

DPI-BASED CONGESTION CONTROL METHOD FOR SERVERS AND NETWORK LINES

Shin-ichi Kuribayashi

Department of Computer and Information Science, Seikei University, Japan

ABSTRACT

The use of Deep Packet Inspection (DPI) equipment in a network could simplify the conventional workload for system management and accelerate the control action. The authors proposed a congestion control method that uses DPI equipment installed in a network to estimate overload conditions of servers or network lines and, upon detecting an overload condition, resolves congestion by moving some virtual machines to other servers or rerouting some communication flows to other routes. However, since the previous paper was focused on confirming the effectiveness of using DPI technology, it assumed some restrictive control conditions.

This paper proposes to enhance the existing DPI-based congestion control, in order to dynamically select an optimal solution for cases where there are multiple candidates available for: virtual machines to be moved, physical servers to which virtual machines are to be moved, communication flows to be diverted, and routes to which communication flows are to be diverted. This paper also considers server congestion for cases where computing power congestion and bandwidth congestion occur simultaneously in a server, and line congestion for cases where the maximum allowable network delay of each communication flow is taken into consideration. Finally, the feasibility of the proposed methods is demonstrated by an evaluation system with real DPI equipment.

KEYWORDS

Deep packet inspection, congestion control, network, server

1. INTRODUCTION

The currently used method of identifying application types used on servers in a data center is expensive because it is necessary, for example, to introduce a packet capturing tool for each server and have engineers well-versed in these application analyses how the applications are used. The use of a DPI device [1]-[3] can eliminate these problems because all that is required is to install the DPI device anywhere in the network rather than installing one for each server. The DPI device can analyze data flows in the network in detail. It makes it possible to monitor application types used by and the amount of traffic carried for each server more easily than before. In addition, DPI technology offers the potential for clearing congestion in a server or in a network line more rapidly than before.

Considering these advantages of using DPI, the authors previously proposed a congestion control method that works as follows [4],[5]. The CPU usage rate of each server is estimated by installing, in a network, a DPI device which constantly monitors the number of simultaneous TCP connections to each server. When the congestion of computing power is detected, it is resolved by moving virtual machines to other servers. In addition, the usage rate of server access bandwidths is estimated by using the DPI device, which constantly monitors the usage rate of bandwidth and application types used by each virtual machine in each server. Furthermore, the DPI device constantly monitors the volume of traffic and application types on each line in the network. If any line congestion is detected, the congestion is resolved by diverting some communication flows on that line to other lines or routes.

Since the main aim of References [4] and [5] was to confirm the effectiveness of using DPI technology, it assumed restrictive control conditions. For example, the server to which virtual machines are moved was fixed and it was assumed that computing power congestion and bandwidth congestion do not occur simultaneously in a server. This paper generalizes the evaluation conditions and proposes to enhance the existing DPI-based congestion control proposed in References [4] and [5], in order to dynamically select an optimal solution for cases where there are multiple candidates available for virtual machines, physical servers, communication flows and alternate routes. The paper also considers server congestion for cases where computing power congestion and bandwidth congestion occur simultaneously in a server, and line congestion for cases where the maximum allowable network delay is defined for each communication flow. In other words, a communication flow is not diverted to any route that cannot satisfy the allowable network delay.

The rest of this paper is organized as follows. Section 2 explains related works. Section 3 proposes to enhance the existing server congestion control method with DPI technology, and confirms the feasibility of the proposed method using an evaluation system with real DPI equipment. Section 4 proposes to enhance the existing line congestion control method with DPI technology. As in Section 3, Section 4 confirms the feasibility of the proposed method using an evaluation system. Section 5 presents the conclusions. This paper is an extension of the study in Reference [19].

2. RELATED WORK

As described in Reference [5], Reference [6] has proposed a network management system with DPI server which supports the network management system to classify the network traffic. For example, if the DPI equipment finds that the user id of the VoIP session is in the black list, the DPI equipment informs the NMS about this SIP session. Then the NMS configures related switches to intercept that VoIP session. Reference [7] has proposed to apply DPI technology to the CDMA mobile network packet switch domain and constructed a DPI-based network traffic monitoring, analysis and management system. Reference [8] has presented a technical survey for the implementation and evaluating of traffic classification modules under a common platform. Reference [9] has described the hardware and software components of the platform of DPI with its four utilization fields. Reference [10] has proposed a method, based on genetic algorithms, that optimizes the cost of DPI engine deployment, minimizing their number, the global network load and the number of unanalyzed flows.

Most of these studies mainly try to restrict the transmission of specific types of traffic. To the best of our knowledge, applying DPI technology to server congestion control and the reduction in server power consumption has not been fully studied.

3. ENHANCED SERVER CONGESTION CONTROL METHOD WITH DPI TECHNOLOGY

3.1 Overview of the Enhanced Method

This section proposes to enhance the existing server congestion control method with DPI technology, in order to handle cases where there are multiple candidates available for: virtual machines to be moved and servers to which virtual machines can be moved. The method also handles cases where computing power congestion and bandwidth congestion occur simultaneously in a server.

Moving a virtual machine to deal with bandwidth congestion is shown in Figure 1. There are n virtual machines (VMs) running on the server and m destination server candidates. The access bandwidth to the server is congested. In this example, one virtual machine, VM2, is dynamically selected and moved to one selected server, server # m . The proposed server congestion control algorithm is described below. This is an extension of the algorithm proposed in References [4] and [5].

<Assumption> There are n virtual machines ($VM_1 \sim VM_n$) running on a server that is being monitored. A DPI equipment installed in the network periodically monitors or estimates the computing power and bandwidth used by each virtual machine, and thereby estimates the usage rates of the total computing power and bandwidth of the server.

<Step 1> If there is any virtual machine whose computing power usage rate or bandwidth usage rate far exceeds a certain value for k_1 times consecutively, a measure is taken to reduce the computing power usage rate or the access bandwidth usage rate to a certain value or lower. Specifically, any new attempt to set up a TCP connection to that virtual machine is rejected or the volume of traffic towards that virtual machine is restricted.

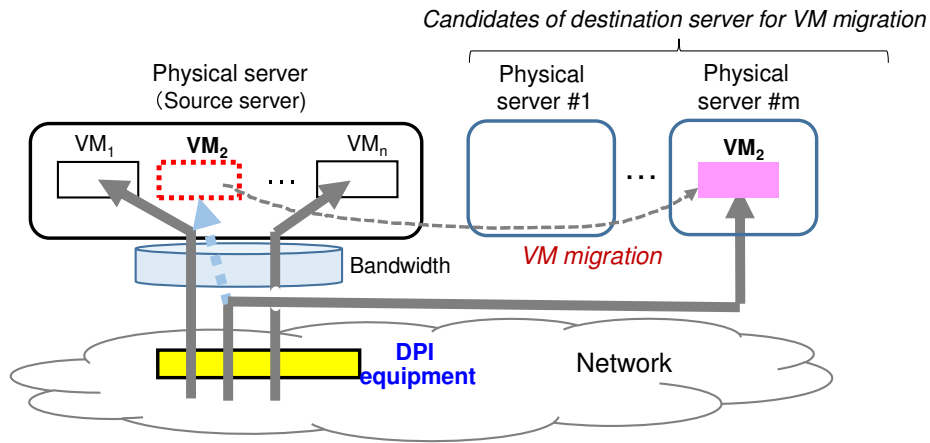


Figure 1. Image of server congestion control by VM migration

<Step 2> If, in spite of the measures taken in Step 1, either the computing power usage rate or bandwidth usage rate of the server exceeds a threshold for k_2 times consecutively, it is determined that either computing power congestion or bandwidth congestion has occurred, and the following measure is taken. Specifically, it is attempted to reduce either the computing power usage rate or the bandwidth usage rate of the server, by selecting one virtual machine in the server and moving it to another server (one server is selected out of n destination server candidates). If, for some reason, it is difficult to move the selected virtual machine, another virtual machine is selected as a candidate to be moved. If it is found that it is difficult to move any of all remaining VM candidates, the DPI device, for example, uniformly reduces the volume of traffic to that server to $\beta\%$ (e.g., $\beta=20$) of the original volume.

The following six alternative methods can be considered for selecting the virtual machine in the server to be moved. To ensure that moving a virtual machine is really effective, only those virtual machines that use $\alpha\%$ (e.g., $\alpha=10$) or more of the total computing power and bandwidth of the server become candidates for selection.

<In the case where the computing power is congested>

- Method 1-1: virtual machine that is using the most computing power
- Method 1-2: virtual machine that is using the least computing power.
- Method 1-3: virtual machine that handles a specific application (ex. P2P)

<In the case where the bandwidth is congested>

- Method 1-4: virtual machine that is using the most bandwidth
- Method 1-5: virtual machine that is using the least bandwidth.
- Method 1-6: virtual machine that handles a specific application

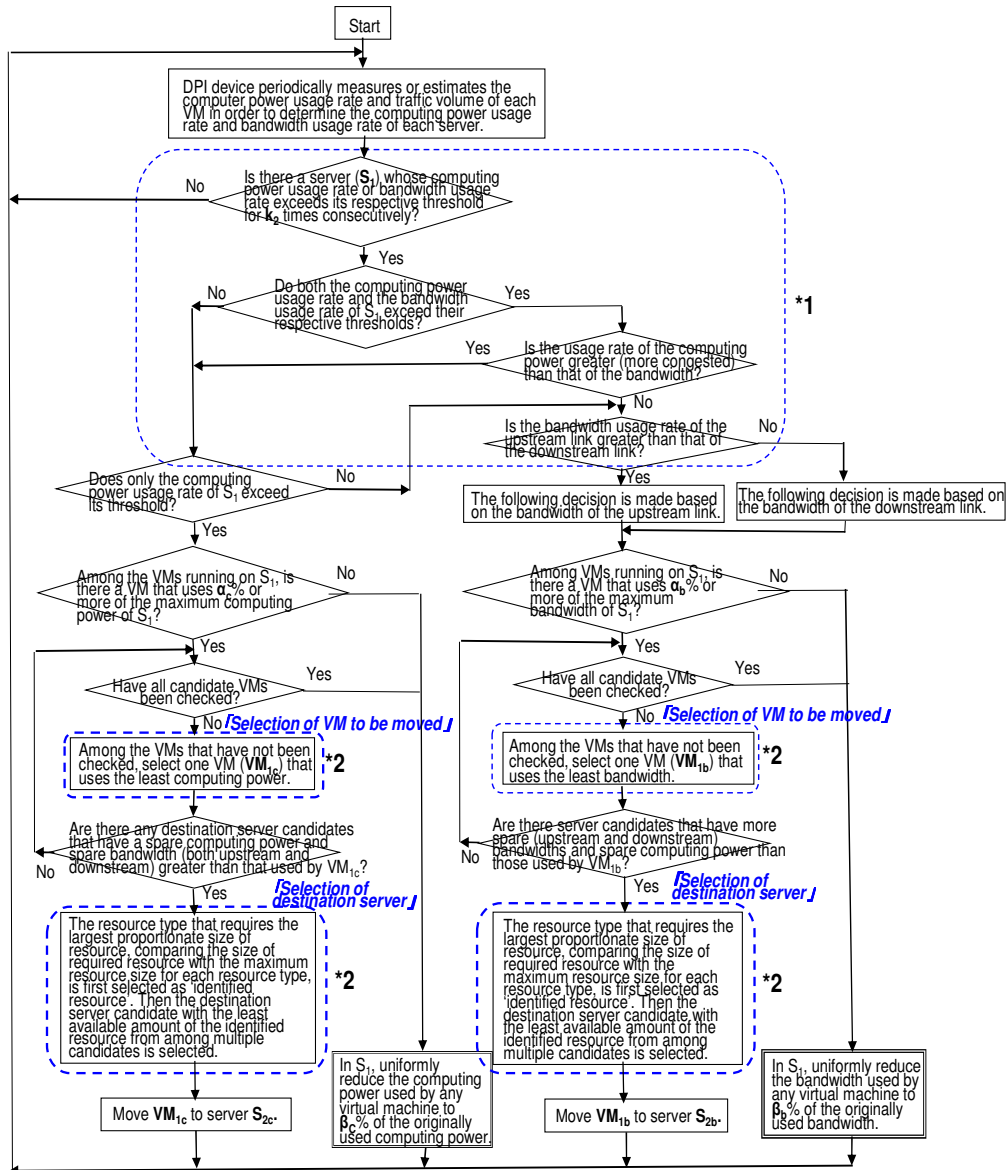
It is assumed that the more computing power or bandwidth a virtual machine uses, the longer time will be required to move that virtual machine. Giving priority to methods with a shorter time to move a virtual machine, we adopt Methods 1-2 and 1-5 this time. If the computing power and the access bandwidth of a server are congested simultaneously, the method related to the resource with a higher usage rate is adopted.

In selecting the destination server, the following three resource types should be simultaneously taking into consideration: computing power of the destination server candidate, access bandwidth of the destination server candidate, and bandwidth of the route to be changed along with the movement of the virtual machine. It is supposed that the route can be determined when the destination server is determined. It is proposed to apply the same idea proposed in References [11] and [12] for the optimal cloud resource allocation. That is, the resource type that requires the largest proportionate size of resource, comparing the size of required resource with the maximum resource size for each resource type, is first selected as 'identified resource'. Then the destination server candidate with the least available amount of the identified resource from among multiple candidates is selected.

Figure 2 illustrates the control flow of the proposed method for server congestion, which is based on Methods 1-2 and 1-5, and the destination server selection method proposed above. This figure does not contain the processing flow of Step 1. The parts that differ greatly from the existing processing flow in References [4] and [5] are indicated by *1 and *2. *1 is the part where the computing power and the bandwidth are considered simultaneously. *2 is the part where a virtual machine to be moved and the destination server are dynamically selected.

3.2 Confirmation of the Operation of the Proposed Method

Here we confirm the operation of the proposed method for a case of bandwidth congestion. The evaluation system used is shown in Figure 3. which is an application program we developed, in the DPI management server is modified to suit the proposed congestion flow in Figure 2. The DPI equipment device is Net Enforcer AC-502 [13] from ALLOT Communications. Virtual Box [14] is used to implement virtualization. Both VM1 and VM2 operating on the source server communicate with terminal #1, and the traffic volumes of VM1 and VM2 are 10 Mbps and 15 Mbps, respectively. It is assumed that the access bandwidth of the source server is congested in this example. The VM running on destination server #1 and terminal #4 are both communicating with terminal #2. Both the VM running on destination server #2 and terminal #5 communicate with terminal #3. Traffic from the destination server #1 and terminal #4 go through Route 1, and the traffic on destination servers #2 and terminal #5 go through Route 2. It is assumed that Route 2 is congested in this example. All communication passes through the DPI device. The DPI tool, which is an application program we developed, in the DPI management server collects traffic data from the DPI device and instructs the live migration of a virtual machine (VM) to the source server and destination server #1, according to the control flow in Figure 2. Live migration is executed by sending a live migration command of Virtual Box from DPI tool to both the source server and the selected destination server with Ps Exec [15] from Microsoft.



*1: Related to considering computing power and bandwidth simultaneously.
 *2: Related to **dynamic selection** to suit the particular conditions.

Figure 2. Control flow of the proposed method for server congestion

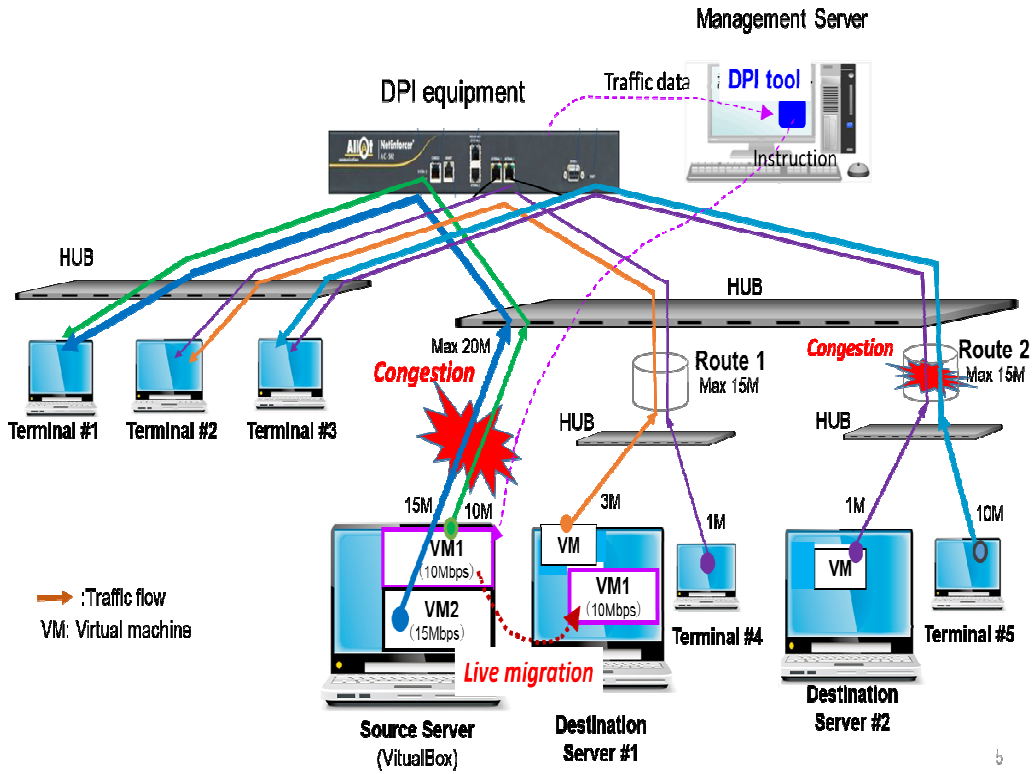


Figure 3. Traffic flows and control instructions for evaluation

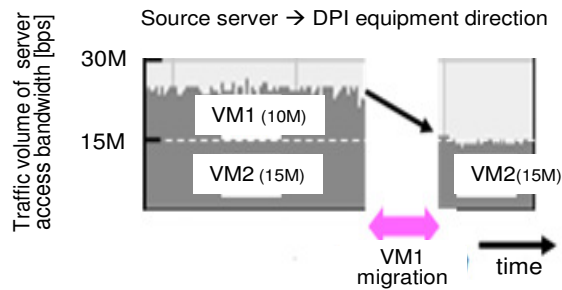


Figure 4. Traffic reduction of congested server access bandwidth by VM migration

It is examined whether the evaluation system operated as expected by the proposed method. Since there are two candidates (VM1 and VM2) for a virtual machine to be moved, two destination server candidates (servers #1 and #2) and two route candidates (Route 1 and Route 2), there are eight combinations. Operational conditions of these eight combinations were set up and it was confirmed that they all operate correctly. Figure 4 illustrates the case where VM1 is selected and access bandwidth at source server is decreased from 25Mbps to 15Mbps (as a result, the access bandwidth congestion at source server is alleviated).

In addition, it was confirmed that there was almost no service interruption due to the movement of the virtual machine.

4. ENHANCED LINE CONGESTION CONTROL METHOD IN A NETWORK WITH DPI TECHNOLOGY

4.1 Overview of the Enhanced Method

As in References [4] and [5], an SDN-based [16] network is assumed. The SDN controller keeps track of the route and line in the network each traffic flow passes through. The DPI management system receives this information (including information about the speed of each line), combines it with traffic data collected by the DPI device to determine whether there is any line congestion in the network. In addition, when the DPI management system determines that a specific line is congested, it requests the SDN controller to divert some communication flows on the line to another route.

The method proposed here assumes that there are multiple candidates for a communication flow to be diverted and multiple route candidates to which the selected flow will be diverted. An example of diverting a communication flow when a line is congested is illustrated in Figure 5. The line linking Node x to Node y is congested. There are two flows (Flow #1 and Flow#2) carried over that line. Flow #1 is dynamically selected and diverted to Route #W, which is one of the candidates to which a communication flow can be diverted, as Flow #2 is not allowed to be diverted to Route #W which has a long network delay.

The enhanced line congestion control algorithm is described below. This is an extension of the line congestion control algorithm proposed in References [4] and [5].

<Assumption> There are g communication flows ($F_1 \sim F_g$) on a line being monitored. The DPI device installed in the network identifies communication flows on each line, and periodically estimates the total volume (V bps). If the usage rate of a line exceeds a certain threshold $\gamma\%$ (e.g., $\gamma=85$) k times consecutively, the DPI device determines that line congestion has occurred. In a manner similar to the server congestion control described in Section 3, if both the upstream and downstream lines are congested simultaneously, a measure is taken to resolve the congestion of the direction with a higher usage rate than the other direction.

<Step 1> If it is determined that a line is congested, some communication flows on that line are diverted to another route in order to reduce the total volume of flows on that line. There are W route candidates to which these flow can be diverted. The optimal route is selected. The routes that cannot satisfy the allowable network delay of each flow are excluded. If it is difficult to divert any communication flow to another route, or if diverting any flow does not sufficiently reduce the total volume of flows, the DPI device takes a different action, such as restricting some traffic, as was the case in Section 3.

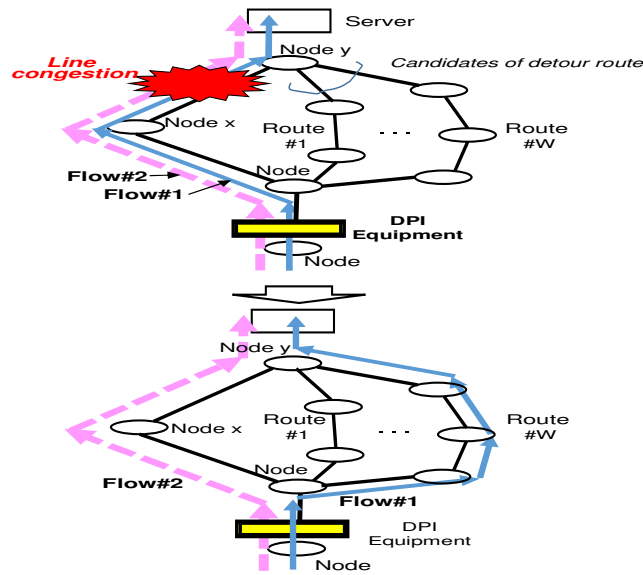


Figure 5. Image of line congestion control by detouring specific flows

There can be five methods of selecting a communication flow on the congested line to be diverted in Step 1. However, as was the case in Section 3 for selecting a virtual machine to be moved, only those communication flows that use $\alpha\%$ (e.g., $\alpha=10$) or more of the bandwidth of the line are considered for selection.

- Method 2-1: flow with the least volume of traffic
- Method 2-2: flow with the most volume of traffic
- Method 2-3: flow with the longest allowable network delay
- Method 2-4: flow with the shortest allowable network delay
- Method 2-5: flow of the specific application

Giving priority to methods that have the least impact on the route to which a flow is to be diverted, it is proposed to adopt Method 2-1. There can be five methods of selecting an alternate route to which a flow is to be diverted. It is noted that a communication flow is not diverted to any route that cannot satisfy the allowable network delay of that flow.

- Method 3-1: route that satisfies the allowable network delay of the flow to be diverted and has the least spare bandwidth
- Method 3-2: route that satisfies the allowable network delay of the flow to be diverted and has the largest spare bandwidth
- Method 3-3: route that satisfies the allowable network delay of the flow to be diverted and has the longest network delay
- Method 3-4: route that satisfies the allowable network delay of the flow to be diverted and has the shortest network delay
- Method 3-5: route that handles a specific application

As we did in Method 2-1, it is proposed to adopt Method 3-1 because we give priority to cram bandwidth into the particular route cramming traffic, in order to be able to meet future demands for large bandwidths.

Figure 6 illustrates the control flow of the proposed method for line congestion which is based on Methods 2-1 and 3-1. This figure does not contain the processing flow of Step 1. The parts that differ greatly from the existing processing flow are indicated by *1 and *2. *1 is the part where the communication flow to be diverted and the route to which the communication flow to be diverted are selected dynamically. *2 is the part where the maximum allowable network delay is considered in the selection of communication flow and alternate route.

4.2 confirmation of the Operation of the Proposed Method

The evaluation system used is more or less the same as that used in References [4] and [5]. The DPI equipment and the management server are the same as those in Figure 3. Each switch is a general-purpose server with VyOS [17] which is a Linux-based network operating system that provides software-based network routing. Line congestion is determined based on the volume of each flow measured by the DPI equipment. When the congestion is detected, DPI tool selects the flow to be diverted and the route to which it is to be diverted. And it sends an instruction for executing this diversion. Again, as in Reference [5], we did not use any SDN controller for rerouting. Instead, we used a plink [18] tool (The SSH-based tool used to remotely execute command lines from a Windows terminal) to remotely rewrite the routing information in the VyOS switch for rerouting.

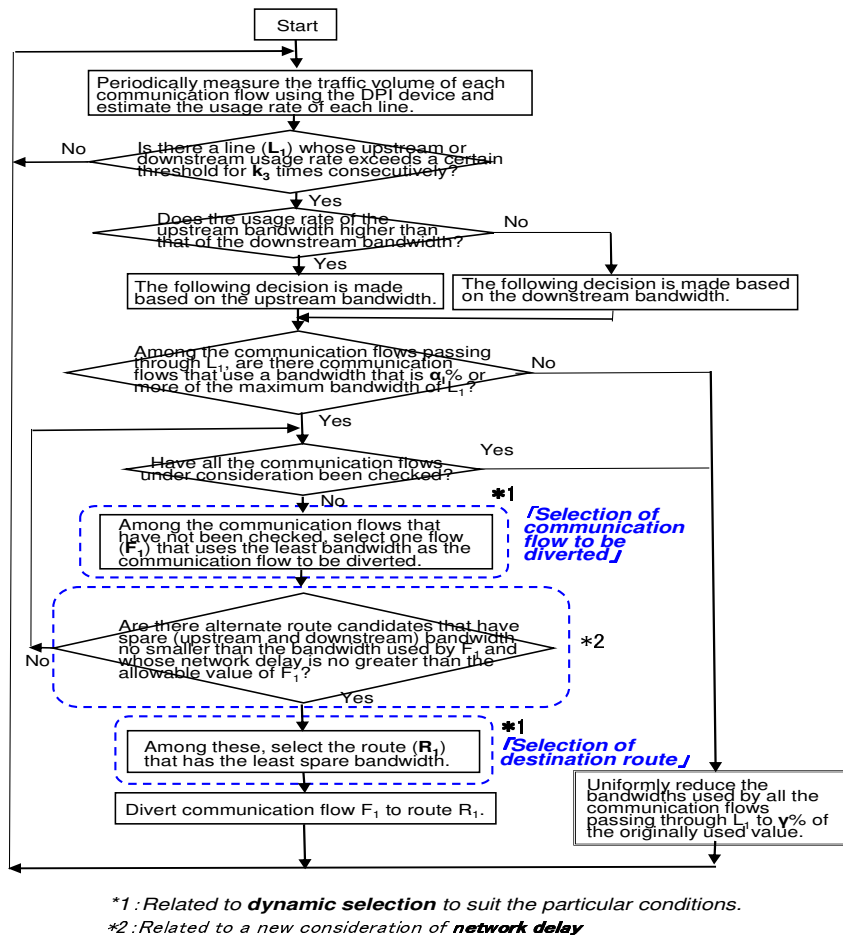


Figure 6. Control flow of the proposed method for line congestion

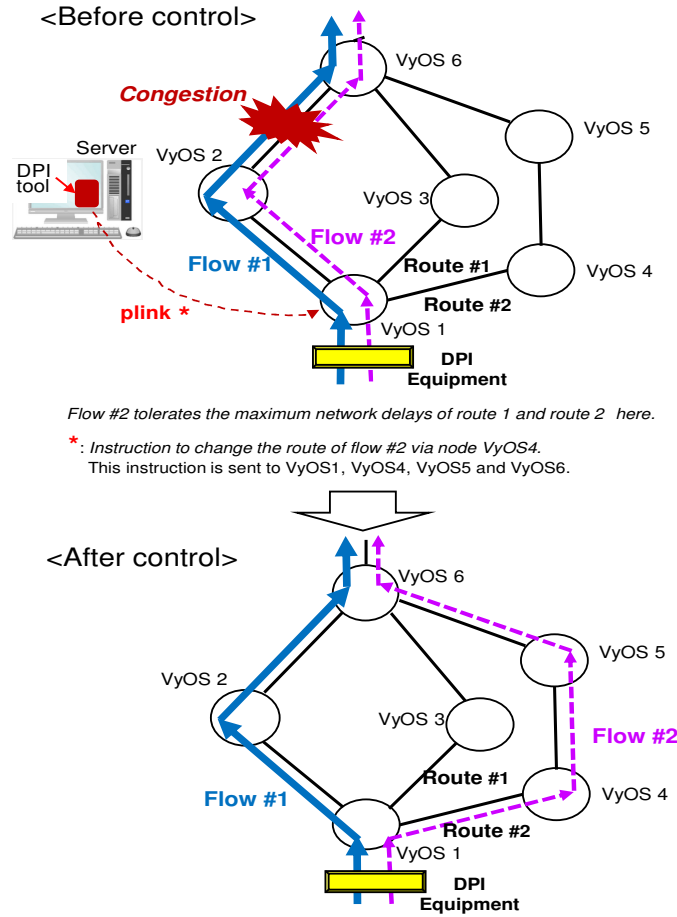


Figure 7. Example of traffic flows and control instructions for evaluation

The relations between the flows considered in this evaluation and routes are shown in Figure 7. Both Flow#1 and Flow#2 are through a line connecting node VyOS2 to node VyOS6. When this line is congested, the congestion is resolved by diverting Flow#2 to Route #2 which goes through nodes VyOS4 and VyOS5. The allowable network delay of the flows is taken into consideration. In Figure 7, both Route#1 and Route#2 satisfies the allowable network delay of Flow#2.

Since there are two flow candidates to be diverted and two alternative route candidates to which a flow would be diverted, there are 4 combinations. Operational conditions of these four combinations are set up and it is confirmed that they all operated correctly. It is also confirmed that diverting a flow causes no interruption to communication.

5. CONCLUSIONS

DPI technology offers potential that congestion on a server or a network line can be estimated more easily and resolved more rapidly than before. This paper has proposed to enhance the existing DPI-based congestion control, in order to dynamically select a solution optimal for the current conditions for cases where there are multiple candidates available for: virtual machines to be moved, physical servers to which virtual machines are to be moved, communication flows to be diverted, and routes to which communication flows are to be diverted. It was proposed to consider the usage status of bandwidth of the route to be changed along with the movement of the

virtual machine. The paper also has considered server congestion for cases where computing power congestion and bandwidth congestion occur simultaneously in a server, and line congestion for cases where the maximum allowable network delay of each communication flow is taken into consideration. The feasibility of the proposed methods has been confirmed by an evaluation system with real DPI equipment.

It will be necessary to study how to determine the optimal control parameter values which will depend on application type and traffic characteristics. As the proposed method can be also applied to DDoS attacks on the servers, it is required to study how to apply it.

ACKNOWLEDGMENT

We would like to thank Mr. Kenichiro HIDA for his help with the evaluation.

REFERENCES

- [1] M.Finsterbusch, C.Richter, E.Rocha, J.A. Muller, and K.Hanßgen, "A Survey of Payload-Based Traffic Classification Approach," *IEEE Communications Surveys & Tutorials*, Vol. 16, No. 2, 2014.
- [2] A. Callado, C. Kamienski, G. Szabo, B. Gero, J. Kelner, S. Fernandes, and D. Sadok, "A Survey on Internet Traffic Identification," *IEEE Commun. Surveys & Tutorials*, vol. 11, no. 3, pp. 37–52, 2009.
- [3] M.Kassner, "Deep Packet Inspection: What you need to know" <http://www.techrepublic.com/blog/data-center/deep-packet-inspection-what-you-need-to-know/>
- [4] K. Yanagisawa and S.Kuribayashi, "Use of DPI Technology for Server Congestion Control and Reduction of Power Consumption by servers", *Proceeding of 2015 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (Pacrim15)*, C1-2, Aug. 2015.
- [5] S.Kuribayashi, "Server Congestion Control and Reduction of Server Power Consumption with DPI Technology", *International Journal of Computer Networks & Communications (IJCNC)* Vol.7, No.5, pp.83-94, Sep.2015.
- [6] C.S. Yang etc. "A Network Management System Based on DPI", *13th International Conference on Network-Based Information Systems (NBIS2010)*, pp.385-388.
- [7] X.Lu, "A Real Implementation of DPI in 3G Network," *2010 IEEE Global Telecommunications Conference (GLOBECOM 2010)*.
- [8] M.Finsterbusch, C.Richter, E.Rocha, J.A.Muller, and K.Hanbgen, "A Survey of Payload-Based Traffic Classification Approaches," *IEEE COMMUNICATIONS SURVEYS & TUTORIALS*, VOL. 16, NO. 2, SECOND QUARTER 2014.
- [9] Y.Lee, J.Oh, J.K.Lee, D.Kang and B.G.Lee, "The Development of Deep Packet Inspection Platform and Its Applications," *3rd International Conference on Intelligent Computational Systems (ICICS'2013)* January 26-27, 2013 Hong Kong.
- [10] M.Bouet, J.Leguay, and V.Conan, "Cost-based placement of virtualized Deep Packet Inspection functions in SDN," *2013 IEEE Military Communications Conference*, pp.992-997, Nov. 2013.
- [11] S.Kuribayashi, "Optimal Joint Multiple Resource Allocation Method for Cloud Computing Environments", *International Journal of Research and Reviews in Computer Science (IJRRCS)*, Vol.2, No.1, pp.1-8, Feb. 2011.

- [12] S.Kuribayashi, "Resource Allocation Method for Cloud Computing Environments with Different Service Quality to Users at Multiple Access", International Journal of Computer Networks & Communications (IJCNC) Vol.7, No.6, pp.33-51, Nov.2015.
- [13] NetEnforcer, Allot Communications <http://www.allot.com/netenforcer.html>
- [14] VirtualBox <https://www.virtualbox.org/>
- [15] Windows Sysinternals " PsExec" <https://technet.microsoft.com/ja-jp/sysinternals/bb897553.aspx>
- [16] H. Kim and N. Feamster, "Improving network management with software defined networking," IEEE Communications Magazine, IEEE, vol. 51, No. 2, pp. 114–119, 2013
- [17] VyOS <http://www.vyos-users.jp/>
- [18] plink <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>
- [19] S.Kato, K.Hida and S.Kuribayashi, "Enhanced congestion control method for servers and network lines with DPI technology", Proceeding of 2017 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (Pacrim2017), Comm-S1, Aug. 2017.

AUTHOR

Shin-ichi Kuribayashi received the B.E., M.E., and D.E. degrees from Tohoku University, Japan, in 1978, 1980, and 1988 respectively. He joined NTT Electrical Communications Labs in 1980. He has been engaged in the design and development of DDX and ISDN packet switching, ATM, PHS, and IMT 2000 and IP-VPN systems. He researched distributed communication systems at Stanford University from December 1988 through December 1989. He participated in international standardization on ATM signaling and IMT2000 signaling protocols at ITU-T SG11 from 1990 through 2000. Since April 2004, he has been a Professor in the Department of Computer and Information Science, Faculty of Science and Technology, Seikei University. His research interests include optimal resource management, QoS control, traffic control for cloud computing environments and green network. He is a member of IEEE, IEICE and IPSJ.

