A Weighted Fair Queue Management on DOCSIS Multi-downstream Channels

Yao-Chiang Yang, Wei-Tsong Lee, Chih-Hsing Chen

Department of Electrical and Computer Engineering, Tamkang University, Taiwan
ychiangy@gmail.com.tw, wtlee@mail.tku.edu.tw, 608450184@gms.tku.edu.tw

Abstract

In the home media network, Cable Modem is a useful device that most people have well-known. People expected network access as much less delayed as possible as what they can obtain. Therefore, the objective of effective bandwidth control becomes more and more important recently. In this paper, we are going to propose a methodology of Multi-downstream Service Algorithm (MDSA) to monitor the variation of input arrival rate to manage the bandwidth by Weighted Fair Queue (WFQ) based on a service workload. Applying this way, it is useful to refer to this paper of the bandwidth management method, assessing the input arrival rate that matches how the relative bandwidth is. Furthermore, the delayed time is effectively reduced accordingly.

Keywords: DOCSIS, WFQ, Multi-downstream

1 Introduction

At present, the services of multi-media deeply enter people's daily lives, it becomes more and more important with us. The scope for multi-media cab be extended from small area, such as Voice over IP (VOIP) [1-2] to Video on Demand, Online services, even being extended to the larger field in mobile communication. Everywhere, we can see the multi-media services exist. Down to the home network, it has a popular standard that most people have been using as the services from Data Over Cable Service Interface Specification (DOCSIS) [3-4].

DOCSIS standard comes from Hybrid Fiber Coaxial (HFC) [5] network, it was developed by Cable Television Laboratories (CableLabs). As shown in Figure 1, The DOCSIS system can allow transparent Internet Protocol (IP) traffic in the system communication as well as a useful property of bidirectional transmission simultaneously. Connected to Wide Area Network (WAN), CableLabs defines a system as Cable Modem Termination System (CMTS) which usually located at a headend in a cable company. From another side of system, a Cable Modem (CM) is defined to connect the Customer Premises Equipment (CPE) as normally being located at home for subscribers that setting up the source device for the home network. In the Cable Network, CMTS exchanges the digital signals with CM for the data transfer through Downstream and Upstream of DOCSIS, all of them are main components in the DOCSIS system.

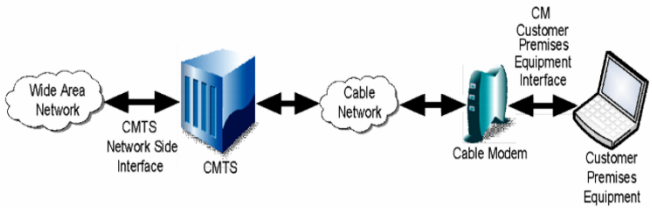


Figure 1. DOCSIS network system [3-4]

Starting from the standard of DOCSIS3.0 [3-4], the first version of channel bonding was released in December 2006, significantly increases data rates of both Downstream and Upstream by channel bonding. It has a bidirectional characteristic, that is because CableLabs defines different frequency bands for Downstream and Upstream of DOCSIS as well as the network traffic is designed as the simultaneous transmission. In addition, DOCSIS 3.0 supports Channel Bonding technology which allows CM through the coaxial cable to split the different channels allocated in the difference frequencies band. The total bandwidths in the channels can be integrated to serve. Logically, CM can utilize the large bandwidth from CMTS in duration. Besides, CMTS is flexibly able to adjust the workload in services of CMs. Through the technology of channel bonding, CMTS provides the services based on the applications in the network. In the packet stream, CMTS is capable of dividing serval segment and arranging in different channels for transmitting. As the explanation in Figure 2 [6-9], packet stream A is served in bother downstream channel #1 and #2, and the same mode of service is going from channel #3 and channel #4 dedicated to the packet stream B.

*Corresponding Author: Wei-Tsong Lee; E-mail: wtlee@mail.tku.edu.tw
DOI: 10.53106/160792642021092205009

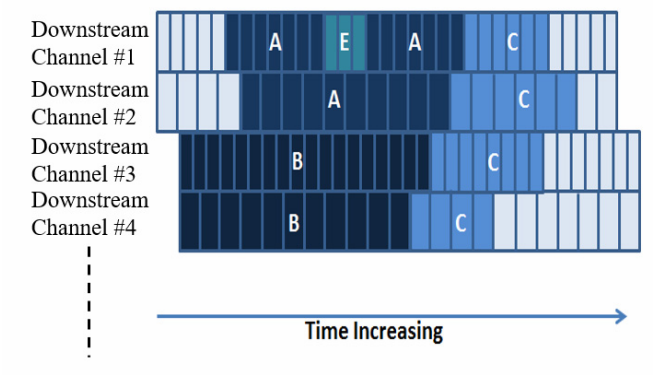


Figure 2. Multiple downstream channels in DOCSIS [6-9]

The research for Multiple channels has been going for a long time, it was a popular subject that a lot of works presented solutions toward that. However, on DOCSIS, there were few studies on the bandwidth management of multi-downstream channels, especially at the point of Quality of Service (QoS). DOCSIS supports total five QoS mechanisms [3-4] which offer users to select them depending on different applications. As the QoS mechanisms are applied on multi-downstream channels, the application is becoming complicated. First of all, when DOCSIS packets enter CMTS as shown in Figure 1, CMTS will classify the characteristic of each packet, schedule it in the queue step by step. Second, even though each packet can be classified by CMTS, the challenge here is how to manage the bandwidth in multi-downstream channels, especially on downstream of DOCSIS, users always expected more bandwidth for their own internet access. Last, in order to maintain the utilization of multiple channels, the consideration of flexibility is an issue leading to adjusting the bandwidth that matches the relative QoS application in the limited output of downstream. How bandwidth to be effectively used becomes our goal in this paper. We are going to propose a methodology to solve this kind of problem.

The rest of this paper is organized as follows. Section 2 provide the overview of the operation on CMTS and CM, QoS and WFQ introduction. In section3, we are going to describe how WFQ is to be implemented in the bandwidth management in Multi-downstream channels. Section 4 provides several simulations to prove the methodology works. Moreover, Section 5 provides the conclusion and further work.

2 Background and Related Work

2.1 Upstream and Downstream of DOCSIS

In DOCSIS specification [3-4], CableLabs defines two layers' specification based on the HFC network, which are "Physical Layer (PHY) Specification" and "Media Access Control (MAC) and Upper Layer Protocols Interface Specification" respectively. It has a bidirectional property of transmission simultaneously, the key is CableLabs defines different frequency band for both Downstream and Upstream as well as the network traffic simultaneously transmits.

2.2 The operation on CMTS and CM

In DOCSIS protocol [10-12], the channels of Upstream and Downstream are divided into many mini-slots of equal size. Two regions of request mini-slots (RMS) and data mini-slots (DMS) are designated by the CMTS. When the CM requests the data transmission through the Upstream channels to the CMTS, the CM first receives a first message of Bandwidth Allocation Map (MAP) from the CMTS, then send a request Protocol Data Unit (PDU) in the RMS region to CMTS. After the CMTS receives the request PDU from CM, CMTS can provide the useful information of scheduling process that including mini-slot allocation results of Upstream channels to the CM by second MAP message. The CM then follow this message until the assigned timing for the DMS to transmit a data PDU to CMTS. The transmission will be done until the CMTS receives the data PDU. The following Figure 3 shows the data transmission process.

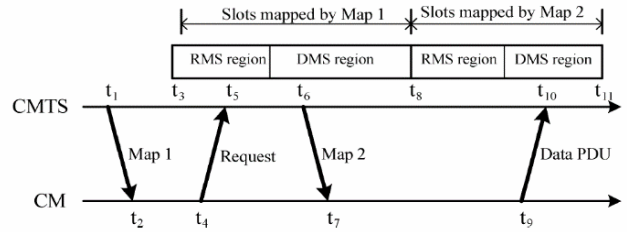


Figure 3. Data transmission process between CMTS and CM [10-12]

- The CMTS send a Bandwidth Allocation Map (MAP) as MAP_1 out that records allocation results of mini-slot for time t_3~t_8 at time t_1.
- MAP_1 then arrives at time t_2 on the CM. The CM can send the request PDU to CMTS in the RMS region at time t_4 if the CM requests to transmit the data.
- The request PDU then is received from the CMTS at time t_5. At time t_6, the CMTS send MAP_2 to the CM that including all the information of schedule process, admission control and mini-slot for the time t_8~t_{11}.
- Data PDU for transmission from the CM is completed at time t_{10}.

2.3 Weight Fair Queue

Weighted Fair Queue (WFQ) [13-15] was extended from Fair Queue (FQ) [16-17]. The purpose of FQ is to balance the resources in the network, as fair as possible to procure the latencies balanced in every single queue. For each queue in the network, it is able to receive an equal bandwidth for fair serving. Overall, the latencies

in queues are balanced. The method suited different types of queues, short and long packet frames were fitted and well-fair of taking care. Moreover, when the overall system latency can be reduced since the short packet frame is served in time. It doesn't waste time to focus on the long packet frame only. From FQ to WFQ, the improvement is an add-on of a weight management for every single queue. Using the weight management, we are able to prioritize the service that designs the weight parameter depending on how important they are between queues. The high-priority queue is able to receive more bandwidth than the queue with low-priority. That means, Higher priority served is going to reduce that latency accordingly. On the other hand, each type of queue is with its own weight management, we can depend on the purpose and application on the queue to arrange the weight, management-controlled further, is Much useful than FQ.

2.4 QoS

In DOCSIS, total five QoS mechanisms [3, 7-8] are supported; They are Unsolicited Grant Service (UGS), Unsolicited Grant Service with Activity Detection (UGS-AD), Real-Time Polling Service (rtPS), Non-Real-Time Polling Service (nrtPS), and Best Effort (BE) Service.

2.4.1 Unsolicited Grant Service, UGS

In Figure 4, it is designed to serve a fixed packet size per fixed period. For instance, like Voice over IP (VOIP), the application needs this kind of instant service. In DOCSIS, both DS and US support UGS, and CMTS is required to automatically offer a fixed bandwidth for serving the relative packet stream per a fixed period. In addition, bandwidth arrangement must match the instant service requirement, which means the response time must be within the limit of time from the application, the user is not able to feel the latency of service, even interruption. In the initiation, CM gest the service after UGS service is bridged with CMTS. After that, CM stops sending the request packet to CMTS, whereas CMTS offers the service per fixed period. This way can not only reduce the percentage of packet collision from intensively transmitting, but also to avoid a waste of time on the request-awaiting.

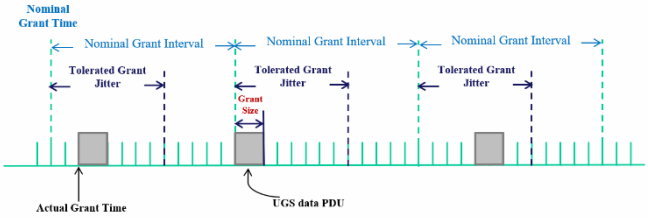


Figure 4. Unsolicited Grant Service, UGS

2.4.2 Real-Time Polling Service, rtPS

In Figure 5, it is designed to offer bandwidth for an unfixed packet frame size per fixed time. For instance, as video broadcasting, DS of DOCSIS supports the service, and schedules for it. It requires CMTS should offer the bandwidth to CM without packet collision. In addition, bandwidth assignment must be within the required time of CM, CMTS provides the service of rtPS until one-time requested application from CM has been finished. Afterward, CMTS inquires the request from CM by polling in every fixed time.

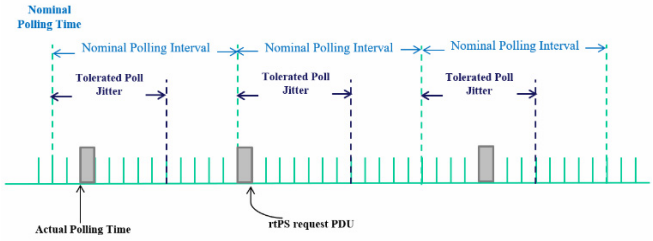


Figure 5. Real-Time Polling Service, rtPS

2.4.3 Unsolicited Grant Service with Activity Detection, UGS/AD

In Figure 6, it is the improvement of UGS. On some services, it is possible that no data is being transmitted for a long time or a period. For instance, While VoIP connection is bridged, CMTS is required to distribute bandwidth for transmitting the data packet like a service of UGS. However, the service will stop if the request from CM remains silence. Like rtPS, CMTS inquires the request from CMs by the way of Polling. This is a combination of UGS/rtPS, the benefit is CM will not occupy the bandwidth all the time, it can be shared with others which are requesting the services.

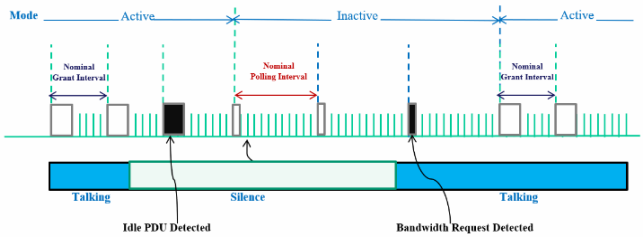


Figure 6. Unsolicited Grant Service with Activity Detection, UGS/AD

2.4.4 Non-Real-Time Polling Service, nrtPS

It is designed to support the unfixed packet-sized frame which is non-instant requirement like FTP. The delay from transmitting is tolerated while it is operating in CMTS. NrtPS features a delay-tolerant packet stream, leaves the small size of bandwidth for the service of avoiding the packet collision.

2.4.5 Best Effort Service, BE

It is designed as the lowest priority of the services, BE follows the general rule in transmission. Besides, the general situation of packet collision is foreseen while transmitting the packet frame.

3 Problem Statement and Proposed Method

3.1 The Problem Statement

In [13], Horng et al proposed a method that utilizing WFQ to effectively control a queueing delay in Time Division Multiple Access (TDMA) Wireless Base Station. It features the weight parameter can arrange the relative bandwidth to the corresponding higher-weight queue. However, in the method, the structure Horng et al proposed was based on the single service channel, besides, it is operated at Wireless Base Station, is not complied with DOCSIS. It brings the reason up why we propose the method in this paper. How to manage the utilization of multiple downstream channels in DOCSIS becomes an issue in which the flexibility to adjust the bandwidth matches the relative QoS application in the limited downstream output. In this paper, we are going to show how WFQ method works in DOCSIS downstream channels, especially for QoS support. We are going to spread the equations based on Queueing Theory [20]. Having a WFQ feature is the same concept with [13], but in our method, single service will not be an only concern, the multiple services will be mainly aimed. In addition, our method features the adaptive parameter of weight in WFQ as dynamic $w_{i,j}$ to the variation, it can be adjusted along with the input arrival rate, the work is not presented in [13].

3.2 Single Downstream Channel in CMTS

In this section, the methodology is going to be introduced. A workload management on the single downstream channel. First of all, it is applied in CMTS, the input arrival rate is coming from WAN, then entering CMTS for management. Queueing length is defined to explain the accumulation of input arrival rate in CMTS that waiting for the service. As shown in Figure 8, one downstream serving is as to present the condition of single downstream channel.

In Figure 7 and Figure 8, queues in the middle Q_1, Q_2, …, Q_m have been defined and described as $q_1(t)$, $q_2(t)$, …, $q_m(t)$, expressing the queueing length that waiting for the service respectively. The definition for each input arrival rate of queues is as $\lambda_1(t)$, $\lambda_2(t)$, …, $\lambda_m(t)$ by top-down view which is distributed from Time Stamp [4] in CMTS. Then, Time Stamp allocates the packet to its belonging route which depends on a corresponding application. The output service of in Downstream is defined as $\mu_1(t)$. Based on the definition, we can build up links between queues and service as $\mu_{1,1}(t)$, $\mu_{2,1}(t)$, …, $\mu_{m,1}(t)$ that are standing for each clear routing-path on $\mu_1(t)$; In addition, a weight of $w_{1,1}$, $w_{2,1}$, …, $w_{m,1}$ can be defined for the flexibility weight of serving management. We have

$$\sum_{i=1}^{m} u_{i,1}(t) \le u_1(t) \tag{1}$$

For each queue, $\mu_{i,1}(t)$

$$u_{i,1}(t) = \frac{w_{i,1}}{\sum_{i=1}^{m} w_{i,1}} u_1(t) \tag{2}$$

Where

- $q_i(t)$ represents queueing length in the system, features $i = 1 \sim m$.
- $\mu_{i,1}(t)$ expresses each output rate from single $q_i(t)$.
- $w_{i,1}$ is the weight value which is to adjust output flow rate, the maximum of $w_{i,1}$ is 1.

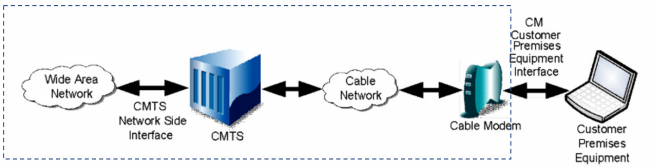


Figure 7. The network from CMTS to CM [3-4]

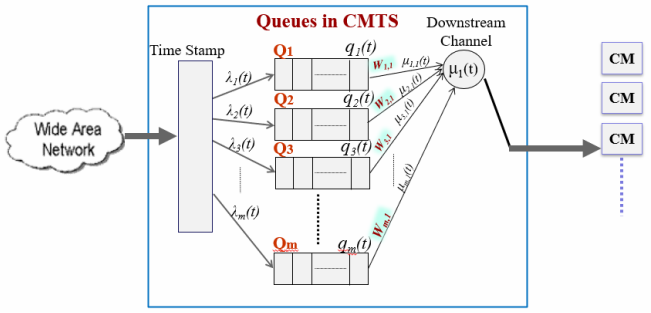


Figure 8. CMTS with single downstream channel

For each single queue $q_i(t)$, if we want to calculate the queueing length from output $\mu_1(t)$ and input $\lambda_i(t)$, we have

$$\frac{dq_i(t)}{dt} = \lambda_i(t) - u_{i,1}(t) \tag{3}$$

$$= \lambda_i(t) - \frac{w_{i,1}}{\sum_{i=1}^{m} w_{i,1}} u_1(t) \tag{4}$$

Since each queue includes a round-robin [18-19] schedule support. According to that, we define a parameter T to express service round time as the duration that the scheduler spends time to deliver packets for all queues in one service round.

In this model, $q_i^{\,k}(t)$ is defined to stand for the queue occupancy at the *k-th* round. We have

$$u_{i,1}^{\,K}(t)T = \frac{w_{i,1}^{k}}{\sum_{i=1}^{m} w_{i,1}^{k}} u_1^{k}(t)T \tag{5}$$

And the dynamic variation can be

$$q_i^{\,K}(t)$$

$$= q_i^{\,K-1}(t) + (\lambda_i^{\,K}(t) - u_{i,1}^{\,K}(t))T \tag{6}$$

$$= q_i^{\,K-1}(t) + \left(\lambda_i^{\,K}(t) - \frac{w_{i,1}^{k}}{\sum_{i=1}^{m} w_{i,1}^{k}} u_1^{\,K}(t) \right) T \tag{7}$$

And the average queueing length of $q_i^{\,k}(t)$ is defined as $E[\,q_i^{\,K}(t)\,]$, which is equal to the sum of total queue length until k-th divided by the number of k. Similarly, $\lambda_i^{\,K}(t)$ is the arrival rate of $q_i^{\,k-1}(t)$ at the end of the k-th service round, and being counted to the k-th, to define the average arrival rate $E[\lambda_i^{\,K}(t)]$ exists. And according to Little Law, the average delay time $d_i^{\,K}(t)$ can be obtained as

$$E[d_i^{\,K}(t)] = \frac{E[q_i^{\,K}(t)]}{E[\lambda_i^{\,K}(t)]} \tag{8}$$

Equation (8) can be used to know queueing delay time in any of queue. On the other hand, we define service workload ρ_i to understand arrival rate vs. service rate that the workload now is running with a light or heavy load. For each service $q_i\,(t)$, we can obtain

$$\rho_i^{k} = \left(\frac{\lambda_i^{\,K}(t)}{u_{i,1}^{\,K}(t)} \right) T \tag{9}$$

$$= \left(\frac{\lambda_i^{\,K}(t)}{\frac{w_{i,1}^{k}}{\sum_{i=1}^{m} w_{i,1}^{k}} u_1^{\,K}(t)} \right) T \tag{10}$$

The parameter ρ_i^{k}, $i = 1\sim m$ is a good notice that is easy to evaluate the current service workload whether the service is under a light or heavy load that $w_{i,1}^{k}$ would be adjusted on the relative service, $\frac{w_{i,1}^{k}}{\sum_{i=1}^{m} w_{i,1}^{k}} u_1^{\,K}(t)$.

3.3 Multiple Downstream Channels in CMTS

In Figure 8, the single downstream channel has been introduced. For the next, Figure 9 is going to present the methodology as how it works in multiple downstream channels. Furthermore, we are going to derive all the formulas for multiple downstream channels.

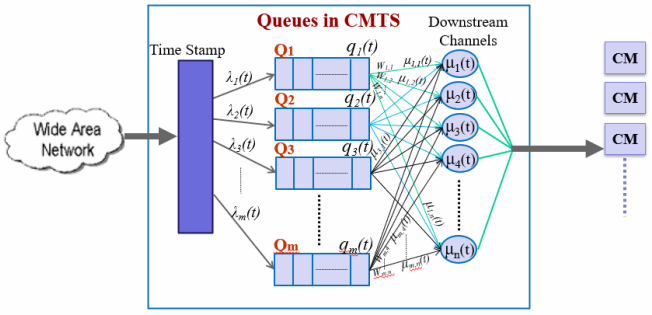


Figure 9. CMTS with multiple downstream channels

In Figure 9, the same definitions of queues in the middle that Q_1, Q_2,…, Q_m have been defined and described as $q_1(t)$, $q_2(t)$, …, $q_m(t)$, expressing the queueing length that waiting for the service respectively. For each input arrival rate of queues is as $\lambda_1(t)$, $\lambda_2(t)$, …, $\lambda_m(t)$ by top-down view which is distributed from Time Stamp [4] in CMTS. To build up the multiple downstream channels, we create the links with $\mu_1(t)$, $\mu_2(t)$, …, $\mu_n(t)$, branches to the connections are as $\mu_{i,1}(t)$, $\mu_{i,2}(t)$, …, $\mu_{i,n}(t)$ for any of queue $q_i(t)$. So we have

$$\sum_{i=1}^{m} u_{i,j}(t) \le u_j(t) \tag{11}$$

$$u_{i,j}(t) = \frac{w_{i,j}}{\sum_{i=1}^{m} w_{i,j}} u_j(t) \tag{12}$$

Where

- $\mu_j(t)$ expresses each downstream channel, j is described as j = 1~n.
- $\mu_{i,j}(t)$ expresses each output rate from any of $q_i(t)$, j is described as j = 1~n.
- $w_{i,j}$ is the weight value which is to adjust output flow rate, the maximum to each $w_{i,1}$, $w_{i,2}$, …, $w_{i,n}$ is 1.

Extended to calculate the queueing length from output $\mu_j(t)$ and input $\lambda_i(t)$, we can obtain

$$\frac{dq_i(t)}{dt} = \lambda_i(t) - \sum_{j=1}^{n} u_{i,j}(t) \tag{13}$$

$$\lambda_i(t) - \sum_{j=1}^{n} \frac{w_{i,j}}{\sum_{i=1}^{m} w_{i,j}} u_j(t) \tag{14}$$

Since each queue has been served in a round-robin schedule. According to that, we define service round time T as the duration that the scheduler spends time to deliver packets for all queues in one service round.

In this model, $q_i^{\,k}(t)$ is defined to stand for the queue

occupancy at the *k-th* round. We have

$$u_{i,j}{}^{K}(t)T = \sum_{j=1}^{n} \frac{w_{i,j}^{k}}{\sum_{i=1}^{m} w_{i,j}^{k}} u_j^k(t)T \quad \textbf{(15)}$$

And the dynamic variation can be

$$q_i{}^{K}(t)$$

$$= q_i{}^{K-1}(t) + \left(\lambda_i{}^{K}(t) - \sum_{j=1}^{n} u_{i,j}{}^{K}(t) \right) T \quad \textbf{(16)}$$

$$= q_i{}^{K-1}(t) + \left(\lambda_i{}^{K}(t) - \sum_{j=1}^{n} \frac{w_{i,j}^{k}}{\sum_{i=1}^{m} w_{i,j}^{k}} u_j^k(t) \right) T \quad \textbf{(17)}$$

Where $\lambda_i{}^{K}(t)$ is the arrival rate of $q_i{}^{k-1}(t)$ at the end of the k-th service round. The average queueing length of $q_i{}^{k}(t)$ is defined as $\mathrm{E}[\, q_i{}^{K}(t)\,]$, which is equal to the sum of total queue length until k-th divided by the number of k. Counting to the k-th, then defining the average arrival rate $\mathrm{E}[\, \lambda_i{}^{K}(t)\,]$ exists. According to Little Law [20], the average delay time $d_i{}^{K}(t)$ can be obtained as

$$\mathrm{E}[d_i{}^{K}(t)] = \frac{\mathrm{E}[q_i{}^{K}(t)]}{\mathrm{E}[\lambda_i{}^{K}(t)]} \quad \textbf{(18)}$$

Equation (18) can be used to know queueing delay time in any of queue. On the other hand, we define service workload ρ_i to understand arrival rate vs. service rate that the workload now is running with light or heavy, so for each service $q_i(t)$,

$$\rho_i^k = \left(\frac{\lambda_i{}^{K}(t)}{\sum_{j=1}^{n} u_{i,j}{}^{K}(t)} \right) T \quad \textbf{(19)}$$

$$= \frac{\lambda_i{}^{K}(t)T}{\sum_{j=1}^{n} \frac{w_{i,j}^{k}}{\sum_{i=1}^{m} w_{i,j}^{k}} u_j^k(t)T} \quad \textbf{(20)}$$

The parameter $\rho_{i,\ i=1\sim m}$ is a good notice that is easy to evaluate the current service workload whether the service is overload that must consider to activate another $\mu_j(t)$ for satisfying the service, reduce the queueing delay time.

3.4 Multi-downstream Service Algorithm in CMTS

In Figure 10, we are going to design an algorithm as being named "Multi-downstream Service Algorithm(MDSA)" in CMTS. In the algorithm, we will explain how we adjust the value of $w_{i,j}$, and when is the timing to create or decrease the number of downstream service channels.

```
Packet classified by Time Stamp
Packet λ_i^K(t) allowed entering Qi

while (K < service round)
    Compute workload ρ_i^k from equation (20)

    # At all Queues
    if (All Queues)
        if (Lower Bound ≤ ρ_i^k ≤ Upper Bound)
            Packet Delivered
        else
            if (ρ_i^k > upper bound)
                w_i,j + α
            else
                if (w_i,j > 1)
                    max = max + 1
                    create u_max(t)

            if (ρ_i^k < lower bound)
                w_i,j − α
            else
                if (w_i,j < Initial State)
                    max = max - 1
                    keep u_max(t)

    # Input Surge At UGS Queue
    else (A Surge happening at UGS Queue)
        if (ρ_UGS^{k+1} > β * ρ_UGS^k)
            if (any w_UGS,j ≠ 1 exists)
                all w_UGS,j are set to 1
            else
                if (w_UGS,j > 1)
                    max = max + 1
                    create u_max(t)

    K = K + 1
```

Figure 10. Multi-downstream Service Algorithm (MDSA)

In Figure 10, first of all, when packet frame $\lambda_i{}^{K}(t)$ enters Time Stamp. It classifies what kind of applications from the packet format, then it assigns the packet to the relative Q_i, i = 1~m. Afterward, the service workload ρ_i^k is going to be computed following the equation (20). The service workload at the time is going to be observed to judge the existing ρ_i^k whether it is within the accepted range or not. If it is true, which means, the existing ρ_i^k is less than the upper bound, but larger than lower bound at the k-th round. We keep the k-th weight value of $w_{i,j}$ for the next round. And if else, which is the ρ_i^k is larger than the upper bound, the k-th weight value of $w_{i,j}$ will be increased by α, then going to the next round for computing by (20) to see whether ρ_i^k is going to be within the accepted range in the next round. Conversely, if ρ_i^k is less than lower bound, the k-th weight value of $w_{i,j}$ will be decreased by α, which is an adjustment parameter. The purpose is to achieve service rate can be adjusted that follows the variation of arrival rate in every k-th round. On the other hand, in queueing theory, the evaluation is going to see the behaviors in variation under static state. So in the end, the weight value will follow the input arrival rate to vary, the setting of how small with decimal point can be set is relating to how accurate and

detail we want to obtain. If it is bigger, which means the response to the variation of input arrival rate is becoming quick but it might not be accurately to arrange the corresponding bandwidth to the service.

The information above, we explained how MDSA works in all queues. However, for UGS type of QoS, it needs a certain support that it is not allowed any packet delayed in the service. It has the real-time requirement that we must secure the traffic without the obvious delay or latency sense by the users. According to that, as long as the QoS queue is defined for UGS support, the weight factor of $w_{i,j}$ must respond as quicker as possible. It is not only including as what the algorithm we did in all queues, but also the algorithm must have a mechanism to react an input surge of arrival. So here we come up with an additional algorithm which is dedicated to the UGS queue. As you can see in the algorithm, after the packet frame $\lambda_i^K(t)$ enters Time Stamp, which packets goes into UGS queue can be identified. In the UGS queue, same process we do is to judge the service workload ρ_i^k is within or larger/less than the existing service rate. The equation (20) is going to be computed first. Then the service workload at the time is observed to judge the existing ρ_i^k whether it is within the accepted range or not. If it is true, the packets are going to be delivered. If not, the service workload ρ_i^k is to be determined by an instant response of rate $\sum_{j=1}^{n} u_{i,j}{}^K(t)$. If the equation exists as $\rho_{UGS}^{k+1} > \beta * \rho_{UGS}^k$, the k-th weight value of $w_{i,j}$ will be increased sharply to be 1 to gain a soonest support for the unclear arrival rate variation. It is not easy to know the k-th arrival rate comes with a slow or sharp input. With the thought, it is obvious as the secure way is to set all the existing weight value of $w_{i,j}$ to the largest 1 in the k-th working services $\sum_{j=1}^{n} u_{i,j}{}^K(t)$, where "n" equals the maximal number of downstream channels we created at the k-th round for service. Afterward, the packets go back to compute by equation (20) again. If the ρ_i^k is within the accepted range, then packets are delivered. However, if not, the maximal number of downstream channels will be plus 1. The number of downstream channels for service is increased at the time. It is a recursive process until the ρ_i^k is within the accepted range, which is Lower Bound $\le \rho_i^k \le$ Upper Bound.

4 Results and Analysis

4.1 Simulation Parameter

As below, the parameter list is set in Table 1. In the simulation, a high-level programming language of Python [21] has been applied. The simulation tool has easy-read and fewer lines properties, as well as it is like C/C++ language that is suitable for general simulation cases. For the paper, we are going to run the different QoS applications and compare each other at average queueing length and average delay time by each scenario below. With the QoS applications, total 5 queues are made of UGS, UGS/AD, rtPS, nrtPS, BE to perform varied service workload ρ_i, and all the simulation cases are performing with an input arrival of Poisson Process. The results are going to be presented based on the single and multiple downstream channels.

Table 1. Parameter list

Parameter	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Queue Type	UGS, UGS/AD, rtPS, nrtPS, BE				
Arrival Rate	Average Rise	UGS/BE Rise	Average Rise	UGS/BE Rise	UGS Surge
Workload, ρ_i	0~0.9				
Initial State, $w_{i,j}$	0.1				
Adjustment of α	0.1, 0.05, 0.01, 0.001	0.05, 0.01	0.05, 0.01	0.05	0.05
Upper/Lower Bound	$0.7 < \rho_i{}^k < 0.9$				
Service Type	Single Downstream Channel		Multiple Downstream Channels		
Service Rate	30Kb/ms per channel				
Adjustment of β	3.0	3.0	3.0	3.0	2.5, 3.0, 3.5, 4.0
Downstream Number	1 channel	1 channel	2 channels	2 channels	4 channels
Round Time, T	1ms				
Service Round, K	10000				

4.2 Simulation Result

4.2.1 Scenario 1

In scenario 1, the result will present input of average variation how it results in each queue under the condition of single downstream channel. At the beginning, we want to explain what that means of average variation, it means corresponding to each queue, each input of arrival rate varies at the same time and trend, one example is showing as Figure 11 that each average arrival rate rises at the same time, feeding the input in Poisson Random Process. The purpose here is to understand when all input arrival rates increase, the results to each queue whether the average queueing length and average delay time can be managed under the control. Furthermore, as which adjusted value of α is unknown to adopt when ρ_i^k achieves upper bound, we are going to simulate what is the better choice of α value selection in each k round of the simulations.

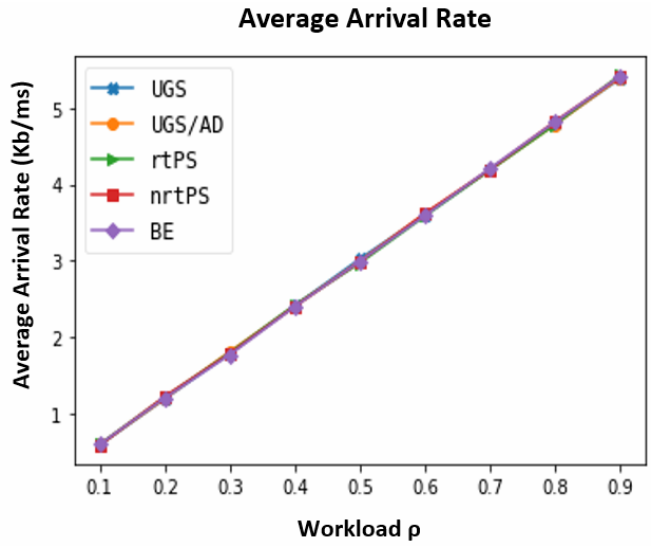


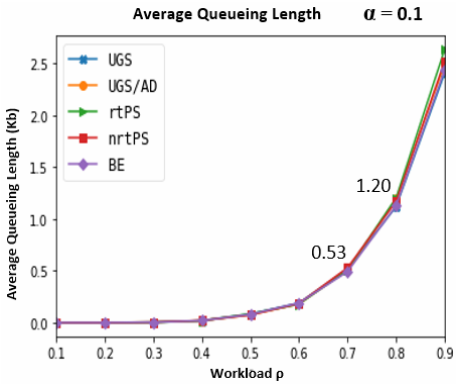
Figure 11. All input rise in single downstream channel

In Figure 12 and Figure 13, the simulations show the different results to present effects of the average queueing length and the average delay time in varied, adjusted value of α, which are running in 0.1, 0.05, 0.01, 0.001. All the results are similar, but a small variation can be seen still. By α =0.1, it means every service round, the $w_{i,j}$ can be adjusted in 0.1 in every k round. It has quick response to react the ρ_i^k variation, it is not only for the arrival rate increases, but it is also suitable when the arrival rate decreases. On the other hand, if choosing α =0.001, that means, the reaction is slower than α =0.1 in varying arrival rate, but it is better to precisely react the bandwidth arrangement without waste. So that is why we present different adjusted values of α in scenario 1. The purpose is trying to find out the right choice of α. In Figure 12, before workload ρ = 0.6, all the average queueing lengths are similar, as well as the average delay time presents the similar results in the simulations. The obvious discrepancy happens from workload ρ = 0.7 (please note that the value we marked is the highest value of all queues, our purpose is to simply compare the discrepancies each other), it responds the longer average queueing length at 0.53kb(even over y-axis scale at 0.5kb) by α =0.1 as it is too quick to react the input, causes the bandwidth arrangement cannot be precisely down to follow the throughput variation. It is having the cross-effect by each queueing length calculation of $\frac{w_{i,1}^k}{\sum_{i=1}^{m} w_{i,1}^k} u_1^k(t)$ since the maximal service rate in single downstream channel is fixed. Therefore α =0.1 is not a good choice to the simulations. Except α = 0.1, we can obtain the similar results by α = 0.05, 0.01 and 0.001, even at ρ = 0.8, but as we mentioned earlier, a larger α will result in the quick reaction to the input variation. As the thought it is, the value of α is good to choose 0.05 for the following simulations as it has the similar precision as α = 0.01 and 0.001, whereas it also has a quicker reaction than α = 0.01 and 0.001. Table 2 is a list in presenting the latest value of $w_{i,j}$, which is the state of K = 10000. It is explaining as when input arrival rate λ is increasing, the values of the corresponding $w_{i,j}$ is following to increase accordingly as well.

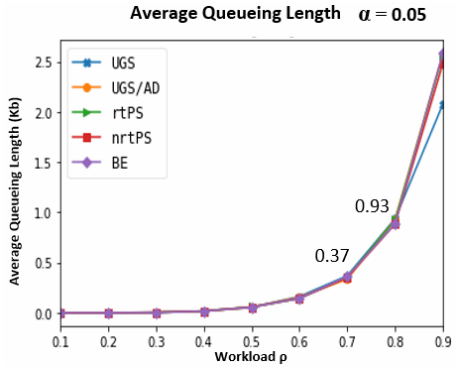
4.2.2 Scenario 2

In scenario 2, we are going to show if the increment happens only in UGS or BE queue, how the corresponding average queueing length and average delay time will be in the single downstream channel. In Figure 14, two conditions show the variations of UGS/BE input arrival rate. In the period of ρ = 0.1~0.5, we set all the input arrival rates of Poisson Process increase to each queue. Until ρ = 0.5, the rest of changes is all on UGS or BE queue in the simulation. The purpose here is to understand how long of the average queueing length and delay time are in the simulations under the condition of α = 0.05 and 0.01.

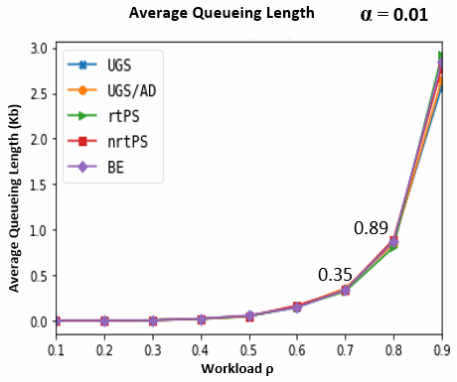
In Figure 15 and Figure 16, they presented the results that performing the input arrival rates of Figure 14. Basically, it is no obvious variation before ρ = 0.5. In Figure 15 and Figure 16, both the average queueing lengths and the average delay times are close if we saw UGS queue in Figure 15 compares to BE queue in Figure 16. And in Table 3 and Table 4, we can see the α values for both UGS and BE queue have similar results at the latest k_{th} round no matter what we are performing with α = 0.05 or 0.01. They achieved well-balanced level before ρ = 0.5. Why we said that here, it is because our MDSA features tolerating a small discrepancy, for instance, in column of ρ = 0.5 of Table 3 α = 0.01, we saw UGS queue has the latest k_{th} $w_{i,j}$ at



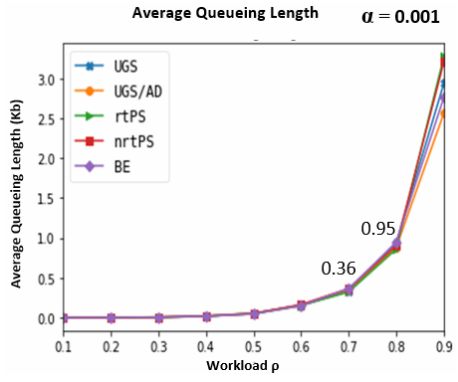
(a) α = 0.1



(b) α = 0.05

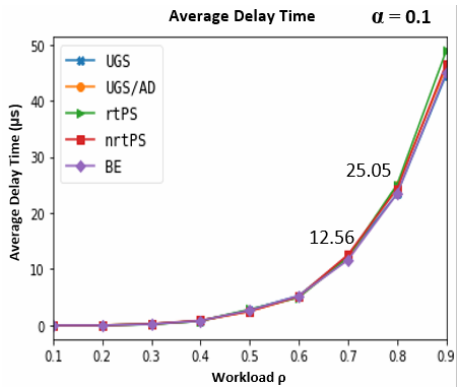


(c) α = 0.01

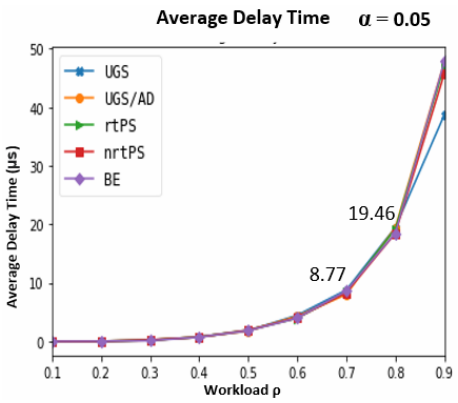


(d) α = 0.001

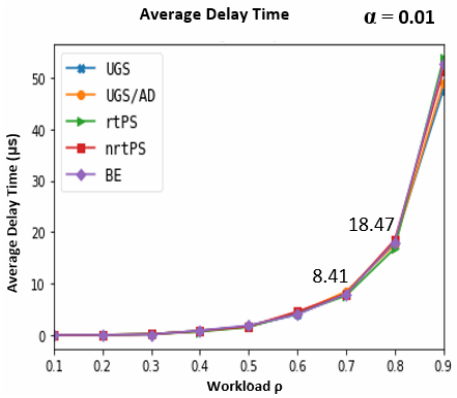
Figure 12. Average Queueing Length with α = 0.1, 0.05, 0.01, 0.001



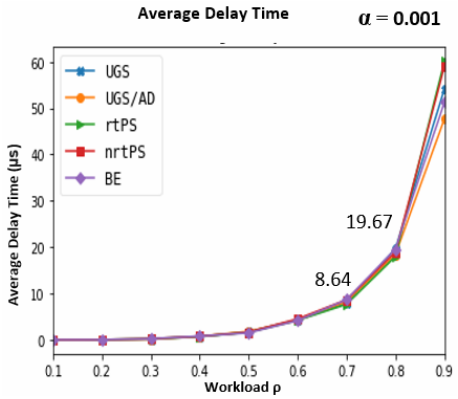
(a) α = 0.1



(b) α = 0.05



(c) α = 0.01



(d) α = 0.001

Figure 13. Average delay time with α = 0.1, 0.05, 0.01, 0.001

Table 2. The latest k_{th} value of $w_{i,j}$ with α = 0.05

α = 0.05

queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.95
UGS/AD	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	1.00
rtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	1.00
nrtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.80
BE	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.95

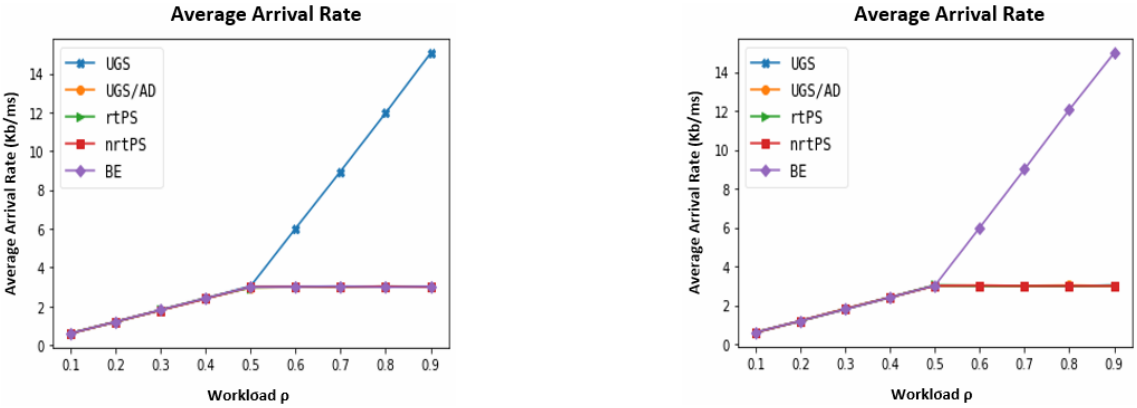
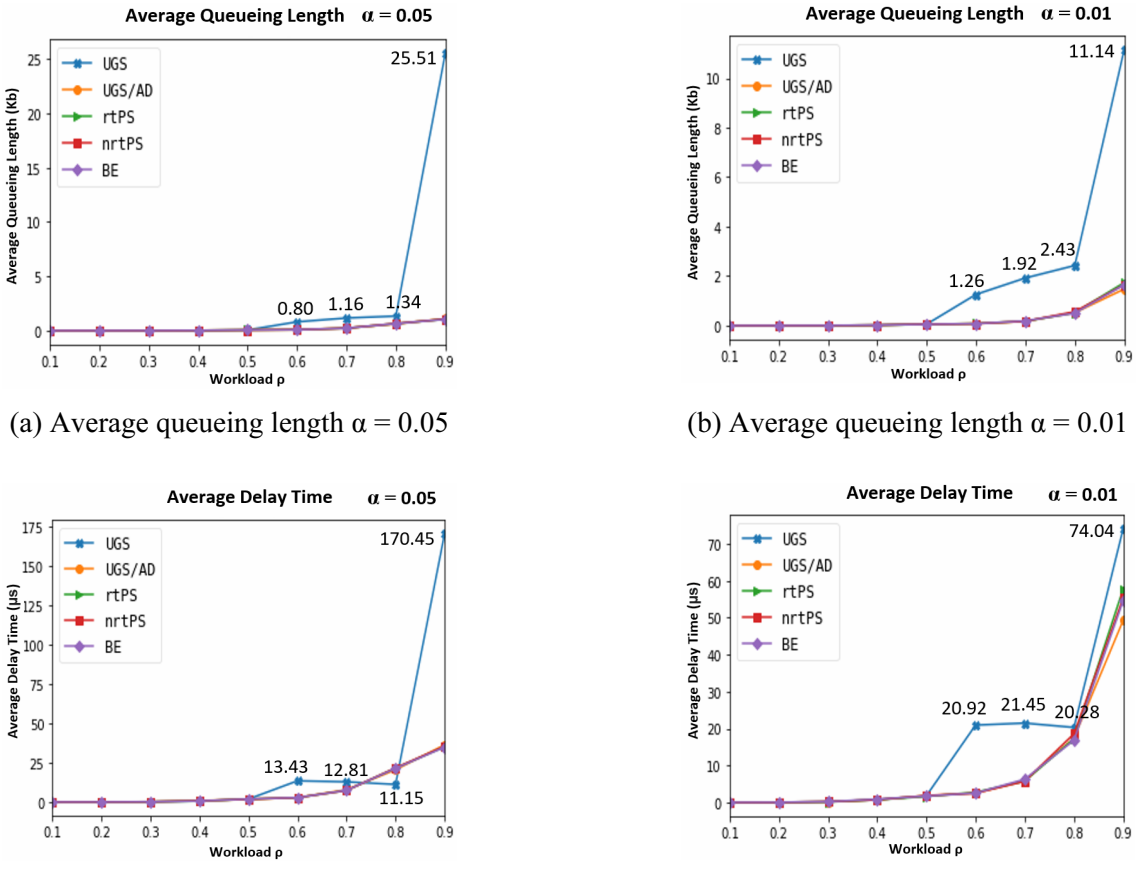


Figure 14. UGS/BE Rise in single downstream channel



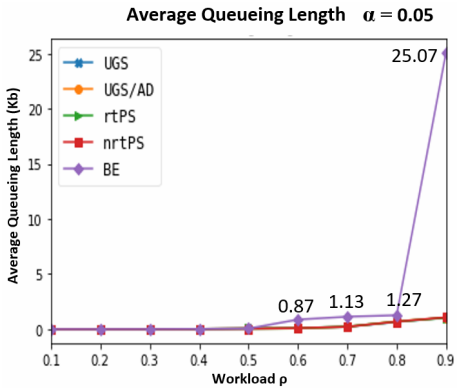
(a) Average queueing length α = 0.05

(b) Average queueing length α = 0.01

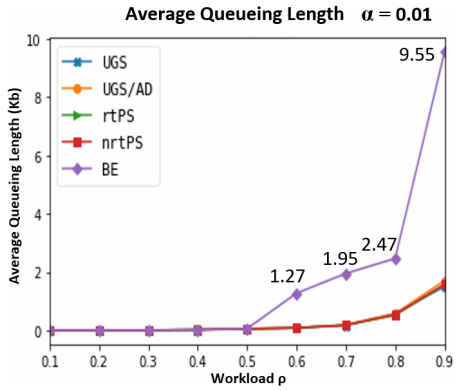
(c) Average delay time α = 0.05

(d) Average delay time α = 0.01

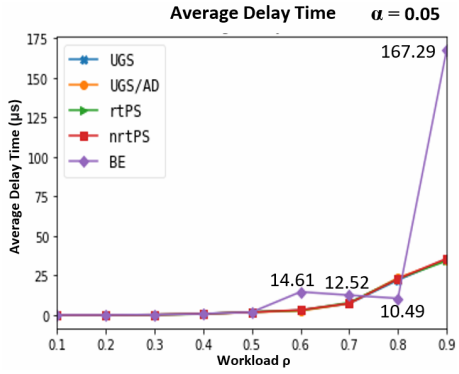
Figure 15. UGS variation in single downstream channel



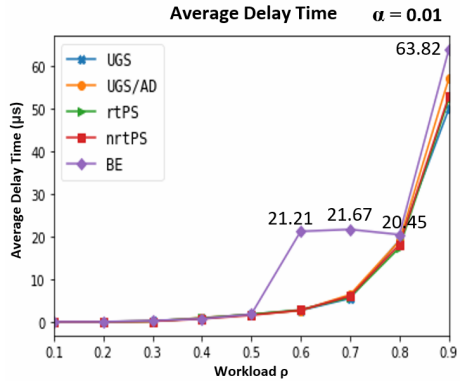
(a) Average queueing length α = 0.05



(b) Average queueing length α = 0.01



(c) Average delay time α = 0.05



(d) Average delay time α = 0.01

Figure 16. BE variation in single downstream channel

Table 3. The latest k_{th} value of $w_{i,j}$ with UGS variation with α = 0.05 and 0.01

(a) α = 0.05

α = 0.05

queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.15	0.15	0.15	0.15	0.15	0.30	0.65	1.00
UGS/AD	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.35
rtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.30
nrtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.20	0.25
BE	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.30

(b) α = 0.01

α = 0.01

queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.15	0.15	0.15	0.16	0.19	0.29	0.53	1.00
UGS/AD	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.16	0.21
rtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.22
nrtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.16	0.27
BE	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.17	0.22

Table 4. The latest k_{th} value of $w_{i,j}$ with BE variation with α = 0.05 and 0.01

(a) α = 0.05

α = 0.05

queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.25	0.20
UGS/AD	0.10	0.15	0.15	0.15	0.15	0.20	0.15	0.15	0.15
rtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.20	0.25
nrtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.20	0.25	0.20
BE	0.10	0.15	0.15	0.15	0.15	0.20	0.35	0.85	1.00

(b) α = 0.01

α = 0.01

queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.15	0.15	0.15	0.16	0.15	0.15	0.16	0.22
UGS/AD	0.11	0.15	0.15	0.15	0.15	0.15	0.15	0.17	0.21
rtPS	0.11	0.15	0.15	0.15	0.15	0.15	0.15	0.17	0.24
nrtPS	0.10	0.15	0.15	0.15	0.15	0.15	0.15	0.16	0.23
BE	0.10	0.15	0.15	0.15	0.16	0.22	0.29	0.60	1.00

0.16, but with α = 0.05, it has 0.15. It does not matter as a small gap of 0.01(0.16-0.15) exists, that is because while performing our method, by the calculation of $\frac{w_{i,1}^k}{\sum_{i=1}^{m} w_{i,1}^k} u_1^k(t)$, nothing obviously observed is with small gap of the calculation.

After ρ = 0.5, UGS and BE queues become risen, relatively, in Table 3, the value of $w_{UGS,1}$ became increased clearly, the same situation happened in Table 4, BE queue. Both UGS and BE queues follow the variations of input arrival rate from Figure 14 in modification. In Figure 15, it is easy to see with α = 0.05, the reaction to the input variation is quicker than α = 0.01. The consequence of average delay time is better also if we choose α = 0.05 as we obtained 11.15~13.43μs on the average delay time, but 20.28~21.45μs in α = 0.01 during ρ = 0.6~0.8 in Figure 15. Turned to see the status of BE queue in Figure 16, the situation is the same, 10.49~14.61μs can be obtained with α = 0.05, but the longer average delay time at 20.45~21.67μs is obtained in α = 0.01. Later, at ρ = 0.9, it is closest to the maximal workload in the condition of one downstream channel. At ρ = 0.9, all the values of $w_{UGS,1}$ and $w_{BE,1}$ equipped with maximal 1 in Table 3 and Table 4, which is running the maximal service rate for the highest input arrival rate against other queues that running with the lower values of $w_{i,j}$. Since the input arrival rate came huge, both the average queueing lengths and delay times are suddenly ramping up that cannot hold and control, causing every queue straight ramps up. But here applied with α = 0.01 is better than α = 0.05 as the bandwidth can be accurately arranged.

4.2.3 Scenario 3

In scenario 3, we are going to simulate the situation as all input arrival rate increase together shown in Figure 17. It is like Figure 11, all input arrival rates are fed with Poisson Random Process. In addition, we increase the rates as double from Figure 11 to observe the results in Multiple services. As the same as Scenario 2, α = 0.05 and 0.01 are adopted for the comparison in the simulations, besides, two different adjustments to $w_{i,j}$ will be considered in the multiple downstream channels. One is to adjust the value of $w_{i,j}$ to all downstream channels at the same time as trying to achieve a concept of loaded balance. Another one is to adjust the value of $w_{i,j}$ to each downstream channel per round making the adjustment separately.

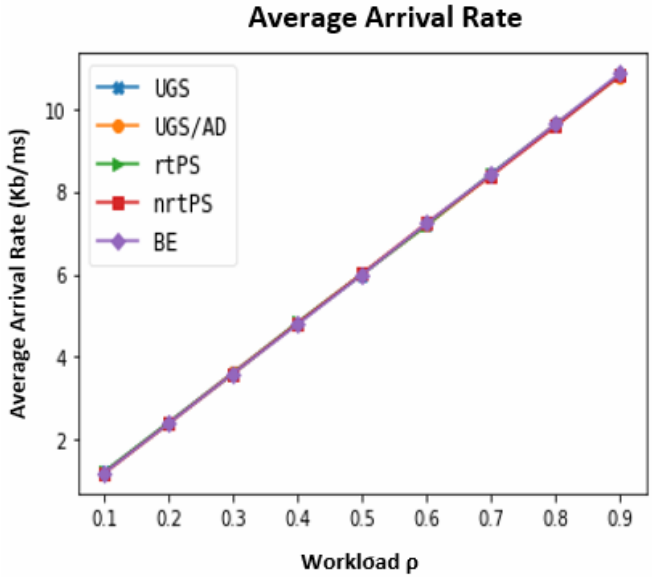
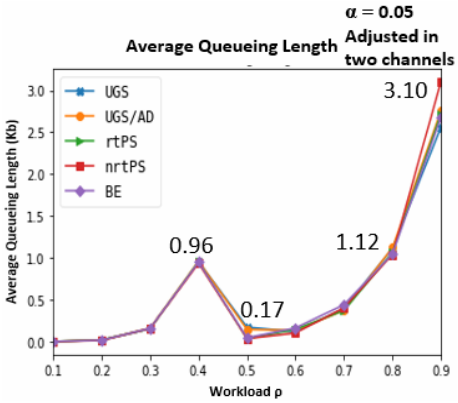


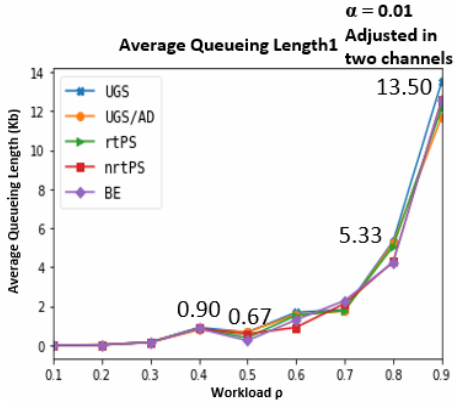
Figure 17. All input rise in dual downstream channels

In Figure 18 and Figure 19, they presented the results that performing the input arrival rates of Figure 17. First, if we compare the results in Figure 18, the result in the average queueing length with α = 0.05 shows a small spike occurs at ρ = 0.4, increasing the average queueing length up to 0.96kb (please note that the value we marked is the highest value of all queues, our purpose is to simply compare the discrepancies each other). Same situation we can see from Figure 11, as the input arrival rate in Figure 17, is running double as Figure 11, that accumulation is mostly like the outcome from Figure 12; when the workload goes up at ρ = 0.8, a few queueing lengths are being accumulated. The similar result we can receive in the simulation of α = 0.01, here we obtained 0.90kb in Figure 18. It shows all the simulation results are similar before the workload ρ = 0.4 no matter which α = 0.05 or 0.01 are adopted. After ρ = 0.05, the situation becomes changed. It is obvious at ρ = 0.05, the average queueing length is lower than the accumulation in α = 0.01. Apparently, the consequence is seeing the longer average delay time at 11.23μs at ρ = 0.05 in α = 0.01 of Figure 18. The same situation happens during ρ = 0.6~0.9, it explains the advantage of adopting α = 0.05 here again. Even though running with α = 0.01 is more precise of managing the bandwidth, the trade-off is we must choose the better solution of α = 0.05 as quicker as possible to react the input average arrival rate variation.

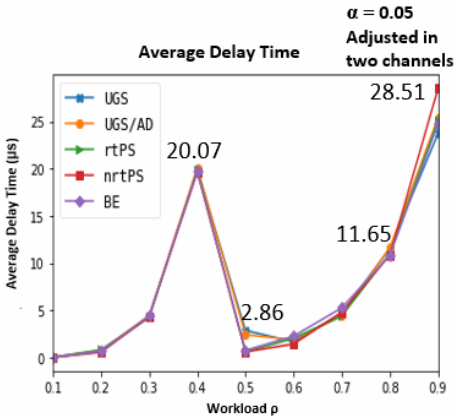
In Figure 19, it shows another way for $w_{i,j}$ adjustment, which is to adjust the value of $w_{i,j}$ to each downstream channel per round making the adjustment separately. Compared to Figure 18, running with α = 0.05, both the average queueing length and delay time do not differ much, the latest kth value of $w_{i,j}$ is listed in Table 5.



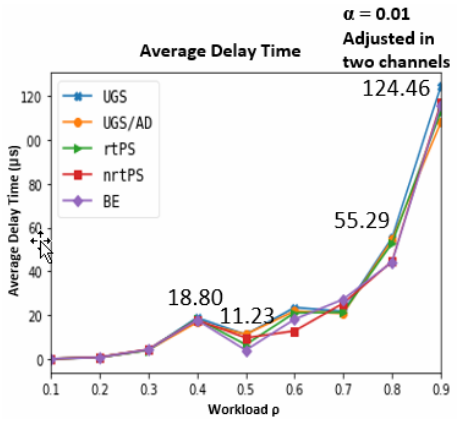
(a) Average queueing length α = 0.05



(b) Average queueing length α = 0.01

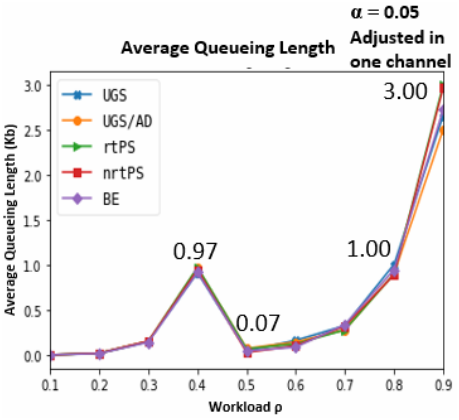


(c) Average delay time α = 0.05

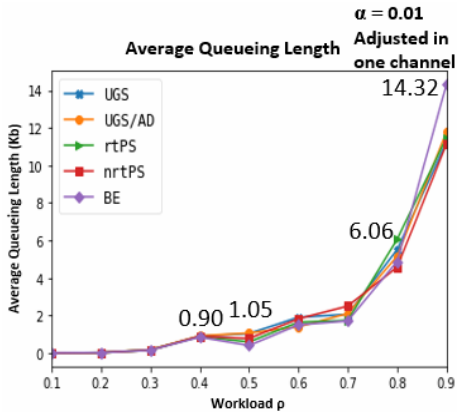


(d) Average delay time α = 0.01

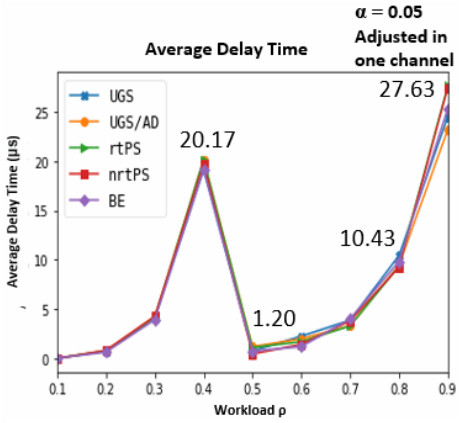
Figure 18. Two downstream channels adjustment with α = 0.05, 0.01



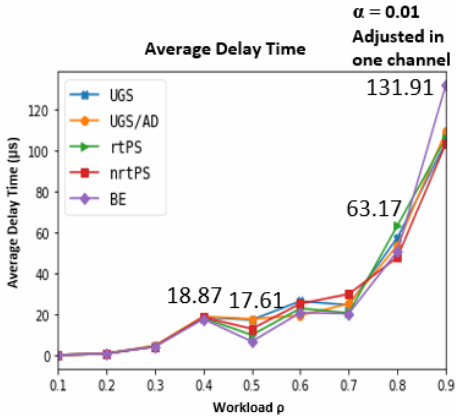
(a) Average queueing length $\alpha = 0.05$



(b) Average queueing length α = 0.01



(c) Average delay time $\alpha = 0.05$



(d) Average delay time α = 0.01

Figure 19. One downstream channel adjustment with α = 0.05, 0.01

Table 5. The latest k_{th} value of $w_{i,j}$ with all Input rise in α = 0.05 with one/two downstream channels adjustment

(a) $w_{i,j}$ adjusted in two channels

$w_{i,j}$ adjusted in two channels α = 0.05

	$w_{i,1}$									$w_{i,2}$								
queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.10	0.10	0.10	0.10	0.15	0.15	0.30	0.95	0.00	0.00	0.00	0.00	0.10	0.15	0.15	0.30	0.95
UGS/AD	0.10	0.10	0.10	0.20	0.10	0.10	0.10	0.25	1.00	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.25	1.00
rtPS	0.10	0.10	0.10	0.15	0.10	0.10	0.10	0.20	0.95	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.20	0.95
nrtPS	0.10	0.10	0.10	0.20	0.10	0.10	0.20	0.30	1.00	0.00	0.00	0.00	0.00	0.10	0.10	0.20	0.30	1.00
BE	0.10	0.10	0.10	0.20	0.10	0.10	0.10	0.30	0.95	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.30	0.95

(b) $w_{i,j}$ adjusted in one channel

$w_{i,j}$ adjusted in one channel α = 0.05

	$w_{i,1}$									$w_{i,2}$								
queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.15	0.15	0.15	0.20	0.75	0.85	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.20	0.95
UGS/AD	0.15	0.15	0.15	0.20	0.85	0.90	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.20	0.90
rtPS	0.15	0.15	0.15	0.20	0.85	0.90	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.15	0.15	0.20	0.20	0.90
nrtPS	0.15	0.15	0.15	0.20	1.00	0.95	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.20	1.00
BE	0.15	0.15	0.15	0.20	0.95	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.15	0.20	0.15	0.25	1.00

4.2.4 Scenario 4

In scenario 4, we are going to present if the increment happens only in UGS or BE queue, the corresponding average queueing length and average delay time how they will be in the two downstream channels. In Figure 20, two conditions show the variations of UGS/BE input arrival rate as double as what we did in Figure 14. In the period of ρ = 0.1~0.5, we set all the input arrival rates of Poisson Process increase to each queue. Until ρ = 0.5, the rest of changes is all on UGS or BE queue in the simulation. The purpose here is to understand which $w_{i,j}$ adjustment way is better for the condition of multiple downstream channels. In the comparison, we are going to show two cases, one is to adjust the value of $w_{i,j}$ to all downstream channels at the same time as trying to achieve a concept of loaded balance, another one is to adjust the value of $w_{i,j}$ to each downstream channel per round. All the simulations are performing in the better value of α = 0.05 based on the result in scenario 3.

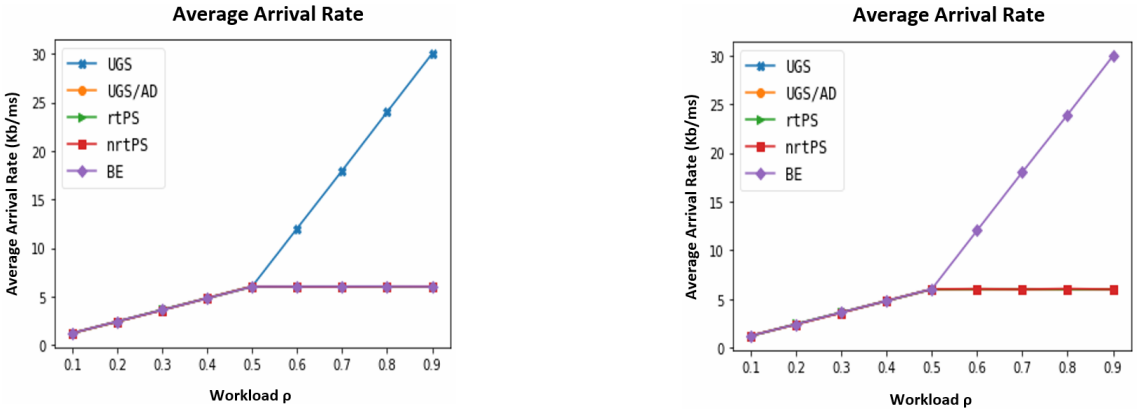
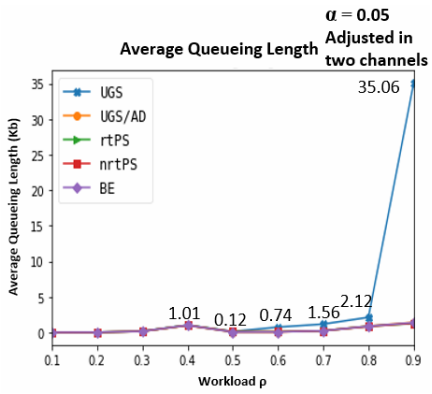


Figure 20. UGS/BE input rise in dual downstream channels

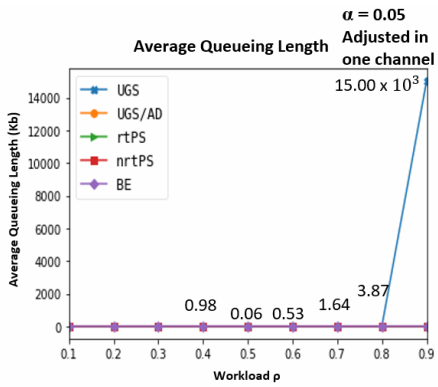
In Figure 21 and Figure 22, they presented the results that performing the input arrival rates of Figure 20. Since the input arrival rate in Figure 20 is double as the rates running in Figure 14. Basically, it is obvious that no variation before ρ = 0.3. In the period, Table 6 and Table 7 show the small change of $w_{i,j}$ at the latest k_{th} round, only one as 1st downstream channel serves the input arrival rate which both values of $w_{UGS,2}$ and $w_{BE,2}$ are 0 that the 2nd downstream channel has not been enabled at the moment. Later on, at ρ = 0.4, one small spike occurs in the average queueing length to both UGS and BE queue. It is like scenario 3, the simulations before ρ = 0.4, all the average input arrival rates increase together to the service, only one as 1st downstream channel. Until ρ = 0.5, the situation becomes changed along with the 2nd downstream is enabled. As you can see in both Table 6 and Table 7, both values of $w_{UGS,2}$ and $w_{BE,2}$ become increased from ρ = 0.5 on, the discrepancy to two types of $w_{i,j}$ adjustment way is becoming clear. First of all, during ρ = 0.5~0.7, we don't see the big differences between the adjustment ways of $w_{i,j}$ in two downstream channels modification or one channel per round. All the values are below 10μs in the average delay time of Figure 21 and Figure 22. However, at ρ = 0.8, the situation starts to change. In Figure 21, the comparisons in UGS queue, the average delay time at 8.81μs (2.12Kb accumulation) by $w_{i,j}$ adjustment way in two channels is obtained lower than 16.11μs (3.87Kb accumulation) that only one-channel $w_{i,j}$ adjustment. The situation in BE queue, Figure 22 is more intense. It has 11.19μs (2.68Kb accumulation) by $w_{i,j}$ adjustment way in two channels against 31.03μs(7.45Kb accumulation) one channel. It is much lower. With the discussion down to here, the statement of advantage to $w_{i,j}$ adjustment way in two channels is not finished. As you can see at ρ = 0.9, both simulations in UGS and BE queue clearly presented if you go with the $w_{i,j}$ adjustment way in two channels, the average queuing lengths are controllable at ρ = 0.9 in Figure 21 and Figure 22, but with the $w_{i,j}$ adjustment way in one channel, the average queueing lengths are out of the control, going straight to the top and divergent.

4.2.5 Scenario 5

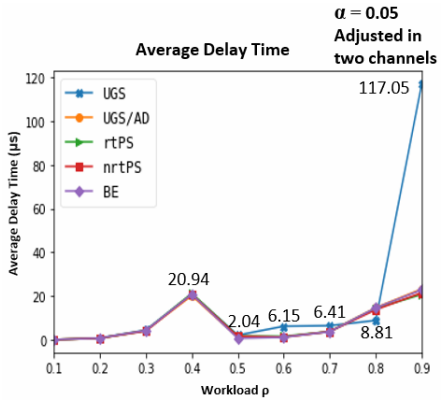
In scenario 5, we are going to present if the input arrival rate in UGS queue with a surge, how our MDSA responses. In Figure 23, it is showing an input arrival rates in the 10000 rounds, where a surge occurs at 30~60% of 10000 rounds, which is a 60Kb/ms surge enters the queue from the 3002nd round to the 6001st round. The rest of rounds is going to be stable running at ρ = 0.8 compared to single downstream channel, an input arrival rate of Poisson Process. In the simulation, the value of $w_{i,j}$ must be adjusted as quick as possible. Moreover, in other word, the number of downstream channels must be enabled along with the input surge. In scenario 4, we have known the adjustment for $w_{i,j}$, the better way is adjusting the value of $w_{i,j}$ in channels at the same time rather than adjusted in one channel each. Also, α = 0.05 was simulated as it is the better value to the simulation in scenario 3.



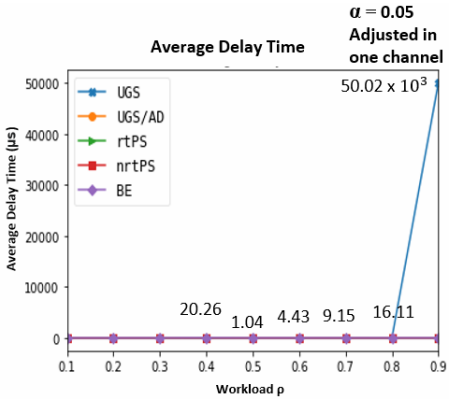
(a) Average queueing length: two channels



(b) Average queueing length: one channel

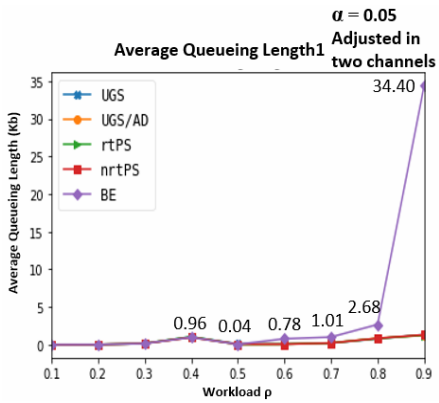


(c) Average delay time: two channels

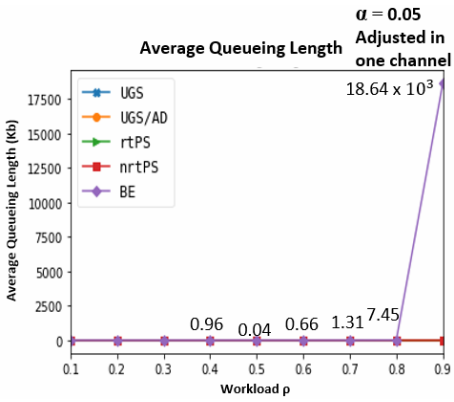


(d) Average delay time: one channel

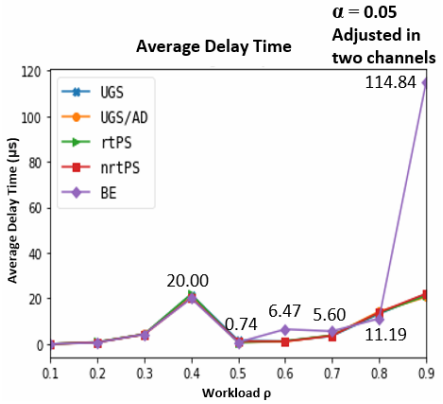
Figure 21. UGS variation in dual downstream channels



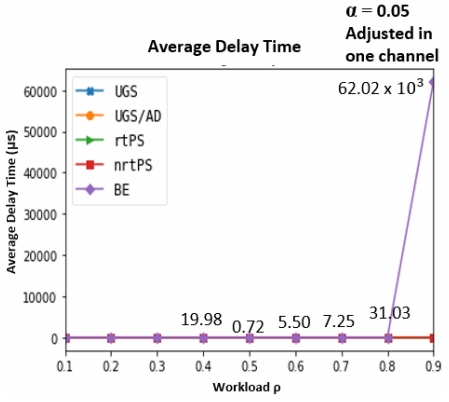
(a) Average queueing length: two channels



(b) Average queueing length: one channel



(c) Average delay time: two channels



(d) Average delay time: one channel

Figure 22. BE variation in dual downstream channel

Table 6. The latest k_{th} value of $w_{i,j}$ with UGS rise in α = 0.05 with one/two downstream channels adjustment

(a) $w_{i,j}$ adjusted in two channels

$w_{i,j}$ adjusted in two channels α = 0.05

	$w_{UGS,1}$									$w_{UGS,2}$								
queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.10	0.10	0.25	0.10	0.20	0.25	0.60	1.00	0.00	0.00	0.00	0.00	0.10	0.20	0.25	0.60	1.00
UGS/AD	0.10	0.10	0.10	0.25	0.10	0.10	0.10	0.10	0.30	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.10	0.30
rtPS	0.10	0.10	0.10	0.20	0.10	0.10	0.10	0.20	0.30	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.20	0.30
nrtPS	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.15	0.30	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.15	0.30
BE	0.10	0.10	0.10	0.15	0.10	0.10	0.10	0.15	0.30	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.15	0.30

(b) $w_{i,j}$ adjusted in one channel

$w_{i,j}$ adjusted in one channel α = 0.05

	$w_{UGS,1}$									$w_{UGS,2}$								
queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.15	0.20	0.15	0.20	0.85	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.15	0.20	0.50	1.00	1.00
UGS/AD	0.15	0.15	0.15	0.25	0.95	0.75	0.80	0.50	0.50	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.25	0.20
rtPS	0.15	0.15	0.15	0.30	0.80	0.60	0.65	0.45	0.55	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.25	0.20
nrtPS	0.15	0.20	0.15	0.30	1.00	0.55	0.70	0.50	0.55	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.20	0.15
BE	0.15	0.20	0.15	0.30	0.90	0.65	0.65	0.55	0.55	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.15	0.15

Table 7. The latest k_{th} value of $w_{i,j}$ with BE rise in α = 0.05 with one/two downstream channels adjustment

(a) $w_{i,j}$ adjusted in two channels

$w_{i,j}$ adjusted in two channels α = 0.05

	$w_{BE,1}$									$w_{BE,2}$								
queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.10	0.10	0.10	0.15	0.10	0.10	0.10	0.10	0.15	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.10	0.15
UGS/AD	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.20	0.15	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.20	0.15
rtPS	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.20	0.30	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.20	0.30
nrtPS	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.15	0.20	0.00	0.00	0.00	0.00	0.10	0.10	0.10	0.15	0.20
BE	0.10	0.10	0.10	0.10	0.10	0.15	0.25	0.55	1.00	0.00	0.00	0.00	0.00	0.10	0.15	0.25	0.55	1.00

(b) $w_{i,j}$ adjusted in one channel

$w_{i,j}$ adjusted in one channel α = 0.05

	$w_{BE,1}$									$w_{BE,2}$								
queue \ ρ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
UGS	0.15	0.15	0.15	0.20	0.95	0.60	0.50	0.55	0.65	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.25	0.20
UGS/AD	0.15	0.15	0.15	0.30	0.90	0.75	0.60	0.55	0.60	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.15	0.25
rtPS	0.15	0.15	0.15	0.30	0.95	0.60	0.55	0.70	0.60	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.25	0.25
nrtPS	0.15	0.15	0.15	0.20	1.00	0.70	0.55	0.70	0.60	0.00	0.00	0.00	0.00	0.15	0.15	0.15	0.20	0.20
BE	0.15	0.15	0.15	0.20	0.75	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.20	0.20	0.35	1.00	1.00

In Figure 24 and Figure 25, they presented the results that performing the input arrival rates of Figure 23. An input arrival surge at 60Kb/ms from 30~60% of 10000 rounds. Here we presented each result based on β = 2.5, 3.0, 3.5, and 4.0. In our MDSA, β is a ratio that representing $\rho_i^{k+1} > \beta * \rho_i^k$, whether the reaction to input surge at next round (ρ_i^{k+1}) is activated against ρ_i^k in Figure 10. In Table 8, As the input surge is entering UGS queue from the 3002nd round on. At the 3002nd round, all the simulations presented the same in β = 2.5, 3.0, 3.5, and 4.0. The value of $w_{UGS,j}$ becomes 1 in the queues, features the maximal service support compared to other UGS/AD, rtPS, nrtPS, and BE queues at the same round. Later, since all ρ_{UGS}^k values are larger than the upper bound of 0.9 (Shown in Table 8), the additional downstream channel is being enabled once and again until the 3006th round (total 4 downstream channels are in service), the value of ρ_{UGS}^k

became stable within the accepted value of 0.7 < ρ_{UGS}^{k} < 0.9, the additional downstream channel is no longer enabled. In Figure 24, the input surge causes an obvious spike happening from the 3002nd round, disappearing until the 3006th round (please note that the value we marked is the highest value in the 3002nd ~3006th round), all ρ_{UGS}^{k} values in β = 2.5, 3.0, 3.5, and 4.0 became stable within 0.7~0.9. In the rounds of 30~60%. 4 downstream channels are enabled in service. From 6001st round on, with the input arrival surge disappeared, our MDSA makes the value of $w_{UGS,j}$ going down in β = 3.0, 3.5, and 4.0 of Table 8. A decrease rate by α = 0.05 is adopted in each round.

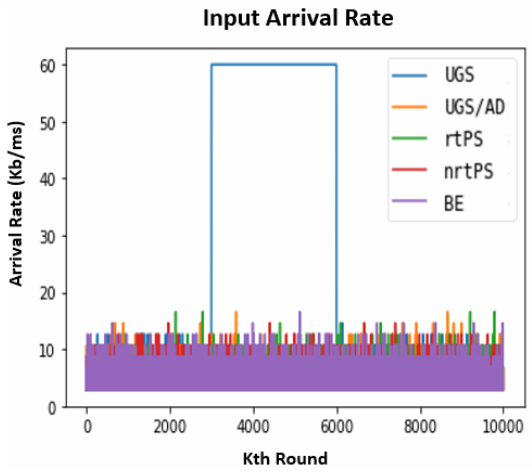
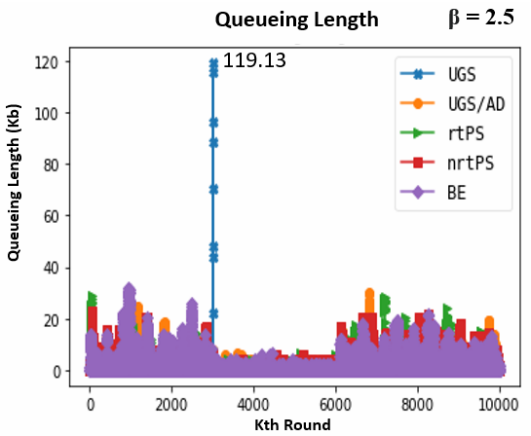
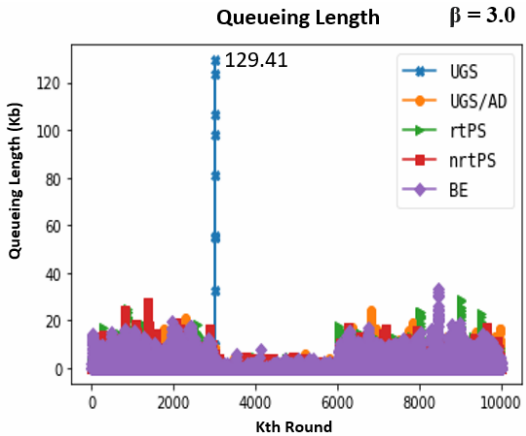


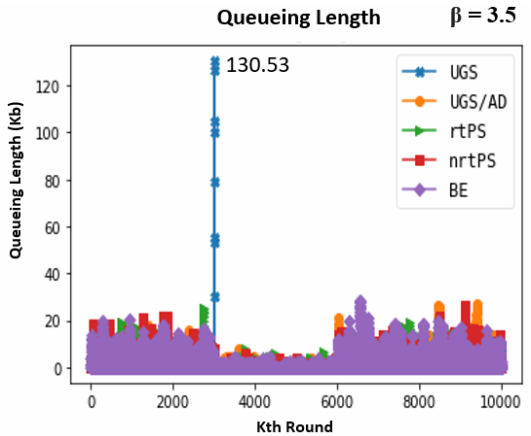
Figure 23. A surge input in UGS queue



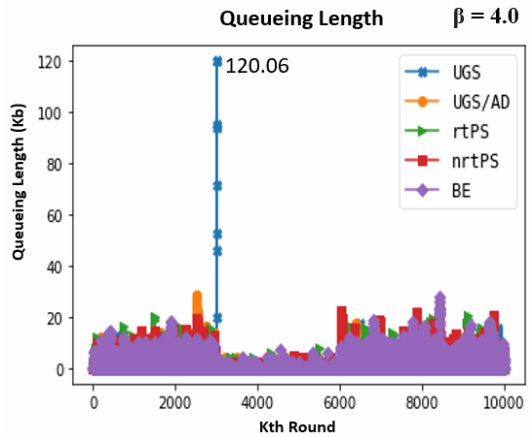
(a) Queueing length with β = 2.5



(b) Queueing length with β = 3.0



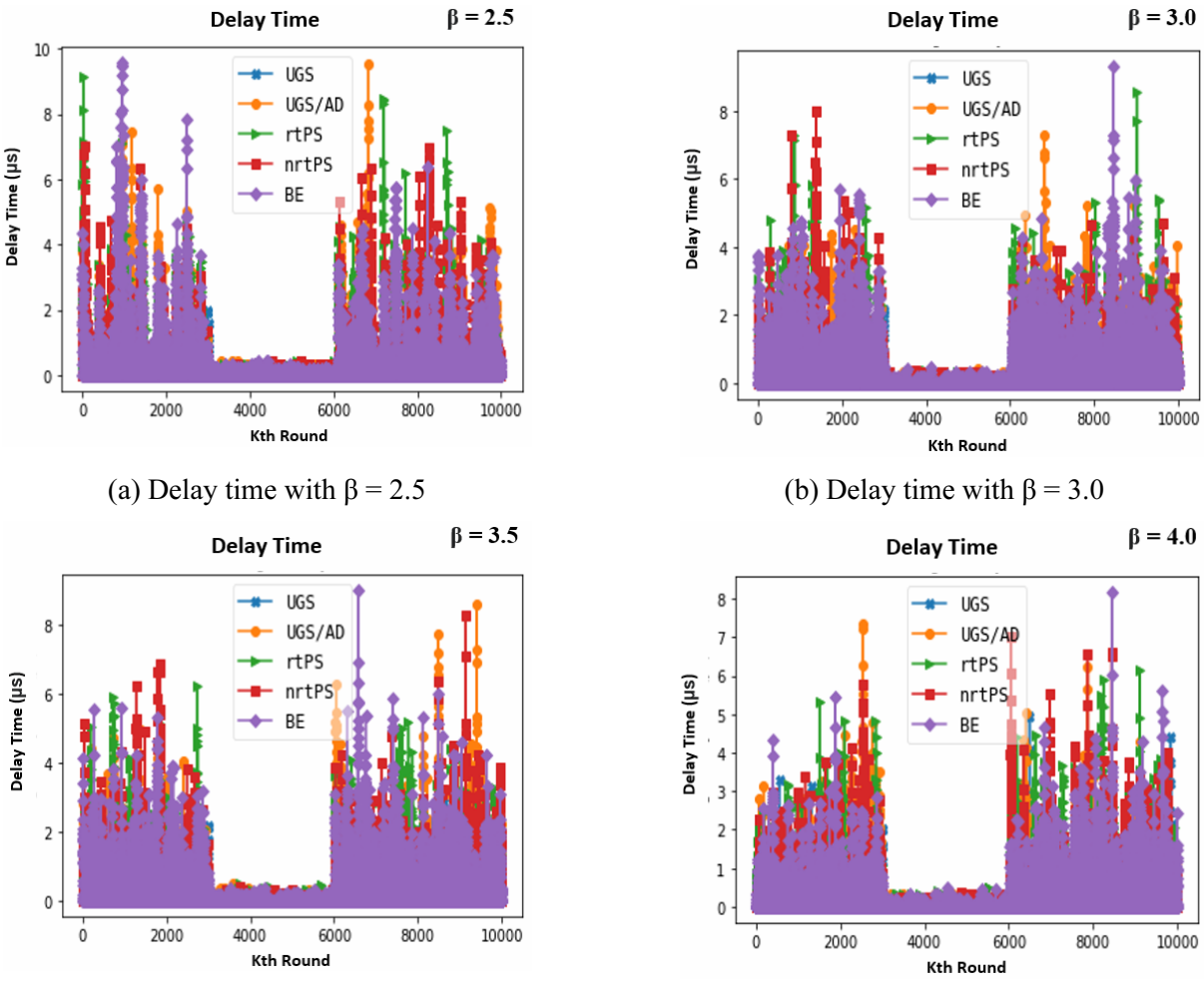
(c) Queueing length with β = 3.5



(d) Queueing length with β = 4.0

Figure 24. Queueing length with β = 2.5, 3.0, 3.5, 4.0 in 10000 rounds

In Table 8, β = 2.5, it is a good example to explain the divergence and sensitivity. The value of $w_{UGS,j}$ for surge is too sensitive to be activated easier if the value is set too low. As you can see from the 6001st round on, the input arrival surge started to disappear. The 6005th to 6006th round, the equation of $\rho_{UGS}^{6005} > 2.5 * \rho_{UGS}^{6006}$ shows 0.13 > 2.5*0.04 that all the values of $w_{UGS,j}$ are set to 1 in our MDSA. The setting of β = 2.5 reacted too sensitive, it results in the activations in 6011th to 6012th and 6035th to 6036th round, the decrease of $w_{UGS,j}$ cannot be executed in the rounds. Even though it presented the lowest average delay time in Table 9, it is still not a good choice toward the power-saving point of view, a waste for the energy as 4 channels are kept enabled so long. Thus, the rest of choices by β = 3.0,



(a) Delay time with β = 2.5

(b) Delay time with β = 3.0

(c) Delay time with β = 3.5

(d) Delay time with β = 4.0

Figure 25. Delay time with β = 2.5, 3.0, 3.5, 4.0 in 10000 rounds

Table 8. The k_{th} value of $w_{UGS,j}$ with β = 2.5, 3.0, 3.5, 4.0

(a) β = 2.5

$W_{i,j}$ adjusted in two channels β = 2.5

	Kth round / ρ_{UGS}, $w_{UGS,j}$	3001	3002	3003	3004	3005	3006	6005	6006	6011	6012	6035	6036
	ρ_{UGS}	0.42	3.68	4.00	1.80	1.07	0.73	0.04	0.13	0.04	0.11	0.06	0.15
	$w_{UGS,1}$	0.60	1.00	1.00	1.00	1.00	1.00	0.75	1.00	0.75	1.00	0.25	1.00
UGS	$w_{UGS,2}$	0.00	0.00	1.00	1.00	1.00	1.00	0.75	1.00	0.75	1.00	0.25	1.00
	$w_{UGS,3}$	0.00	0.00	0.00	1.00	1.00	1.00	0.75	1.00	0.75	1.00	0.25	1.00
	$w_{UGS,4}$	0.00	0.00	0.00	0.00	1.00	1.00	0.75	1.00	0.75	1.00	0.25	1.00

(b) β = 3.0

$W_{i,j}$ adjusted in two channels β = 3.0

	Kth round / ρ_{UGS}, $w_{UGS,j}$	3001	3002	3003	3004	3005	3006	6000	6001	6017	6018	6019	6020	6021
	ρ_{UGS}	0.66	13.00	3.50	1.75	1.10	0.73	0.73	0.03	0.17	0.27	0.28	0.16	0.36
	$w_{UGS,1}$	0.10	1.00	1.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.10	0.10
UGS	$w_{UGS,2}$	0.00	0.00	1.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.10	0.00
	$w_{UGS,3}$	0.00	0.00	0.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.00	0.00
	$w_{UGS,4}$	0.00	0.00	0.00	0.00	1.00	1.00	1.00	0.95	0.15	0.10	0.00	0.00	0.00

Table 8. The k_{th} value of $w_{UGS,j}$ with β = 2.5, 3.0, 3.5, 4.0 (continue)

(c) β = 3.5

$W_{i,j}$ adjusted in two channels β = 3.5

ρ_{UGS}, $w_{UGS,j}$ \ Kth round		3001	3002	3003	3004	3005	3006	6000	6001	6017	6018	6019	6020	6021
	ρ_{UGS}	0.47	13.30	3.90	1.80	1.07	0.71	0.73	0.04	0.12	0.09	0.12	0.16	0.26
	$w_{UGS,1}$	0.10	1.00	1.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.10	0.10
UGS	$w_{UGS,2}$	0.00	0.00	1.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.10	0.00
	$w_{UGS,3}$	0.00	0.00	0.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.00	0.00
	$w_{UGS,4}$	0.00	0.00	0.00	0.00	1.00	1.00	1.00	0.95	0.15	0.10	0.00	0.00	0.00

(d) β = 4.0

$W_{i,j}$ adjusted in two channels β = 4.0

ρ_{UGS}, $w_{UGS,j}$ \ Kth round		3001	3002	3003	3004	3005	3006	6000	6001	6017	6018	6019	6020	6021
	ρ_{UGS}	0.53	8.00	3.50	1.70	1.00	0.71	0.71	0.04	0.07	0.09	0.20	0.27	0.40
	$w_{UGS,1}$	0.20	1.00	1.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.10	0.10
UGS	$w_{UGS,2}$	0.00	0.00	1.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.10	0.00
	$w_{UGS,3}$	0.00	0.00	0.00	1.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.00	0.00
	$w_{UGS,4}$	0.00	0.00	0.00	0.00	1.00	1.00	1.00	0.95	0.15	0.10	0.10	0.00	0.00

Table 9. Average queueing length and delay time with β = 2.5, 3.0, 3.5, 4.0

(a) Average queueing length

queue \ β	2.5	3.0	3.5	4.0
UGS	0.13	0.39	0.37	0.61
UGS/AD	1.34	1.39	1.44	1.20
rtPS	1.46	1.47	1.36	1.26
nrtPS	1.38	1.45	1.50	1.27
BE	1.38	1.59	1.50	1.14

(b) Average delay time

queue \ β	2.5	3.0	3.5	4.0
UGS	0.01	0.05	0.04	0.09
UGS/AD	0.26	0.26	0.28	0.22
rtPS	0.29	0.29	0.26	0.24
nrtPS	0.28	0.28	0.29	0.24
BE	0.29	0.32	0.29	0.21

3.5, and 4.0 of Table 8, they are all good for adoption in the simulation as they feature convergent in Table 8. From the 6001st round on, they start to decrease the value of $w_{UGS,j}$ by α = 0.05. And starting to disable the downstream channel if $w_{UGS,j}$ is down to 0.10, which is the initial state we set in Table 1. In the simulations, the UGS queue can be securely served down to one downstream channel from the 6021st round.

5 Conclusion and Future Work

In this paper, the challenge is how to manage the bandwidth of multi-downstream channels in DOCSIS. People are always expecting the good service of internet. Now, the useful and efficient methodology to bandwidth management becomes more and more important. In [13], Horng et al proposed a method that utilizing WFQ to effectively control a queueing delay in Wireless Base Station; it features the higher-weight value can arrange the relatively higher bandwidth to the corresponding queue. But it didn't operate as flexible which owns the dynamic weight parameter to react the different QoS applications. In this paper, we presented the variable weight value that observing the input arrival rate to adjust the weight accordingly by the variation of workload ρ_i^K. The feature made the bandwidth be effectively used in the QoS queues. In addition, we also presented how the methodology works in both single and multiple downstream channels as shown in section 4. First, we compared the different values of α and found out the better value of 0.05 for the adjustment as it is quicker in response of the input variation. Later, we found the benefit to choose the way as to adjust the value of $w_{i,j}$ to all downstream channels at the same time, trying to achieve a concept of loaded balance. Finally, through the simulation, the parameter β≥3.0 has been discovered as the solution in the scenario 5, featuring power-saving as the downstream channels can be effectively disabled. In the future, how to normalize the parameters of α and β by formulas will be the next work. It needs to study the knowledge of extrema, trying to create an idea to obtain the maxima and minima in mathematical analysis.

Acknowledgements

This work is supported partially by the Ministry of Science and Technology, Taiwan, R.O.C., under the grant No. MOST 109-2221-E-032-023.

References

[1] P. Addesso, M. Cirillo, M. Di Mauro, M. Longo, V. Matta, Adversarial Detection of Concealed VoIP Traffic, *International Conference on Computing, Networking and Communications (ICNC)*, Honolulu, HI, USA, 2019, pp. 437-441.

[2] E. Cipressi, M. L. Merani, An Effective Machine Learning (ML) Approach to Quality Assessment of Voice Over IP (VoIP) Calls, *IEEE Networking Letters*, Vol. 2, No. 2, pp. 90-94, June, 2020.

[3] Cable Television Laboratories, Inc., *Data-Over-Cable Service Interface Specifications, DOCSIS3.0, Physical Layer Specification*, CableLabs, CM-SP-PHYv3.0-C01-171207, December, 2017.

[4] Cable Television Laboratories, Inc., *Data-Over-Cable Service Interface Specifications, DOCSIS3.0, MAC and Upper Layer Protocols Interface Specification*, CableLabs, CM-SP-MULPIv3.0-C01-171207, December, 2017.

[5] Y.-D. Lin, C.-Y. Huang, W.-M. Yin, Allocation and Scheduling Algorithms for IEEE 802.14 and MCNS in Hybrid Fiber Coaxial Networks, *IEEE Transactions on Broadcasting*, Vol. 44, No. 4, pp. 427-435, December, 1998.

[6] A.-C. Liu, *Dynamic Channel Allocation Scheme in Multi-Channel CATV Network*, Master's Thesis, Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan, 2001.

[7] C.-C. Chen, *A Scheduling Mechanism in Multi-channel HFC Network for non-UGS Services*, Master's Thesis, Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan, 2004.

[8] C.-P. Tan, *A Research of Bandwidth and Channel Assignment Scheme in Multi-Channel Cable Network for UGS Service Provisioning*, Master's Thesis, Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan, 2003.

[9] H.-S. Koo, D.-J. Choi, N. Hur, Dynamic Load Balancing Algorithm for Upstream Bandwidth Allocation in DOCSIS 3.0, *International Conference on ICT Convergence (ICTC)*, Jeju, Korea, 2012, pp. 710-713.

[10] W.-T. Lee, K.-C. Chung, K.-C. Chu, J.-Y. Pan, DOCSIS Performance Analysis Under High Traffic Conditions in the HFC Networks, *IEEE Transactions on Broadcasting*, Vol. 52, No. 1, pp. 21-30, March, 2006.

[11] T. Kusunoki, T. Kurakake, Y. Kawamura, K. Imamura, K. Saito, Development of in-building transmission device utilizing DOCSIS standard and IP encapsulation method for 4K/8K multi-channel IP broadcas, *IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, 2020, pp. 1-5.

[12] M. J. Emmendorfer, Cable Operator's Access Architecture from Aggregation to Disaggregation and Distributed, *IEEE Photonics Society Summer Topical Meeting Series (SUM)*, Ft. Lauderdale, FL, USA, 2019, pp. 1-2.

[13] M. F. Horng, Y. H. Kuo, J. P. Hsu, R. H. Cheng, Adaptive slot allocation to control queueing delay in TDMA wireless base station, *Journal of Information Science and Engineering*, Vol. 20, No. 5, pp. 845-868, September, 2004.

[14] C. K. Jha, H. Y. Zorkta, A. H. Al-Saleh, F. N. Fakhrow, New Queuing Technique for Improving Computer Networks QoS, *International Conference for Emerging Technology (INCET)*, Belgaum, India, 2020, pp. 1-5.

[15] L. Xu, H. Li, F. Dong, Y. Zhou, Assessment of Data Flow Control Methods and their Performance on IEC61850 based Digital Substation, *IEEE 8th International Conference on Advanced Power System Automation and Protection (APAP)*, Xi'an, China, 2019, pp. 1408-1412.

[16] C. You, Hierarchical Multi-resource Fair Queueing for Network Function Virtualization, *INFOCOM 2019 - IEEE Conference on Computer Communications*, Paris, France, 2019, pp. 406-414.

[17] X. Wang, Y. Pi, A. Tang, Scheduling of Electric Vehicle Charging via Multi-Server Fair Queueing, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 28, No. 11, pp. 3298-3312, November, 2017.

[18] A. A. Alsulami, Q. A. Al-Haija, M. I. Thanoon, Q. Mao, Performance Evaluation of Dynamic Round Robin Algorithms for CPU Scheduling, *2019 SoutheastCon*, Huntsville, AL, USA, 2019, pp. 1-5.

[19] B. Shen, Z. Wang, D. Wang, H. Liu, Distributed State-Saturated Recursive Filtering Over Sensor Networks Under Round-Robin Protocol, *IEEE Transactions on Cybernetics*, Vol. 50, No. 8, pp. 3605 - 3615, August, 2020.

[20] J. F. Shortle, J. M. Thompson, D. Gross, C. M. Harris, *Fundamentals of Queueing Theory, Fifth Edition*, John Wiley & Sons, Inc., 2018.

[21] V. Skorpil, V. Oujezsky, M. Tuleja, Testing of Python Models of Parallelized Genetic Algorithms, *International Conference on Telecommunications and Signal Processing (TSP)*, Milan, Italy, 2020, pp. 235-238.

Biographies

Yao-Chiang Yang received his MS degree in Electrical Engineering from Tamkang University, Taiwan. He is currently a Ph.D. student in the Department of Electrical and Computer Engineering, Tamkang University, New Taipei City, Taiwan, His research interests include wireless networking, and computer networks.

Wei-Tsong Lee received his BS, MS and PhD degrees in Electrical Engineering from National Cheng Kung University, Tainan, Taiwan. He is currently a professor in the Department of Electrical and Computer Engineering, Tamkang University. His research interests include computer architecture, micro-processor interface and computer networks.

Chih-Hsing Chen received his BS degree in Electrical Engineering from Tamkang University, Taiwan. He is currently a MS student in the Department of Electrical and Computer Engineering, Tamkang University, New Taipei City, Taiwan, His research interests include computer networks.