Cooperated Traffic Shaping With Traffic Estimation and Path Reallocation to Mitigate Microbursts in IoT Backhaul Network

KAZUAKI HONDA[1], **NAOTAKA SHIBATA**[1], **RINTARO HARADA**[1], **YOTA ISHIDA**[2],
KUNIO AKASHI[2,3], **SHIN KANEKO**[1], **TOSHIYUKI MIYACHI**[2], **AND JUN TERADA**[1]

[1]NTT Access Network Service Systems Laboratories, NTT Corporation, Yokosuka, Kanagawa 239-0847, Japan
[2]National Institute of Information and Communications Technology, Nomi, Ishikawa 923-1211, Japan
[3]Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan

Corresponding author: Kazuaki Honda (kazuaki.honda.ku@hco.ntt.co.jp)

ABSTRACT An aggregating switch (SW) network offers cost-effective accommodation to Internet-of-Things (IoT) traffic by aggregating the traffic. In such a network, it is crucial to eliminate the discarded traffic caused by the simultaneous transmission of massive IoT devices, namely microburst. Traffic shaping is a technique of storing traffic in one SW to mitigate microbursts. Conventional traffic shaping is limited because only one SW performs shaping with a limited queue length. Thus, we propose cooperated traffic shaping using multiple SWs to accommodate more traffic. We formulated equations to derive the minimum queue length with shaping rates which gradually decrease in geometric progression. To acquire the queue length using general SWs without short-cycle monitoring, we propose a scheme for estimating the instantaneous input rate and data size of microburst traffic required for our equations. If the calculated queue length cannot be prepared in the current path, we propose reallocating the path to another one with more SWs. We experimentally demonstrated the proposed coordinated traffic shaping technique by implementing it in commercial SWs with 125 emulated IoT devices. The results showed that the difference between the experimental and numerical results was below 4.2%, and the queue length can be reduced by 40% when there are three SWs. In addition, a path with two SWs was successfully reallocated to one with three SWs.

INDEX TERMS 5G mobile communication, the Internet of Things, optical fiber networks, traffic shaping.

I. INTRODUCTION

As 5G mobile services have begun to spread worldwide, the amount of data traffic has been increasing rapidly. These services take advantage of 5G features such as high speed/large capacity, massive connected devices, ultra low latency, and ultra reliability [1]. Internet-of-Things (IoT) devices are deployed for various applications such as camera monitoring and sensing. They use cellular systems as well as non-cellular wireless systems such as Long-Range Wide Area Network (LoRaWAN) [2], depending on the application requirements.

IoT devices are connected to IoT servers via IoT gateways (IoT-GWs), which provide a wireless coverage area via cellular or non-cellular connections. The gateways aggregate the IoT traffic in their coverage areas and send the aggregated data to the server via an optical network known as IoT backhaul. While each device only generates a small amount of traffic, the total amount of traffic is large because of the massive devices. In order to accommodate the traffic, a network consisting of Layer-2 or Layer-3 aggregation switches (SWs) would be suitable for IoT backhaul [3]. The network (NW) can cost-effectively accommodate such traffic by using statistical multiplexing.

Microbursts are sudden increases in traffic caused by the simultaneous transmission of massive devices. This presents a problem when accommodating IoT traffic in an aggregated NW [4]. When microbursts occur, the amount of traffic can exceed the queue length in SWs, resulting in discarded packets. In other words, an increase in delay occurs due to packet loss and retransmission. Retransmission should be avoided for IoT devices because they have limited battery life. A method of controlling the transmission timing for

The associate editor coordinating the review of this manuscript and approving it for publication was Hosam El-Ocla.

each service can be used to eliminate microbursts. However, because the NW accommodates the traffic of various independently-operated IoT service providers, it is not practical to adjust the timing for all of them.

In this paper, we propose cooperated shaping using SWs to accommodate a greater amount of traffic with a limited queue length. Our previous study [5] assumed that the input data size of microburst traffic was obtained. Since the accurate estimation of the input data size needs the short-cycle monitoring in SWs, the previous study has a limitation with respect to type of SWs. It is necessary to propose a new method that estimates not only input rate of microburst traffic but also the input data size. Therefore, we derive the minimum queue length through improved theoretical investigations. Then we introduce a method of acquiring the queue length by using general SWs without short-cycle monitoring. We improved the algorithm for cooperated shaping presented in our previous paper [5] to estimate the parameters required for the instantaneous input rate and data size of microburst traffic. If the calculated queue length cannot be prepared in the current path, we propose reallocating the path to one with more SWs in order to reduce the required queue length.

The rest of this paper is organized as follows. In Section II, we discuss related work on techniques for dealing with microbursts. Then in Section III, we describe the details of the proposed cooperated traffic shaping. In Section IV, we present our experimental results. Finally, Section V concludes this paper.

II. RELATED WORK

Several solutions have been proposed to mitigate microbursts, particularly in data center networks [6]–[11]. These solutions include transmission control protocol (TCP) pacing mechanisms to avoid TCP incast in SWs, predictive packet drop mechanisms to slow down the transmission rate of TCP flows, and explicit congestion notification (ECN) marking schemes from switches to servers.

End-host batching schemes can cause microburst traffic in a network. ECNs are widely used in data centers to mitigate microburst traffic by maintaining a low queue occupancy. However, the current ECN marking scheme based on instantaneous queue length may lead to problems such as buffer underflow [6]. Specifically, the current ECN marking scheme used in data centers is prone to spurious congestion signals, which may cause overreaction of the sender and oscillation of the switch queue length.

Adaptive pacing (AP) is a protocol that was proposed to dynamically adjust burstiness on the basis of flow concurrency to reduce the probability of TCP incast [10]. AP can be integrated transparently into different TCP protocols, such as NewReno and data center TCP (DCTCP). The design of AP is based on uniformly distributed traffic from the characteristics of the data center NW. However, IoT devices may access the network in a highly synchronized manner, for example, after a power outage [12]. Thus, the performance of AP may be affected in an aggregated NW.

A tiny packet program (TPP) was proposed in which the end host embeds a small program in a packet to actively query and manipulate the internal state of the network [13]. The TPP interface enables end hosts to have unprecedented visibility into the behavior of the network. However, coordinating with the network controller to control communication timing would make IoT devices more complex.

Traffic shaping is a method in which traffic is stored in SW buffer memories to reduce traffic flow to the IoT server. Compared with other methods, shaping can be applied to any protocol and is suitable for a network that accommodates various service providers. Conventional traffic shaping is limited because only one SW performs shaping [14].

We previously proposed a cooperated traffic shaping scheme for reducing the required queue length of each SW [5]. To minimize the queue length in the scheme, the optimal settings for each SW can be calculated by optimization models [15], [16]. However, it is difficult to measure some parameters with commercial SWs due to the burstiness of the traffic. The input data size of microbursts was assumed to be known in ref. [5], so a method for estimating the input data size is needed for cooperated shaping using general SWs without short-cycle monitoring. In addition, our previous study only considered linear topology. By considering the control of the path in a ring or mesh NW, it may be possible to further eliminate the discarded frames caused by microbursts.

III. COOPERATED SHAPING WITH TRAFFIC ESTIMATION IN AGGREGATED NETWORK

A. FORMULATION OF COOPERATED SHAPING

Figure 1 shows the architecture of an aggregated NW using cooperated traffic shaping techniques. Our proposed aggregated NW consists of SWs and a network controller (NWC) which configures the SW settings such as queue length and shaping rate. Each SW aggregates the uplink traffic from the IoT-GW and forwards it to the IoT server by statistical multiplexing. The traffic shaper and queue range of each SW is set on the basis of the buffer memory size. The queue here indicates the control for each VLAN-ID. The upper limit of the buffer memory size is assumed to be the same for all SWs. The IoT server sends a request for data transmission to each device via the IoT-GW, and the application is assumed to send data in response. In this case, the IoT devices transmit data simultaneously, and a microburst occurs with an input data size (d_{in}) and an input rate (r_{in}).

Here, the queue length of all SWs is calculated to be the minimum so that the buffer memory required for SWs is equalized for efficiency. q_k, the queue length of SW #k, should be set as follows

$$q_k = \frac{d_{in}(r_{k-1} - r_k)}{r_{k-1}}. \tag{1}$$

For $k = 1$, r_0 denotes the input rate r_{in} of microburst traffic to the first SW. r_n at the last SW #n is equal to r_t. The right-hand side of (1) indicates the accumulation size of the queue
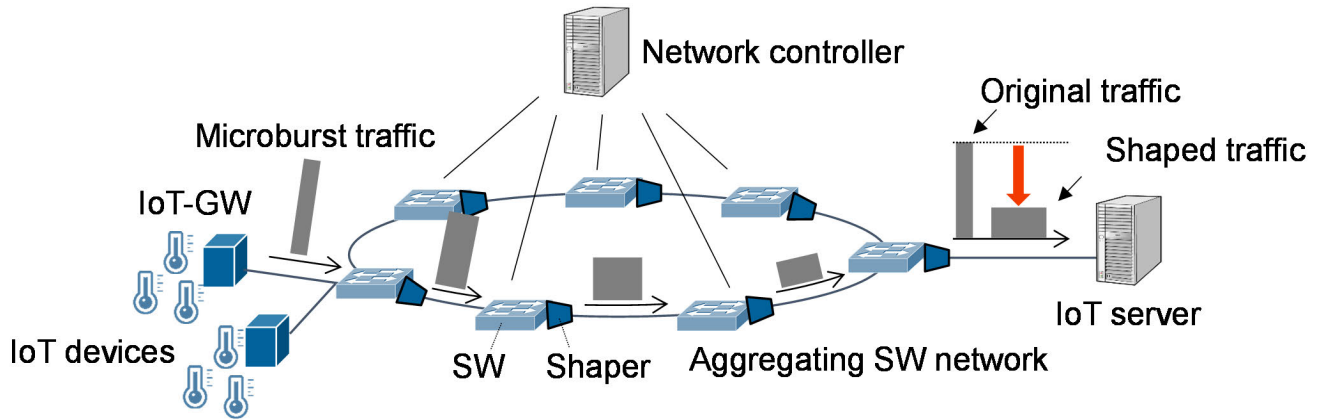
FIGURE 1. IoT backhaul network with cooperated traffic shaping deployed in ring topology.

that accumulates until the end of the burst. In other words, it is the multiplication of the accumulation rate in the queue, $r_{k-1} - r_k$, and the burst length input to SW #k, d_{in}/r_{k-1}. To shape the IoT traffic, r_k becomes smaller as #k becomes larger. In (1), the required q_k increases monotonically as r_{k-1} increases, and it decreases monotonically as r_{k-1} decreases. Considering q_{k+1} decreases monotonically as r_k increases, the longest queue length is minimized under when the queue length is constant (i.e., $q_k = q_{k+1}$). Calculating the shaping rate from this condition, we get $r_k = \sqrt{r_{k-1}r_{k+1}}$, which is a geometric progression. Thus, optimizing the shaping rate and the queue length yields

$$r_k = \left(\frac{r_t}{r_{in}}\right)^{\frac{k}{n}} r_{in}, \tag{2}$$

$$q_{min} = \left\{1 - \left(\frac{r_t}{r_{in}}\right)^{\frac{1}{n}}\right\} d_{in}, \tag{3}$$

where q_{min} is the minimized length of the longest queue.

B. IMPLEMENTATION OF COOPERATED TRAFFIC SHAPING
To know the values of r_{in} and d_{in}, the traffic at all of the ports in each SW should be continuously monitored because it is impossible to predict where microburst traffic will occur. Moreover, the monitoring cycle should be relatively short in order to calculate the instantaneous value accurately. However, short-cycle monitoring increases the SW cost. Our previous study [5] assumed that d_{in} was obtained accurately. A new estimation method is needed to estimate d_{in} and r_{in} without short-cycle monitoring so that it can be performed even by general SWs.

We propose estimating d_{in} and r_{in} through two-time estimation. Because the two unknown variables are the input rate and data size, the solution can be obtained with simultaneous equations from two different parameters. In this method, each SW measures the input rate and data size periodically within a specific time interval. A microburst of traffic is assumed to occur under the same conditions of r_{in} and d_{in}.

First, the NWC configures the shaping rate and queue length to the target rate r_t and the initial queue length q_{init}, respectively, for all SWs. The initial queue length q_{init} is determined from the minimum queue length in the SW specifications. In regular operation, the traffic counter in each SW measures discarded data size w_1 at a specific time interval. Once a SW observes dropped data (i.e., $w_1 > 0$), the discarded data size w_1 and queue length q_{init} are expressed by

$$w_1 + q_{init} = (r_{in} - r_t)\frac{d_{in}}{r_{in}}. \tag{4}$$

The equation includes two unknown values, d_{in} and r_{in}, so it cannot be solved. The NWC tentatively defines the SW as SW #0 and corrects the estimated input rate $r_{estimated}$ and input data size $d_{estimated}$ measured by the SW #1 traffic monitor. We set the estimated input rate $r_{estimated}$ as the average traffic rate in a monitoring interval. Then, using these estimates, the shaping rate of SW #k is set as follows,

$$r'_k = \left(\frac{r_{target}}{r_{estimated}}\right)^{\frac{k}{n}} r_{estimated} \tag{5}$$

$$q'_{init} = (r_{estimated} - r_{target}) * \frac{d_{estimated}}{r_{estimated}} - w_1. \tag{6}$$

Because the burst length is smaller than the monitoring interval, the estimated input rate $r_{estimated}$ becomes smaller than the instantaneous rate of the microburst that is the actual required value. Thus, the calculated queue length is shorter than the length actually required, and the second data discursion w_2 will occur at SW #1. From this result,

$$w_2 + q'_{init} = (r_{in} - r'_1) * \frac{d_{in}}{r_{in}} \tag{7}$$

can be obtained. From (4) and (7), we obtain the two unknown values d_{in} and r_{in} by the following:

$$r_{in} = \frac{(q'_{init} + w_2) r_t - (q_{init} + w_1) r'_1}{(q'_{init} + w_2) - (q_{init} + w_1)} \tag{8}$$

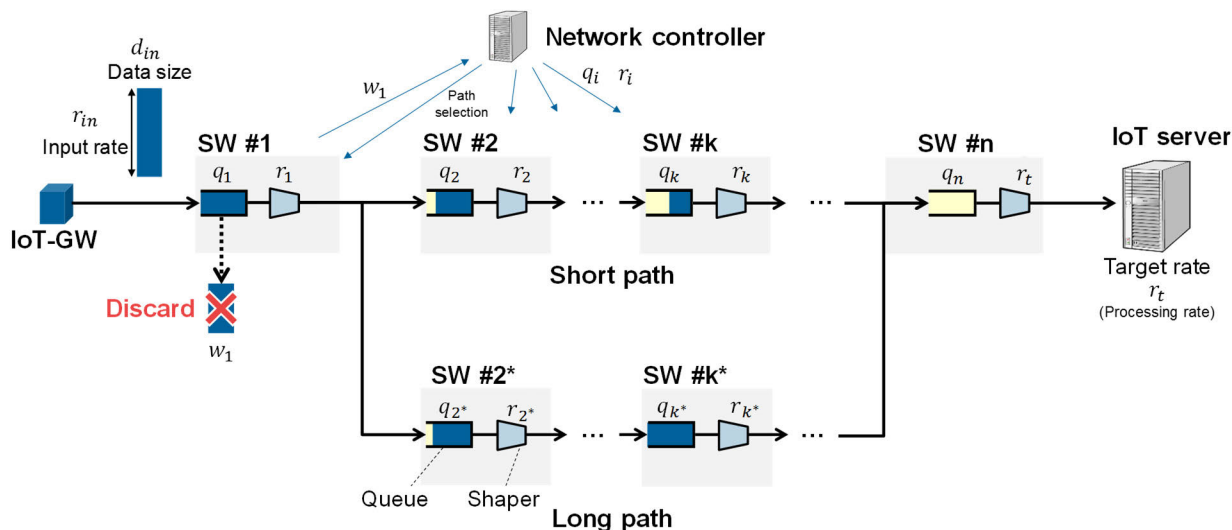$$d_{in} = \frac{(q'_{init} + w_2) r_t - (q_{init} + w_1) r'_1}{r'_1 - r_{in}} \tag{9}$$

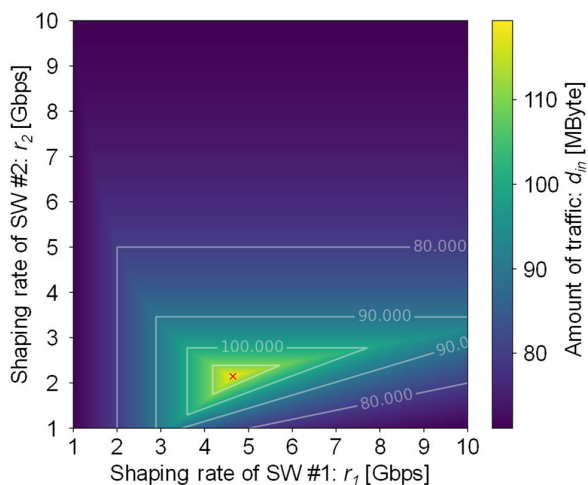**FIGURE 2.** Queue length and traffic shaping calculation model.

FIGURE 3. Maximum d_{in} determined by cooperated traffic shaping. Red x: theoretical shaping rate calculated by (2).

Then the NWC calculates the shaping rate and queue length of each SW by using (8) and (9) and sends the values to each SW.

C. PATH REALLOCATION

In the IoT backhaul, the transmission path candidates of the flow are based on the topology of the NW. As shown by (3), as the number of SWs increases, the required queue length per SW decreases. The number of SWs is determined by the path that is selected. The NWC generally sets the shortest path to each IoT flow to use the resources of the queues in the NW efficiently. If the required queue length calculated by the shortest path exceeds the upper limit of the SW, we propose changing the path to reduce the queue length required for each SW. This will increase the amount of traffic that can be accommodated.
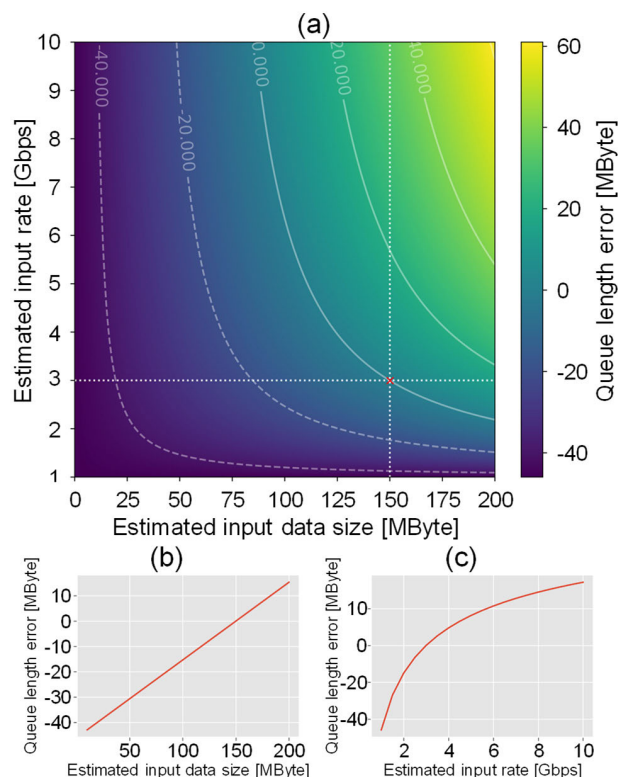

FIGURE 4. (a) Queue length error for various estimation quantities. (b) Calculated queue length error with incorrect estimation of data size (vertical line in (a)). (c) Calculated queue length error with incorrect estimation of input rate (horizontal line in (a)).

IV. NEUMERICAL AND EXPERIMENTAL EVALUATION
A. NUMERICAL ANALYSIS
Figure 3 shows the maximum d_{in} calculated by cooperated traffic shaping without frame dropping when a microburst occurs to a three-SW series (r_{in}: 10 Gbps). Each point was
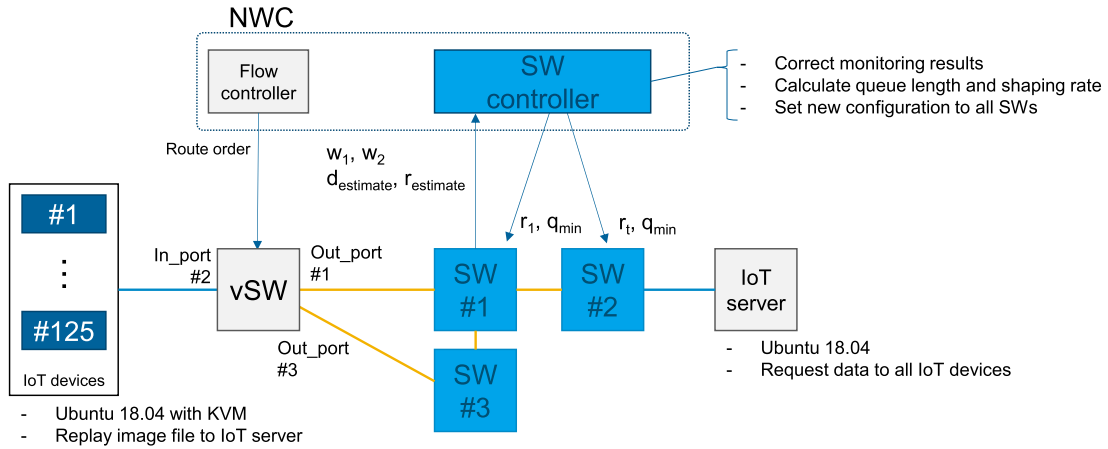
NWC

Flow
controller

SW
controller

- Correct monitoring results
- Calculate queue length and shaping rate
- Set new configuration to all SWs

Route order

w_1, w_2
$d_{estimate}, r_{estimate}$

r_1, q_{min} r_t, q_{min}

#1
⋮
#125

IoT devices

- Ubuntu 18.04 with KVM
- Replay image file to IoT server

In_port
#2

vSW

Out_port
#1

Out_port
#3

SW
#1

SW
#3

SW
#2

IoT
server

- Ubuntu 18.04
- Request data to all IoT devices

FIGURE 5. Experimental setup and picture of switches for shaping.

calculated using (1) by increasing the input data size d_{in} so as not to exceed the maximum queue length (i.e., buffer memory size). We assumed that the buffer memory size of each SW is up to 64 MBytes and that the IoT-GW and all SWs are connected via 10-Gigabit Ethernet (10 GbE). In the cooperated traffic shaping, the NWC orders SW #1, #2, and #3 to store the overload data. In conventional traffic shaping, one SW stores the overload data. Thus, the proposed aggregated NW with the cooperated traffic shaping can forward larger microburst traffic (113 MBytes) than the conventional method (64 MBytes).

The required queue length depends on the measurement accuracy of the observable d_{in} and r_{in}, as shown in (3). Figure 4 (a) shows various estimated queue length errors under the condition that r_{in} is 3 Gbps, d_{in} is 150 MBytes, and that there are three SWs. In this case, q_{min} is calculated to be about 46 MBytes. Figure 4 (b) shows the calculated queue length when d_{in} is misestimated and r_{in} is correctly estimated as 3 Gbps. The result coincides with the dotted horizontal line in Fig. 4 (a). Figure 4 (c) shows the calculated required queue length when r_{in} is misestimated and d_{in} is correctly estimated as 150 Mbytes. The result coincides with the dotted vertical line in Fig. 4 (a). In the case of underestimation, the queue length allocation is reduced, and the data is discarded. In the case of overestimation, the allocation of the queue length becomes large, and the queue becomes excessive. No data is discarded, but the number of queues that can be allocated decreases when considering the entire buffer. Note that the estimated input rate increases and decreases logarithmically, while the estimated amount of data increases and decreases linearly.

B. EXPERIMENTAL SETUP AND RESULTS

We experimentally demonstrated the coordinated traffic shaping technique by implementing it in commercial SWs [17]. The experimental NW consisted of IoT devices, an IoT server, three SWs, a virtual SW, and a NWC, as shown in Figure 5. The NWC consists of a flow controller and a SW controller.
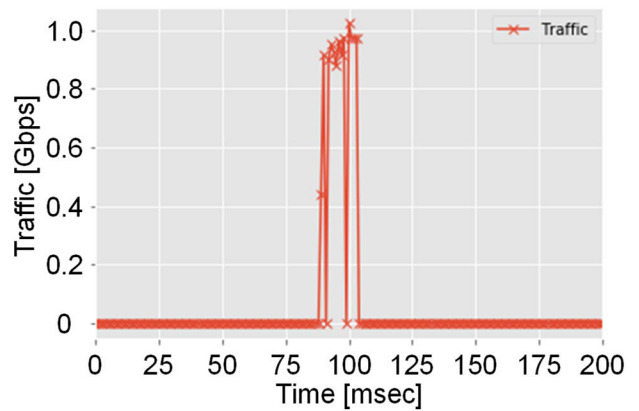

FIGURE 6. Amount of traffic in microburst.

We emulated massive IoT devices to generate a microburst. This was performed in StarBED, a large-scale emulation platform [18]. The 125 IoT devices were implemented on Ubuntu with Kernel-based Virtual Machine (KVM). Each device periodically sent 5 kB of data (a JPEG file) with 1,500-byte user datagram protocol (UDP) packets and 12-byte interframe gaps, assuming periodic data transmission such as sensor data. The IoT server sent requests at a specific interval to control the transmission timing of the IoT devices. The interval of the microburst traffic was set to 10 seconds. Figure 6 shows the amount of traffic in a microburst with a 1-millisecond resolution. The SWs periodically monitored w_1 within 10 seconds and reported to the SW controller of the NWC. The SW controller was implemented as a software controller on an Ubuntu server. The SW controller has the path information for each flow and SW specifications such as minimum queue length. The SW controller calculated the queue length and shaping rates as shown in the previous chapter and configured them to all of the SWs. r_{target} and the initial shaping rate of each SW were both set to 100 Mbps. The queue length q_{init} of each SW was initially set to 10 kB, and the maximum input data size that could be transferred without frame loss
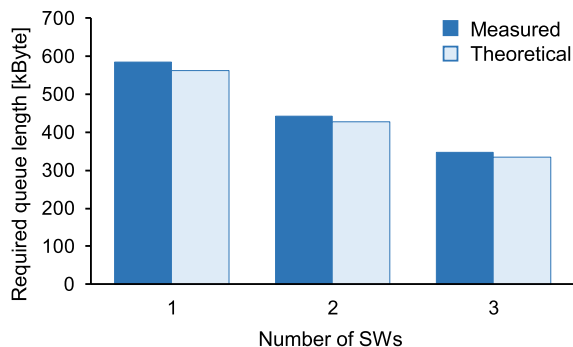
FIGURE 7. Experimental results of cooperated traffic shaping.

FIGURE 8. Traffic offload example.

was measured by varying the maximum queue length of the SW. For comparison, a SW with a fixed queue length was prepared as a conventional method by keeping the shaping rate at 100 Mbps.

The experimental NW was constructed by a virtual SW using Lagopus [19], and Ryu [20] was used as a flow controller. The SW controller calculated the queue length required for eliminating discarded frames; if it is greater than the rest of the SWs' queue in the current path, the other path is allocated by the flow controller.

Figure 7 shows the experimental results of cooperated traffic shaping. When the number of SWs was set to one, the shaping rates of SW #1 and #2 were set to r_{target} for the conventional traffic shaping method, i.e., the baseline. As the number of SWs increases, cooperative shaping improves the performance of required queue length because the required queue length decreases with the number of SWs. The required queue length was 348 kBytes in the proposed method using three SWs while the required queue length was 585 kBytes in the conventional method. That is, our proposed cooperated traffic shaping method reduced the queue length required in the conventional traffic shaping (using one SW) by 40%. As a result, the data size of the microburst which can be accommodated is 1.68 times larger than that of the conventional method with the same queue length. Figure 7 also shows the numerical analysis results when r_{in} and d_{in} are known. The results were computed using (2) and (3). The difference between the experimental and numerical results was below 4.2% for up to 3 SWs. Because the difference is negligible, we can conclude that the estimation scheme in (8) and (9) worked accurately.

Next, we demonstrated the path reallocation by adjusting the rest of the queue length of SW #2. The vSW outputs the data from IoT devices to port #1 first. The queue length calculated by the NW controller was larger than the rest of the queue length at SW #2 so the path was reallocated by the flow controller to an additional path through SW #3. Figure 8 shows an example of offloaded traffic, in which 10% of the traffic is offloaded to output port #3. The NWC uses the traffic offload function of Lagopus and Ryum and controls the traffic for each path by setting the weight. When the weight is set to 100% for output port #3, the path can be reallocated to
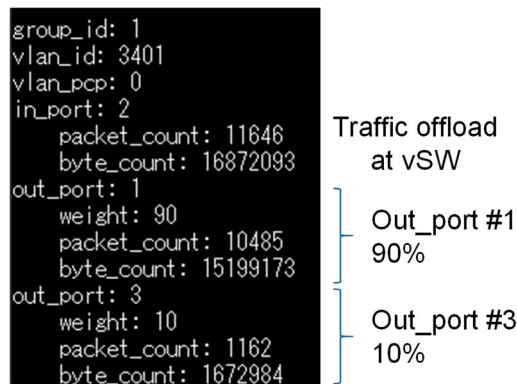
the longer one. The number of SWs in the path increased from two to three, so the required queue length was reduced and the queue length was allocated to eliminate discarded frames.

However, the path reallocation algorithm presents some limitations. For example, IoT traffic fluctuates if the number of IoT devices changes; we assumed the number of IoT devices was constant in this paper. The optimal path should be changed in accordance with such time dependent factors. An adaptive reallocation algorithm will be required to further improve the proposed method.

V. CONCLUSION

Internet-of-Things (IoT) devices will be deployed for various applications such as camera monitoring and sensing in 5G mobile networks and beyond. A network consisting of Layer-2 or Layer-3 aggregation switches (SWs) can accommodate the traffic for IoT backhaul; however, the problem of microbursts arises when accommodating IoT traffic in an aggregated NW. Thus, we proposed cooperated traffic shaping using SWs to accommodate more traffic with a limited queue length. We formulated equations to derive the minimum queue length with shaping rates which gradually decrease in geometric progression. To acquire the queue length using general SWs without short-cycle monitoring, we proposed a scheme for estimating the instantaneous input rate and data size of microburst traffic required for the equations. If the calculated queue length cannot be prepared in the current path, then the path is reallocated to one with more SWs. We experimentally demonstrated the proposed coordinated traffic shaping technique by implementing it in commercial SWs with 125 emulated IoT devices. We found that the difference between the experimental and numerical results was below 4.2%, and the queue length could be reduced by 40% when there were three SWs. In addition, the path was successfully reallocated from a path with two SWs to one with three SWs. In the future, an adaptive reallocation algorithm will be required to address the dynamic microburst traffic in an actual network.

REFERENCES

[1] *White Paper 5G Evolution and 6G*, NTT Docomo, Tokyo, Japan, Jan. 2020. [Online]. Available: https://www.nttdocomo.co.jp/english/corporate/technology/whitepaper_6g/

[2] *LoRaWAN Specification V1.1*, LoRa Alliance, Fremont, CA, USA, Oct. 2017.

[3] N. Shibata, P. Zhu, K. Nishimura, Y. Yoshida, K. Hayashi, M. Hirota, R. Harada, K. Honda, S. Kaneko, J. Terada, and K.-I. Kitayama, "First demonstration of autonomous TSN-based beyond-best-effort networking for 5G NR fronthauls and 1,000+ massive IoT traffic," in *Proc. Eur. Conf. Opt. Commun. (ECOC)*, Dec. 2020, pp. 1–4.

[4] T. Tojo, H. Baba, S. Yasukawa, and Y. Okazaki, "Micro-burst analysis and mitigation for 5G low-latency communication services," in *Proc. IEEE 2nd 5G World Forum (GWF)*, Sep. 2019, pp. 232–236.

[5] K. Honda, N. Shibata, R. Harada, S. Kaneko, and J. Terada, "Cooperated traffic shaping technique for efficient accommodation of microbursts in IoT backhaul network," *IEICE Commun. Exp.*, vol. 10, no. 6, pp. 307–312, 2021.

[6] D. Shan and F. Ren, "ECN marking with micro-burst traffic: Problem, analysis, and improvement," *IEEE/ACM Trans. Netw.*, vol. 26, no. 4, pp. 1533–1546, Aug. 2018.

[7] D. Shan, W. Jiang, and F. Ren, "Absorbing micro-burst traffic by enhancing dynamic threshold policy of data center switches," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2015, pp. 118–126.

[8] D. Shan, F. Ren, P. Cheng, R. Shu, and C. Guo, "Observing and mitigating micro-burst traffic in data center networks," *IEEE/ACM Trans. Netw.*, vol. 28, no. 1, pp. 98–111, Feb. 2020.

[9] D. Shan, W. Jiang, and F. Ren, "Analyzing and enhancing dynamic threshold policy of data center switches," *IEEE Trans. Parallel Distrib. Syst.*, vol. 28, no. 9, pp. 2454–2470, Sep. 2017.

[10] S. Zou, J. Huang, J. Wang, and T. He, "Flow-aware adaptive pacing to mitigate TCP incast in data center networks," *IEEE/ACM Trans. Netw.*, vol. 29, no. 1, pp. 134–147, Feb. 2021.

[11] A. M. Abdelmoniem and B. Bensaou, "T-RACKs: A faster recovery mechanism for TCP in data center networks," *IEEE/ACM Trans. Netw.*, vol. 29, no. 3, pp. 1074–1087, Jun. 2021.

[12] X. Chen, Z. Li, Y. Chen, and X. Wang, "Performance analysis and uplink scheduling for QoS-aware NB-IoT networks in mobile computing," *IEEE Access*, vol. 7, pp. 44404–44415, 2019.

[13] V. Jeyakumar, M. Alizadeh, C. Kim, and D. Mazières, "Tiny packet programs for low-latency network control and monitoring," in *Proc. 12th ACM Workshop Hot Topics Netw.*, College Park, MD, USA, Nov. 2013, pp. 1–7. Accessed: Jun. 17, 2021.

[14] G. Jereczek, G. L. Miotto, and D. Malone, "Analogues between tuning TCP for data acquisition and datacenter networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, U.K., Jun. 2015, pp. 6062–6067, doi: 10.1109/icc.2015.7249288.

[15] J. Guo, M. Liang, and Z. Liu, "A load balancing algorithm based on end-to-end rerouting," in *Proc. IEEE Int. Conf. Electron. Technol., Commun. Inf. (ICETCI)*, Aug. 2021, pp. 1–6.

[16] N. Maksic, "Topology independent multipath routing for data center networks," *IEEE Access*, vol. 9, pp. 128590–128600, 2021.

[17] APRESIA Systems. Accessed: Jul. 14, 2021. *Apresia 26004*. [Online]. Available: https://www.apresia.jp/en/products/sp/ap26000/#ap26004

[18] *StarBED4 Project*. Accessed: Jul. 2, 2021. [Online]. Available: https://starbed.nict.go.jp/en/index.html

[19] *Lagopus Switch and Router*. Accessed: Jul. 14, 2021. [Online]. Available: https://www.lagopus.org/

[20] *Ryu SDN Framework*. Accessed: Jul. 14, 2021. [Online]. Available: https://ryu-sdn.org/

KAZUAKI HONDA received the B.S. degree from Kyoto University, Kyoto, Japan, in 2013. In 2013, he joined the Access Network Service Systems Laboratories, NTT Corporation, Kanagawa, Japan, where he has been involved in research on IoT aggregating network and WDM-PON systems. He is a member of IEICE.

NAOTAKA SHIBATA received the B.E. degree in electrical and electronic engineering and the M.E. degree in communications and computer engineering from Kyoto University, Kyoto, Japan, in 2007 and 2009, respectively. In 2009, he joined Nippon Telegraph and Telephone (NTT) Corporation Access Network Service Systems Laboratories, where he was engaged in research on wireless communication systems. Since 2012, he has been engaged in research on optical-wireless converged networks. He received the Young Engineer Award from the Institute of Electronics, Information, and Communication Engineers (IEICE), Japan, in 2015. He is a member of IEICE.

RINTARO HARADA received the B.E. degree in computer science and engineering and the M.E. degree in computer science and communications engineering from Waseda University, Tokyo, Japan, in 2015 and 2017, respectively. In 2017, he joined the NTT Access Network Service Systems Laboratories, where he has been engaged in research on optical access networks. He received the Young Researcher's Award from IEICE, Japan, in 2020. He is a member of IEICE.

YOTA ISHIDA received the M.E. degree in mechanical engineering from Kanazawa University, Ishikawa, Japan, in 2014. In 2014, he joined LIFULL Company Ltd., where he worked on real estate information services. In 2020, he joined the National Institute of Information and Communications Technology (NICT), Japan, where he worked on network emulation technology.

KUNIO AKASHI received the M.S. and Ph.D. degrees in information science from the Japan Advanced Institute of Science and Technology (JAIST), Ishikawa, Japan, in 2010 and 2017, respectively. From 2017 to 2021, he joined the National Institute of Information and Communications Technology (NICT), Japan, where he worked on network emulation technology. In 2021, he joined The University of Tokyo, Japan, as an Assistant Professor. His current research interests include computer networks and network security.

SHIN KANEKO received the B.E. and M.E. degrees in electronics engineering from The University of Tokyo, Tokyo, Japan, in 2002 and 2004, respectively. In 2004, he joined NTT Access Network Service Systems Laboratories, Chiba, Japan. His current research interests include next-generation optical access networks and systems. He is a member of IEICE.

TOSHIYUKI MIYACHI received the M.S. and Ph.D. degrees in information science from the Japan Advanced Institute of Science and Technology, Ishikawa, Japan, in 2002 and 2007, respectively. In 2007, he joined the National Institute of Information and Communications Technology (NICT), Japan, where he has been engaged in the research on evaluation of network technologies especially on actual node-based network testbed. He is currently the Director of the Hokuriku StarBED Technology Center, NICT.

JUN TERADA received the B.E. degree in science and engineering and the M.E. degree in computer science from Keio University, Yokohama, Japan, in 1993 and 1995, respectively. In 1995, he joined NTT LSI Laboratories, where he was involved in the research and development of low-voltage analog circuits, especially A/D and D/A converters. In 1999, he was involved in developing small and low-power wireless systems for sensor networks. In 2006, he was involved in researching high-speed front-end circuits for optical transceivers. He is currently a Senior Research Engineer and a Supervisor with NTT Access Network Service Systems Laboratories, Yokosuka, Japan, where he is responsible for research and development management for optical and wireless converged access networks. He is a member of IEICE. He has served as a member of the Technical Program Committee of the Symposium on VLSI Circuits. He has been participating in the Asian Solid-State Circuits Conference (A-SSCC), since 2012.

• • •