國 立 交 通 大 學

電信工程研究所

碩 士 論 文

時槽式光叢集交換環中波長分配與訊務控制

Wavelength Assignment and Traffic Control (WATC) in Slotted OBS Rings

研 究 生：謝耀庭

指導教授：張仲儒 博士

中 華 民 國 九 十 九 年 七 月

時槽式光叢集交換環中波長分配與訊務控制

Wavelength Assignment and Traffic Control (WATC) in Slotted OBS Rings

研 究 生：謝耀庭　　　　Student：Yao-Ting Hsieh

指導教授：張仲儒 博士　　Advisor：Dr. Chung-Ju Chang

國立交通大學

電信工程研究所

碩士論文

A Thesis

Submitted to Department of Communication Engineering

College of Electrical and Computer Engineering

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of Master of Science

in

Communication Engineering

July 2010

Hsinchu, Taiwan

中華民國九十九年七月

時槽式光叢集交換環中波長分配與訊務控制

研究生：謝耀庭　　　　　　指導教授：張仲儒

國立交通大學電信工程研究所碩士班

摘要

　　光叢集交換(Optical Burst Switching)環主要是設計應用於高速的都會型區域網路，並希望能支援的多種類型的服務。在光叢集交換環中，頻寬利用率、公平性、和穩定性都是很重要的議題。在本篇論文中，我們提出了一個波長分配與訊務控制(WATC)的方案來達成上述的考量。系統採用了預約的機制來避免時槽式系統中資料的碰撞，而且為了減少光電之間的轉換，每個節點中都沒有過境緩衝器。首先，排程器會依據入口緩衝器中佇列的權重來分配頻寬，然後由波長分配器(WA)決定適合的位置。波長分配器會適時地調整將要產生的資料叢集的大小，藉此有效利用空隙並提升頻寬利用率。另一方面，為了減輕一個下游節點可能遭受到頻寬挨餓的問題，一個乏晰保護性預約產生器(FPRG)會參考過境訊務的情況，動態地預留一些頻寬給入境的聲音與影像訊務。然後，公平流速產生器(FRG)再根據剩餘的頻寬產生一個評估過的公平流速，並由訊務控制器(TC)決定出最終宣傳的公平流速來規範上游的節點。模擬的結果顯示所提出的方案在波長分配的表現上有較高的頻寬利用率。在訊務控制方面，它也能使頻寬的使用達到公平。由於觀察窗口的應用，收斂時間可以比 DBA 縮短許多。波長分配與訊務控制的方案在各種不同的網路情景下都能展現出它穩定性。因此，它不失為一個可用在光叢集交換環的不錯方案。

Wavelength Assignment and Traffic Control (WATC) in Slotted OBS Rings

Student: Yao-Ting Hsieh   Advisor: Chung-Ju Chang

Institute of Communication Engineering

National Chiao Tung University

Abstract

The optical burst switching (OBS) ring network is mainly designed for high-speed metropolitan area network, which is expected to support many kinds of services. In the OBS rings, the issues of bandwidth utilization, service differentiation, fairness, and stability are important. In this thesis, the wavelength assignment and traffic control (WATC) scheme is proposed to achieve these considerations. The reservation mechanism is adopted to avoid burst collisions in the time-slot based system, and no transit buffers are deployed at each node for the reason of bypassing the traffic directly. The wavelength assigner (WA) cooperates with the scheduler to generate and then transmit data at the determined position, thereby increasing the bandwidth utilization and supporting service differentiation. In order to alleviate the bandwidth starvation problem a downstream node may suffer from, a fuzzy protective reservation generator (FPRG) preserves some bandwidth for ingress voice and video traffic referring to the network conditions. Afterwards, the fair rate generator (FRG) produces an estimated fair rate, and the traffic controller (TC) determines the advertised fair rate which regulates the upstream node. Since the observation window is applied, the convergence time is reduced. Besides, the WATC scheme shows stability in different kinds of network scenarios. The simulation results also show that the bandwidth utilization is high. Consequently, it is concluded that the WATC scheme is promising for the OBS rings.

誌 謝

能夠順利的完成這篇碩士論文，必須感謝非常多人的鼓勵與幫忙。首先我要感謝張仲儒教授，除了指導我論文的方向與撰寫技巧外，也在待人處世方面給予了許多的建議，讓我在學業和生活上都獲益良多。其次我要感謝文祥學長在論文研究上提供了許多的建議，並且助我解決面臨到的一些問題。還要特別感謝抽空回來指導我們的詠翰和芳慶學長，提供我們最新的資訊和專業上的看法。感謝文敬、聖章、文祥、志明、耀興、振宇、吉成、益興、奇恩學長們，在我遇到困難的時候提供寶貴的意見。感謝常常帶給我們實驗室歡樂的學長姐欣毅、和儁、志遠、盈仔，學弟竣威、苡仲、俊魁，以及辛苦的助理玉棋。當然，也要感謝在這兩年來和我一起努力的正忠、蕊綺、心瀅，謝謝你們平時的照顧。

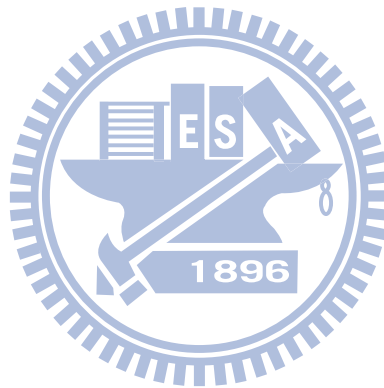最後我要感謝我的父母及、親人和女友，若沒有你們的體諒和支持，我無法專注在課業上並順利完成論文。

<div align="right">

謝耀庭 謹誌

民國九十九年七月

</div>

Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Motivation and Objective

Optical burst switching (OBS) is a promising system technique currently under study that can be used to transport data over a wavelength division multiplexing (WDM) optical network [1], [2]. The OBS occupies the middle spectrum between the coarse-grain bandwidth optical circuit switching (OCS) scheme and the fine-grain bandwidth optical packet switching (OPS) scheme [3]-[5]. It can provide a flexible infrastructure for carrying future Internet traffic in an efficient and practical manner. On the other hand, ring topology has been a good choice for solving the current metro gap problem between core network and access network owing to its simplicity and scalability [4]-[6]. It can serve as a metropolitan area network (MAN) that interconnects a number of access networks.

Each network node in a ring network employs OBS techniques to send and receive data traffic, respectively. There are several architectures with variants combinations of transmitters and receivers in OBS nodes [7]-[10]. Fixed transmitters or receivers can only transmit or receive data on a particular fixed wavelength. With the fixed transmitter and tunable receiver (FT–TR) architecture as in [7], a node can transmit bursts without worrying about channel collision, but several bursts may arrive at the same node simultaneously and result in receiver collision. While with the

tunable transmitter–fixed receiver (TT–FR) architecture in [8], each destination node can receive bursts without worrying about the receiver collision. However, since each could use all wavelengths for transmitting a burst to the same destination, channel collision would not be avoided. Also, the two types of architectures would waste bandwidth because the node only transmits or receives the bursts in the fixed wavelengths due to the fixed transmitter or fixed receiver. A more scalable and flexible system with tunable transmitter–tunable receiver (TT–TR) architecture was also proposed [9], [10]. Its advantages come at the expense of a higher resource contention possibility and a higher packet loss probability. In this thesis, the TT-TR architecture is adopted for the reason of scalability.

In an OBS ring, control and data planes are separated by means of dedicating one wavelength for the control plane and the remaining wavelengths for the data plane. In the OBS ring, the several packets with the same destination in an ingress node would be assembled into a data burst, named by DB, by an assembly algorithm and the ingress node also generates the associated control burst of the DB, named by CB, containing the related information about the DB. The assembly algorithms are divided into three types: length based algorithm, time based algorithm and hybrid algorithm [11]. An actual data bursts (DBs) pass through the intermediate nodes up to the destination all optically and it must be transmitted according to a specific medium access control (MAC) protocol to avoid burst collisions. On the contrary, a control burst (CB) is sent on a separate control wavelength requiring optical-to-electronic (O/E) and electronic-to-optical (E/O) conversions at every node. This can restrict O/E and E/O conversions to be only needed by the control wavelength instead of the large number of data wavelengths.

Several MAC protocols in the OBS rings have been proposed and they can be

classified into two major categories: token based scheme and time-slot based scheme [12], [13], [14]. In token based schemes [12], [13], tokens are used to control the medium access and thus avoid burst collisions. For each data wavelength, one or multiple tokens circulating around the ring in the control wavelength are used to control medium access. Only the node which holds the token can transmit its DBs on the data wavelength. Prior to any data transmission, a control burst is sent to reserve resources of every node on the path. The CB contains some important information, such as wavelength used, offset time, burst length, source node, and destination node. Although the token based schemes indeed avoid burst collisions, their low channel utilization is still a big problem to be solved. In time-slot based scheme [14], all the wavelengths are divided into many frames, and a frame contains a fixed number of time-slots. For each frame, there is a corresponding control message rotating in the control wavelength for record of slots usage. The status of each time-slot is represented by a bit in the control message. An empty time-slot is marked as bit "0" and an occupied time-slot is marked as bit "1." Every node which has DBs to transmit must reserve time-slots in advance and initiate data bursts transmission in the next frame time. This time-slot based scheme has higher utilization than token-based schemes. However, it does not support service differentiation [14].

Also, the future optical networks are expected to support integrated multimedia services. Several MAC schemes with quality of service (QoS) assurance have been proposed for the OPS WDM networks [15]-[17]. The real-time traffic, such as constant bit rate (CBR) and variable bit rate (VBR), is subject to the call admission control (CAC) which accepts connections only when all demands are guaranteed to be satisfied [18], [19]. While the non real-time traffic, such as available bit rate (ABR), exploits all the remaining bandwidth. The multimedia-MAC protocol was proposed to

integrate MAC protocols to accommodate various types of multimedia traffic with different QoS demands [15]. The mathematical analysis is also shown to evaluate the performance. In the frame-based slot reservation scheme proposed, the real-time services are satisfied by reserved slots through the connection setup and termination, and the data services are met by reserving the minimum bit rate [16]. Another QoS assurance scheme proposed is based on priorities, in which the scheme favors high-priority packets [17]. Moreover, it differentiates the packets' schedule order by prioritizing the long-length over the short-length packets.

Another important work in OBS networks is the wavelength assignment. An appropriate wavelength assignment method not only makes better utilization of link bandwidth but also supports service differentiation. Some wavelength assignment methods have been proposed for OBS network [20], [21], [22]. With consideration of extremely high speed environment, the OBS-GS is proposed to minimize the voids while maintaining its simplicity [21]. A utility-based fuzzy wavelength assignment (UFWA) was proposed to further reduce the voids between DBs and utilize the wavelengths more efficiently [22].

Moreover, there is still a problem of bandwidth sharing among the nodes for ring networks. When traffic load increases, the downstream nodes may suffer from bandwidth starvation. A traffic control method could prevent this situation and provide fair access to the link bandwidth. In addition to fairness, the issues of traffic control include stability and convergence time. The stability would avoid the oscillation of regulated traffic flows, which would decrease the throughput. The convergence time is the time interval between the start of the overused link and the instant that the throughputs of regulated traffic flows approach the fair rate. Some existing methods of traffic control have been proposed for the resilient packet ring

(RPR), which is also a ring based network, and the concept of these designs is good for references [23], [24], [25].

Recently, intelligent techniques such as fuzzy logic control systems have been widely used to control nonlinear and time-varying systems owing to their appealing properties. A major property is that they can continuously adapt themselves to the dynamic, uncertain, and bursty environment. Fuzzy and neural fuzzy implementations of congestion control method, admission control method, and route control were studied in the literature [23], [24], [25], [26]. In the fuzzy logic control systems, it is easy to incorporate the human experience into the design of the fuzzy logic controller (FLC). Thus, the FLC computes values of control variables from the observation of state variables such that a desired system performance is achieved. It can provide a simple and efficient solution to our problem of traffic control.

In this thesis, we propose an intelligent wavelength assignment and traffic control (WATC) to determine the wavelengths for ingress traffic and adjust the traffic frame by frame in the OBS rings. The WATC consists of the wavelength assigner (WA), the local fair rate generator (LFRG), and the traffic controller (TC). The WA is introduced to determine the wavelengths of the ingress DBs for the purpose of high channel utilization and service differentiation. In order to control the traffic to achieve the fairness in OBS rings, an advertised fair rate is send to the upstream node to regulate the maximum transmission rate. The LFRG calculates such a fair rate from the local view point. Besides, a fuzzy protective reservation generator (FPRG) is introduced to reserve appropriate bandwidth for future ingress traffic with regard to the uncertainty of network. Finally, the traffic controller (TC) is designed to adjust the advertised fair rate according to the traffic condition.

Using the same traffic control method, the wavelength assignment of the WATC

has link utilization 5% higher than OBS-GS. On the other hand, when traffic control methods are compared, simulation results show that the WATC scheme takes 25% less time than DBA to become stable. Moreover, different traffic scenarios simulated also show the robustness of the proposed scheme. Thus, the WATC scheme can be regarded as a promising candidate for the OBS rings.

1.2 Thesis Organization

The remainder of this thesis is organized as follows. In Chapter 2, we introduce the OBS ring network topology, node architecture, scheduler, reservation mechanism, and fairness reference model. In Chapter 3, we describe the proposed wavelength assignment and traffic control in WATC for the OBS ring network. Chapter 4 shows the simulation results and discussions. Finally, the conclusions are given in Chapter 5.
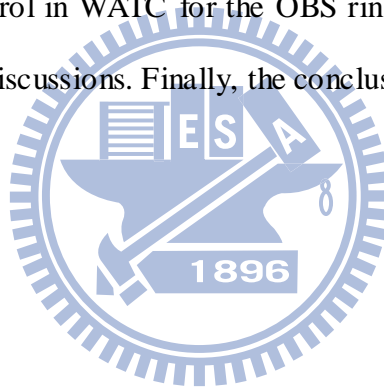
Chapter 2
System Model

An optical burst switching (OBS) ring contains two unidirectional ringlets and N nodes arranged in the clockwise manner as shown in Fig. 2.1. For simplicity, assume that these N nodes are uniformly distributed on the OBS ring. An OBS node is directly connected with two neighboring nodes, that is, an upstream neighbor and a downstream neighbor. For example, node i is directly connected with node i-1 and i+1 through fiber links. Each node is an ingress node which deals with the packets from the access networks. Here, we consider four service classes: voice, denoted by Vo, video, denoted by Vi, hyper-text transfer protocol (HTTP), denoted by H, and file transfer protocol (FTP), denoted by F. The packet size of the Vo service is fixed with 70 bytes, and each packet size of the other services is uniformly distributed with an variable mean, which ranges from 64 to 1518 bytes.
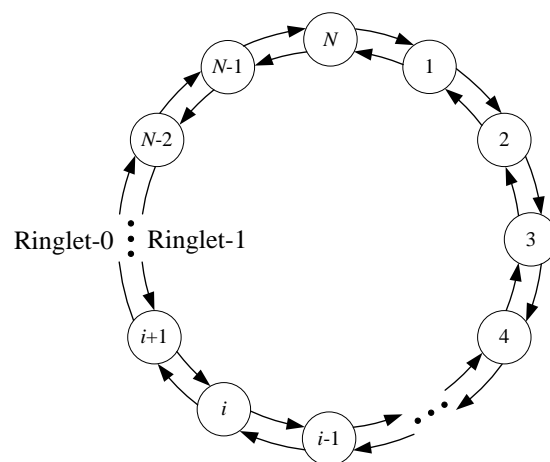
Figure 2.1: The topology of an OBS ring

2.1 Slotted OBS Rings

On each ringlet, there are totally (W+1) wavelengths: W data wavelengths and one control wavelength. As shown in Fig 2.2, all the wavelengths are divided into frames, a frame is composed of S slots, and a slot size is 2000-bit long. The data wavelength, denoted by λ_i, $1 \leq i \leq W$, is used for transmitting DBs. The control wavelength, denoted by λ_c, contains wavelength reservations (WRs) rotating around the ring for bandwidth reservation.

The WR comprises of four kinds of messages for the next frame: total empty slots, denoted by E, the advertised fair rate, denoted by F_a, the status of the ith
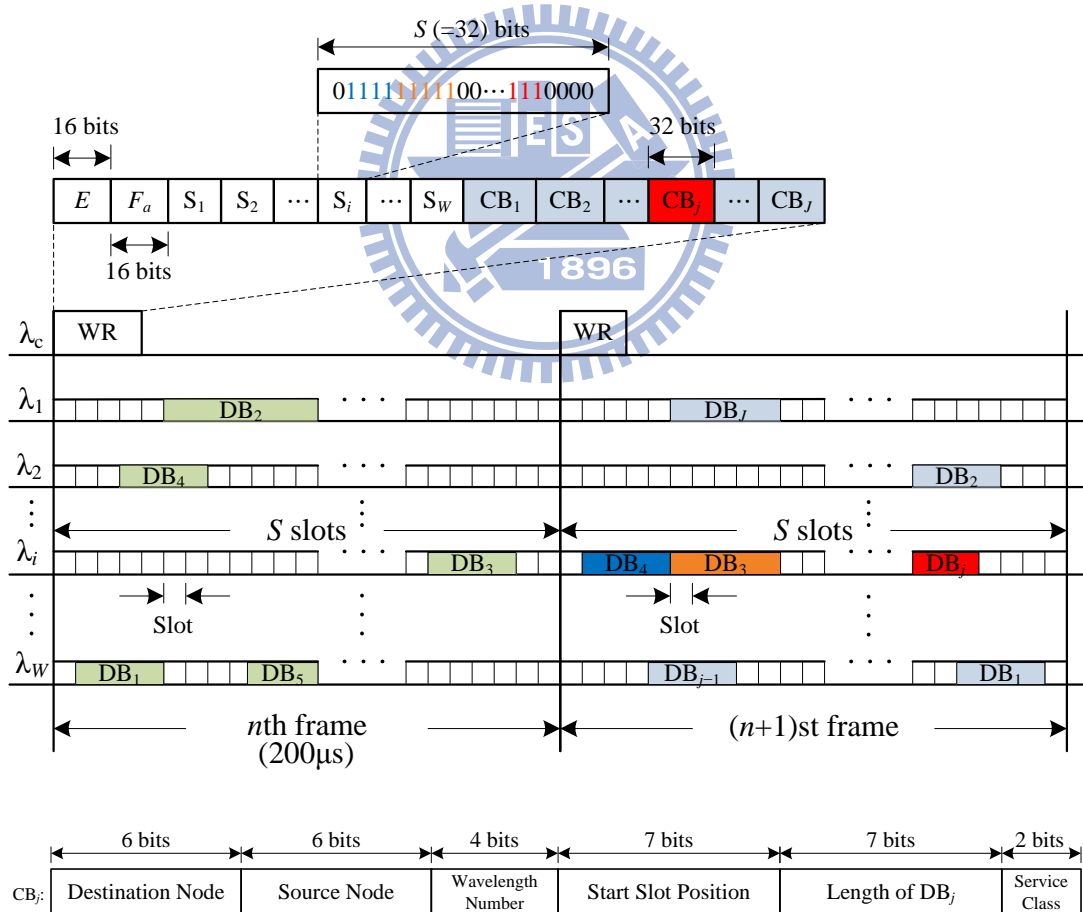
Figure 2.2: Frame structure in slotted OBS rings

wavelength, denoted by S_i, $1 \leq i \leq W$, and the CB_j relating to the DB_j, $1 \leq j \leq J$, where J is the number of the transit DBs in the next frame. The length of E and F_a are both 16 bits. Note that F_a is used to avoid the congestion and to achieve the fairness between each node by limiting the amount of the ingress traffic into ringlet, and it is transmitted to upstream nodes. The status of the ith wavelength S_i contains S bits, and each bit represents the status of a corresponding slot in λ_i at the next frame. An empty slot is marked as bit "0" and an occupied slot is marked as bit "1." The CB_j comprises information such as destination node, source node, wavelength number, start slot position, length of DB_j in unit of slots, and service class, $1 \leq j \leq J$. For simplicity, we assume that there is no synchronization problem among the wavelengths in every node throughout the thesis.

2.2 Node Architecture

An OBS node, shown in Fig. 2.3, contains two optical add/drop multiplexers (OADMs), four optical-to-electronic (O/E) converters, four electronic-to-optical (E/O) converters, a fiber delay line (FDL), $N{-}1$ ingress buffers, some egress buffers, a central processor (CP) with wavelength assignment and traffic control (WATC), and a scheduler. An OADM is used to add traffic to the fiber links at the transmitter or drop traffic from the fiber links at the receiver. Thus, this OBS node can be viewed as that TT-TR architecture is deployed. For simplicity, assume that the wavelength transition time can be ignored. An O/E converts the traffic data from the optical domain to electronic domain, and an E/O converts the traffic data from the electronic domain to optical domain. The FDL is used to delay the transit traffic on data wavelengths to synchronize with the control wavelength with the delay duration about 20~30μs. Ingress buffers and egress buffers are used to temporarily hold the packets for later operations. The WATC helps CP determine how to reserve the empty slots for the
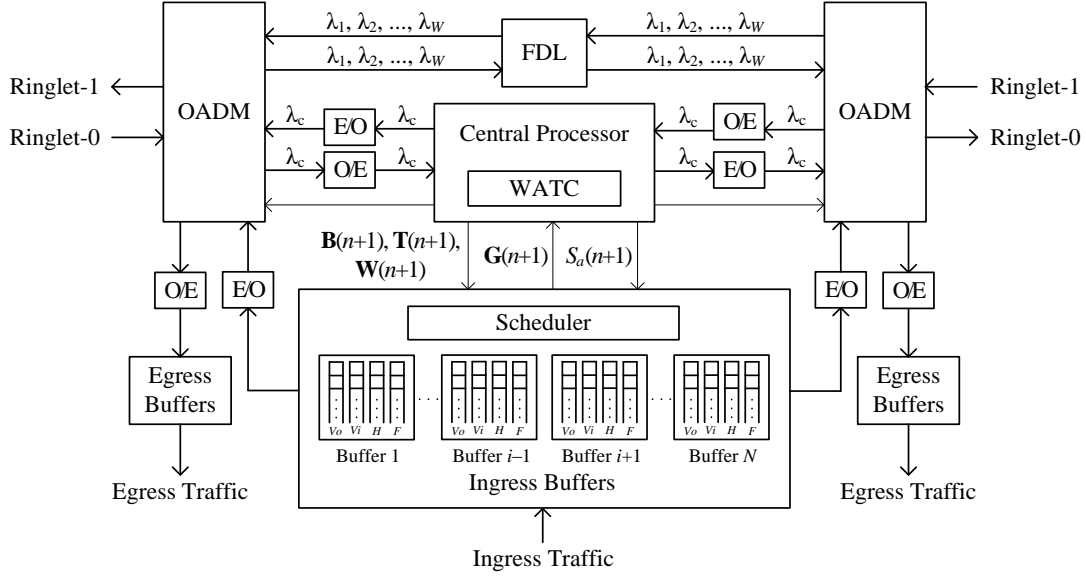
Figure 2.3: Architecture of node i at the nth frame

ingress traffic and to advertise the upstream nodes of traffic limitation in the next frame.

Each node adds its ingress traffic to the ringlet and drops down egress traffic from the ringlet. As for the transit traffic, the node simply bypasses it to next node. The ingress packets are first patched into ingress buffers according to their destinations and are stored in the different queue based on their service classes. Each ingress buffer contains Vo, Vi, H, and F queues. The packets in the same ingress buffer would be assembled into several DBs based on the result of the wavelength assignment by WATC. When the transit DB arrives at its destination, it is disassembled into original packets, which are stored in the egress buffers. Note that a traffic flow means the traffic aggregating in the same source node and destining to the same destination node. Each node deals with traffic on the two ringlets in the same way separately except that the advertised fair rate is sent via the other ringlet. For simplicity, we only illustrate the operations on one ringlet throughout this thesis.

2.3 Scheduler

The schedule is used to decide the allowance bandwidth of each queue and its operation is as follows. For example, at the nth frame, the scheduler receives the total available slots for the $(n+1)$st frame, $S_a(n+1)$, which is calculated by CP. This value is the maximum number of slots the scheduler can grant. Based on the weighted round robin method [27], the scheduler determines the granted bandwidth of each queue in unit slots in ingress buffers. If no HTTP packet, which is at the head of a queue, exceeds its maximum tolerance delay time, denoted by T_d, the weights for the four traffic types are defined as

$$Vo : Vi : H : F = \alpha : \beta : \gamma : \delta, \ \alpha > \beta > \gamma > \delta > 0. \qquad (2.1)$$

Otherwise, the weight of the H queue is set to be β. Denote the weight of queue j at the nth frame by $WE_j(n)$. The granted bandwidth of queue j in unit slots at the $(n+1)$st frame is obtained by

$$g_j(n+1) = \min\left\{ q_j(n), \left\lfloor \frac{WE_j(n)}{TW} \cdot S_a(n+1) \right\rfloor \right\}, \ j = 1, \ 2, \ \ldots, \ 4(N-1), \qquad (2.2)$$

where $q_j(n)$ is the length of queue j at the nth frame; $\lfloor \cdot \rfloor$ represents the floor function, and the TW is the sum of the total queue weights, $TW = \sum_{k=1}^{4(N-1)} WE_k(n)$.

The granted bandwidth of all queues is arranged in order of their service classes. For example, $g_i(n+1)$, $1 \leq i \leq N-1$, represents the granted bandwidth of Vo queue of node i; $g_{i+(N-1)}(n+1)$ is for Vi queue of node i; $g_{i+2(N-1)}(n+1)$ is for H queue of node i; $g_{i+3(N-1)}(n+1)$ is for F queue of node i. The scheduler generates the granted bandwidth vector, which is defined as

$$\mathbf{G}(n+1) = \left[g_1(n+1), \ g_2(n+1), \ \dots \ g_{4(N-1)}(n+1) \right], \ g_i(n+1) \le S, \ i = 1, \ 2, \dots, 4(N-1). \ (2.3)$$

This vector is sent to WATC. The WATC will choose appropriate burst sizes to assemble the packets in each queue according to its granted bandwidth. The collection of the length of the assembled DBs is presented by a vector $\mathbf{B}(n+1)$ similar as $\mathbf{G}(n+1)$, the start time of DBs is presented by a vector $\mathbf{T}(n+1)$, and the located wavelengths of DBs is presented by a vector $\mathbf{W}(n+1)$. Finally, the WATC marks the status bits of the nth frame in the WR_n by changing related 0's to 1's.

2.4 Beforehand Reservation

Whenever node i has packets to transmit, it must reserve appropriate slots in advance and then initiate transmissions at the next frame time. Since this is a one-way reservation, no acknowledgement is considered here. Besides, once the DBs are transmitted, their transmission positions cannot be changed by intermediate nodes. The mechanism of beforehand reservation is as follows. At the beginning of the nth frame of node i, WR_n^i is converted to electronic domain and forwarded to CP, shown in Fig. 2.4. After processing the information of WR_n^i, CP determines total available slots $S_a(n+1)$ and sends it to the scheduler. The scheduler would determine the granted bandwidth $\mathbf{G}(n+1)$ immediately and this result would be sent to WATC. Then, WATC chooses appropriate burst size $\mathbf{B}(n+1)$, start time $\mathbf{T}(n+1)$ of the DBs, and wavelengths $\mathbf{W}(n+1)$, and marks the status bits in the WR_n^i by changing related 0's to 1's. The timing sequence in an ingress node is shown in Fig. 2.5. Afterwards, CP discards original WR_n^i and generates $\text{WR}_n^{i\,'}$, which is sent to the next node. At the same time, the node initiates all the transmissions of DBs, denoted by $(\text{DBs})_n^i$, that have been reserved at the precious frame. The whole operations continue at every frame time. Note that, the advertised fair rate for current node is sent by the upstream node on the

other ringlet. In order to ensure that operations can work normally, the start time of the frames on ringlet-1 differ from those on ringlet-2 half a frame time.
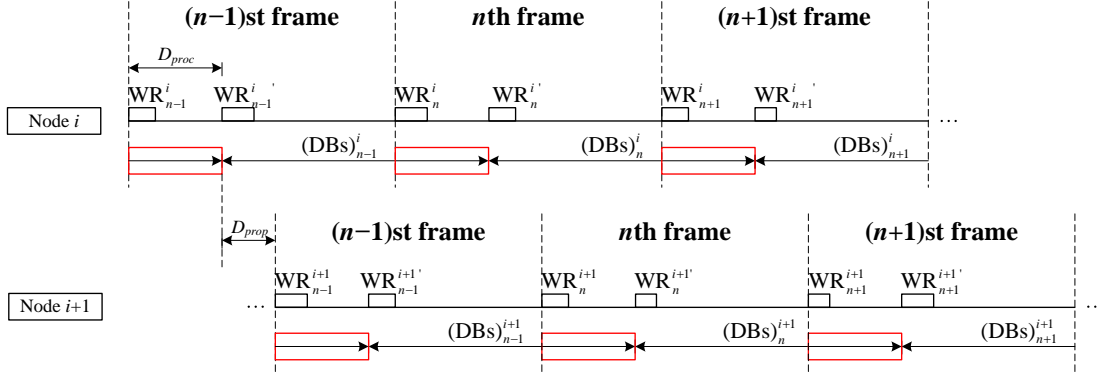
Figure 2.4: Operation of the reservation scheme
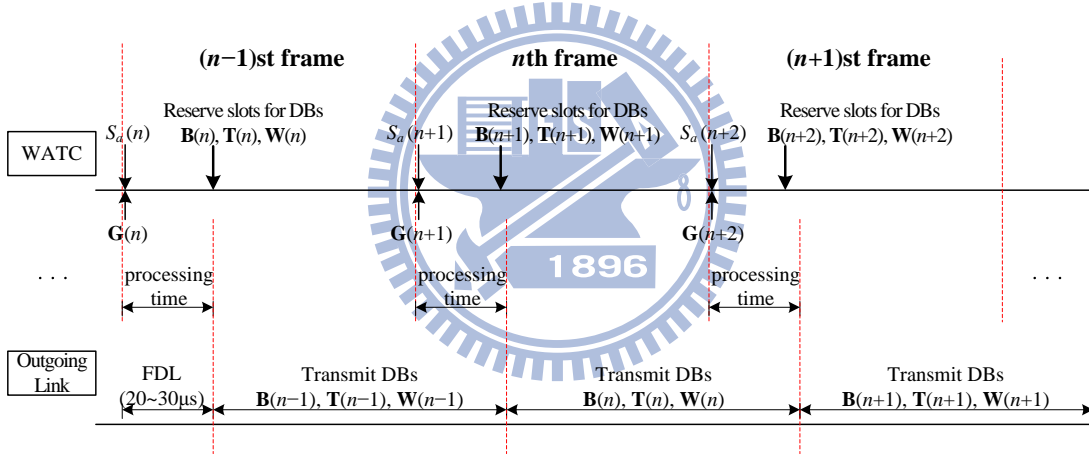
Figure 2.5: Timing sequence in an ingress node

2.5 Fairness Reference Model

In the OBS ring network, when the nodes transmit more data than the link capacity can sustain, an unfair bandwidth allocation problem occurs. The upstream nodes have more chance to transmit their data. In order to achieve fairness bandwidth allocation, some approaches have been proposed for Resilient Packet Ring networks [28], [29]. The concept is to generate an advertised fair rate at each node to regulate

transit traffic flows. In order to define the meaning of fairness, an appropriate fairness reference model is important. Here, we adopt the ring ingress aggregated with spatial reuse (RIAS) reference model [30], [31]. In this model, the available bandwidth in current link will be fairly allocated among all ingress aggregated (IA) flows, where IA flow represents the aggregation of all flows originating from a specific ingress node. Furthermore, RIAS ensures maximal spatial reuse. That is, the unused bandwidth can be reallocated to other IA flows while maintaining fairness among the IA flows.

Chapter 3

Wavelength Assignment and

Traffic Control (WATC)

3.1 Introduction

In this thesis, we propose an intelligent wavelength assignment and traffic control (WATC) scheme to determine the wavelengths for ingress traffic and adjust the traffic frame by frame. The WATC scheme consists of a wavelength assigner (WA) to schedule the ingress DBs, a local fair rate generator (LFRG) to generate a local fair rate, denoted by $F_l(n)$, which satisfies the RIAS fairness, and a traffic controller (TC) to decide an advertised fair rate, denoted by $F_a(n)$, shown in Fig. 3.1. The WA assigns not only the bandwidth to the queues according to the priority of the traffic but also the proper void to the granted bandwidths of queues as possible. Also, the LFRG would protectively reserve some bandwidth for future voice and video ingress traffic
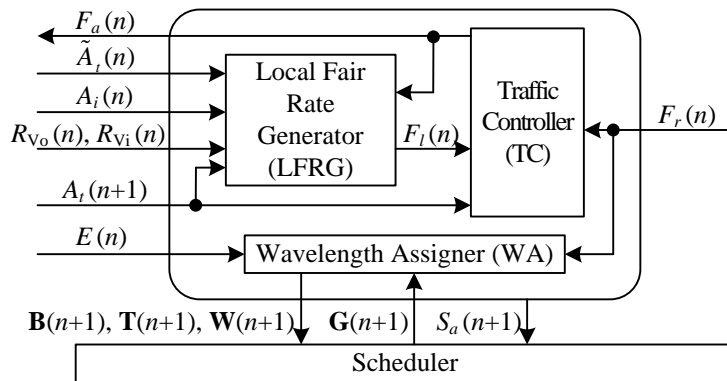

Figure 3.1: The structure of WATC

15

since there is no transit traffic buffer. Finally, the TC generates $F_a(n)$ to inform the upstream nodes of the traffic limitation for effectively utilizing the bandwidth and enhancing the fairness between nodes. The WATC scheme avoids the bandwidth starvation problem due to no transit traffic buffers and makes better use of link bandwidth in terms of throughput and fairness.

3.2 Wavelength Assigner (WA)

At each nth frame, the WA first computes the total available slots for the $(n+1)$st frame, $S_a(n+1)$, which could not be reserved at the next frame, at the beginning of nth frame. The $S_a(n+1)$ can be obtained by

$$S_a(n+1) = \min\{E(n+1), F_r(n)\}, \ 0 \le S_a(n+1) \le C, \tag{3.1}$$

where C is the link capacity in unit slots per frame, $E(n+1)$ is the total empty slots at the $(n+1)$st frame, and $F_r(n)$ is the fair rate received from the downstream nodes. Note that $F_r(n)$ is obtained from WR on the other ringlet. After the scheduler generates the granted bandwidth $\mathbf{G}(n+1)$ referring to $S_a(n+1)$, the WA will assign the slots to each queue according to its grated bandwidth. The procedure of wavelength assignment in the WA is shown in Fig. 3.2. For simplicity, the notation of frame time is not shown in the procedure.

For the value of the (remaining) granted bandwidth of queue j, denoted by M, the WA searches for possible voids, whose length is longer than $g_j(n+1)$, shown in Fig. 3.3. The results are divided into two cases. In case 1, shown in Fig. 3.4 (a), if at least one possible void exists, the WA chooses one of the smallest possible voids as the best position, denoted by (λ_b, t_b), for the queue j. As shown in Fig. 3.4 (b), the size of the ith assembled DB of queue j, denoted by $b_j^i(n+1)$, is set as M. Besides, the

wavelength assigned for this DB, denoted by $w_j^i(n+1)$, is set as the wavelength on which the best position located, and the start time of this DB, denoted by $t_j^i(n+1)$, is set as that of the best position.

```
for j ← 1 to 4(N−1)
    initialize i ← 1, M ← g_j
    while (M ≠ 0)
        search for possible voids
        if (exist a possible void)      // case 1
            choosed one of the smallest possible voids as the best position (λ_b, t_b)
            set b_j^i ← M, w_j^i ← λ_b, t_j^i ← t_b, M ← 0
        else                            //case 2
            choose the biggest void V with length l_V as the best position (λ_b, t_b)
            set b_j^i ← l_V, w_j^i ← λ_b, t_j^i ← t_b, i ← i+1, M ← M − l_V
        end if
    end while
    j ← j+1
end for
return B, W, T
```

Figure 3.2: Pseudocode of wavelength assignment
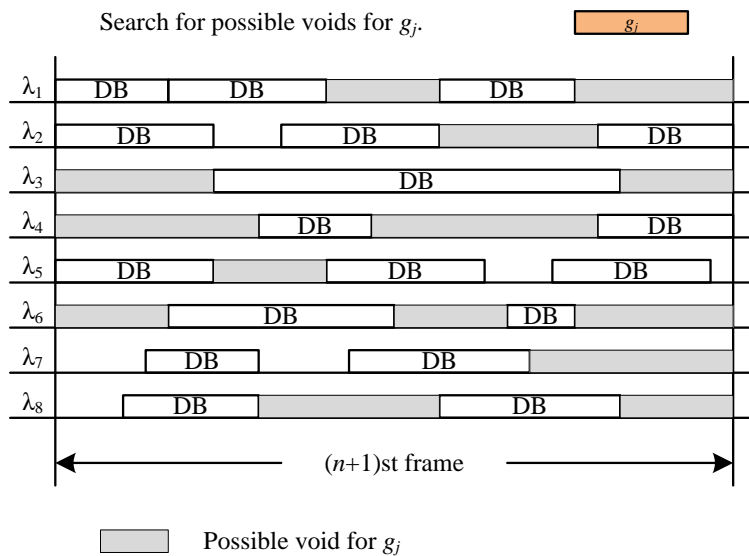
Figure 3.3: Searching for possible voids in WA

Case 1: exist a possible void for g_j [g_j]
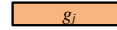
Choose the smallest ones from the possible voids.

Possible void for g_j

Smallest possible void for g_j
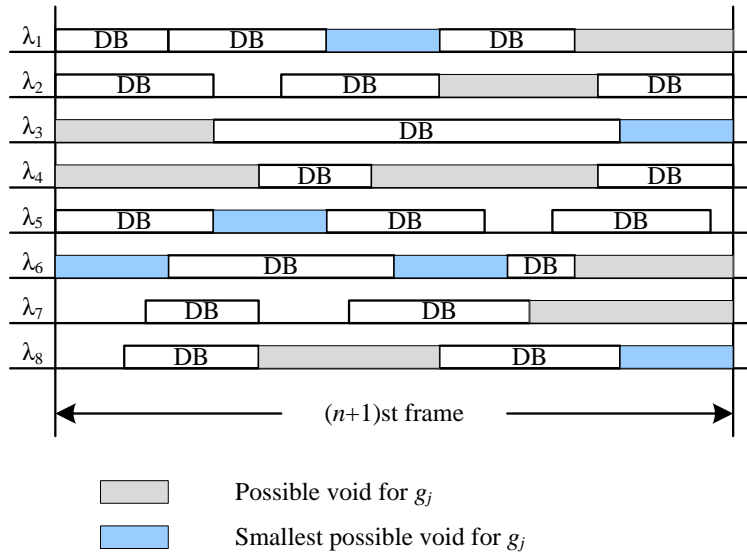
(a) Choosing the smallest ones from possible voids

Regard one of smallest possible voids as the best position.

best position for g_j: λ_b [g_j]

Set $b_j^i = g_j$, $w_j^i = \lambda_b$, $t_j^i = t_b$ t_b
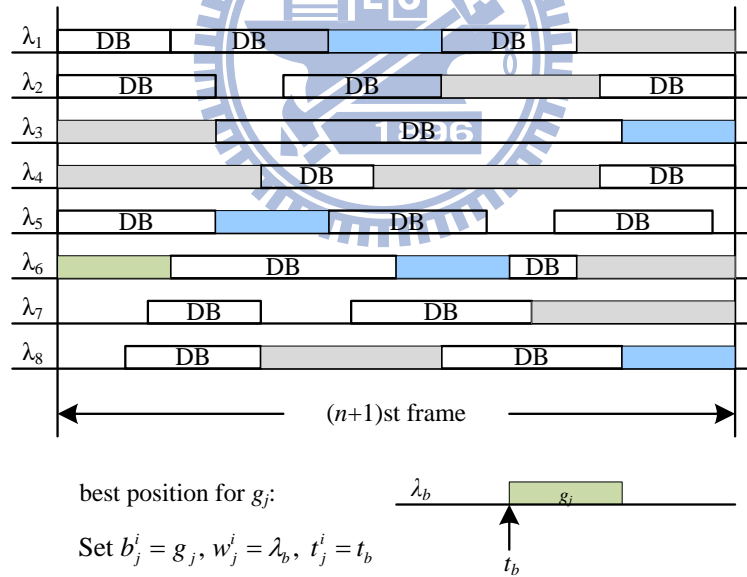
(b) Regarding one of the smallest possible voids as the best position

Figure 3.4: Case 1 of WA: (a) choosing the smallest ones from possible voids (b) regarding one of the smallest possible voids as the best position

In case 2 as shown in Fig. 3.5 (a), since no possible void exists, it is impossible to generate only one DB for transmitting packets of queue j. Thus, the WA will assign two or more DBs for queue j. First, the WA will choose the biggest void, denoted by V, with length l_v as the best position for the first assembled DB of queue j. and sets $b_j^i(n+1)$, $w_j^i(n+1)$, and $t_j^i(n+1)$ in the same way as in case 1. In Fig. 3.5 (b), the rest of the granted bandwidth also follows the previous procedure of wavelength assignment and could be categorized into the two cases. The whole wavelength assignment procedure is repeated until each $g_j(n+1)$ is appropriate assigned.

The size of each DB, which will be transmitted at the $(n+1)$st frame, is defined as

$$\mathbf{B}(n+1) = \left[b_1(n+1),\ b_2(n+1),\ \ldots,\ b_j(n+1),\ \ldots,\ b_{4(N-1)}(n+1) \right], \tag{3.2}$$

$$b_j(n+1) = \left[b_j^1(n+1),\ b_j^2(n+1),\ \ldots,\ b_j^i(n+1),\ \ldots,\ b_j^{n_j(n+1)}(n+1) \right], \tag{3.3a}$$

$$0 < b_j^i(n+1) \leq g_j(n+1), \tag{3.3b}$$

and it is constrained by

$$\sum_{i=1}^{n_j(n+1)} b_j^i(n+1) = g_j(n+1), \tag{3.4}$$

where $b_j^i(n+1)$ is the size of ith DB for queue j, $n_j(n+1)$ is the number of DBs for queue j at the $(n+1)$st frame, $0 \leq n_j(n+1) \leq g_j(n+1)$. The start slot position relating to each DB, which will be transmitted at the $(n+1)$st frame, is obtained by

$$\mathbf{T}(n+1) = \left[t_1(n+1),\ t_2(n+1),\ \ldots,\ t_j(n+1),\ \ldots,\ t_{4(N-1)}(n+1) \right], \tag{3.5}$$

$$t_j(n+1) = \left[t_j^1(n+1),\ t_j^2(n+1),\ \ldots,\ t_j^i(n+1),\ \ldots,\ t_j^{n_j(n+1)}(n+1) \right], \tag{3.6a}$$

$$1 \leq t_j^i(n+1) \leq S, \tag{3.6b}$$

Case 2: No possible void for g_j
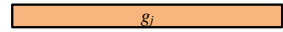
Regard it as the best position for ith DB for queue j.

$(n+1)$st frame

 The biggest unused void V

best position for b^i_j: λ_b [green bar l_V]

Set $b^i_j = l_V$, $w^i_j = \lambda_b$, $t^i_j = t_b$

t_b
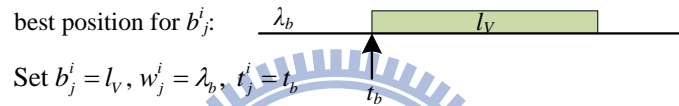
(a) Choosing the biggest unused void

Repeat the previous procedure of wavelength assignment until g_j is fully scheduled.

$(n+1)$st frame

best position for b^i_j: λ_b [green bar]

Set $b^i_j = g_j - l_V$, $w^i_j = \lambda_b$, $t^i_j = t_b$

t_b
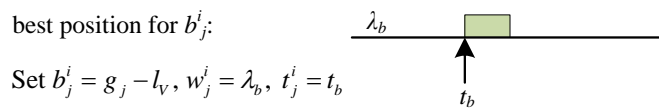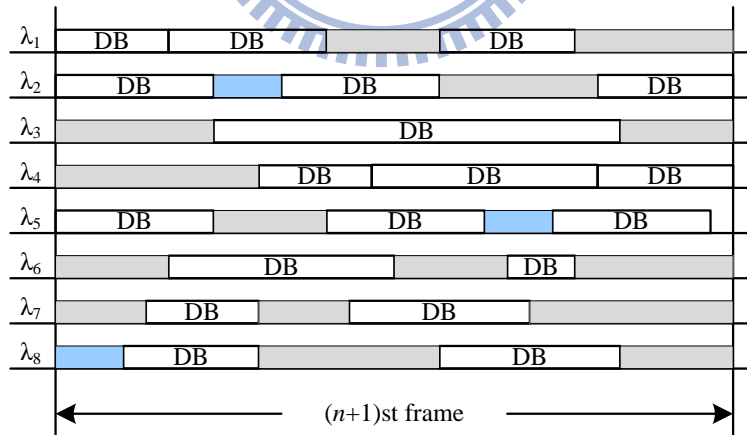
(b) Repeating the previous procedure of wavelength assignment

Figure 3.5: Case 2 of WA: (a) choosing the biggest unused void (b) repeating the previous procedure of wavelength assignment

20

where $t_j^i(n+1)$ is the start slot position of ith DB for queue j. The wavelength used by each DB, which will be transmitted at the $(n+1)$st frame, is obtained by

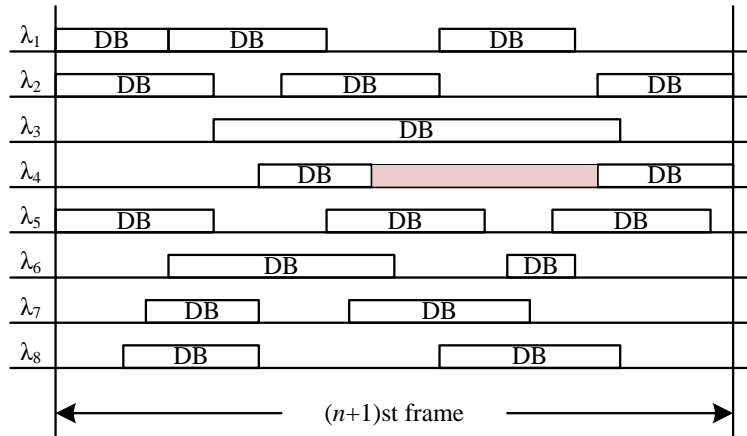$$\mathbf{W}(n+1) = \left[w_1(n+1),\ w_2(n+1),\ \ldots,\ w_j(n+1),\ \ldots,\ w_{4(N-1)}(n+1) \right], \qquad (3.7)$$

$$W_j(n+1) = \left[w_j^1(n+1),\ w_j^2(n+1),\ \ldots,\ w_j^i(n+1),\ \ldots,\ w_j^{n_j^{(n+1)}}(n+1) \right], \qquad (3.8a)$$

$$1 \leq w_j^i(n+1) \leq W, \qquad (3.8b)$$

where $w_j^i(n+1)$ is the wavelength used by ith DB for queue j.

Since the length of each void is an integer in the range from 0 to S, the counting sort could be used here, where S is the maximum number of slots on a wavelength per frame. This sorting algorithm takes only $\theta(X)$ time at the expense of requiring extra memory, where X is the number of the voids. The binary search algorithm is used to search the possible voids, and it takes the only $O(\lg X)$ time. Therefore, the complexity of the total wavelength assignment algorithm is $O(X + N \lg X)$, which would be acceptable in the system.

3.3 Local Fair Rate Generator (LFRG)

In order to achieve RIAS fairness mentioned in chapter 2, a traffic control algorithm is proposed. This algorithm is performed frame by frame in each node. The concept is to judge the situations of the link and adjust the fair rate according to observed statistics. As shown in Fig. 3.6, the structure of LFRG contains two parts: fuzzy protective reservation generator (FPRG) and fair rate generator (FRG).
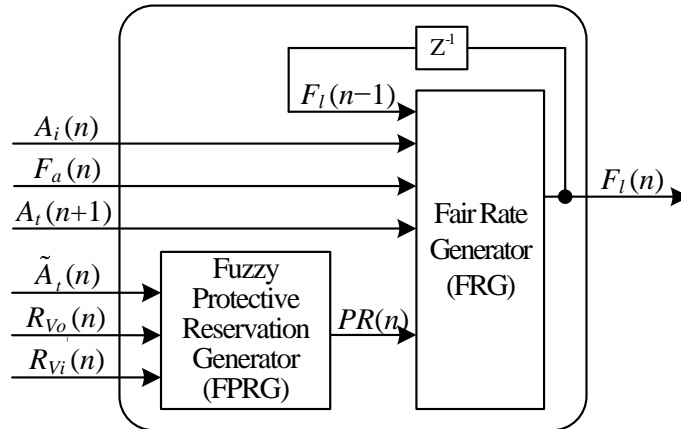
Figure 3.6: The structure of LFRG

3.3.1 Fuzzy Logic Control System (FLCS)

The typical structure of a fuzzy logic control system, shown in Fig. 3.7, comprises of four principal components: a fuzzifier, a fuzzy rule base, an inference engine, and a defuzzifier. The fuzzifier transforms crisp measured input variables $\mathbf{X}$ into suitable fuzzy linguistic terms. These fuzzy linguistic terms are specified by membership functions $\mu(\mathbf{X})$ and defined in a term set $T(\mathbf{X})$. The fuzzy rule base stores the empirical knowledge, which is described by a collection of fuzzy control rules (e.g., IF-THEN rules) involving linguistic variables. These rules describe the relationship between the input variable $\mathbf{X}$ and the control action $\mathbf{Y}$. The inference engine is the kernel of a FLC. It simulates human decision-making by performing an inference method to yield a desired control, which is described as fuzzy linguistic terms. These fuzzy linguistic terms are specified by membership functions $\mu(\mathbf{Y})$ and defined in a term set $T(\mathbf{Y})$. The defuzzifier is used to transform the inferred fuzzy control action to a non-fuzzy control action $\mathbf{Y}$.
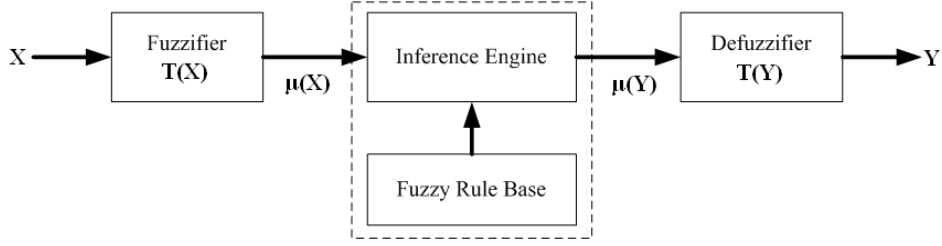
Figure 3.7: The basic structure of a FLC

3.3.2 Fuzzy Protective Reservation Generator (FPRG)

In the considered network, the transit traffic will bypass current node directly without any O/E and E/O conversions. Therefore, once the frame is fully used by upstream nodes, current node would hardly have bandwidth to transmit its own data, even if the service class of the data is more important than those of the transit traffic. Although this problem can be alleviated by the advertised fair rate, it takes much propagation time to see the effect. In order to further alleviate the mentioned problem, we protectively reserve some bandwidth, denoted by $PR(n)$, in advance for future voice and video traffic in ingress nodes.

Here, a fuzzy inference system is used to determine the protective reservation. FPRG has three inputs: the moving average of transit traffic, denoted by $\tilde{A}_t(n)$, the ratio of moving average of voice transit traffic to moving average of total transit traffic at the nth frame, denoted by $R_{Vo}(n)$, and the ratio of moving average of video transit traffic to moving average of total transit traffic at the nth frame, denoted by $R_{Vi}(n)$. $\tilde{A}_t(n)$ is obtained by

$$\tilde{A}_t(n) = \frac{1}{L} \sum_{k=n-L+1}^{n} A_t(k), \tag{3.9}$$

where $A_t(n)$ is the arrival rate of transit traffic in unit slots at the nth frame, L is the

observation window. In other words, $\tilde{A}_t(n)$ is a moving average of $A_t(n)$. $R_{Vo}(n)$ and $R_{Vi}(n)$ is obtained by

$$R_{Vo}(n) = \frac{\tilde{A}_t^{Vo}(n)}{\tilde{A}_t(n)}, \quad 0 \le R_{Vo}(n) \le 1, \tag{3.10}$$

$$R_{Vi}(n) = \frac{\tilde{A}_t^{Vi}(n)}{\tilde{A}_t(n)}, \quad 0 \le R_{Vi}(n) \le 1, \tag{3.11}$$

respectively, where $A_t^{Vo}(n)$ is the arrival rate of voice transit traffic in unit slots at the nth frame, $A_t^{Vi}(n)$ is the arrival rate of video transit traffic in unit slots at the nth frame. The output of FPRG is $PR(n)$, which is the protective reservation in unit slots at the nth frame for future voice and video traffic in the ingress node.

We define the term set for $\tilde{A}_t(n)$ as $T(\tilde{A}_t(n)) = \{$Low (L), Medium (M), High $(H)\}$; for $R_{Vo}(n)$ as $T(R_{Vo}(n)) = \{$Low (L), High $(H)\}$; for $R_{Vi}(n)$ as $T(R_{Vi}(n)) = \{$Low (L), Medium (M), High $(H)\}$; for $R_{Vi}(n)$ as $T(R_{Vi}(n)) = \{$Low (L), Medium (M), High $(H)\}$.

We use the triangular function $f(x; x_0, a_0, a_1)$ and the trapezoidal function $g(x; x_0, x_1, a_0, a_1)$ to define the membership functions for terms in the term set. These two functions are given by

$$f(x; x_0, a_0, a_1) = \begin{cases} \dfrac{x - x_0}{a_0} + 1, & \text{if } x_0 - a_0 < x \le x_0, \\[2mm] \dfrac{x_0 - x}{a_1} + 1, & \text{if } x_0 < x < x_0 + a_1, \\[2mm] 0, & \text{otherwise,} \end{cases} \tag{3.12}$$

and

$$g(x; x_0, x_1, a_0, a_1) = \begin{cases} \dfrac{x - x_0}{a_0} + 1, & \text{if } x_0 - a_0 < x \leq x_0, \\[2mm] 1, & \text{if } x_0 < x \leq x_1, \\[2mm] \dfrac{x_1 - x}{a_1} + 1, & \text{if } x_1 < x < x_1 + a_1, \\[2mm] 0, & \text{otherwise,} \end{cases} \tag{3.13}$$

where x_0 in $f(\cdot)$ is the center of the triangular function; x_0 (x_1) in $g(\cdot)$ is the left (right) edge of the trapezoidal function; a_0 (a_0) is the left (right) width of the triangular or the trapezoidal function. The center, edge, or width of the triangular or trapezoidal membership function is set intuitively but based on the characteristics of the linguistic variables.

The corresponding membership functions of L, M, and H in $T(\tilde{A}_t(n))$ are denoted as

$$\mu_L(\tilde{A}_t(n)) = g(\tilde{A}_t(n); 0,\ 0.45C,\ 0,\ 0.2C), \tag{3.14}$$

$$\mu_M(\tilde{A}_t(n)) = f(\tilde{A}_t(n); 0.65C,\ 0.2C,\ 0.2C), \tag{3.15}$$

$$\mu_H(\tilde{A}_t(n)) = g(\tilde{A}_t(n); 0.85C,\ C,\ 0.2C,\ 0), \tag{3.16}$$

where C is the channel capacity in unit slots. The corresponding membership functions of L and H in $T(R_{Vo}(n))$ are denoted as

$$\mu_L(R_{Vo}(n)) = g(R_{Vo}(n); 0,\ 0.3,\ 0,\ 0.2), \tag{3.17}$$

$$\mu_H(R_{Vo}(n)) = g(R_{Vo}(n); 0.5,\ 1,\ 0.2,\ 0). \tag{3.18}$$

The corresponding membership functions of L, M, and H in $T(R_{Vi}(n))$ are denoted as

$$\mu_L(R_{Vi}(n)) = g(R_{Vi}(n); 0,\ 0.2,\ 0,\ 0.2), \tag{3.19}$$

$$\mu_M(R_{Vi}(n)) = g(R_{Vi}(n); 0.4,\ 0.5,\ 0.2,\ 0.2), \tag{3.20}$$

$$\mu_H(R_{Vi}(n)) = g(R_{Vi}(n); 0.7, 1, 0.2, 0). \tag{3.21}$$

For the reason of simplicity in computation of defuzzification, let the membership functions for Null (*Nu*), Very Low (*VL*), Low (*L*), Medium (*M*), High (*H*), Very High (*VH*) in $T(PR(n))$ be fuzzy singletons. Define these membership functions as

$$\mu_{Nu}(PR(n)) = f(PR(n); 0, 0, 0). \tag{3.22}$$

$$\mu_{VL}(PR(n)) = f(PR(n); 10, 0, 0), \tag{3.23}$$

$$\mu_{L}(PR(n)) = f(PR(n); 20, 0, 0), \tag{3.24}$$

$$\mu_{M}(PR(n)) = f(PR(n); 30, 0, 0), \tag{3.25}$$

$$\mu_{H}(PR(n)) = f(PR(n); 40, 0, 0), \tag{3.26}$$

$$\mu_{VH}(PR(n)) = f(PR(n); 50, 0, 0). \tag{3.27}$$
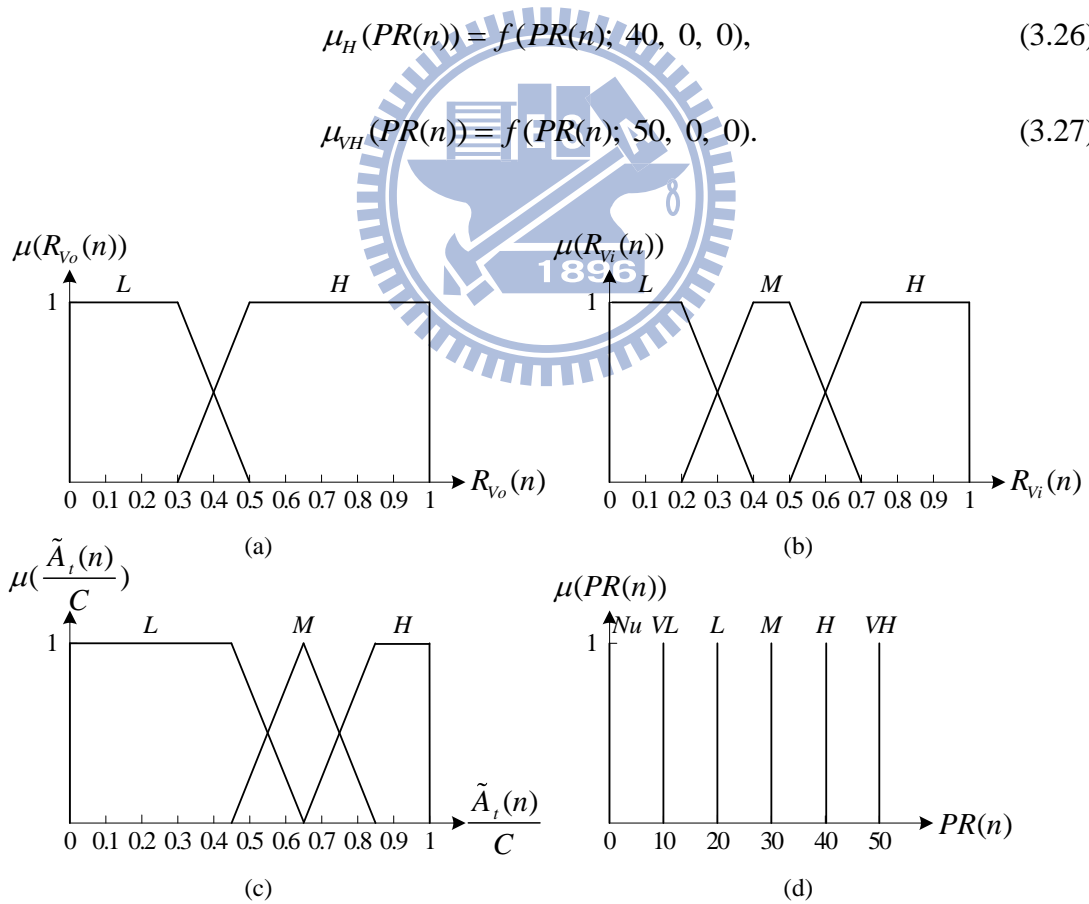

Figure 3.8: The membership functions of the term set (a) $T(R_{Vo}(n))$ (b) $T(R_{Vi}(n))$ (c) $T(\tilde{A}_t(n))$ (d) $T(PR(n))$

Rule	$\tilde{A}_t(n)$	$R_{Vo}(n)$	$R_{Vi}(n)$	$PR(n)$	Rule	$\tilde{A}_t(n)$	$R_{Vo}(n)$	$R_{Vi}(n)$	$PR(n)$
1	L	L	L	L	10	M	H	L	L
2	L	L	M	L	11	M	H	M	L
3	L	L	H	VL	12	M	H	H	VL
4	L	H	L	VL	13	H	L	L	VH
5	L	H	M	Nu	14	H	L	M	VH
6	L	H	H	Nu	15	H	L	H	H
7	M	L	L	H	16	H	H	L	H
8	M	L	M	M	17	H	H	M	M
9	M	L	H	M	18	H	H	H	M

Table 3.1: The rule base of FPRG

The membership functions of $T(PR(n))$ are discrete uniformly distributed between 0 and 50. All membership functions of FPRG are shown in Fig. 3.8.

The rule base of FPRG is shown in Table 3.1. The order of significance of the input linguistic is $\tilde{A}_t(n)$, $R_{Vo}(n)$, and then $R_{Vi}(n)$. When the arrival rate of transit traffic is high, the protective reservation would be set high to prevent the voice and video ingress traffic from bandwidth starvation. And if the important portion of the transit traffic, including voice and video traffic, is relatively low, the protective reservation would be set higher, too.

The FPRG adopts the max-min inference method for inference engine. To explain max-min inference method, we consider rule 5 and rule 6 which have the same control action "$PR(n)$ is Nu." Applying "min" operator, we obtain the membership function values of the control action "$PR(n)$ is Nu" of rule 5 and rule 6 respectively by

$$m_5(n) = \min\{\mu_L(\tilde{A}_t(n)), \mu_H(R_{Vo}(n)), \mu_M(R_{Vi}(n))\}, \qquad (3.28)$$

$$m_6(n) = \min\{\mu_L(\tilde{A}_t(n)), \mu_H(R_{Vo}(n)), \mu_H(R_{Vi}(n))\}. \qquad (3.29)$$

Then, applying the "max" operator, the overall membership value of the control action "$PR(n)$ is Nu", denoted as $w_{Nu}(n)$, is obtained by

$$w_{Nu}(n) = \max\{m_5(n), m_6(n)\}. \tag{3.30}$$

Similarly, the overall membership function of the control action VL, L, M, H, and VH, denoted as $w_{VL}(n)$, $w_L(n)$, $w_M(n)$, $w_H(n)$, and $w_{VH}(n)$, respectively, can be obtained. After inferring all the rules, FPRG uses center of area (COA) method for defuzzifier. The output value $PR(n)$ is obtained by

$$PR(n) = \frac{0 \times w_{Nu}(n) + 10 \times w_{VL}(n) + 20 \times w_L(n) + 30 \times w_M(n) + 40 \times w_H(n) + 50 \times w_{VH}(n)}{w_{Nu}(n) + w_{VL}(n) + w_L(n) + w_M(n) + w_H(n) + w_{VH}(n)}.$$

$$\tag{3.31}$$

3.3.3 Fair Rate Generator (FRG)

Here, we define a parameter, denoted by $M(n)$, to measure the equivalent number of IA transit flows traversing node i, whose rate is equal to previous advertised fair rate. This parameter is determined by

$$M(n) = \frac{A_i(n) + \tilde{A}_t(n)}{\tilde{F}_a(n-1)}, \tag{3.32}$$

where $A_i(n)$ is the arrival rate of ingress traffic in unit slots at the nth frame, $A_i(n) = \min(F_r(n), \sum_{j=1}^{4(N-1)} q_j(n))$., $q_j(n)$ is the length of queue j at the $(n+1)$st frame; $\tilde{F}_a(n-1)$ is the moving average of $F_a(n-1)$, which is the advertised fair rate generated at the $(n-1)$st frame, $\tilde{F}_a(n-1) = (1/L)\sum_{k=n-L}^{n-1} F_a(k)$. The local fair rate at the nth frame is calculated as

$$F_l(n) = \min\left\{C - PR(n),\ F_l(n-1) + \frac{1}{M(n)}(C - PR(n) - (A_i(n) + A_t(n+1)))\right\}, (3.33)$$

where $PR(n)$ is the protective reservation in unit slots at the nth frame for future voice and video traffic in ingress node.

3.4 Traffic Controller (TC)

The traffic condition in the fiber link is always changing. In order to alleviate the overuse of a link as well as make better use of the link capacity, TC is design to adapt the IA flows. In order to decide the traffic condition, TC observes the incoming transit traffic, say $A_t(n+1)$, and compares it with the minimum of the local fair rate $F_l(n)$ and the received advertised fair rate from the other ringlet, called $F_r(n)$. The main reason of using minimum operation is to be a little more conservative in order not to incur overuse of a link too often. Depending on the traffic conditions, TC takes two possible actions.

3.4.1 Traffic Suppression

If $A_t(n+1)$ is bigger than or equal to the minimum of $F_l(n)$ and $F_r(n)$, the link is considered as overused. So, TC will decrease the rates of IA flows, and choose the minimum of $F_l(n)$ and $F_r(n)$ as the advertised fair rate $F_a(n)$.

3.4.2 Traffic Promotion

If $A_t(n+1)$ is smaller than or equal to the minimum of $F_l(n)$ and $F_r(n)$, the link is considered as not sufficiently used. So, TC will increase the rates of IA flows, and choose the maximum of $F_l(n)$ and $F_r(n)$ as the advertised fair rate $F_a(n)$.

Chapter 4
Simulation Results

4.1 Simulation Environment

In the simulations, the general settings for environment include link capacity of 10Gbps, 7 data wavelengths, one control wavelength, 200μs propagation delay between nodes, 200μs frame time, and 1.6μs slot time. In terms of unit slot which is 2.5k bits, the link capacity per frame (C) is 875 slots and the number of slots per data wavelength in a frame (S) is 125. All the simulation parameters are listed in Table 4.1. The network traffic includes four service classes, each occupies one fourth of the total amount. Simulation results are recorded per frame time to show the bandwidth usage on an overused link.

The wavelength assignment of WATC scheme is compared to OBS-GS [21]. The OBS-GS groups the DBs in a collecting period, and assigns them proper positions to maximize the link utilization. It is assumed that the DBs scheduled by OBS-GS are generated using the volume-based burst assembly algorithm. For simplicity, the traffic scenario is assumed homogeneous, which means that each node generates equal amount of traffic, and the nodes are all greedy. On the other hand, the performance of the traffic control of the WATC scheme is compared to DBA [28]. The DBA estimates the equivalent number of IA flows referring to current arriving traffic and allocate the residual bandwidth equally to these IA flows.

Circumference of ring	200 km	320 km
Number of nodes (N)	5	8
Node architecture	TT-TR	
Link capacity	10 Gbps	
Light velocity	2×10^8 m/s	
Number of data channels (W)	7	
Number of control channels	1	
Frame time	200 μs	
Number of slots per frame per wavelength (S)	125	
Slot time	1.6 μs	
Link capacity per frame in unit slots (C)	875 slots	
Weights of four service classes: voice, video, HTTP, FTP	$\{\alpha, \beta, \gamma, \delta\}=\{4, 3, 2, 1\}$	

Table 4.1: General settings for environment

4.2 Comparisons of Wavelength Assignment Methods

In this section, simulations are conducted to compare the performances of the wavelength assignment of the WATC scheme with OBS-GS. Figure 4.1(a) shows a large parking lot scenario where there are eight greedy nodes. This scenario assumes that the flows are generated from node 1, 2, 3, 4, 5, 6, and 7 but terminated at node 8. The simulated traffic is homogeneous and composes of four service classes. Cooperating with the traffic control of the WATC scheme, the total throughputs of the two wavelength methods, denoted by WATC and OBS-GS_TC, are shown in Fig.

4.1(b). The throughputs recorded are in the outgoing link of node 7, and they are normalized to the link capacity. It can be seen that the throughput of the WATC scheme approximate the link capacity. However, the throughput of the OBS-GS_TC scheme only achieves 95% of the link capacity. This is due to the fact that OBS-GS only searches and utilizes the earliest possible voids. Whenever a granted queue has no possible void to put on its data, OBS-GS would leave the data behind and wait for reservation next time. Instead, the WA would divide the granted queue into two smaller pieces of data and assign each of them an approximate void, thus the WATC scheme has better performance.
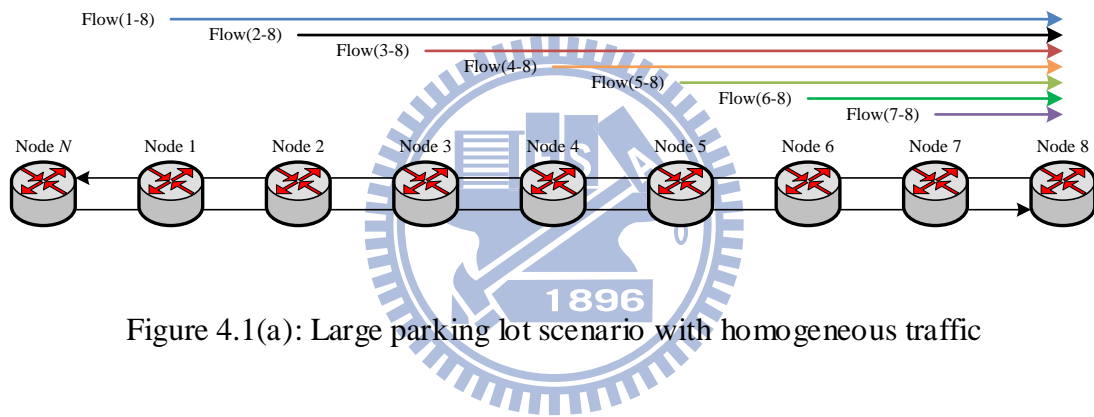

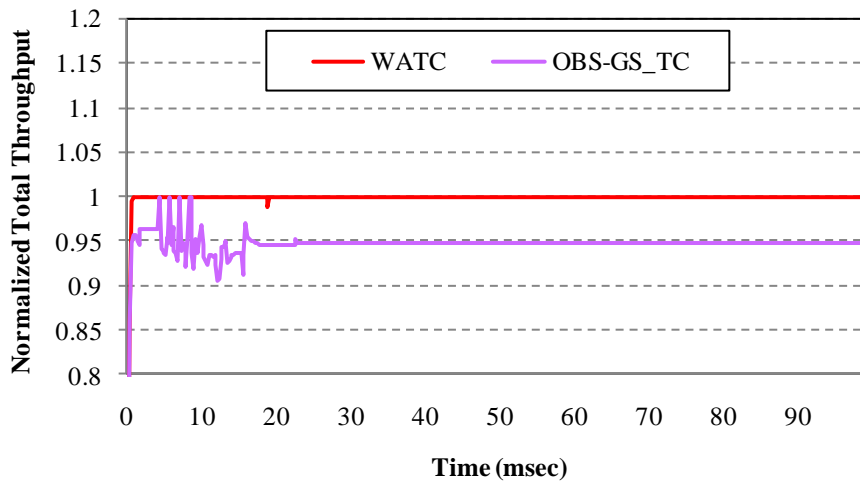Figure 4.1(a): Large parking lot scenario with homogeneous traffic

Figure 4.1(b): The normalized throughputs of WATC and OBS-GS_TC

Alternatively, the throughputs of the two wavelength methods cooperating with DBA, denoted by WA_DBA and OBS-GS_DBA, are shown in Fig. 4.1(c). It is found that the wavelength assignment of the WATC scheme also has high throughput. Unfortunately, the OBS-GS_DBA not only has lower throughput, but also suffers from throughput oscillation. This reveals the disadvantage of DBA, that is, the flows can hardly converge in many scenarios. The oscillation makes the available bandwidth time-varying. Besides, OBS-GS cannot fully utilize these bandwidth and further worse the oscillation. Therefore, the curve of the oscillation is not smooth. Obviously, the wavelength assignment of the WATC scheme is better than OBS-GS for OBS ring networks.
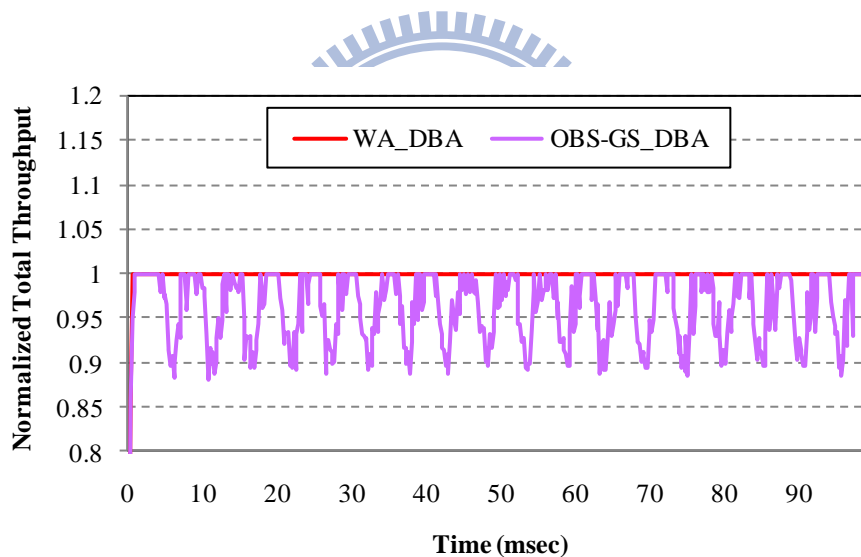

Figure 4.1(c): The normalized throughputs of WA_DBA and OBS-GS_DBA

4.3 Comparisons of Traffic Control Methods

4.3.1 Homogeneous Traffic Generated by Constant-bit-rate Traffic Pattern

Figure 4.2(a) shows a small parking lot scenario where there are five greedy

nodes. Figures 4.2(b) and 4.2(c) present the normalized throughput of each flow in the outgoing link of node 4 controlled by WATC and WA_DBA. This scenario assumes that flows are generated from node 1, 2, 3, and 4 but terminated at node 5. The propagation delay is small. The simulated traffic includes four service classes, but only the HTTP and FTP traffic are controlled to meet the fairness. The throughputs are normalized to the link capacity. These flows all start at 0 ms. As shown in Figs. 4.2(b) and 4.2(c), the flows of WATC and WA_DBA take 17 and 26.4 ms to become stable, respectively. The WATC improves the convergence time over the WA_DBA by 35.6%. It is due to the fact that the WA_DBA generates the fair rate to regulate the flows of upstream node referring to the short-term arriving traffic, but it takes some propagation time to see the actual influence of each adjustment of the advertised fair rates. This would incorrectly limit the amount of the transit traffic to the next node and cause the WA_DBA to make an error decision. As a result, the WA_DBA may adjust the flows too much, incurring a longer convergence time. On the contrary, the WATC calculates the moving average of arriving traffic within an observation window to mitigate the effect of propagation delay. This can avoid rapid flow adjustment and thus improve the convergence time.

In Fig. 4.2(b), it is found that the Flow(4-5) has slightly more throughput than the other flows. This is due to the fact that the WATC scheme applies the protective reservation, which allows the ingress node have more bandwidth for its voice and video traffic. The difference between the Flow(4-5) and the other traffic flows is only the throughput of voice and video traffic. The amount of the protective reservation is dynamically generated the FPRG according to the network conditions. It can also be observed that, at steady state, the throughputs of all flows are approximately the same as required, thus achieving fairness.
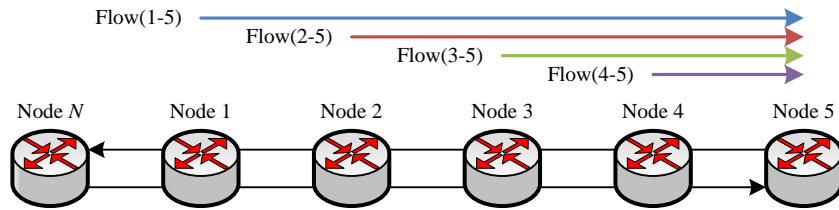
34

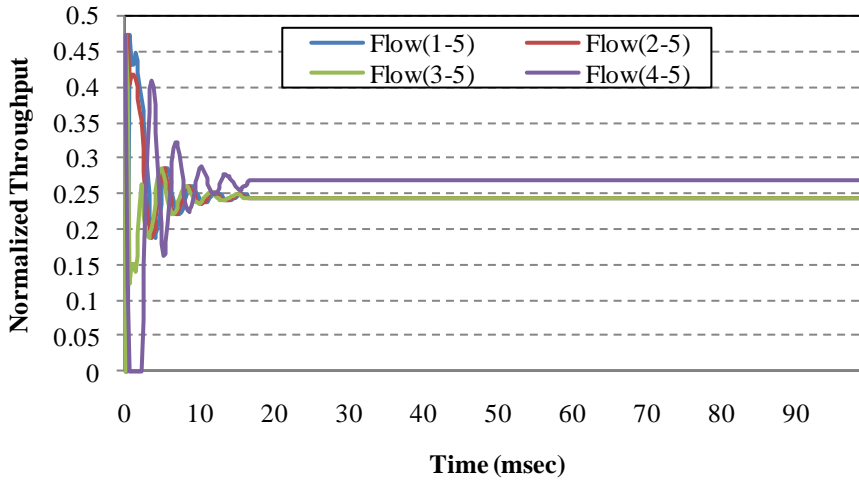Figure 4.2(a): Small parking lot scenario with homogeneous traffic

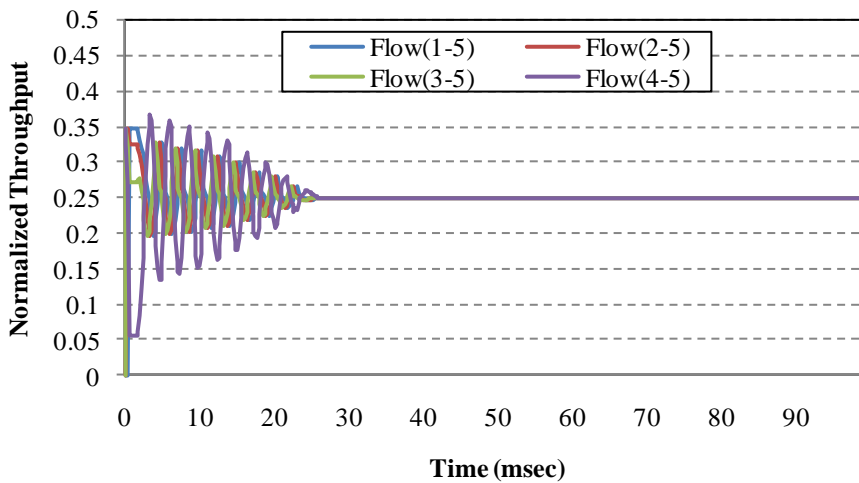Figure 4.2(b): The normalized throughputs of WATC

Figure 4.2(c): The normalized throughputs of WA_DBA

In Figs. 4.2(b) and 4.2(c), it is found that the oscillation of Flow(4-5) is different from the other flows. The Flow(4-5) increases when the other flows decreases and vice versa. The reason is that the node architecture does not have transit buffers, thus the node 4 can use only the residual bandwidth after node 1 to node 3 have reserved their bandwidth. Besides, since the overused link occurs at the outgoing link of node 4, all the flows from upstream nodes are regulated by the fair rate generated by node 4 and being changed in turn.

Figure 4.3(a) shows a large parking lot scenario where there are eight greedy nodes. Figures 4.3(b) and 4.3(c) present the normalized throughputs of Flow(1-8), Flow(3-8), Flow(5-8), and Flow(7-8) in the outgoing link of node 7 controlled by WATC and WA_DBA. This scenario assumes that flows are generated from node 1, 2, 3, 4, 5, 6, and 7 but terminated at node 8. Again, only the HTTP and FTP traffic are controlled to meet the fairness, and the throughputs are normalized to the link capacity. These flows all start at 0 ms. In Fig. 4.3(b), it can be seen that the flows of WATC take 21 ms to become stable. Unfortunately, it is hard for WA_DBA to make the flows become stable. This is due to the fact that the WA_DBA computes the number of effective IA flows referring to both the short-term traffic and the previous local fair rate to generate the current local fair rate. However, due to the large propagation delay, the correlation between the short-term traffic and the previous local fair rate becomes low. Therefore, the WA_DBA cannot generate a correct local fair rate to regulate the flows. As a result, the flows oscillate and barely converge.
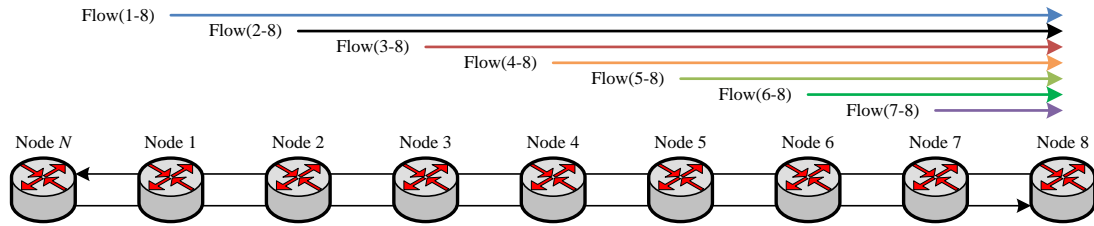
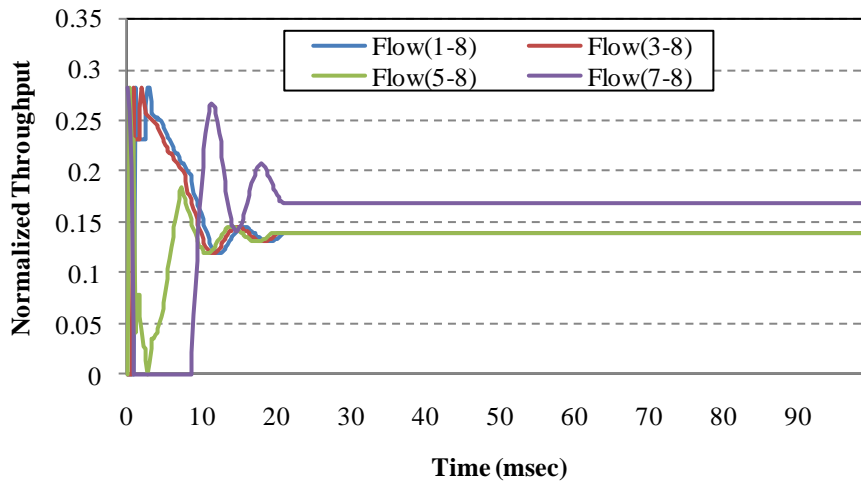Figure 4.3(a): Large parking lot scenario with homogeneous traffic

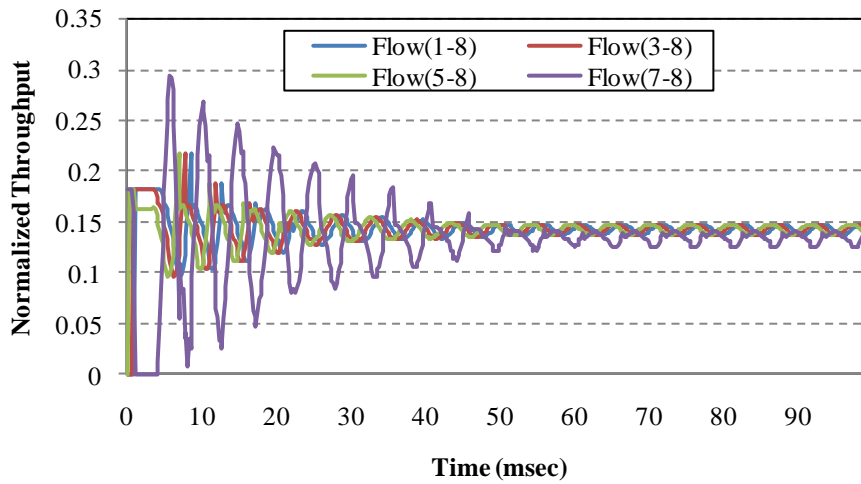Figure 4.3(b): The normalized throughputs of WATC

Figure 4.3(c): The normalized throughputs of WA_DBA

4.3.2 Homogeneous Traffic Generated by Constant-bit-rate Traffic Pattern in a More Realistic Scenario

In this section, simulations are conducted to see the performances of WATC and WA_DBA in a more complicated scenario, in which two overused links will be caused by the traffic flows. Figure 4.4(a) is the considered scenario where there are eight greedy nodes. This scenario assumes that flows are generated from node 2, 3, 4, 5, 6, and 7 but terminated at node 8. In addition, a flow is generated from node 1 but terminated at node 4. In order to keep the figure simple and clear, Figs. 4.4(b) and 4.4(c) only show the normalized throughputs of Flow(1-4), Flow(3-8), Flow(5-8), and Flow(7-8) controlled by WATC and WA_DBA, respectively.

Figure 4.4(b) shows the simulation result of the WATC scheme. These flows all start at 0 ms. The outgoing link of node 2 and node 7 are overused at 0.4 ms, and they send fair rates to upstream nodes to suppress the flows. As can be seen from Fig. 4.4(b), Flow(1-4) is suppressed at 0.6 ms and 1.6 ms, and it becomes stable at 6 ms. All the other flows, such as Flow(3-8), Flow(5-8), and Flow(7-8), reach stable at 15 ms. The flows are influenced both by the fair rates determined by node 2 and node 7. For example, when the Flow(2-8) is suppress by node 7, the transit traffic at node 4 is decreased. Since the utilization of the outgoing link of node 4 is decreased, an increased advertised fair rate is generated by node 4, which would be sent to increase the Flow(1-4) and Flow(2-8). This will in turn influences the transit traffic at node 7. Therefore, the curves of these flows are different from those mentioned in previous section. In Fig. 4.4(c), the flows except Flow(1-4) have great oscillations due to the propagation delay. Besides, it can be seen that Flow(7-8) suffers from bandwidth starvation periodically, which would be harmful to voice and video traffic. Hence, the performance of the WATC scheme is obviously better than the WA_DBA scheme.
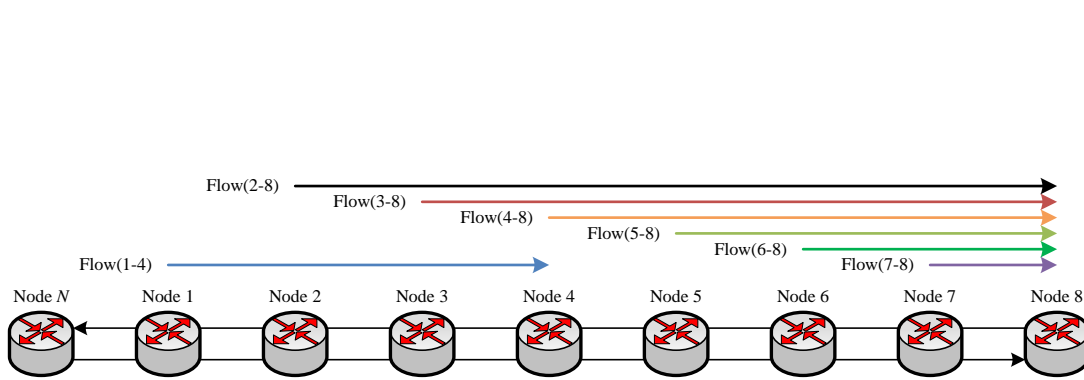
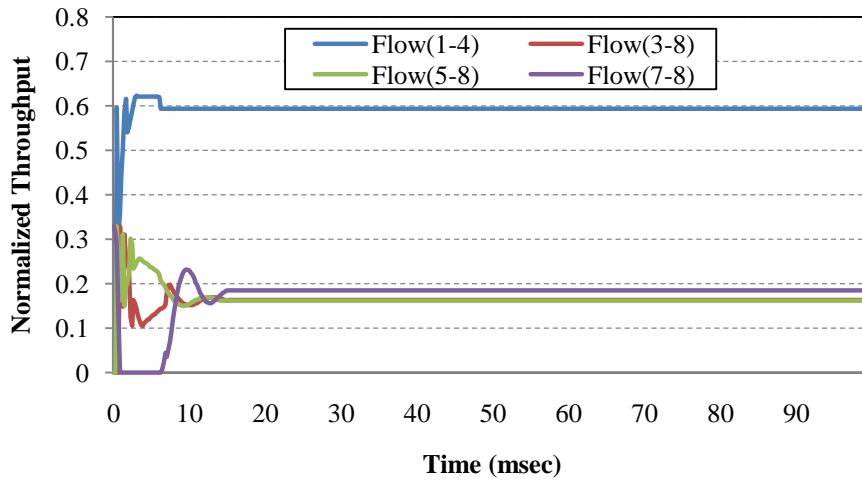Figure 4.4(a): A more complicated scenario with homogeneous traffic

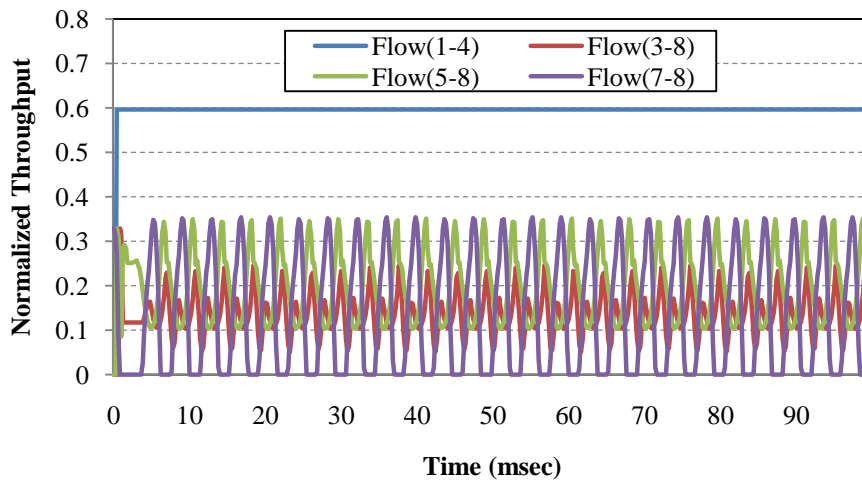Figure 4.4(b): The normalized throughputs of WATC

Figure 4.4(c): The normalized throughputs of WA_DBA

4.3.3 Homogeneous Traffic Generated by Exponentially Distributed Traffic Pattern

In contrast to the traffic patterns applied in previous sections, the traffic patterns of four service classes are not all constant-bit-rate here. Except that the voice traffic is constant-bit-rate, the traffic patterns of video, HTTP, and FTP traffic are exponential distributions. This probability distribution is used to simulate the uncertainty and bursty property of these three kinds of traffic. The considered scenarios are shown in Fig. 4.5(a) and Fig. 4.5(d), and the homogeneous traffic is simulated. The settings of the scenarios are the same as mentioned in section 4.3.1 except the traffic pattern. Figs. 4.5(b), 4.5(c), 4.5(e), and 4.5(f) present the normalized throughput of each flow in the overused link controlled by WATC and WA_DBA. These flows all start at 0 ms.
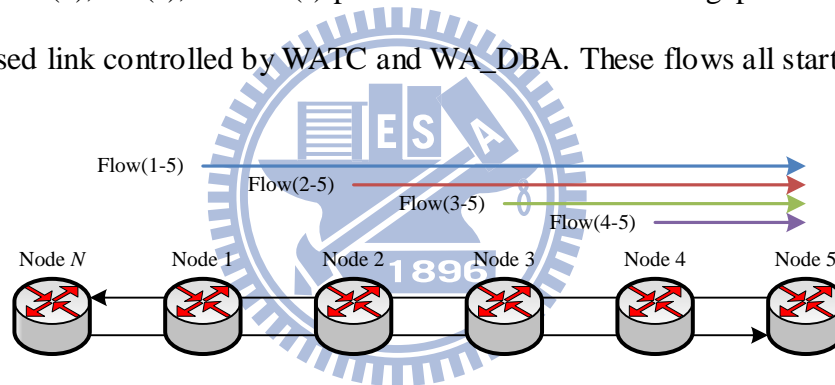

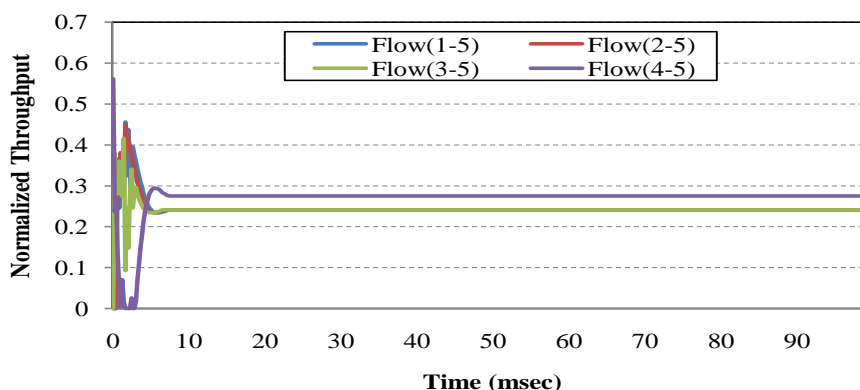Figure 4.5(a): Small parking lot scenario with homogeneous traffic

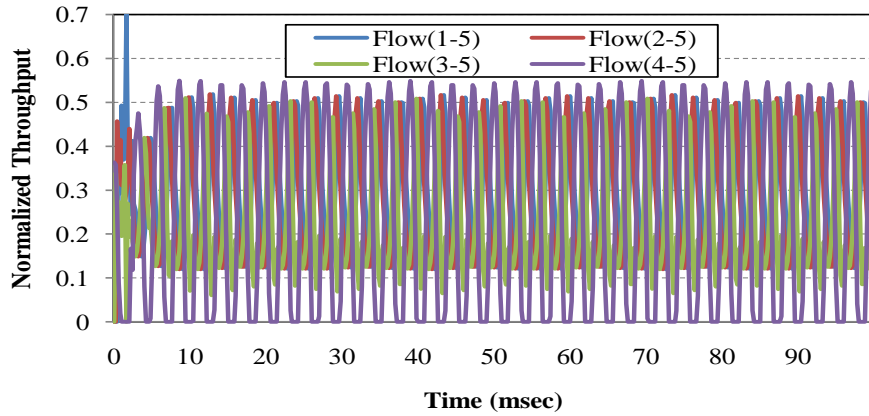Figure 4.5(b): The normalized throughputs of WATC

Figure 4.5(c): The normalized throughputs of WA_DBA

In Fig. 4.5(b), the WATC scheme takes only 7.4 ms to become stable. However, from the Fig. 4.5(c), it is observed that the WA_DBA scheme fails to achieve stability and has a great oscillation problem. That is due to the fact that WA_DBA always adjusts the fair rate only referring to the current arriving traffic. If the arrival rate is time variant instead of constant, the length of each queue is also time-varying. Thus, the observed traffic at each node may change dramatically, resulting in the great variation of generated fair rate. On the contrary, the WATC scheme applies the observation window to alleviate the traffic variation. Since the moving average of the arriving traffic is considered, the WATC scheme is more stable than the WA_DBA scheme. In Figs. 4.5(e) and 4.5(f), the influence of the time-varying traffic is more obvious in the larger scenario. Since the links are shared by all the nodes, the uncertainty property would be increased with the number of nodes in the scenario. Consequently, the downstream nodes are greatly influenced by the uncertainty of the traffic due to the fact that no transit buffer is deployed at each node. As shown in Fig. 4.5(f), the great oscillations cause that Flow(5-8) and Flow(7-8) suffer from bandwidth starvation time after time. On the contrary, the WATC scheme can efficiently mitigate the uncertainty and show its stability.
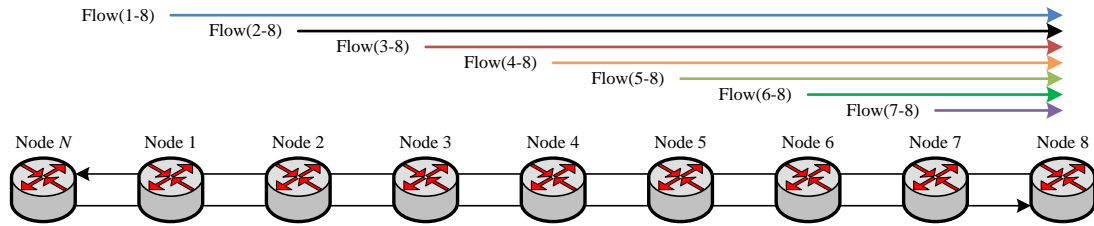
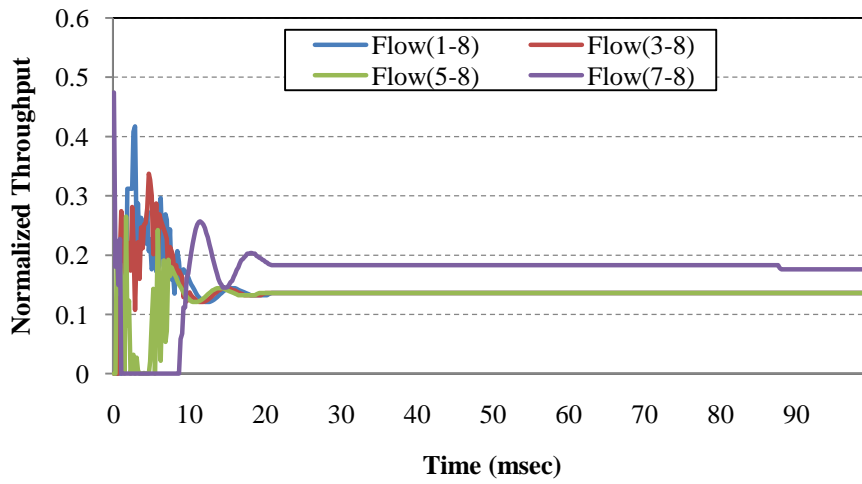Figure 4.5(d): Large parking lot scenario with homogeneous traffic

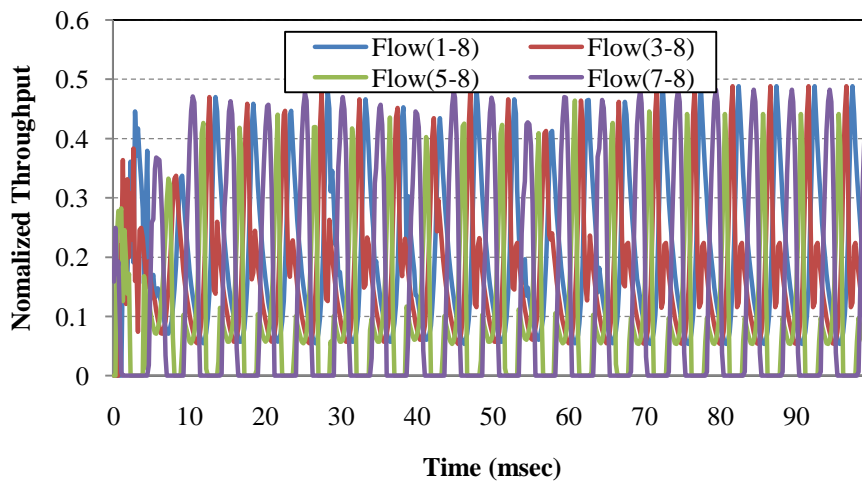Figure 4.5(e): The normalized throughputs of WATC

Figure 4.5(f): The normalized throughputs of WA_DBA

4.3.4 Non-homogeneous Traffic Generated by Constant-bit-rate Traffic Pattern

In this section, the non-homogeneous traffic is simulated to see the performances of WATC and WA_DBA. Different from the homogeneous traffic simulated in previous sections, the required data rates of the flows are not all the same here. Figure 4.6(a) is the considered scenario, which assumes that flows are generated from node 1, 2, 3, 4, 5, 6, and 7 but terminated at node 8. Node 1, 2, 4, 5, 6, and 7 require a mean rate of 0.1 multiplied by link capacity, but node 3 requires a mean rate of 1.2 multiplied by link capacity. In other words, node 3 has much more traffic than all the other flows. Figs. 4.6(b) and 4.6(c) show the normalized throughputs of Flow(1-8), Flow(3-8), Flow(5-8), and Flow(7-8) in the outgoing link of node 7 controlled by WATC and WA_DBA. These flows all start at 0 ms. Again, only the HTTP and FTP traffic are controlled to meet the fairness, and the throughputs are normalized to the link capacity.

Since only the node 7 transmits more traffic than the fair rate, the arrival rate of ingress traffic at node 7 should be suppressed, but the other flows should be maintained. In Fig. 4.6(b), the flows experience consecutive adjustments and take 15.8 ms to become stable and meet the requirements. The Flow(3-8) is suppressed to 0.337 multiplied by link capacity, and each of the others is about 0.093 multiplied by link capacity. There is a slightly drop of link utilization due to the insufficiently used protective reservation. In Fig. 4.6(c), WA_DBA cannot efficiently control the Flow(3-8), which causes the bandwidth starvation problem of downstream nodes. It reveals the oscillation problem of WA_DBA, and shows that the non-homogeneous traffic would worsen the problem. So, WATC has better stability than WA_DBA at the expense of slightly lower link utilization.
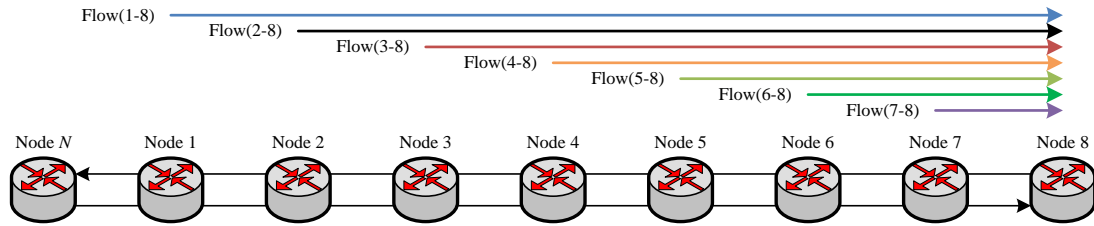
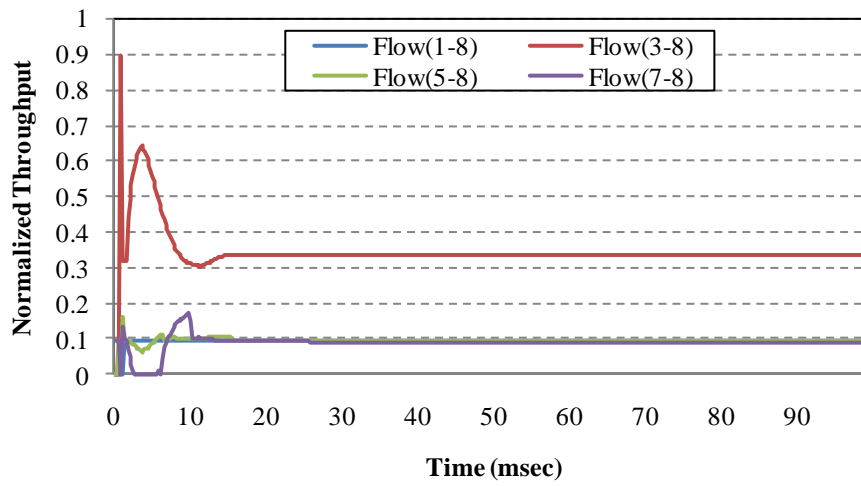Figure 4.6(a): Large parking lot scenario with non-homogeneous traffic

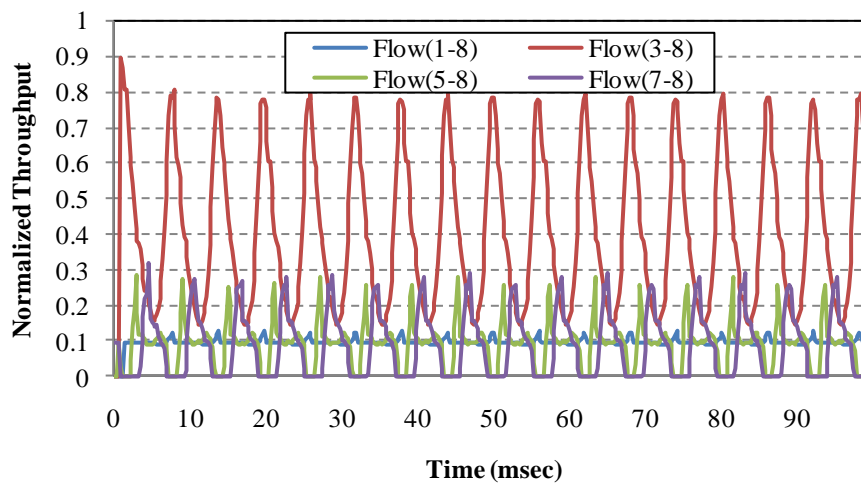Figure 4.6(b): The normalized throughputs of WATC

Figure 4.6(c): The normalized throughputs of WA_DBA

Chapter 5
Conclusions

In this thesis, the wavelength assignment and traffic control issues are studied. The WATC scheme is proposed to increase the link utilization, satisfy fairness criterion as RIAS fairness, make flows converges to the fair rate, and guarantee stability. The time-slot reservation mechanism is adopted to achieve a collision free network. In order to alleviate the bandwidth starvation problem in the transit bufferless environment, the concept of protectively reservation bandwidth is proposed. With regard to the uncertainty of network traffic, the WATC scheme implements fuzzy logic control system to generate the protective reservation.

The structure of WATC is composed of three function blocks, specifically, WA, LFRG, and TC. The WA assigns ingress traffic appropriate wavelength and position referring to the bandwidth granted by scheduler. The algorithm can allocate bandwidth according to the current voids, thus it can make maximum use of the link bandwidth. The LFRG generates an estimated fair rate by modifying DBA, thus it can alleviate large propagation delay. Moreover, it generates the protective reservation, allowing the ingress voice and video traffic has more bandwidth to use. The TC judges the current network condition and decides either to promote traffic or suppress traffic. The WATC scheme can smartly generate protective reservation against the changing traffic conditions.

Simulations are conducted to see the normalized throughputs of different

wavelength assignment methods. It is shown that the wavelength assignment of the WATC scheme has the better performance and stability than OBS-GS. On the other hand, the WATC and WA_DBA schemes are applied in small and large scale scenarios to see their performance on throughput, stability and convergence time. The WA_DBA scheme reaches stability only in small scale scenario due to the influence of propagation delay. In contrast, the WATC scheme has 25% smaller convergence time and better scalability than the WA_DBA. When non-homogeneous traffic scenarios are considered, simulation results shows that the WATC scheme still has good stability. Finally, it is concluded that the WATC scheme has good performance on fairness, convergence time, link utilization, and stability.

Bibliography

[1] B. Mukherjee, "WDM optical communication networks: progress and challenges," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 10, pp. 1810-1824, Oct. 2000.

[2] P. Green, "Progress in optical networking," *IEEE Communication Magazine*, vol. 39, no. 1, pp. 54-61, Jan. 2001.

[3] M. Maier and M. Reisslein, "Trends in optical switching techniques: a short survey," *IEEE Network Magazine*, vol. 22, no. 6, pp. 42-47, Nov. 2008.

[4] S. Yao, S. Yoo, B. Mukherjee, and S. Dixit, "All-optical packet switching for metropolitan area networks: opportunities and challenges," *IEEE Communications Magazine*, vol. 39, no. 3, pp. 142-148, Mar. 2001.

[5] M. Yuang, I. Chao, and B. Lo, "HOPSMAN: an experimental optical packet-switched metro WDM ring network with high-performance medium access control," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 2, no. 2, pp. 91-101, Feb. 2010.

[6] M. Herzog, M. Maier, and M. Reisslein, "Metropolitan area packet-switched WDM networks: a survey on ring systems," *IEEE Communications Surveys & Tutorials*, vol. 6, no. 2, pp. 2-20, 2004.

[7] L. Xu, H. G. Perros, and G. N. Rouskas, "A simulation study of optical burst switching and access protocols for WDM ring network," *Computer Networks*, vol. 41, no. 2, pp. 143-160, Feb. 2003.

[8] Y. Arakawa, N. Yamanaka, and I. Sasase, "Performance of optical burst switched WDM ring network with TTFR system," *International Federation for Information Processing (IFIP '05)*, vol. 164, pp. 95–102, 2005.

[9] L. M. Peng, B. B. Li, W. H. Yang, Y. C. Kim, and X. W. Li, "Novel queue-grouping based round-robin protocols for reducing collision of OBS ring networks," *International Conference on Networking and Services (ICNS'09)*, pp. 97-101, 2009.

[10] L. M. Peng, and Y. C. Kim, "Investigation of the design of MAC protocols for TT-TR-based WDM burst-switched ring networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 1, no. 2, pp. 25-34, Jul. 2009.

[11] K. Ramantas, K. Vlachos, Ó. G. de Dios, and C. Raffaelli, "Window-based burst assembly scheme for TCP traffic over OBS," *IEEE/OSA Journal of Optical Networking*, vol. 7, no. 5, pp. 487-495, Apr. 2008.

[12] J. Kim, M. Maier, T. Hamada, and L. G. Kazovsky, "OBT: optical burst transport in metro area networks," *IEEE Communications Magazine*, vol. 45, no. 11, pp. 44-51, Nov. 2007.

[13] L. M. Peng, K. M. Yoo, K. E. Han, and Y. C. Kim, "Design and performance comparison of multiple-token based MAC protocols for optical burst switched ring networks," *Photonic Network Communications*, vol. 15, no.3, pp. 213-225, Jun. 2008.

[14] W. P. Chen, W. F. Wang, and W. S. Hwang, "A novel and simple beforehand bandwidth reservation (BBR) MAC protocol for OBS metro ring networks," *Journal of High Speed Networks*, pp. 59-72, Jan. 2008.

[15] L. Wang, M. Hamdi, R. Manivasakan, and D. Tsang, "Multimedia-MAC protocol: its performance analysis and applications for WDM networks," *IEEE*

Transactions on Communications, vol. 54, no. 3, pp. 518-531, Mar. 2006.

[16] H. Lin, W. Chang, and H. Wu, "FARE: An efficient integrated MAC protocol for differentiated services in WDM metro rings," *Computer Communications*, vol. 30, no. 6, pp. 1315-1330, Mar. 2007.

[17] P. Sarigiannidis, S. Petridou, G. Papadimitriou, M. Obaidat, and A. Pomportsis, "Supporting quality-of-service scheduling in a TT-FR WDM system," *IEEE Systems Journal*, vol. 2, no. 4, pp. 525-535, Dec. 2008.

[18] S. Chatziperis, P. Koutsakis, and M. Paterakis, "A new call admission control mechanism for multimedia traffic over next-generation wireless cellular networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 95-112, Jan. 2008.

[19] Z. Ali, W. Sheikh, E. Chong, and A. Ghafoor, "A scalable call admission control algorithm," *IEEE/ACM Transactions on Networking*, vol. 16, no. 2, pp. 424-434, Apr. 2008.

[20] J. Li, C. Qiao, and Y. Chang, "Recent progress in the scheduling algorithms in optical-burst-swtiched networks," *Journal of Optical Networking*, vol. 3, no. 4, pp. 229-241, Mar. 2004.

[21] S. Charchranoon, T. S. El-Bawab, J. D. Shin, and H. C. Cankaya, "Group-scheduling for multi-service optical burst switching (OBS) networks," *Photonic Network Communications*, vol. 11, no. 1, pp. 99-110, May 2005.

[22] H. J. Lin, *Utility-based fuzzy wavelength assignment in OBS network. Master Thesis*, 2009.

[23] R. G. Cheng and C. J. Chang, "Design of a fuzzy traffic controller for ATM networks," *IEEE/ACM Transaction on Networking*, vol. 4, no. 3, pp. 460–469, Jun. 1996.

[24] W. S. Tang and C. J. Chang, "Intelligent interring route control in bridged resilient packet rings," *IEEE/OSA Journal of Lightwave Technology*, vol. 27, no. 24, pp. 5631–5638, Dec. 2009.

[25] IEEE Standard 802.17, "Resilient packet ring (RPR) access method and physical layer specification," 2004.

[26] C. T. Lin and C. S. Lee, *Neural fuzzy systems*. Englewood Cliffs, NJ: Prentice Hall, 2003.

[27] T. G. Kwon, S. H. Lee, and J. K. Rho, "Scheduling algorithm for real-time burst traffic using dynamic weighted round robin," *International Symposium on Circuits and Systems (ISCAS'98)*, vol. 6, pp. 506-509, Jun. 1998.

[28] F. Alharbi and N. Ansari, "Distributed bandwidth allocation for resilient packet ring networks," *Computer Networks*, vol. 49, no. 2, pp. 161-171, Oct. 2005.

[29] W. C. Wang, *An effective fuzzy local fair rate generator for resilient packet ring. Master Thesis*, 2008.

[30] W. S. Tang, C. J. Chang, P. L. Tien, and W. C. Wang, "FLAG: a fuzzy local fairrate generator for resilient packet ring," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 2, no. 5, pp. 1-11, May 2010.

[31] F. Davik, A. Kvalbein, and S. Gjessing, "An analytical bound for convergence of the resilient packet ring aggressive mode fairness algorithm," *International Conference on Communications (ICC'05)*, vol. 1, pp. 281-287, May 2005.

Vita

Yao-Ting Hsieh was born in 1983 in Tainan, Taiwan. He received the B.E. degree in department of electrical engineering from National Cheng-Kung University, Tainan, Taiwan, in 2006, and the M.E. degree in institute of communication engineering, college of electrical and computer engineering from National Chiao-Tung University, Hsinchu, Taiwan. His research interests include optical burst switching, resource management, and intelligent techniques involving fuzzy logic.