Cooperated traffic shaping technique for efficient accommodation of microbursts in IoT backhaul network

Kazuaki Honda\(^{1,\text{a)}}\), Naotaka Shibata\(^1\), Rintaro Harada\(^1\), Shin Kaneko\(^1\), and Jun Terada\(^1\)

\(^1\)NTT Access Network Service Systems Laboratories, NTT Corporation, 1–1 Hikarinooka Yokosuka, Kanagawa, 239–0847, Japan

\(^{\text{a)}}\)kazuaki.honda.ku@hco.ntt.co.jp

Abstract: An aggregating switch (SW) network offers cost-effective accommodation to IoT (Internet of Things) traffic by aggregating the traffic. For more efficient accommodation, the microburst traffic must be mitigated, which occurs when massive IoT devices transmit data simultaneously. In this paper, we propose cooperated traffic shaping by SWs with estimation of the input rate from the IoT devices to increase the accommodated data size of microburst traffic and derive equations for optimal configurations. Experimental results show that the data size can be increased by 60%.

Keywords: IoT, layer-2 network, traffic shaping

Classification: Network System

References

[1] G.A. Akpakwu, B.J. Silva, G.P. Hancke, and A.M. Abu-Mahfouz, “A survey on 5G networks for the internet of things: communication technologies and challenges,” IEEE Access, vol. 6, pp. 3619–3647, 2018. DOI: 10.1109/ACCESS.2017.2779844

[2] H.S. Altwassi, Z. Pervez, K. Dahal, and B. Ghaleb, “The RPL load balancing in IoT network with burst traffic scenarios,” 2018 12th International Conference on Software, Knowledge, Information Management & Applications (SKIMA), Phnom Penh, Cambodia, pp. 1–7, 2018. DOI: 10.1109/SKIMA.2018.8631520

[3] Z. Zhou, Y. Yan, M. Berger, and S. Ruepp, “Analysis and modeling of asynchronous traffic shaping in time sensitive networks,” Proc. of 2018 14th IEEE International Workshop on Factory Communication Systems (WFCS), Imperia, pp. 1–4, 2018. DOI: 10.1109/WFCS.2018.8402376

[4] S. Thangamuthu, N. Concer, P.J.L. Cuijpers, and J.J. Lukkien, “Analysis of Ethernet-switch traffic shapers for in-vehicle networking applications,” 2015 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, pp. 55–60, 2015. DOI: 10.7873/DATE.2015.0045
1 Introduction

5G mobile services have started handling rapidly expanding traffic. Massive machine type communication (mMTC), one use case of 5G, leads to widespread use of Internet of Things (IoT) devices [1]. IoT devices are connected to IoT-gateways (IoT-GWs) by wireless connections. IoT-GWs aggregate the IoT traffic in their coverage areas and send the aggregated IoT data to an IoT server via an optical network called the “IoT backhaul.” Each IoT device generates a small amount of traffic, but massive devices are dispersed in a wide area. Thus, an aggregating network (NW) composed of aggregating switches (SWs) is suitable for an IoT backhaul given aggregating NW’s cost-effective accommodation of such traffic by aggregation with statistical multiplexing gain. The SWs aggregate the IoT traffic in layer-2 or layer-3.

In uplink transmissions, many IoT devices send the data to the IoT server periodically since typical IoT devices send sensor data for its application. When massive IoT devices send data simultaneously, a large, instantaneous amount of traffic, called “microburst traffic,” occurs [2]. Moreover, it should be considered to accommodate multiple IoT service providers in one telecom carrier network. If there are independent flow controllers for individual IoT flows, the microburst traffic may occur because each IoT flow does not consider other IoT flows. If the data rate of microburst traffic is higher than the processing speed of the IoT server, overloaded frames are discarded in the IoT server. Then, the IoT server requests to resend the lost frames to the IoT devices, ultimately shortening the battery life of the IoT devices. Moreover, microburst traffic can occur repeatedly since IoT devices send data periodically.

To address these problems, traffic shaping is generally used in layer-2 networks [3, 4]. In a conventional method, the shaping rate of each SW is set equal to the processing rate of the IoT server. When the input data rate is higher than the shaping rate, a part of the received frames are stored in a queue in a single SW momentarily. Therefore, a significantly large queue length is required to accommodate the high-rate microbursts without frame loss. Given, it is impossible to know which SW the microburst traffic flows to, each SW should be equipped with a large buffer memory, which ultimately increases the SW cost.

In this paper, we propose a novel cooperated traffic shaping technique for aggregating NWs that enhances the accommodated data size of microburst traffic with a given buffer size. First, we theoretically optimize the shaping rates to minimize queue lengths for SWs. Since the traffic shaping of one SW affects the rate and length of the burst traffic input to the next SW, we calculate the optimal configuration using a recurrence formula. Next, we verify our optimization methods including the input data-rate estimation experimentally by implementing them in commercial SWs.

2 Cooperated shaping in aggregating switch network

Figure 1 shows the aggregating NW architecture with our cooperated shaping technique. Our proposed aggregating NW is composed of SWs and a network controller (NWC). The NWC is connected to the SWs and controls their settings (e.g., queue length, and the shaping rate, etc.). Each SW aggregates uplink traffic from the IoT-GWs and transfers them to an IoT server with statistical multiplexing. The traffic shaper and queue ranges of each SW are based on buffer memory size. Microburst
traffic often occurs periodically since almost all IoT applications send data (e.g., sensor data) in a certain cycle.

The co-operated shaping mitigates the microburst traffic as follows. During the initial setup, the shaping rates of all SWs are set to the processing rate of the IoT server (target rate \( r_{\text{target}} \)), and the queue lengths of all SWs are set to the initial queue size, \( q_{\text{init}} \). Here, each SW monitors the input data size, \( d_{\text{in}} \), and the discarded data size, \( d_{\text{dis}} \), in the SW during a certain cycle. When an SW detects frames being discarded, it reports the size of \( d_{\text{in}} \) and \( d_{\text{dis}} \) to the NWC. The NWC calculates the shaping rate, \( r_k \), and queue length, \( q_k \), for SW \( k \) (1 \( \leq k \leq n \)) to avoid discarding the frames on condition that \( r_k \) equals to \( r_{\text{target}} \). Here, SW \#1 is the switch where the frame discard occurs and SW \#n is the last SW which is on the transfer route. After the calculation, the NWC sets the shaping rates and queue lengths to the SWs.

We calculate the queue length among all SWs to be minimized so that the required buffer memories in SWs are equalized for an easy operation. The queue length of the SW \#k (\( q_k \)) must be set to

\[
q_k [\text{Byte}] = (r_{k-1} [\text{bit/sec}] - r_k [\text{bit/sec}]) \frac{d_{\text{in}} [\text{Byte}]}{r_k [\text{bit/sec}]}.
\]

In the case of \( k = 1 \), \( r_{k-1} \) denotes the input rate, \( r_{\text{in}} \), of the microburst traffic to the first SW. The right side of Eq. (1) is the multiplication of the accumulation speed at the queue, \( r_{k-1} - r_k \), and the burst length input to the SW \#k, \( d_{\text{in}}/r_{k-1} \). \( r_k \) gets smaller as the \#k gets larger. From Eq. (1), required \( q_k \) decreases monotonically as \( r_{k-1} \) increases and increases monotonically as \( r_k \) increases. By considering that \( q_{k+1} \) decreases monotonically as \( r_k \) increases, the longest queue length is minimized at the constant queue length condition (i.e., \( q_k = q_{k+1} \)). The shaping rate is calculated from this condition as \( r_k = \sqrt{r_{k-1}/r_{k+1}} \), which gives a geometric progression. Thus, the shaping rate and the queue length are optimized as

\[
r_k = \left( \frac{r_{\text{target}}}{r_{\text{in}}} \right)^{1/n} r_{\text{in}}, \quad q_{\text{min}} = \left\{ 1 - \left( \frac{r_{\text{target}}}{r_{\text{in}}} \right)^{1/n} \right\} d_{\text{in}},
\]

where \( q_{\text{min}} \) denotes the minimized longest queue length.

To know the value of \( r_{\text{in}} \), the traffic of all the ports at each SW should be continuously monitored since it is impossible to know where microburst traffic will occur in advance. Moreover, the monitoring cycle should be significantly short in order to calculate the instantaneous value accurately. However, short-cycle monitoring increases SW cost, so, we propose a way of estimating \( r_{\text{n}} \) without short-cycle monitoring, that can even be performed by general SWs. Once a certain SW

**Fig. 1.** IoT backhaul network with cooperated traffic shaping.
observes data being discarded, the discarded frames and queue length are described by
\[ d_{\text{dis}} + q_{\text{init}} = (r_{\text{in}} - r_{\text{target}}) \frac{d_{\text{in}}}{r_{\text{in}}}. \]  
(3)

Eq. (3) is derived from the initial setup (i.e., shaping rate: \( r_{\text{target}} \), queue length: \( q_{\text{init}} \)) and Eq. (1) by applying \( q_{\text{init}} + d_{\text{dis}} \) as the required queue length. Then, the NWC estimate \( r_{\text{in}} \) as
\[ r_{\text{in}} = \frac{d_{\text{in}}}{d_{\text{in}} - d_{\text{dis}} - q_{\text{init}} r_{\text{target}}}. \]  
(4)

To estimate \( r_{\text{in}} \), the SWs can sufficiently monitor the values with the same frequency as IoT devices’ data transmission.

Figure 2 illustrates our traffic shaping technique when the microburst (\( r_{\text{in}} \): 5 Gbps, \( d_{\text{in}} \): 7.2 MB) is input to the SW #1. It is assumed that the buffer memory size in each SW is limited to 3 MB and the IoT-GW and the SWs are connected with 10 Gigabit Ethernet (10GbE). In our cooperated shaping, the NWC orders the SW#1, #2 and #3 to store the overloaded data. From Eq. (3), the required queue length per SW becomes 3 MB, which is the limit of the buffer memory size. On the other hand, discarding 2.8 Mbytes at SW #1 is inevitable using a conventional method, which forces a single SW to store the overloaded data. Thus, the aggregating NW with the proposed cooperated shaping can transfer the larger microburst traffic (7.2 MB) than the conventional method (4.4 MB).

![Fig. 2. (a) Conventional shaping. (b) Cooperated shaping in aggregating SW network.](image-url)
3 Experimental setup

We demonstrated our traffic shaping technique experimentally by implementing it in commercial SWs. The experimental network was composed of 3 SWs connected in a series as described in Fig. 2. We used the topology because the SWs set in each flow are independent from other flows and logically connected in a series even if the network has a ring or a mesh topology. A traffic generator and an analyzer were connected to SWs #1 and #3, respectively. The microburst traffic was generated by the traffic generator, which was composed of 1,500-Byte frames and 12-Byte inter frame gaps. The cycle of microburst traffic was set to 10 seconds by assuming periodic data transmission such as sensor data. The SW #1 periodically monitored the \( d_{in} \) and the \( d_{dis} \) within 10 seconds and reported them to the NWC for the estimation of \( r_{in} \). The value of \( r_{target} \) was set to 100 Mbps, and the initial shaping rate at each SW was also set to 100 Mbps. The queue length for each SW, \( q_{init} \), was set to 10 kB at the initial state. We measured the maximum input data size transferable without any frame loss by changing the maximum queue length in the SWs. For comparison, as a conventional method, we prepared an SW with the queue fixed at while keeping the shaping rate of 100 Mbps.

4 Experimental results and discussion

Figure 3 compares the accommodatable input data size in microburst traffic against the maximum queue length. Our cooperated shaping received 75 MB when the buffer memory was set to 40 MB for each SW; the conventional method received 44 MB. Assuming that an IoT device sends 128 kB of data (e.g., a 640 × 480 jpeg file) from a network camera, our technique accommodates about 585 IoT devices with no discarded frames in 40 MB of buffer memory while the conventional method accommodates only about 343 devices. Our cooperated shaping method increased the accommodatable data size to 1.6 times the amount using a conventional method at any size of buffer memory. We experimentally confirmed our shaping technique can increase the input data size. Figure 3 also shows the numerical analysis of using \( r_{in} \) without estimations. The results of the conventional method and the cooperated shaping were calculated by Eq. (1) and Eq. (2) respectively. The

![Fig. 3. Effect of cooperated shaping in aggregating SW network.](image-url)
negligible differences between the experimental and numerical results indicate that we performed our estimation scheme in Eq. (4) correctly. When the number of SWs increases, the cooperated shaping improves performance because the required queue length shortens based on the number of SWs.

5 Conclusion

We proposed cooperated shaping with estimation of the input rate of the microburst traffic in an aggregating NW monitored by a NWC. The feasibility of the cooperated shaping with estimation was also experimentally demonstrated with comatial SWs. Our method increased the accommodatable input data size to 1.6 times the conventional amount for any buffer memory size by setting the shaping rate in the SWs’ geometric progression based on the calculated optimal configuration.

Acknowledgments

A part of this work was conducted under the R&D contract “Wired-and-Wireless Converged Radio Access Network for Massive IoT Traffic” with the Ministry of Internal Affairs and Communications, Japan, for radio resource enhancement (JPI000254).