF f, and the amount of server resources allocated to the flow packets with SFC request c. r_{v_1} and r_{u_1} are the rate coefficients for Reaction Equation (1), and r_{v_2}, r_{u_2} and r_w are the rate coefficients for Reaction Equation (2). Reaction Equation (1) indicates that server resources are allocated in accordance with the demand of each VNF, and that the allocation is controlled by the concentration of $RSRC$. Reaction Equations (2) and (3) indicates that VNF f is executed on the basis of the amount of allocated resources. Reaction (4) indicates that VNF decays at a rate proportional to its concentration.

2.2.2 Diffusion of VNFs

To describe the diffusion of highly-demanded VNFs to other servers, Reaction Equation (5) is described.

$$VNF_f \xrightarrow{r_{mf}} VNF_f^{\leadsto} \tag{5}$$

r_{ms} is the rate coefficient for Reaction Equation (5). This Reaction Equation indicates that a highly-demanded VNF in a server diffuses to the surrounding connected servers at a rate proportional to its concentration. This diffusion destination of VNFs is stochastically determined in accordance with the concentrations of VNF at connected tuple spaces. As a result, highly-demanded VNFs are distibuted to multiple servers.

2.2.3 Packet Forwarding

When packets remain unprocessed in a server due to a lack of server resources for corresponding VNF, it is required that the packets move to another server that can process the corresponding VNF. Furthermore, the forwarding direction of packets should be determined so that the packets would approach a server executing the corresponding VNFs with enough server resources. To achieve these behaviors, a gradient field is exploited to determine the moving directions of packets. A gradient field for each VNF is constructed based on the demand of VNFs and the available resources on each server. The moving direction of packets is then determined in accordance with the gradient field. For that purpose, Reaction Equations (6)-(10) are introduced.

$$VNF_f \,|\, RSRC_t \xrightarrow{r_{rg}} VNF_f \,|\, RSRC_t \,|\, GRAD_f \tag{6}$$

$$VNF_f \,|\, RS\_VNF_f \xrightarrow{r_{rg}} VNF_f \,|\, RS\_VNF_f \,|\, GRAD_f \tag{7}$$

$$GRAD_f \xrightarrow{r_{dg}} 0 \tag{8}$$

$$GRAD_f \xrightarrow{r_{mg}} GRAD_f^{\rightsquigarrow}(GRAD_f^{-}) \tag{9}$$

$$PKT_c \xrightarrow{r_{mp}} PKT_c^{\rightsquigarrow}(GRAD_f^{+}) \tag{10}$$

Substance $GRAD_f$ establishes a gradient field for VNF f. r_{rg} is the rate coefficient for Reaction Equation (6) and (7), and r_{dg}, r_{mg} and r_{mp} are the rate coefficients for Reaction Equation (8), (9) and (10), respectively. Reaction Equation (6) and (7) indicate that *GRAD* is generated at a rate proportional to the concentrations of *VNF*, *RSRC*, and *RS_VNF*. Reaction Equation (8) indicates that *GRAD* decays at a rate proportional to its concentration. Reaction Equation (9) indicates that *GRAD* spreads to the surrounding servers with smaller concentration of *GRAD*. Therefore, the gradient field is constructed so that the server providing VNFs with enough resources becomes a summit with the largest concentration of *GRAD*, and the surrounding servers have smaller concentration of *GRAD* in accordance with the distance from the summit. Reaction Equation (10) describes the movement of *PKT* to the surrounding servers with large concentration of *GRAD*. The forwarding direction of packets are stochastically determined at a proportional to the concentrations of *GRAD* at connected tuple spaces.

2.2.4 Coexistence of Multiple VNFs

When multiple VNFs coexist on a single server, it is required to share server resources by allocating them in accordance with the demand of each VNF. Therefore, the abovementioned biochemical reaction equations are defined for each VNF.

3. Simulation Experiments

In this section, we assess the performance of the NFV system based on the method described in Section 2.. The basic behaviors of the proposed method, such as placement of VNFs on servers, resource allocation to each VNF, and flow routing in accordance with SFC requests, have been confirmed in [6]. We then confirm that the proposed method can cope with dynamical changes in the NFV system.

In order to simulate the model with biochemical reactions, we exploit τ-leaping method [8], which is one of stochastic simulation algorithms that can capture the inherent stochasticity in many biochemical systems.

3.1 Common Parameter Settings

The initial values of the concentrations of substances *VNF* for the VNFs placed on servers are set to 2,000. The initial values of the concentrations of other chemical substances except *RSRC* are set to 0. Unless otherwise specified, the reaction rate coefficients of Reaction Equations (1)-(10) are set as $r_{u1} = 0.0003$, $r_{v1} = 0.278$, $r_{u2} = 0.1$, $r_{v2} = 0.001$, $r_w = 0.05$, $r_d = 0.01$, $r_{mf} = 0.003$, $r_{rg} = 0.0001$, $r_{dg} = 0.03$, $r_{mg} = 0.005$, $r_{mp} = 0.3$, as used in [6]. The value of time step in τ-leaping method is set to 0.6 [msec], which is identical to that in [6].

3.2 Scenario 1: Placement of VNFs Considering Flow Priorities

3.2.1 Application Scenario

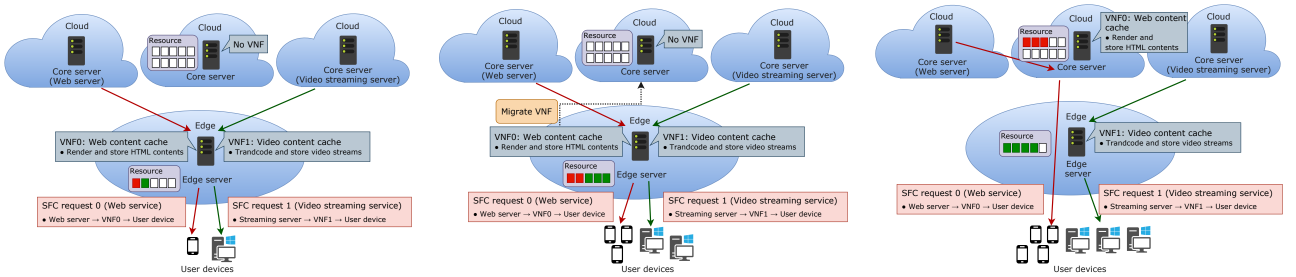
In this scenario, we consider the situation where there are two kinds of application flows with SFC requests that have different priorities on the latency requirements. Figure 2 depicts this scenario. In the figure, Web service and video streaming service are provisioned in edge and cloud computing environments. In the beginning, a Web server and a video streaming server are placed in the cloud. The both servers receive requests from user devices, and send content packets to the user devices. Flows between the Web server and user devices require functions of rendering and caching in the network. On the other hand, transcoding and caching functions are applied to flows between the video streaming server and the user devices. Consequently, these VNFs are deployed in the network, that are called as VNF 0 and VNF 1, respectively. Furthermore, the SFC requests of the flows in the NFV system are {Web server→VNF 0→User device} and {Streaming server→VNF 1→User device}. These SFCs are respectively denoted by SFC 0 and SFC 1. We assume that the flows for the video streaming service have higher priority in being executed at the edge server to meet the latency requirements.

As depicted in Figure 2(a), in case of edge server being not busy, VNF 0 and VNF 1 are deployed on the edge server to realize contents caching for both services near the user devices. In Figure 2(b), since the number of user devices increases, the edge server becomes busy and all packets cannot be processed only at the edge server. Then, as depicted in Figure 2(c), VNF 0 is migrated to the cloud server, while VNF 1 remains on the edge server to avoid the degradation of the quality of both services.

3.2.2 Network Topology and Parameter Settings for Simulation Experiments

Figure 3 depicts the network topology for simulation experiments of Scenario 1, that consists of two nodes and a link. Node 0 and node 1 correspond to the edge server and the cloud server, respectively. VNF_{f_0} and VNF_{f_1} correspond to VNF 0 and VNF 1 in Figure 2, respectively. There are two flows with SFC requests $c_0 = \{f_0\}$ and $c_1 = \{f_1\}$, corresponding to the flows with SFC 0 and SFC 1. These flows are denoted by flow 0 and flow 1, respectively.

The simulation time is 2,000 [msec]. VNF_{f_0} and VNF_{f_1} are initially deployed on node 0, and their initial concentrations are set to 2,000. The initial concentrations of *RSRC* at node 0 and node 1 are set to 500 and 1,000, respectively, which means that the cloud server has larger and sufficient resource than the edge server. Table 1 shows the temporal change in flow rates. In the table, t is defined as simuation time. For $0 \leqq t \leqq 1,000$, packets of flow 0 and flow 1 arrive at node 0 at 5 packets per time step, corresponding to 8.3 [Kpps]. At $t = 1,000$, the rates of both flows are increased to 20 packets per time step, corresponding to 33.3 [Kpps]. Note that for $0 \leqq t \leqq 1,000$, the edge server can process all incoming packets, and for $1,000 < t \leqq 2,000$, the edge server


Cloud
Core server
(Web server)
Resource
No VNF
Core server
Core server
(Video streaming server)
Edge
VNF0: Web content cache
• Render and store HTML contents
VNF1: Video content cache
• Trandcode and store video streams
Edge server
SFC request 0 (Web service)
• Web server → VNF0 → User device
SFC request 1 (Video streaming service)
• Streaming server → VNF1 → User device
User devices
Migrate VNF
VNF0: Web content cache
• Render and store HTML contents

(a) Edge server being not busy (b) Edge server being busy (c) Migrating the Web service VNF to the cloud server

Figure 2 Scenario1: Placement of VNFs considering flow priorities

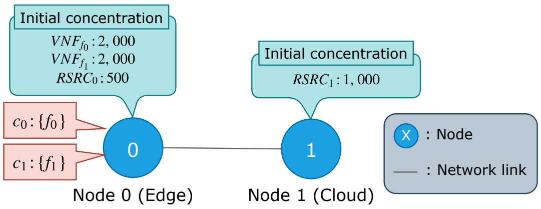


Figure 3 Scenario1: Network topology for simulation experiments

Table 1 Scenario1: Temporal change in rate of flows

Flow	Priority	Rate	
		$0 \leqq t \leqq 1,000$ [msec]	$1,000 < t \leqq 2,000$ [msec]
flow 0	low	8.3 [Kpps]	33.3 [Kpps]
flow 1	high	8.3 [Kpps]	33.3 [Kpps]

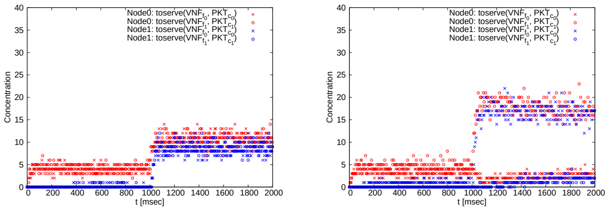
cannot process all packets.

To prioritize the execution of VNF 1 at node 0, the rate coefficient r_{u1} in Reaction Equation (1) is adjusted. We utilize $r_{u1} = 0.0003$ for the VNF_{f_0}, and $r_{u1} = 0.003$ for the VNF_{f_1} at node 0, to prioritize flow 1. We also perform simulation experiments with $r_{u1} = 0.0003$ for VNF_{f_0} and VNF_{f_1} at node 0 for comparison purposes.

3.2.3 Simulation Results and Discussion

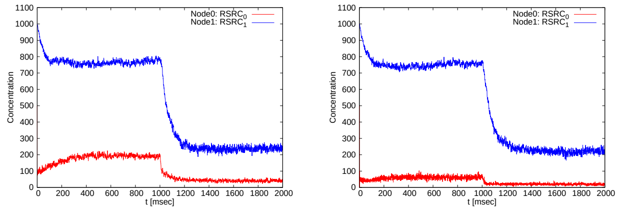
Figure 4 plots the average number of executions of Reaction Equation (2), that corresponds to the executions of VNFs to flow packets, as a function of simulation time step. Figures 4(a) and 4(b) show simulation results with $r_{u1} = 0.0003$ and $r_{u1} = 0.003$ for VNF_{f_0} at node 0, respectively. Figure 5 shows the temporal change in the concentrations of $RSRC$ at node 0 and node 1.

For $0 \leqq t \leqq 1,000$, VNF_{f_0} and VNF_{f_1} are executed almost only at node 0. This is because node 0 has sufficient resources to execute both VNFs to flow packets. It can be confirmed from the concentration of $RSRC_0$ in Figure 5. For $1,000 < t \leqq 2,000$, VNF_{f_0} and VNF_{f_1} are executed at node 0 and node 1 in a distributed manner, when using the same value of r_{u1} for VNF_{f_0} and VNF_{f_1}. This is because node 0 has insufficient resources to execute both VNFs due to the increase in flow rates. It can be confirmed from the concentration of $RSRC_0$ in Figure 5(a). On the other hand, when setting r_{u1} in accordance with flow priorities, VNF_{f_0} and VNF_{f_1} are executed at node 1 and node 0, respectively. This behavior realizes that the VNF in video streaming service with higher priority is preferentially executed at the edge server, as depicted in Figure 2.



(a) $r_{u1} = 0.0003$ (b) $r_{u1} = 0.003$

Figure 4 Scenario1: Average number of executions of Reaction Equation (2)



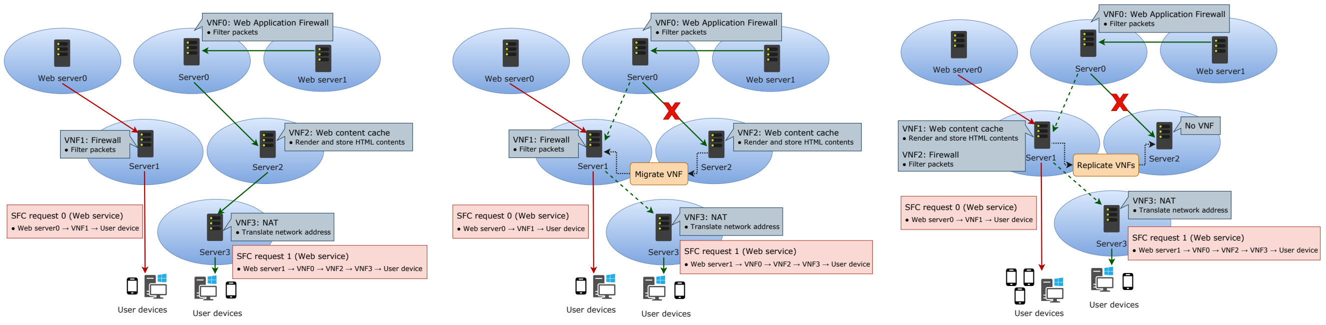
(a) $r_{u1} = 0.0003$ (b) $r_{u1} = 0.003$

Figure 5 Scenario1: Temporal change in the concentrations of $RSRC$ at node 0 and node 1

3.3 Scenario 2: Route changes and VNF migrations on network failures

3.3.1 Application Scenario

In this scenario, we consider the situation where failures of network link occur. Figure 6 depicts this scenario. In the figure, two Web services are provisioned in cloud computing environment. The both of Web server 0 and Web server 1 receive requests from user devices, and send content packets to the user devices. Flows between the Web servers and user devices require a function for filtering, monitoring, and blocking HTTP traffic, a function for monitoring and controling incoming and outgoing network traffic, a function for rendering and caching in the network, and a function for translating network addresses. Consequently, four VNFs exist in the network, that are denoted by VNF 0, VNF 1, VNF 2, and VNF 3. Initially, VNF 0, VNF 1, VNF 2, and VNF 3 are respectively deployed on server 0, server 1, server 2, and server 3. VNF 1 is applied to the flow between Web server 0 and user devices, and VNF 0, VNF 2, VNF 3 are sequentially applied to the flow Web server 1 and user devices. The SFC requests of the two flows are {Web server→VNF1→User device} and {Web server→VNF0→VNF2→VNF3→User device}, that is called


VNF0: Web Application Firewall
• Filter packets
Web server0
Server0
Web server1
VNF1: Firewall
• Filter packets
Server1
VNF2: Web content cache
• Render and store HTML contents
Server2
SFC request 0 (Web service)
• Web server0 → VNF1 → User device
VNF3: NAT
• Translate network address
Server3
SFC request 1 (Web service)
• Web server1 → VNF0 → VNF2 → VNF3 → User device
User devices
Migrate VNF
VNF1: Web content cache
• Render and store HTML contents
VNF2: Firewall
• Filter packets
No VNF
Replicate VNFs

(a) System being operated normally (b) System failures occur (c) The server being busy

Figure 6 Scenario2: Route changes and VNF migrations on network failures

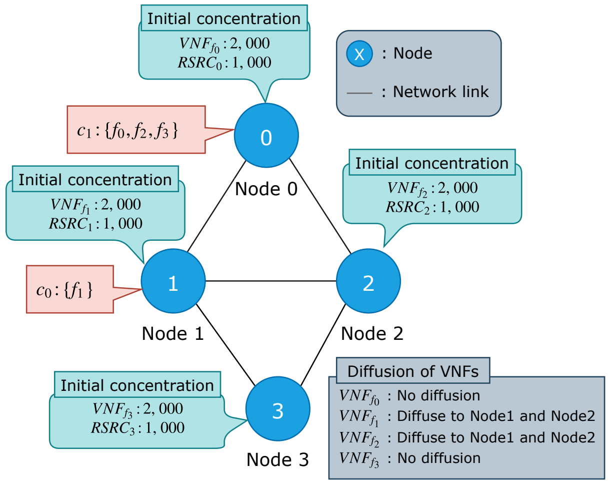


Figure 7 Scenario2: Network topology for simulation experiments

SFC 0 and SFC 1.

Figure 6(a) shows the situation where the system is operated normally. In Figure 6(b), a network link between server 0 and server 2 is disconnected due to network failures. Then, server 0 forwards flow packets via server 1 to continue the service. Additionally, VNF 2 is migrated to server 1 to reduce the number of hops. In Figure 6(c), the number of user devices increases and server 1 becomes busy, that means all packets cannot be processed only at server 1. Then, VNF 1 and VNF 2 are executed at server 1 and server 2 in a distributed manner.

3.3.2 Network Topology and Parameter Settings for Simulation Experiments

Figure 7 depicts the network topology for Scenario 2, that consists of four nodes and five links. Node 0, node 1, node 2, and node 3 correspond to server 0, server 1, server 2, and server 3 in Figure 6, respectively. VNF_{f_0}, VNF_{f_1}, VNF_{f_2}, and VNF_{f_3} correspond to VNF 0, VNF 1, VNF 2, and VNF 3, and are initially deployed on node 0, node 1, node 2, and node 3, respectively. There are two flows with SFC requests $c_0 = \{f_1\}$ and $c_1 = \{f_0, f_2, f_3\}$, corresponding to the flows with SFC 0 and SFC 1, that are called flow 0 and flow 1, respectively. When VNF_{f_0} is executed to the flow with c_1, c_1 changes as $c_2 = \{f_2, f_3\}$. When VNF_{f_2} is executed to the flow with c_2, c_2 changes as $c_3 = \{f_3\}$. The flows with SFC requests c_2 and c_3 are called flow 2 and flow 3, respectively.

Table 2 Scenario2: Temporal change in rate of flows

Flow	Rate	
	$0 \leqq t \leqq 2,000$ [msec]	$2,000 < t \leqq 3,000$ [msec]
flow 0	16.6 [Kpps]	50 [Kpps]
flow 1	33.3 [Kpps]	33.3 [Kpps]

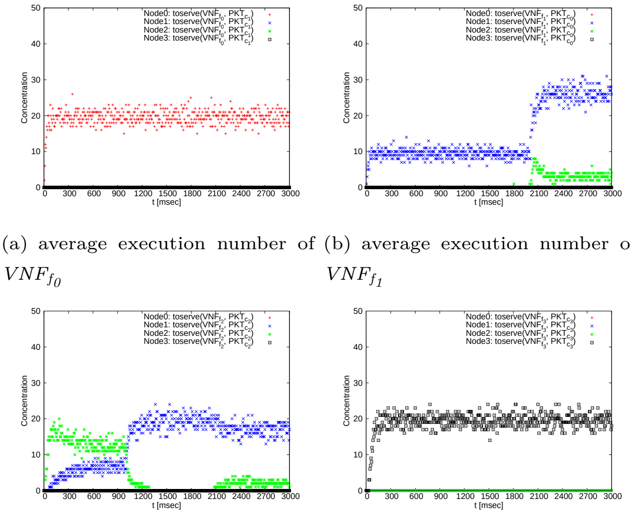
The simulation time is 3,000 [msec]. The initial concentrations of VNF_{f_0}, VNF_{f_1}, VNF_{f_2}, and VNF_{f_3} are set to 2,000. The initial concentrations of $RSRC$ at all nodes are set to 1,000. Table 2 shows the temporal change in flow rates. In the table, t is defined as simulation time. For $0 \leqq t \leqq 2,000$, packets of flow 0 arrive at node 1 at 10 packets per time step, corresponding to 16.6 [Kpps]. Packets of flow 1 arrive at node 0 at 20 packets per time step, corresponding to 33.3 [Kpps]. At $t = 2,000$, the rate of flow 0 is increased to 30 packets per time step, corresponding to 50 [Kpps]. Note that for $0 \leqq t \leqq 2,000$, node 1 processes all incoming packets, and for $2,000 < t \leqq 3,000$, node 1 cannot process all packets. In addition, at $t = 1,000$, a network link between node 0 and node 2 is disconnected.

When we configure the diffusion of VNFs so that all VNFs can be diffused to any other nodes, the concentrations of VNF_{f_1} and VNF_{f_2} increase at node 0 by executions of Reaction Equation (2) because packets of flow 1 arrive at node 0. Then, VNF_{f_1} and VNF_{f_2} are executed at node 0, and we cannot confirm the behaviors in Scenario 2. Therefore, the diffusion areas of VNF_{f_1} and VNF_{f_2} are limited to node 1 and node 2, and the reaction rate coefficient r_{mf} for VNF_{f_0} and VNF_{f_3} of Reaction Equation (5) is set to 0, so that the route of flow 1 is adequately changed on network failure, and that VNF_{f_1} and VNF_{f_2} are executed at node 1 and node 2 in a distributed manner, when the amount of traffic increases.

3.3.3 Simulation Results and Discussion

Figure 8 plots the average number of executions of Reaction Equation (2). Figures 8(a), 8(b), 8(c), and 8(d) are results for VNF_{f_0}, VNF_{f_1}, VNF_{f_2}, and VNF_{f_3}, respectively. Figure 9 shows the temporal change in the concentrations of VNF_{f_0}, VNF_{f_1}, VNF_{f_2}, and VNF_{f_3}. Figure 10 shows the temporal change in the concentrations of $RSRC$ at all nodes.

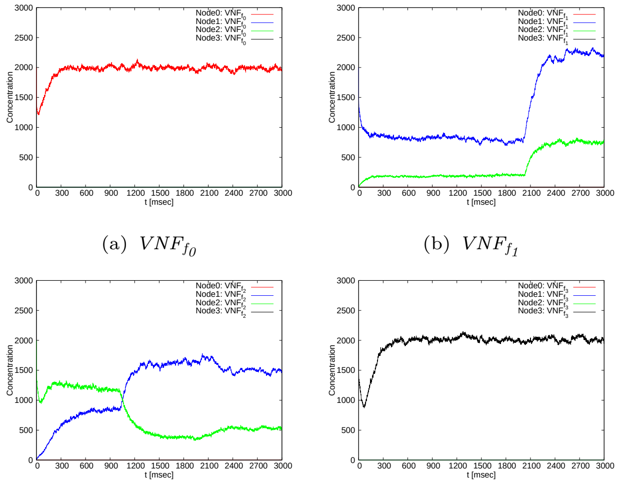
For $0 \leqq t \leqq 1,000$, VNF_{f_1} is executed at node 1 in Figure 8(b). It can be confirmed from the concentration of VNF_{f_1} in Figure 9(b). VNF_{f_0}, VNF_{f_2}, and VNF_{f_3} are executed at node 0, node 2, and node 3 in Figures 8(a), 8(c), and 8(d). For $1,000 < t \leqq 2,000$, VNF_{f_2} is executed at node 1. This is because VNF_{f_2} is migrated to node 1, where



(a) average execution number of VNF_{f_0} (b) average execution number of VNF_{f_1}

(c) average execution number of VNF_{f_2} (d) average execution number of VNF_{f_3}

Figure 8 Scenario2: Average number of executions of Reaction Equation (2)



(a) VNF_{f_0} (b) VNF_{f_1}

(c) VNF_{f_2} (d) VNF_{f_3}

Figure 9 Scenario2: Temporal change in the concentration of *VNF*

packets of flow 1 arrive after VNF_{f_0} is applied. This behavior realizes that server 0 forwards flow packets via server 1 and VNF 2 is migrated to server 1, as depicted in Figure 6. For $2,000 < t \leqq 3,000$, VNF_{f_1} and VNF_{f_2} are executed at both of node 1 and node 2 in a distibuted manner. This is because node 1 has insufficient resources to execute VNF_{f_1} and VNF_{f_2}. It can be confirmed from the concentration of $RSRC_1$ in Figure 10. From the above results, we confirmed that the behaviors in Scenario 2 can be achieved.

4. Conclusion and Future Work

In this report, we evaluated the performance of the NFV system based on biochemically-inspired tuple space model, and presented its implementation design. Specifically, we explained the tuple space model using biochemical reactions and how to apply the model to NFV system. We then performed computer simulation experiments assuming two situations in the NFV system. We confirmed that the proposed

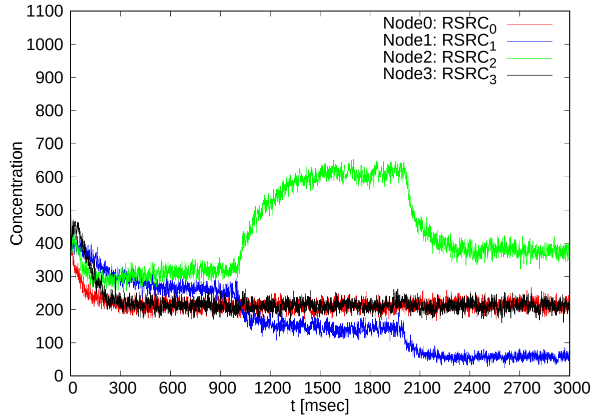


Figure 10 Scenario2: Temporal change in the concentrations of *RSRC* at all nodes

method can cope with dynamical environmental changes in the NFV system.

For future work, we plan to extend the proposed method to include more factors of the actual network environment, such as the effect of the propagation delay and the link bandwidth between tuple spaces.. It is also necessary to achieve discrete resource allocation to VNFs to accommodate the CPU core-based resource control in the current virtualized computing environment. Furthermore, it is also important to implement and evaluate the NFV system based on the proposed method.

References

[1] ETSI, “Network Function Virtualisation - White Paper 1.” available at `https://portal.etsi.org/nfv/nfv_white_paper.pdf`.

[2] J. G. Herrera and J. F. Botero, “Resource Allocation in NFV: A Comprehensive Survey,” *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 518–532, Sep. 2016.

[3] A. Engelmann and A. Jukan, “A Reliability Study of Parallelized VNF Chaining,” in *Proceedings of IEEE International Conference on Communications (ICC)*, pp. 1–6, May 2018.

[4] Z. Allybokus, K. Avrachenkov, J. Leguay, and L. Maggi, “Multi-path Alpha-fair Resource Allocation at Scale in Distributed Software-Defined Networks,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 12, pp. 2655–2666, Dec. 2018.

[5] M. Viroli, M. Casadei, S. Montagna, and F. Zambonelli, “Spatial Coordination of Pervasive Services through Chemical-inspired Tuple Spaces,” *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 6, no. 2, pp. 1–24, June 2011.

[6] Koki Sakata, “Adaptive and Autonomous Placement Method of Virtualized Network Functions based on Biochemical Reactions,” Master’s thesis, Osaka University, Feb. 2018.

[7] R. Goldberg, Y. B Tewari, and T. Bhat, “Thermodynamics of Enzyme-catalyzed Reactions,” *Science Direct*, vol. 20, no. 16, pp. 2874–2877, Dec. 2004.

[8] H. Li, Y. Cao, L. Petzold, and D. T Gillespie, “Algorithms and Software for Stochastic Simulation of Biochemical Reacting Systems,” *Biotechnology Progress*, vol. 24, pp. 56–61, Feb. 2008.